Performances of Hill-Type and Neural Network Muscle Models—Toward a Myosignal-Based Exoskeleton

Jacob Rosen,* Moshe B. Fuchs,† and Mircea Arcan*,†

*Department of Biomedical Engineering and †Department of Solid Mechanics, Materials and Structures, Faculty of Engineering, Tel Aviv University, Ramat Aviv 69978, Tel Aviv, Israel

Muscle models are the essential components of any musculoskeletal simulation. In addition, muscle models which are incorporated in neural-based prosthetic and orthotic devices may significantly improve their performance. The aim of the study was to compare the performances of two types of muscle models in terms of predicting the moments developed at the human elbow joint complex based on joint kinematics and neuromuscular activity. The performance evaluation of the muscle models was required to implement them in a powered myosignal-driven exoskeleton (orthotic device). The experimental setup included a passive exoskeleton capable of measuring the joint kinematics and dynamics in addition to the muscle myosignal activity (EMG). Two types of models were developed and analyzed: (i) a Hill-based model and (ii) a neural network. The task, which was selected for evaluating the muscle models performance, was the flexion–extension movement of the forearm with a hand-held weight. For this task the muscle model inputs were the normalized neural activation levels of the four main flexor–extensor muscles of the elbow joint, and the elbow joint angle and angular velocity. Using this inputs, the muscle model predicted the moment applied on the elbow joint during the movement. Results indicated a good performance of the Hill model, although the neural network predictions appeared to be superior. Relative advantages and shortcomings of both approaches were presented and discussed. © 1999 Academic Press

1. INTRODUCTION

Integrating humans and robotic machines in one system offers a world of opportunities for creating a new generation of assistance technology that can be used in biomedical, industrial, and aerospace applications. The human contributes its natural and highly developed control algorithms that utilize advanced decision making and specialized fuzzy sensing mechanisms, whereas the robotic system offers technological advantages such as power, accuracy, and speed.

An ongoing project at the Biomechanics Laboratory of the Faculty of Engineering, Tel Aviv University, is the study, design, and implementation of a myosignal-based exoskeleton for the human arm. The purpose of the powered device is to amplify the human muscular system. Alternatively it can be construed as an orthotic device for assisting disabled persons, with impaired elbow muscles, to be

used in regular daily activities. The powered exoskeleton arm is a robotic manipulator, attached in parallel to the human arm, with links and joints corresponding to the elbow and shoulder joints. The torque generated by an actuator located at the elbow joint is used to amplify the moment produced by the elbow muscles such that a major part of the joint torque is produced by the actuator, leaving a fraction of the load to be carried by the human. The powered device thus increases the body muscle strength while maintaining human control of the movement.

The concept is not new and several such devices have been designed and implemented in the past (1, 2). However, what sets the present exoskeleton apart from its predecessors is that it uses myosignals from the elbow muscles (3) in addition to a set of feedback control signals to generate a command signal to the exoskeleton actuator. Using this approach the human–machine interface level (the junction in the system where the human and the machine exchange information) is raised to the neuromuscular junction. The myosignals produced during the elbow muscle contractions are used to predict the immanent muscle moments on the elbow joint. A major advantage of establishing an interface at the physiological muscle level is the ability to estimate the forces that will be generated by the muscles before the contraction phenomenon is fully developed. During the short time delay between the appearance of the myoelectric signals in the elbow muscles and the intended movement of the arm, the exoskeleton control system acquires the myosignals and the force feedback signals, processes, and interprets them, feeding an appropriate command signal to the elbow joint actuator, in time, to add its torque to the operator's muscular moment to move the arm under external load.

It is clear that the entire concept depends on the ability to acquire and process the myosignals in due time and in the quality of the predicted muscle forces. In other words, the technique requires reliable myoprocessors (4), a term which defines the module of the system responsible for estimating the moments about to be applied to the joint, based on the myoelectric signals and the joint kinematics, which are then fed into the actuator. This is the context of the present study. The specific objective of the research was to establish the feasibility of a Hill-based and/or neural network-type muscle model for predicting the muscle forces in the musculoskeletal system of the human elbow joint as a function of muscle neural activity and the joint kinematics.

Modeling neuromusculoskeletal systems is a major research subject in biomechanics. It is at the heart of any attempt to model and quantify the human body movement (5). Muscle models are also important in practical bioengineering research of orthotic, prosthetic, and functional neuromuscular electrical stimulation systems, for restoring lost or impaired motor function in disabled persons. A classification of muscle models, based on the level of structure that they address, was proposed by Zahalak (6). Using this approach the muscle models can be categorized as: (i) microscopic models, (ii) fiber models, and (iii) macroscopic models. The boundaries between these groups are not always distinct and some muscle models bear characteristics of more than one group. This is the case of the distributed-moment models (7–9) for instance, which may be construed as bridging the microscopic and the macroscopic domains. The muscle models which are

considered herein belong to the macroscopic whole-muscle category. This group is further divided into: (*iii-a*) viscoelastic models which consider muscle to have viscoelastic properties, (*iii-b*) Hill-based models which are variations of the proto-type model proposed by A. V. Hill in his classic paper (*10*), and (*iii-c*) system models which regard muscles as black boxes, the contents of which are to be determined by formal parameter identification procedures.

In the context of the present study the performances of two types of macroscopic muscles models were compared. The first model was the Hill-based (HB) muscle model (*iii-b*). It was based on Hill's classical muscle model which has been the subject of numerous studies. Its practical implementations had many variants; however, the current study was based on recent work of Winters and colleagues (*11–15*) using the direct modeling approach (input; myoelectric activity and joint kinematics; output, joint moment). The second muscle model was the neural net-work (NN) muscle model (*iii-c*). Neural networks, which were inspired by nervous systems, typically consisted of a large network of simple elements, capable of solving complicated and ambiguous problems and also to "learn" to recognize new patterns (*16*).

Applications of artificial NN for studying biological and biomechanical systems have appeared only within the past few years. Sepulveda *et al.* (*16*), for instance, used NN with a back-propagation algorithm to model the relationship between muscle activity and lower-limb dynamics during human gait. Similarly, Holzreiter and Kohle (*17*) applied a trained NN to distinguish between healthy and pathological gait. In the current study NN was used to predict the muscle moments with respect to the human elbow joint. The predicted moments of the HB and NN muscle models were compared with actual measured data.

The subsequent sections include descriptions of the elbow exoskeleton setup followed by fundamental definitions of the HB and the NN muscle models. The experimental data were first used for training the NN in addition to performance evaluation of both the HB and the NN (Section 2). The performance evaluation was accomplished by comparing the predicted elbow moment with the measured data (Section 3). This comparison is followed by a discussion of the relative merits and shortcomings of both models and the paper closes with conclusions (Section 4).

2. METHOD

This section includes a detailed description of the exoskeleton experimental setup and the various signals which were measured to create the experimental database. This is then followed by defending the architecture of the present variant of the HB muscle model and of the selected NN model. All the numerical simula-tions were performed in Matlab environment using Simulink and the Neural Net-work toolboxes.

2.1. The Exoskeleton Experimental Setup

The exoskeleton structure, used in the current study (Fig. 1), was a two degrees of freedom mechanism with links and joints corresponding to the upper (*SE*) and the lower (*EH*) right arm links and to the shoulder (*S*) and elbow (*E*) joints of
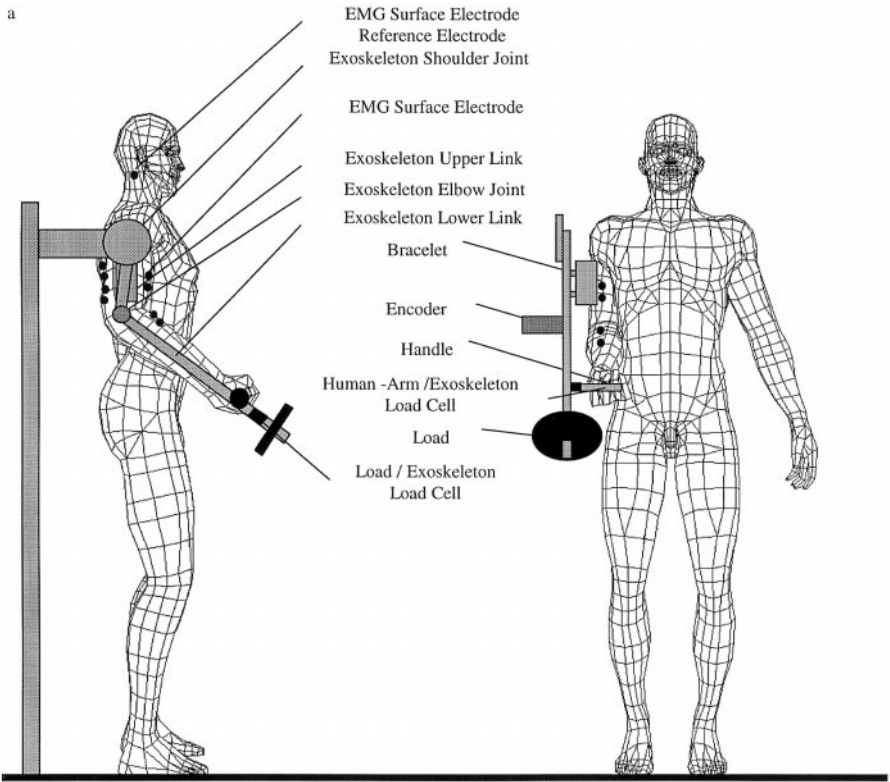
FIG. 1. (a) Frontal and lateral view of the experimental setup: The exoskeleton and human operator. (b) Schema of the two-link, two-joint exoskeleton. (c) The human operator using the exoskeleton devise.

