Abstract—Using computer simulation, the theoretical feasibility of functional electrical stimulation (FES) assisted standing up is demonstrated using a closed-loop self-adaptive fuzzy logic controller based on reinforcement machine learning (FLC-RL). The control goal was to minimize upper limb forces and the terminal velocity of the knee joint. The reinforcement learning (RL) technique was extended to multicontroller problems in continuous state and action spaces. The validated algorithms were used to synthesize FES controllers for the knee and hip joints in simulated paraplegic standing up. The FLC-RL controller was able to achieve the maneuver with only 22% of the upper limb force required to stand-up without FES and to simultaneously reduce the terminal velocity of the knee joint close to zero. The FLC-RL controller demonstrated, as expected, the closed-loop fuzzy logic controller and on-line self-adaptation capability of the RL was able to accommodate for simulated disturbances due to voluntary arm forces, FES induced muscle fatigue and anthropometric differences between individuals. A method of incorporating a priori heuristic rule based knowledge is described that could reduce the number of the learning trials required to establish a usable control strategy. We also discuss how such heuristics may also be incorporated into the initial FLC-RL controller to ensure safe operation from the onset.

Index Terms—Adaptive fuzzy logic controller, computer simulation, FES, paraplegia, reinforcement learning, standing up.

I. INTRODUCTION

Standing up is a prerequisite for walking, reaching objects, transfers and face to face interaction with others. The regular use of functional electrical stimulation (FES) for standing may also provide important therapeutic benefits [1]–[5]. There is a requirement for improved FES control that requires a minimum of tuning or set up and can reduce upper limb effort and terminal knee velocities. In this paper we present preliminary results of a computer simulation to demonstrate the feasibility of robust and self-adaptive control using reinforcement learning (RL). RL has its origins in animal learning and is based on the commonsense notion that if an action is followed by a satisfactory outcome, then the tendency to produce that action should be strengthened, i.e., reinforced.

Kralj and Bajd [2] proposed a procedure for paraplegic FES standing up in which constant intensity stimulation produces knee extension moment that differ significantly from normal. Normally, high knee moment occur at seat lift-off that are progressively reduced resulting in knee angular velocities close to zero at full extension [6], [7]. The technique has been widely adopted and works adequately for many people, however, it is a compromise that requires manual adjustment to minimize the terminal knee velocities and the upper limb forces. The latter have been estimated to be as high as two-thirds of the body weight [8], [9]. The long term effects of repetitive high transient loading of the arm during activities such as transfers and wheeling have been implicated in the high incidence of overuse syndromes in the spinal injured population [10], [11]. Therefore, it is reasonable to develop safer FES systems that would also minimize upper limb pain when standing up for those with overuse syndromes. Reducing the upper limb force requirements may also extend the techniques to the frailer patient or those with impaired upper limb function. Higher than normal terminal knee velocities may also be cause for concern since it is not yet known if repetitive insults to the ligaments that abruptly limit knee hyperextension may eventually be damaging.

Khang and Zajac [12], [13] used a linearized model of paraplegic standing, formulated about vertical set point, to design an output feedback control law for unassisted standing in which the upper body forces were considered to be external disturbances. The model was able to maintain standing and recover when disturbed or from an initial position halfway between sitting and standing, however, the sit-to-stand maneuver was not described. The optimization technique used to determine the control law was computationally demanding procedure and not suitable for practical application. Donaldson and Yu [14] proposed a controller for minimizing the handgrip reaction forces by referring them to the equivalent leg joint moments. Static equilibrium equations must be solved for the equivalent leg joint moments and inverse models of the muscle recruitment curves are required to find the appropriate muscle stimulation levels. Ewins et al. [15] used PID type closed loop knee controllers for FES standing up, smoother trajectories were observed, however, the terminal knee velocities and upper limb loading were not reported. Quintern [16] proposed a combination of a closed-loop PID and an open-loop feed-
forward controller for paraplegic standing. The feedforward control signal was either a simple ramp-up or was calculated using a model of standing. They suggest that more attention must be given to man–machine synchronization to improve the quality of FES movements. The necessity of coordination between the artificial controller and the intact natural motor control has also been emphasized in a work by Hunt et al. [17] and Munih et al. [18] where they used linear quadratic Gaussian (LQG) controller to restore unassisted standing in paraplegia. Andrews [19] and Mulder et al. [20], [21] used closed-loop ON/OFF control based on a predetermined phase plane switching curve of the desired knee velocity versus knee angle. Andrews used a bell shaped switching curve to turn ON or OFF stimulation by comparing the actual knee velocity and angle with the template. In comparison to the basic maximal stimulation of the knee extensors, considerable improvements were reported in decreasing the terminal velocity of the knee joint and the amount of stimulation but the sit-stand maneuver took longer and more arm force was required.