the human body. The mechanical integration between the human arm and the exoskeleton was obtained by using a semicircular hollow cross section bracelet attached to the upper arm and handle (H) that was grasped by the operator. Weight plates (load) could be attached to the tip of the exoskeleton forearm link (T). The mechanism was fixed to the wall and positioned parallel to the sagittal plane of the operator. The two-degrees-of-freedom mechanism was defined by the shoulder joint angle (θ_S) and the elbow joint angle (θ_E). Black dots in Fig. 1a indicate locations where EMG surface electrodes were positioned to capture myosignals. In normal operating mode a rotational actuator was mounted on the elbow joint of the exoskeleton. When the human operator exerted a moment on the elbow joint, to lift a load, myosignals were generated and recorded by the surface electrodes. These signals were transferred to a myoprocessor for proper processing. The command signals were fed into the elbow actuator which supplied an additional torque at (E). The combined action of the human and the exoskeleton allowed the system to generate moments at the elbow joint, well beyond the normal loading

FIG. 1. Continued
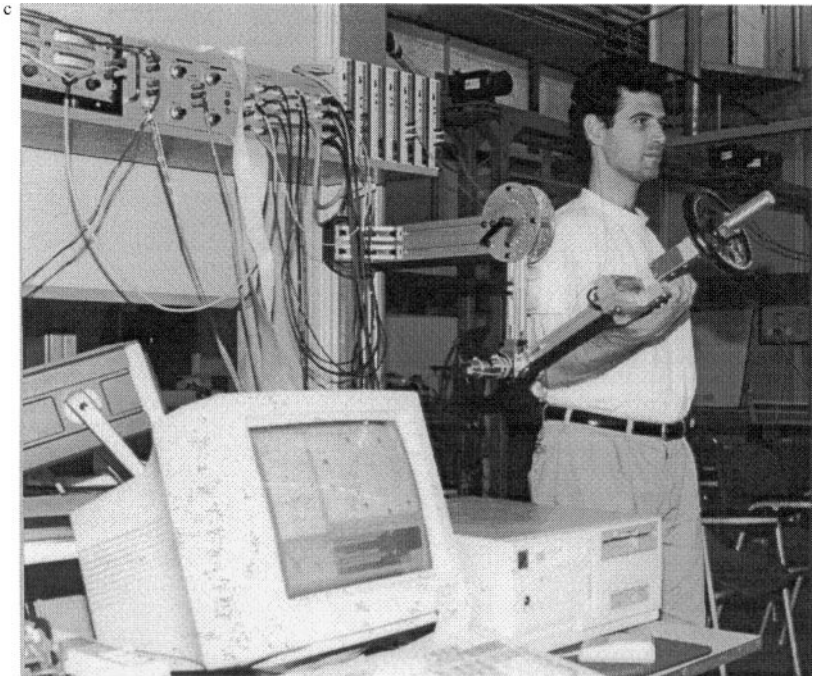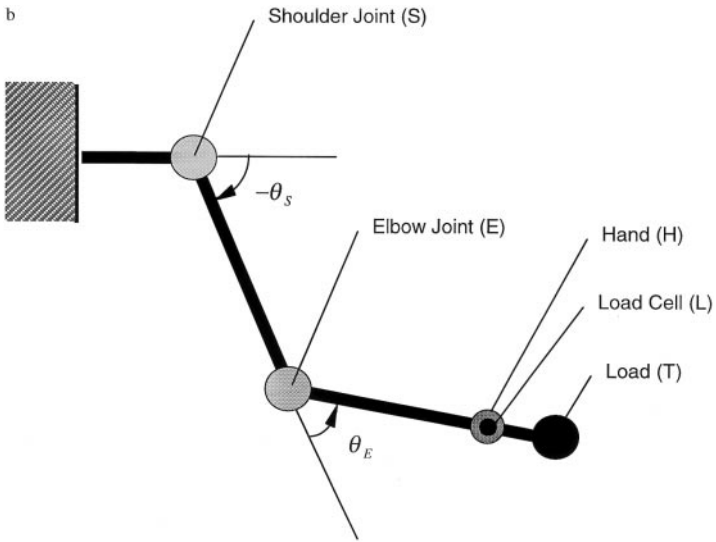
envelope of the human. As indicated earlier, the muscle model laid at the heart of the myoprocessor.

For the purpose of assessing the quality of the muscle models the exoskeleton was employed in passive mode. In this mode, the actuator at (E) was disconnected and the processed myosignals, instead of passing through the actuator, were recorded for off-line analysis. In addition, the system was reduced to a one-degree-of-freedom mechanism by fixing the system shoulder joint (S) at specific angles. Consequently the upper link of the exoskeleton and the upper arm (SE) were also fixed. The only possible movement was a rotation of the lower link and forearm (AH) about the elbow joint (E). The system had a built-in mechanical constraint limiting the elbow angle (θ_E) to remain within an average human antropometric range.

During the flexion/extension movement of the forearm, the following types of signals were sampled simultaneously: (i) the myosignals of the elbow joint muscles, (ii) the elbow joint kinematics, and (iii) the actual moment applied by the external loads on the exoskeleton elbow joint. The myosignals produced by the muscles of the elbow joint were measured by reusable 8-mm silver–silver chloride (Ag–AgCl) surface electrodes (BIOPAC EL208S). The electrodes were attached to the subject skin by adhesive disks at locations recommended by Basmajian and Blumenstein (*18*) for measuring the signals of the following muscles: biceps–brachii, brachiora-dialis, triceps brachii medial-head, and triceps brachii lateral-head. The signals were gained by EMG amplifiers (BIOPAC EMG100A) using a gain factor in the range of 2000–5000 (depending on the subject). All the channels were connected in parallel, sharing a same reference electrode (see Fig. 1a) and protected from environmental interference by using shielded cables.

As indicated earlier the shoulder joint angle (θ_S) was kept constant at specific values during the experimental session. The elbow angle (θ_E) was measured by an optical encoder (HP 5500) with an accuracy of $\pm 0.03°$, integrated into the exoskeleton joint. The elbow joint angular velocity ($\dot{\theta}_E$), which is also needed as an input by the muscle models, was computed numerically by means of finite differences. The moment applied by the external load at the elbow joint was measured by using a load cell (TEDEA 1040) with an accuracy of $\pm 3\%$, located on an extension (HT) of the lower segment of the exoskeleton (Fig. 1a). The load cell measured directly the dynamic shear load in a plane normal to the lower segment axis. Consequently the total moment relative to the elbow joint was obtained by multiplying the measured shear force by the moment arm (ET).

All the signals were acquired simultaneously. The electromyosignals and the load cell signal were sampled using an A/D convector (Scientific Solution Lab Master 12 bit internal PC card), whereas the encoder signals (measuring the elbow joint angle) were counted by digital counter.

2.2. Experimental Database

A typical arm movement session was full flexion followed by full extension of the elbow for a stationary position of the shoulder joint. The forearm in full supination position was free to move in the 2-D sagittal plane of the elbow joint.

Starting from full elbow joint extension ($\theta_E = 0°$) the forearm was rotated until full elbow joint flexion ($\theta_E = 145°$) followed by full extension, ending in the starting position ($\theta_E = 0°$). The experiments were repeated for three positions of the upper arm: two vertical ($\theta_S = -90°, 90°$) and one horizontal ($\theta_S = 0°$). For each joint setup the experiment was performed while carrying weight plates connected to the exoskeleton at (T). The movements where repeated for loads of up to 6 kg. Resting periods of 5 min between each experimental session were imposed such as to alleviate any fatigue effects

The signals which were recorded during a session were defined in Table 1. The first four signals, $I_1–I_4$, were the electromyosignals of the elbow muscles recorded by surface electrodes. The first two (*biceps brachii and brachioradialis*) were agonist muscles and the remaining two (*triceps brachii medial and lateral head*) were antagonist ones. Note, the elbow muscle complex included a third agonist muscle, the *brachialis*. However, since the *brachialis* muscle cannot be measured by noninvasive techniques, the *biceps-brachii* activation was also used for the *brachialis* when the HB model was used. It should be noted that muscle activation level, which was used as one of the muscle model input, was represented by a normalized value in the range of $\langle 0, 1 \rangle$. The muscle activation level was calculated as the ratio between the current neural activity and the same activity during maximal voluntary isometric muscle contraction. The algorithm for processing the raw mayoelectric signal (EMG) included four steps which were implemented in software: (i) high-pass fourth-order Butterworth filters with a cutoff frequency of 10 Hz, (ii) full-wave rectification, (iii) low-pass fourth-order Butterworth filters with a cutoff frequency of 6 Hz, and (iv) normalizing with respect to the maximal value of the muscle activation level, during isometric maximal voluntary contraction.

The three signals, $I_5–I_7$ in Table 1, were respectively the elbow angle, the elbow angular velocity, and the shoulder angle. Based on the measurement of the elbow joint angle, the angular velocity was computed by finite differences and the shoulder angle was determined at the beginning of the session. All these signals were used as input data to the muscle models from which the moment on the elbow joint was computed. The last recorded signal (O in Table 1) was the actual moment M

TABLE 1

Description of *I/O* Parameters of the Muscle Models

Model *I/O*	Symbol	Description
I_1	*BIC*	Biceps brachii muscle myosignal
I_2	*BRD*	Brachioradialis muscle myosignal
I_3	*TRIM*	Triceps brachii medial-head muscle myosignal
I_4	*TRIL*	Triceps brachii lateral-head muscle myosignal
I_5	θ_E	Elbow joint angle [rad]
I_6	$\dot{\theta}_E$	Elbow joint angular velocity [rad/s]
I_7	θ_S	Shoulder joint angle [rad]
O	M	Total moment on elbow joint [nm]

relative to the elbow joint, including dynamic and gravitational effects, exerted by the load while moving the forearm.

All these data constituted the experimental database. Its purpose was primarily to compile a set of input/output signals of the musculoskeletal system for training the NN muscle model. Indeed, the HB model used the Winters and co-worker approach (*14*) and needed no additional measurements. Finally, an experimental session was selected for evaluating the performance of muscle models by using the input signals I_1–I_7 with the HB and NN for predicting the moment on the elbow joint. This prediction was compared with the measured moment O.

2.3. The Hill-Based Muscle Model

The HB model was derived from the classical model originally introduced by A. V. Hill in 1938 (*10*). HB models provided a simplified representation of skeletal muscles. Although they have well-known limitations, they were widely used because of their ease of implementation (*8*). Many researchers had contributed to the development of the Hill model. The current formulation of HB model was a synthesis set of ideas and mathematical expressions from several references. The most consistent source of the HB model was Winters and colleagues (*11–15, 19, 20*) which included a modern engineering version of the model. All physiological parameters were taken from these publications. However, the contractile element (CE) force–velocity formulation and the algorithm for solving the model numerically were a synthesis of Winters' approach and several other references (*21–23*). For the benefit of providing a full and comprehensive description of the HB muscle formulation, this section includes all the definitions of the HB model components that were used in the current study.

The model consisted of three elements: a contractile element (CE), a series element (SE), and a parallel element (PE) as shown in Fig. 2a. The SE and PE components represented passive soft connective tissue including the tendon and the nonactive muscle fibers. The force/extension relations of these tissues were defined by [1] and [2]

$$F_{SE} = \frac{F_{SE\max}}{e^{SEsh} - 1}\left(e^{SE_{sh}\Delta L_{SE}/\Delta L_{SE\max}} - 1\right) \tag{1}$$

$$F_{PE} = \frac{F_{PE\max}}{e^{PEsh} - 1}\left(e^{PE_{sh}\Delta L_{PE}/\Delta L_{PE\max}} - 1\right) \tag{2}$$

where F_{SE}, ΔL_{SE} and F_{PE}, ΔL_{PE} are respectively the forces and extensions of elements SE and PE subscript max indicates corresponding maximal values, and SE_{sh} and PE_{sh} are SE and PE shape function parameters.

The CE component represented the active muscle fibers. Two properties characterized this element: CE force–length, CE force–velocity relations. The total force that was generated by the CE component was defined by a product relationship and could be described in the general form by [3]
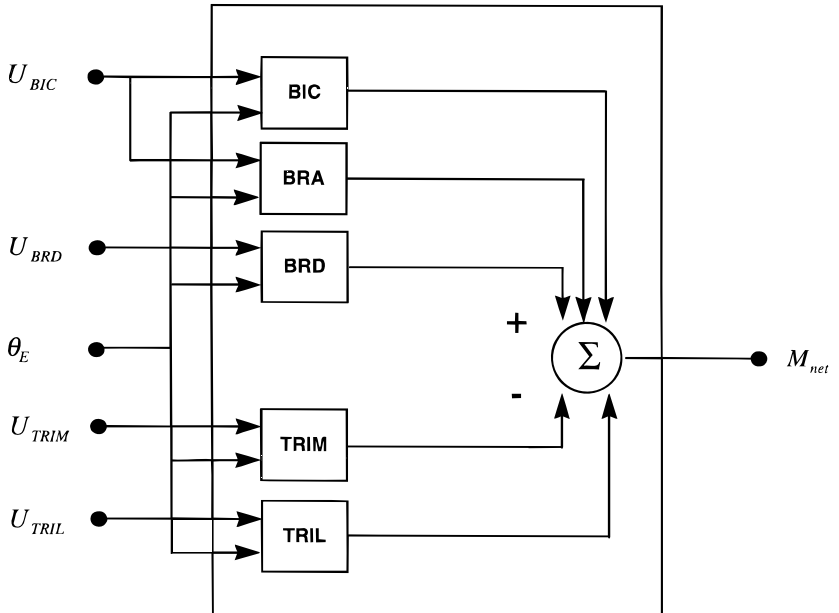
a
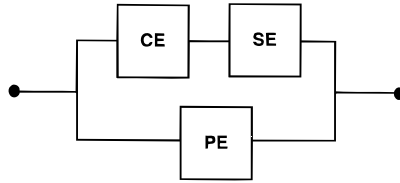
b
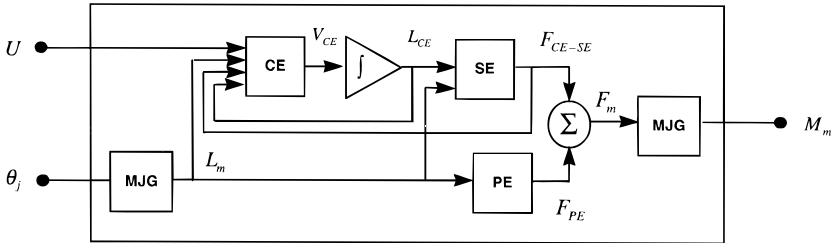
c

FIG. 2. Musculoskeletal system model. (a) Hill-based muscle model main components, (b) schema of the direct modeling approach, (c) model of the elbow joint complex.

$$F_{CE} = f_{FV}(V_{CE}) \cdot f_{FL}(L_{CE}) \cdot F_{max} \cdot U, \qquad [3]$$

where F_{CE} is the force in the CE component, f_{FV} and f_{FL} are dimensionless force–velocity and force–length functions, F_{max} is the maximal force in CE, V_{CE} and L_{CE} are respectively the lengthening (or shortening) velocity and length of CE, and U is the normalized activation level of the CE component.

The dimensionless force–length function in [3] was defined by [4]

$$f_{FL} = e^{-0.5((L_{CE}/L_0 - 1.05)/0.19)^2}, \qquad [4]$$

where L_{CE} and L_0 are the length and resting length of the CE component.

It should be noted that Hill's classical hyperbolic formulation of the CE force–velocity function was valid in the muscle-shortening phase only. The generalized form of the CE force–velocity characteristics was valid for both shortening and lengthening regions (21, 22)

$$f_{FV} = \frac{0.1433}{0.1074 + e^{-1.409 \sinh(3.2 V_{CE}/V_{max} + 1.6)}} \qquad [5]$$

$$V_{max} = 0.5(U + 1)V_0, \qquad [6]$$

where V_{max} and V_0 are maximal velocities of the CE at specific muscle activation levels and at maximal activation level, respectively, and U is the activation level of CE.

Furthermore, Eq. [6] was a modification according to Winters (15) in order to include the dependency of the maximal velocity on the muscle activation.

The structure of the algorithm for solving the HB muscle Eqs. [1]–[6], shown in the block diagram of Fig. 2b, followed the approach developed by Crowe et al. (23). The method, applied to the elbow joint, implemented the direct modeling approach, where the inputs were defined as the muscle activation level (U) and the elbow joint angle (θ_E) while the output was the muscle moment (M_m). This approach was implemented for every muscle of the elbow complex separately. Each muscle moment (M_m) applied on the joint was computed based on the muscle force (F_m) and the joint geometry. The subscript m indicated the individual muscle. The muscle length (L_m) was a function of the elbow joint angle and the muscle–joint geometry (MJG). The latter defined the transformation between the joint angular coordinates and the muscle local Cartesian coordinates (14). The algorithm for solving the HB model made use of the fact that the force generated by the contractile element was equal to the force generated by the serial element since they were connected in series ($F_{CE-SE} = F_{CE} = F_{SE}$). The total muscle force was defined as the sum of its active and passive parts ($F_m = F_{CE-SE} + F_{PE}$). Finally, the total moment (M_{net}) with respect to the elbow joint [3] was the summation of individual

muscle moments with the proper sign $[+]$ for agonist muscles and $[-]$ for antagonist muscles (Fig. 2c).

$$M_{net} = \sum_{1}^{n} M_{mi} \qquad [7]$$

2.4. The Neural Network Muscle Model

In contrast to the semianalytical HB muscle model which might be considered as a simplified approximation of the biology and the mechanics of the skeletal muscle, the NN muscle model was a mathematical model of the process mapping the input set to the output set, without the need to formulate the system functionality. This modern mathematical tool took its name from the namesake arrangement found in biological nervous systems. These two systems consist of a large number of simple elements that are collectively able to learn the mapping of the process and can then proceed to solve complex problems. The rationale behind the NN technique was that a large array of typical inputs and related outputs of a system were presented to the network. After the learning or training phase the NN was expected to predict the output when given an arbitrary input set. In the context of the HB muscle model the representation of elbow muscle/joint complex was achieved by training the NN with a set of input signals I_1–I_7 and the related moment output O was recorded while the subject moved the load by rotating the forearm. Having learned typical relationships between $\{I\}$ and $\{O\}$, the NN was capable to provide a fair approximation of the elbow moment for any arbitrary input set. By comparing this method with the classical HB approach, it was clear that the two techniques were located at opposite ends of possible muscle models' scale.

The primary consideration in developing the NN was to define its architecture. A typical network architecture for function approximation was the feed-forward multilayer NN composed of an input layer, an output layer, and one or more intermediate hidden layers (*24*) (Fig. 3). The size of the input and output was determined by the number of input/output signals; however, the number of neurons in the middle layers determined the power of the system to generate the I/O mapping (*21*). The selected NN architecture included one hidden layer with seven neurons in the input layer (I_1 to I_7) and one unit in the output layer (the total moment O exerted by the load on the elbow joint). The network architecture had a tangential–sigmoid hidden layer [8] and a linear output layer [9] (Fig. 5).

$$\{H\} = \text{TanSig}([W_1]\{I\} + \{B_1\}) \qquad [8]$$

$$\{O\} = [W_2]\{H\} + \{B_2\}, \qquad [9]$$

where $\{H\}$ is the output vector ($n \times 1$) of the hidden layer, $[W_1]$ is the weight matrix ($n \times 7$) of the hidden layer, $\{B_1\}$ is the bias vector ($n \times 1$) of the intermediate hidden layer, $[W_2]$ is the weight matrix ($1 \times n$) of the output layer, $\{B_2\}$ is the bias vector (1×1) of the output layer, and n is the number of neurons (size) in the hidden layer.

a
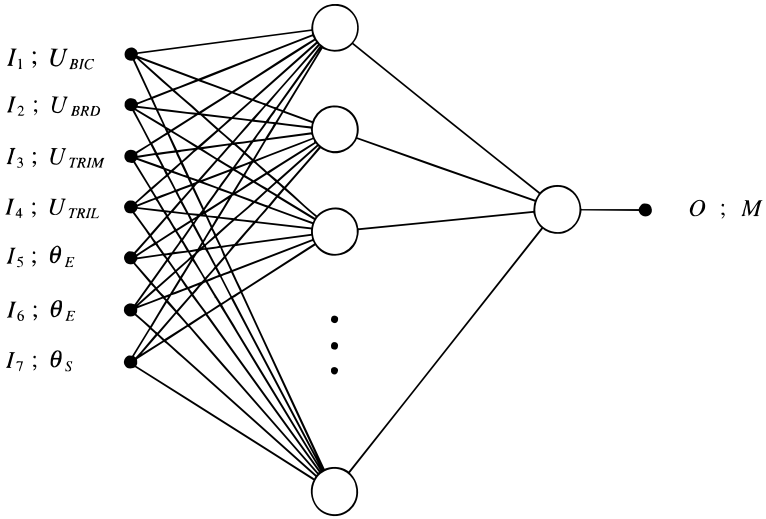
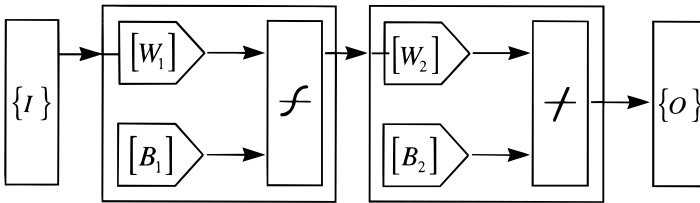Input Signals   Middle (Hidden) Layer   Output Layer   Output Signal

b

FIG. 3.   Fully connected two layer neural network myoprocessor. (a) NN schematic representation. (b) NN compact form of transfer functions [8]–[10]. *I*, input; *O*, output; *U*, muscle activation levels; *θ*, joint angle; *BIC, biceps brachii; BRD, brachioradialis; TRIM, triceps brachii medial head; TRIL, triceps brachii lateral head; E, elbow; S, shoulder; M, moment on elbow hinge; W, weight matrices; B, bias vectors.*

The tangential–sigmoid transfer function (TanSig) was defined by Eq. [10]

$$\text{TanSig} = \text{Tanh}(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \qquad [10]$$

This function produced values in the range $\langle -1, 1 \rangle$, whereas the biases and the linear transfer function let the network generate values outside that range.

The purpose of the network was to compute the output vector $\{O\}$ (1×1) based on the input vector $\{I\}$ (7×1). Equation [8] transforms $\{I\}$ into $\{H\}$ of

the hidden layer. These *n* values constituted the inputs to the second transformation [9] which produced the output *O*. All this depended of course on the entries of the weight and bias matrices. Determining best values for these constants was performed during the learning phase of the process. The network was presented with a set of ({*I*}, {*O*}) pairs, the training set. For a given set of coefficients of the matrices components [W_1], {B_1}, [W_2], and {B_2} in [8] and [9] the network could calculate the current approximation of the output $\tilde{O}$ and this could be compared to the measured output *O*. Repeating this for all the ({*I*}, {*O*}) pairs in the training set allowed to calculate the sum squared error (SSE)—Eq. [10], where the index (*i*) runs over all the ({*I*}, {*O*}) pairs.

$$SSE = \sum_i (O_i - \tilde{O}_i)^2 \qquad [11]$$

The *SSE* was used as an indication of the NN performance for the current values of the weighting and bias matrices. Modifying these matrices to minimize the *SSE* was performed by the back-propagation algorithm. If the minimum value of *SSE* was less than a target value, or if the number of the training sessions had reached a limiting value, the learning process was stopped. If the *SEE* had not reached its target value during the iterative process before the preset number of iterations was performed, the number of the training iteration could increased. Another alternative was to modify the network architecture by adding a hidden layer and/or changing the number of neurons in the intermediate layers. The process was then repeated with the purpose of pushing the *SSE* below the threshold value.

Selecting the number of neurons in the hidden layers is influenced by two contradictory considerations. The size or number of neurons in the hidden layers determines the power of the NN. A large number of neurons enhances the network's ability in mapping the *I/O* gates thereby curtailing the error [11]. On the other hand, reducing the number of neurons decreases the time needed for the learning procedure and also the number of floating point operations needed to calculate {*O*}, a key factor in any real-time approach.

Several techniques were used to accelerate the training procedure of a NN with an architecture based on a neural layer and a tangential–sigmoid transfer function (*24*). The Nguyen–Widrow method (*26*) was used for selecting such initial conditions which shorten the training time. In addition, the adaptive learning rate algorithm was used. It decreased the time needed for the learning process while keeping the network from bouncing around, thus maintaining a stable learning procedure (*25*). Finally, the momentum technique allowed the network not only to respond to the local gradient, but also to comply with recent trends in the error surface while ignoring small features in the error. It was recognized that with a momentum, a network can slide through shallow local minima (*24*).

In applying the NN for modeling the muscles, the *I/O* signals were the same as those used for the HB model. The reason was primarily that this type of *I/O* configuration was based on the standard simplified assumption, that a muscle was generating a moment relative to a joint as a function of the muscle activation level,

and the joint kinematics. Moreover, selecting identical *I/O* ports for both the HB and NN models provided a common ground base to perform a comparison between them. However, it should be noted that whereas the HB model defined each muscle as a different entity to be modeled separately (the final result was the superposition of all the elements) in the NN model, the muscles were lumped into a unified set and the superposition was inherent to the system.

3. EXPERIMENTAL RESULTS

This section includes a description of the experimental results and the model performances. The first subsection describes the *I/O* set and the target tasks for training of the NN muscle model. This is then followed by second and the third subsections defining the quantitative performance of the NN model and comparing it to the HB model.

3.1. Recording a Typical Session

In a typical data acquisition session the upper arm was kept fixed at a constant shoulder angle. The forearm in full supination position was free to move in the 2-D sagittal plane of the elbow joint. Starting from full elbow joint extension the forearm was rotated until full elbow joint flexion followed by full extension, ending in the starting position. During the session the subject was wired as described in Section 2.

An example of a data set of a typical experimental session is plotted in Fig. 4. The subject who participated in the study was a 30 year old male in good physical conditions. A typical task included moving the forearm from full extension to full flexion and back to full extension. In this particular task (Fig. 4), the upper arm was kept in a vertical extension position ($\theta_S = -90°$) and a weight of 5 kg was attached to the exoskeleton tip.

In Fig. 4a the raw EMG signal and the muscle activation level ($\langle 0, 1 \rangle$) are plotted as a function of time for each one of the flexor muscles (*biceps brachii, brachioradialis*) and the extensor muscles (*triceps brachii medial head, triceps brachii lateral head*). Figure 4a indicates that the flexor muscles are active both in flexion ($t = 2.5-5$ s) and in the extension ($t = 5-8$ s) movements. The minor cocontraction of the extensor muscle is performed to overcome the passive elasticity of the flexor muscles keeping the arm in a full flexion position ($t = 0-2.5$, $8-12$ s) and to stabilize the elbow joint during the contraction of the flexor muscles' activity ($t = 2.5-8$ s). Figures 4b and 7a show the kinematics and dynamics of the flexion/extension of the elbow joint respectively as a function of time.

The elbow joint moment (Fig. 7a) is characterized by two peaks which occur when the forearm was perpendicular to the (vertical) upper arm and the system has to carry the full thrust of the load moment. The two peaks in the elbow joint moment are created once during the flexion movement and once during extension. Similar phenomena (two peaks) but with a negative sign also occur when $\theta_E = 90°$, as can be seen in Fig. 5. Note that in these shoulder positions the moment signals have the same value but with opposite signs.
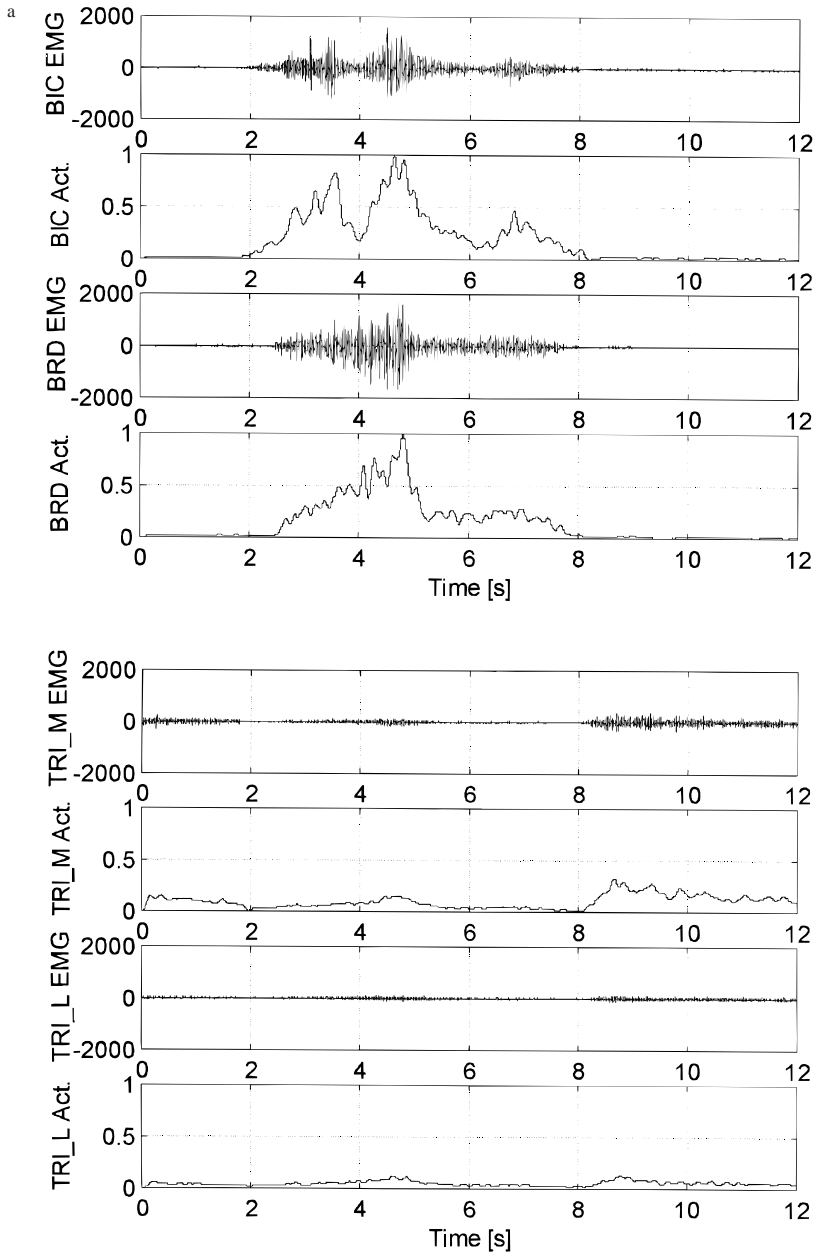
FIG. 4. Typical session recordings. (a) EMG of the elbow flexor and extensor muscles: *BIC, biceps brachii; BRD, brachioradialis; TRI_M, triceps brachii medial head; TRI_L, triceps brachii lateral head; EMG*, raw signal; *Act*, normalized activation level. (b) Elbow kinematics: elbow joint angle, angular velocity, and angular acceleration.
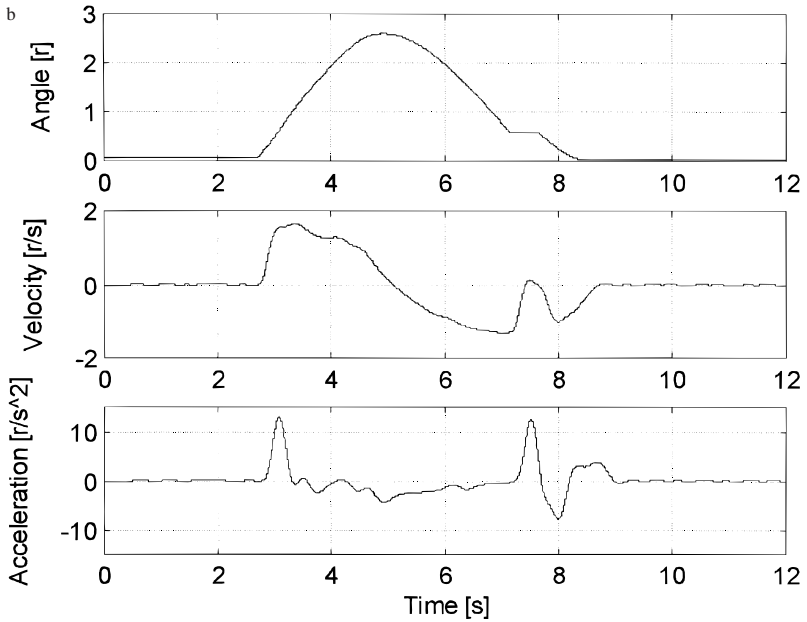
FIG. 4.   Continued

3.2. Neural Network Training

The HB model (Section 2.3) from a system engineering perspective is a phenome-nological-based, lumped parameter model in which its mathematical formulation defines a simplified characterization of the skeletal muscle. Once the model is defined, a single set of inputs is sufficient for predicting its output. In contrast, the NN-based model requires a training set to compute best values of the weight matrices ($[W_1]$, $[W_2]$) and bias vectors ($\{B_1\}$, $\{B_2\}$) of the network transfer functions [8] and [9]. For that purpose the 13 sessions appearing in Table 2 were used.

The parameters of the training sessions were the shoulder position and the applied load (Table 2). In the first five sessions the upper arm was supported vertically in a flexion position ($\theta_S = -90°$). In the next four sessions the upper arm was supported horizontally ($\theta_S = 0°$), and in the last four sessions the upper arm was supported vertically in extension position ($\theta_S = 90°$). The system was loaded from 0 kg, with increments of 1 kg, up to 5 kg. In all cases the forearm moved from full elbow extension ($\theta_E = 0°$) to full flexion ($\theta_E = 145°$) and back to full extension. Every session was recorded similarly to what was described in Section 3.1.

Concatenated experimental results for all 13 sessions are plotted in Fig. 5. The first 7 diagrams represent the input components $\{I\}$ and the last diagram is the measured total moment relative to the elbow joint O. The 13 sessions are concate-nated and presented as one continuous output, although 5-min resting periods are
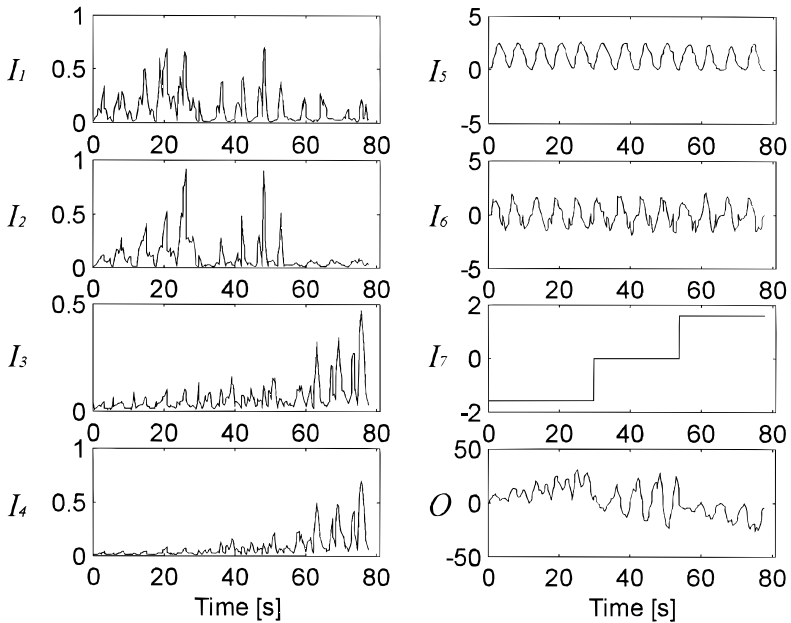
FIG. 5. Concatenated *I/O* signal recordings of the 13 experimental sessions for training the NN muscle model. I_1, normalized *biceps brachii* activation level; I_2, *brachioradialis*; I_3, *triceps brachii medial head*; I_4, *triceps brachii lateral head*; I_5, elbow joint angle; I_6, elbow joint angular velocity; I_7, shoulder joint angle; *O*, moment on elbow joint.
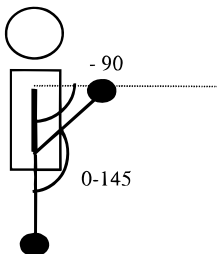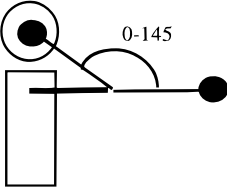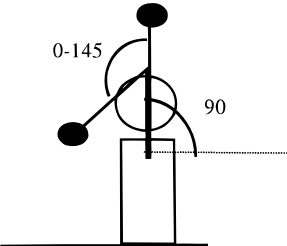
imposed between the sessions to alleviate any fatigue effects. Signals I_1 to I_4 are the normalized muscle activities, I_5 and I_6 are the elbow angle and angular velocity, I_7 is the shoulder joint angle, and *O* is the measured moment applied by the load on the elbow.

This concise overview of the entire database gave some insight on the whole process. One might notice, for instance, the step function in I_7 (the shoulder angle). This angle was kept fixed at 3 angular positions for a recording time of about 20 s in each position. In addition, when the shoulder was in full flexion position, the agonist muscles' activities I_1, I_2 were higher than the activities of the antagonist muscles I_3, I_4. The opposite prevails when the shoulder was in full extension. In sessions where the upper arm was supported horizontally ($\theta_S = 0°$), as expected, the agonist muscles were active in the joint angle range of $0° < \theta_S < 90°$ and the antagonist muscles were active in the joint angle range of $90° < \theta_S < 145°$. The elbow joint kinematics I_5 was a rather smooth curve and, as indicated, I_6 was the time derivative of I_5. Finally, the measured elbow moment and the NN predicted values (*O*) were plotted in Fig. 6.

The network training data were composed of 260 (13 sessions × 20 samples each) sets of vectors yielding a 7×260 input matrix and a 1×260 output vector. As noted earlier, the sampling rate was much higher but the recorded data were resampled to include 20 data sets per session to bring it in line with the computational power of Matlab and NN toolbox. The training set was presented to the

TABLE 2

Configuration of the Experimental Sessions of the NN Training Set

Elbow joint angle (°)	Shoulder joint angle (°)	Weight (kg)	
0–145	−90	1	
0–145	−90	2	
0–145	−90	3	
0–145	−90	4	
0–145	−90	5	
0–145	0	1	
0–145	0	2	
0–145	0	3	
0–145	0	4	
0–145	90	1	
0–145	90	2	
0–145	90	3	
0–145	90	4	

network, and optimal values of the weight matrices ($[W_1],[W_2]$) and bias vectors ($\{B_1\},\{B_2\}$) were determined. The actual calculation was performed in batch mode. The entire set was presented to the network and the constants were determined in iterative manner. For each input presentation (epoch) the network computed its estimated output $\tilde{O}$. The network learning performance was tested using the sum squared error (SSE) between the network estimation $\tilde{O}$ and the target output O (11).

During the training session which included 200,000 epochs of a NN with a 50 neuron in the middle layer, the SEE decreased exponentially with a final value of 14.8 $(nm)^2$. This value corresponded to an average error of 0.015 nm. By increasing the number of neurons to 100, the SEE was decreased to 12.8 $(nm)^2$. Reducing the number of neurons in the middle layer to 25 caused the SEE to increase dramatically to a value of 167.6 $(nm)^2$. Figure 6 shows the performance of the NN with 50 neurons in the middle layer in predicting the joint moment. The 260 target values of the output (+) are plotted against the NN prediction, given by a
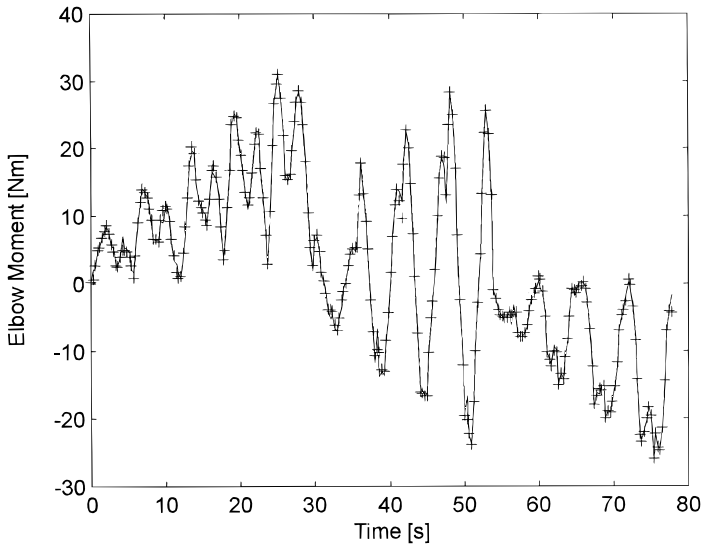
FIG. 6. Neural network estimation of the elbow moment for the training set (line) and measured (target) values (+).

continuous line. As shown in Fig. 6, the NN estimates the moments of the training set in a very satisfactory manner.

3. HB and NN Muscle Models Performance Comparison

To determine the qualities and compare the predictions of the HB and NN models of the human elbow joint complex, a typical movement of the lower arm, the target movement, was selected. This movement, which was not included in the training set of the NN, was described in detail in Section 3.1. Note that although the target movement was performed by the same subject and under the same conditions (arm position and loading) as defined by session 4 of the training set, the data of session 4 in the training set and the target movement were different. Although a same operator performed both movements, they were recorded separately at different times, and indeed, a variability in all the *I/O* signals exists.

The measured and the predicted moments by the HB and NN models for the selected arm movement are plotted in Fig. 7. The measured moment (Fig. 7a) presents typically two peaks separated by a local minimum. The peaks correspond to approximate horizontal positions of the forearm where the load has a maximum lever arm with respect to the elbow joint. The minimum value of the moment corresponds to the maximum flexion angle when the lever arm is, locally, minimum. The HB model (Fig. 7b) follows the general trend, although the first maximum becomes here a double hump with the first peak larger than the second one. This discrepancy is most probably due to the activity of the *biceps brachii* muscle during flexion. The average error between the HB model prediction and the measured data
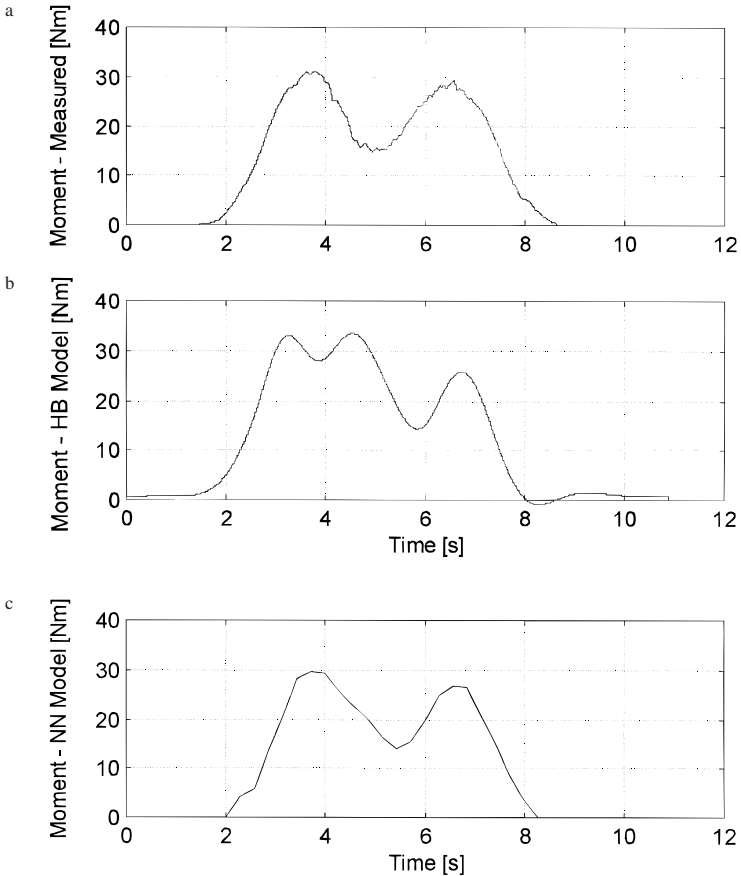
FIG. 7. The moment developed at the elbow joint during flexion/extension of target movement of the forearm while carrying a weight of 5 kg. (a) Measured data, (b) Hill-based model estimation, and (c) neural network model estimation.

is 4.2 nm. The NN model (Fig. 7c) gives, in general, better predictions for the elbow joint load moment. The moment peaks are almost exactly where they appear in Fig. 7a (measured data). The amplitude of the first peak fits perfectly, although the second peak of the NN model is somewhat lower than the corresponding measured value. The average error between the NN model prediction and the measured data is 0.012 nm.

4. DISCUSSION AND CONCLUSIONS

The present study is an attempt to evaluate the performances of two types of muscle models with a view of employing them as myoprocessors in an orthotic device—a powered exoskeleton for the human elbow joint. The task of a muscle model (myoprocessors) in the exoskeleton application is to predict the moment

that should be developed by the muscle for manipulating an external load as a function of the corresponding muscles activation levels and the joint kinematics. The selected models are a classical Hill-based muscle model and a neural network model. The two examples are extreme cases of possible alternative solutions. The HB model is a simplified phenomenological-based, lumped parameter model of the skeletal muscle, whereas the NN model is a black-box *I/O* mapping without any attempt to follow the internal knowledge of the muscle complex.

It is clear that the NN predictions, for a well-defined set of data, are in this case better than what the HB model could provide. These results are at first glance rather startling. A black box (NN model), with no knowledge of the underlying physical phenomena ruling its *I/O* ports, is capable of predicting the *I/O* mapping at least as well as a semiempirical model (HB model) based on years of minute observations, testing, and a good understanding of the principles guiding the phenomena. This newcomer in muscle modeling is outperforming time-honored techniques such as the HB muscle model. The broader picture is however not so definite.

Due to the fact that NN models rely on training sets, NN is very much task dependent. This feature turns out to be both its advantage and its disadvantage. Indeed, the power of the NN as a myoprocessor, for instance, lies in its ability to accommodate predefined *I/O* sets. The NN is capable of representing the physiology of one specific operator for tasks defined in the learning set, but its scope is limited to performances within this set only. As the range of applications broadens, by performing tasks outside the space defined by the learning set or by applying the predefined NN to a different subject, NN models are known to loose their effectiveness. Models of the HB type, on the other hand, are universal. They are valid for most individuals and they can handle a large variety of situations. The drawback of the HB approach is that its predictive power is somewhat hampered by its global nature, leading to results inferior to those of specially trained NN models.

In addition, the HB model does not need any training sessions, and due to its compact form and small number of parameters, it does not require high computational power. This is an important consideration. Handling the current network architecture with 50 neurons in the middle layer was not a simple numerical task. One should bear in mind that Fig. 7c was obtained off-line and that the computation time was thus of little relevance. It was however rapidly recognized that computing power was an important consideration in an on-line mode. Adopting, for instance, the 100-neuron-size middle layer was out of the question, and as indicated, moving to a more tractable 25-neuron middle layer was very detrimental for the quality of the moment predictions. It is noteworthy that in the context of the arm exoskeleton the HB model was eventually adopted because of its generality and independence on training sessions. Training the NN with handicapped persons proved rather difficult.

Table 3 summarizes some characteristics of the NN and the HB models. As already mentioned, the HB and NN models belong to the macroscopic muscle model group. The architecture of the HB model is complex and requires one to solve a set of differential equations; however, the number of model parameters is relatively small and these parameters are related to the anatomy/physiology of an average human. The NN formulation, on the other hand, is relatively simple and

TABLE 3

Comparing Neural Network and Hill-Based Model Characteristics

Category	Characteristic	Neural network	Hill-based
Type		System—Black box	Phenomenological
Architecture	Formulation	Simple	Complex
	Parameter number	Large	Small
	Parameter value	Adjustable	Fixed
	Parameter meaning	None	Anatomy/physiology
Training		Yes	No
Computational power		High	Low
Performance	I/O in training set	High	Acceptable
	I/O out of training set	Low	Acceptable
	Scope of application	Limited	Wide

its architecture is based on two types of neuron transfer functions (tangential–sigmoid and linear). But the cost of using simple building blocks is the large number of neurons in the intermediate layer which is necessary to provide the NN with sufficient flexibility for learning the various instances of the I/O spectrum. The NN parameters are thus adjusted to the specific conditions by means of back-propagation algorithm which tuned the constants of the transfer functions. One the other hand, the HB model does not need any training sessions, and due to its compact form and small number of parameters does not require high computational power. A further flexibility of the network models is that the quality of the model can be enhanced by increasing the training set and by modifying the network architecture (number and size of the hidden layers). But care must be exercised, since overtraining may cause the system to fluctuate very rapidly to simulate as close as possible the instances of the training set.

A known limitation of the present study was due to the fact that all the simulations were performed using data from a single subject. Although a single subject provided enormous versatility in terms of the kinematics, dynamics, and myosignals activity, a full understanding of both the HB and the NN might be achieved by using a number of subjects. The approach outlined in this study might be replicated for different joints with more subjects as well as different task movements.

The interim evidence that may be deduced from the present results is that the classical HB model may provide a limited but reasonable estimation of the physiological muscle performances. Making allowance for the fact that the model is universal and is not task dependent is an additional argument in favor of classical HB muscle models. On the other hand, the results have also shown that NN muscle models are serious contenders in the quest for efficient models. The architecture that is used in the present network in conjunction with the training set produced moment predictions which are in many respects superior to the HB data. What limits the NN method is the need to keep the applications inside the training space in addition to the necessity for substantial computing power. Neural networks can be numerically cumbersome. Further research is obviously warranted, both in

applying the technique to other muscular systems and in employing improved NN architectures. It is to be expected that, with increased computational power, NN models may become successful contenders in the quest for reliable and efficient neuromuscular models.

APPENDIX

List of Symbols

$\{B_1\}$—Intermediate (hidden) layer bias vector ($n \times 1$)

$\{B_2\}$—Output layer bias vector (1×1)

CE—Contractile element of HB model

F_{CE}—Force of the CE

F_m—Muscle force

F_{max}—Maximal force of the CE

F_{PE}—Force of the PE

F_{PEmax}—Maximal force of the PE

F_{SE}—Force of the SE

F_{SEmax}—Maximal force of the SE

$\{H\}$—Intermediate (Hidden) layer output vector ($n \times 1$)

$\{I\}$—Input vector (7×1)

f_{FL}—Force–length dimensionless function

f_{FV}—Force–velocity dimensionless function

L_0—Resting length of the CE.

L_{CE}—Length of the CE

ΔL_{PE}—PE extension

ΔL_{PEmax}—Maximal PE extension

ΔL_{SE}—SE extension

ΔL_{SEmax}—Maximal SE extension

M_m—Moment applied by a muscle on the joint

M_{net}—Total net moment applied by all the muscles on the joint

n—Number of neurons in the intermediate (hidden) layer (size of the layer)

$\{O\}$—Output vector (1×1)

$\{\tilde{O}\}$—Output vector (1×1) approximation during the NN training

PE—Parallel element of the HB model

PE_{sh}—PE shape function parameter

SE—Serial element of the HB model

SE_{sh}—SE shape function parameter

SSE—Sum squared error

U—Activation level ($\langle 0, 1 \rangle$)

V_0—Maximal velocity of the CE at maximal activation level

V_{CE}—Lengthening or shortening CE velocity

V_{max}—Maximal velocity of the CE at specific muscle activation

$[W_1]$—Intermediate (hidden) layer weighting matrix ($n \times 7$)

$[W_2]$—Output layer weighting matrix ($1 \times n$)

REFERENCES

1. Rosheim, M. E. "Robot Revolution," Wiley, New York, 1994.

2. Kazerooni, H. The human power amplifier technology at the University of California, Berkeley. *Robotics Autonomous Syst.* **19,** 179–187 (1996).

3. Rosen, J. "Natural Integration of a Human Arm/Powered Exoskeleton System," Ph.D. thesis, Faculty of Engineering, Tel-Aviv University, Israel, 1997.

4. Prutchi, D. "Relationships between Mechanical and Electrical Activity of the Muscle," Ph.D. thesis, Tel-Aviv University, Israel, 1992.

5. Ailon, A., Langholz, G., and Arcan, M. An approach to control laws for arm motion. *IEEE Transact. Biomed. Eng.* **31**(9), (1984).

6. Zahalak, G. I. An overview of muscle modeling. *In* "Neural Prostheses" (R. B. Stein, P. H. Peckham, and D. B. Popovic, Eds.), Oxford Univ. Press, New York, 1992.

7. Zahalak, G. I. A distribution-moment approximation for kinetic theories of muscular contraction. *Math. Biosci.* **55,** 89–114 (1981).

8. Zahalak, G. I. Modeling muscle mechanics (and energetics). *In* "Multiple Systems: Biomechanics and Movement Organization" (J. M. Winters and S.L-Y Woo, Eds.), pp. 1–23, Springer-Verlag, Berlin, 1990.

9. Zahalak, G. I., and Ma, S. P. Muscle activation and contraction: Constitutive relations based directly on cross-bridge kinetics. *J. Biomech. Eng.* **112,** 52–62 (1990).

10. Hill, A. V. The heat of shortening and the dynamic constants of muscle. *Proc. R. Soc. London Ser. B* **126,** 136–195 (1938).

11. Winters, J. M., and Stark, L., Analysis of fundamental human movement patterns through the use of in-depth antagonistic muscle models. *IEEE Transact. Biomed. Eng.* **BME-32**(10), 826–839 (1985).

12. Winters, J. M., and Stark, L. Task-specific second-order movement models are encompassed by a global eighth-order nonlinear musculo-skeletal model. *In* "IEEE Proceeding on System, Man, and Cybernetics," pp. 1111–1115, November, Arizona, 1985.

13. Winters, J. M., and Bagley, A. M., Biomechanical modeling of muscle–joint system. *IEEE Eng. Med. Biol. Magazine* **September,** 17–21, (1987).

14. Winters, J. M., and Stack, L. Estimated mechanical properties of synergistic muscle involved in movements of a variety of human joints. *J. Biomech.* **21**(12), 1027–1041 (1988).

15. Winters, J. M. Hill-based muscle models: A systems engineering perspective. *In* "Multiple Systems: Biomechanics and Movement Organization" (J. M. Winters, and S.L-Y. Woo, Eds.), pp. 69–93, Springer-Verlag, Berlin, 1990.

16. Sepulveda, F., *et al.* A neural network representation of electromyography and joint dynamics in human gait, *J. Biomech.* **26**(2), 101–109 (1993).

17. Holzreiter, S. F., and Kohle, M. E. Assessment of gait patterns using neural networking, *J. Biomech.* **26**(6), 645–651(1993).

18. Basmajian, J. V., and Blumenstein, R. "Electrode Placement in EMG Biofeedback." Williams & Wilkins, Baltimore London, 1980.

19. Winters, J. M., and Kleweno, D. G. Effect of internal upper-limb alignment on muscle contributions to isometric strength curves, *J. Biomech.* **26**(2), 143–153 (1993).

20. Zajac, F. E., and Winter, J. M. Modeling musculo-skeletal movement system: joint and body segmental dynamics, musculo-skeletal actuation, and neuromuscular control. *In* "Multiple Muscle System" (J. M. Winters, and S. L. Y. Woo, Eds.), Spring-Verlag, Berlin, 1990.

21. Hatze H. "Myocybernetic Control Models of Skeletal Muscle." University of South Africa Press, Pretoria, 1981.

22. Kaufman, J., *et al.* Physiological prediction of muscle forces. I. Theoretical formulation. *Neuroscience* **40**(3), 781–792 (1991).

23. Crawe A., *et al.* Simulation studies contracting skeletal muscles during mechanical stretch, *J. Biomech.* **13,** 333–340 (1980).

24. Demuth, H., and Beale, M. Neural Network Toolbox User's Guide" (for use with MATLAB), The Math Works Ins., Massachusetts, 1993.
25. Freeman, J. A., and Skapura, D. M. "Neural Networks" Addison–Wesley, Reading, MA, 1992.
26. Nguyen, D., and Widrow, B. Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights. *Int. Joint Conf. Neural Networks* **3,** 21–26 (1990).