A fuzzy logic controller (FLC) for FES standing up has been demonstrated, in computer simulation, by the authors to have improved performance, in terms of the trajectory smoothness, knee end velocity and the required arm forces, compared with either PID control or ON/OFF phase plane switching techniques [22]. However, the amount of manual tuning required to optimize the fuzzy controller precluded its practical application. We have also determined the feasibility of optimizing the parameters of the fuzzy logic controllers using a genetic algorithm (GA) optimization method. However, the large number of trials before convergence to the optimal solution (>600) and computational overhead of the present GA algorithms make them unsuitable for on-line tuning. This experience led us to consider a self-adaptive scheme that uses RL to tune the parameters of the fuzzy logic controllers, i.e., the FLC-RL.

In order to conduct preliminary investigations into the feasibility of FLC-RL, we required a test environment that could be precisely repeated, particularly as this adaptive technique learns over several trials. We therefore chose to conduct our investigations using computer simulation on a biomechanical model. Using this model we have explored the potential for FLC-RL control in FES, in particular, arm assisted FES standing up after spinal injury. In this paper we present our preliminary results related to the following questions. Can RL be extended to problems with more than one controller and continuous states and action spaces in particular the application to FES control of standing up? Can RL manage the coordination of the artificial FES controllers with the voluntary upper body forces to achieve a common objective? Can RL recover from transient disturbances typical of those encountered in practical FES standing up? Can RL accommodate system changes due to muscle fatigue or anthropometric differences between patients? Can RL provide safe control and self-adapt quickly enough, particularly in the start up phase, for FES applications? Can RL simultaneously minimize upper limb loading and the terminal velocity of the knee joints?

II. METHODS

A. Computer Model of Paraplegic Standing Up

The sagittal plane model incorporates body segments, muscles, passive joint properties, and voluntary use of the upper limbs. The shank, thigh, trunk-head, upper arm, and forearm are assumed to be connected by frictionless pin joints in a closed chain configuration. Segmental centers of mass are assumed to lie on the line connecting the two adjacent joints and the physical parameters were scaled to body mass and height according to Winter [23]. The resultant knee and hip joint moments in response to stimulation intensity were modeled by first-order transfer functions that include: saturation, neural transmission delays and a model of FES induced muscle fatigue described in [24]. Stimulation intensity was varied between zero and maximum (constant current with pulse width varied between 0–500 µs) producing joint moment according to the sign convention of Fig. 1. For example, if the knee joint controller output is positive stimulation is applied to the knee extensors and when negative to the knee flexors. The moment transmitted by the passive leg joint structures has both elastic and damping terms as described in [25], [26].

The voluntary upper body force actions were divided into active and passive parts. The active part consists of the activities of the arms musculature and is represented by equivalent forces at the shoulder joint (Fig. 1). The passive part was achieved by leaving the arms in the model to provide the kinematic constraints and passive resistance. This was important to prevent the model from assuming impossible configurations. Linear springs and dampers were used to limit the range of movements of the shoulder and elbow joints to physiological limits. Both springs and dampers become effective when the joint angles reach the offset angles before the joint limits (Table I). The horizontal and vertical components of the equivalent force at the shoulder joint were generated.
Fig. 2. The rule base of the fuzzy logic controllers modeling the voluntary arm forces $F_X(A)$ and $F_Y(B)$. $X$, $Y$, $V_X$, and $V_Y$ are the position and velocity components of the shoulder joint in the coordinate system shown in Fig. 1. $X_C = -0.1 \text{ m}$ (center of the foot support area) and $V_{SET} = 0.35 \text{ m/s}$ are used as setpoints for the horizontal position and the vertical velocity of the shoulder joint. The outputs of the controllers are the normalized $F_X$ and $F_Y$ that are then linearly scaled to their maximum values. The maximum values of $F_X$ and $F_Y$ are set at 150 N and total body weight, respectively. $N$—Negative, $P$—Positive, $Z$—Zero, $S$—Small, $M$—Medium, $L$—Large, and $V$—Very.

by fuzzy logic algorithms with the rules defined heuristically based on the assumption that these forces primarily provide balance and help in lifting the body. $F_X$ moves the shoulder joint toward the center of the foot support area and is a function of the horizontal position and velocity of the shoulder joint. $F_Y$ maintains a minimum upward speed and prevents downward movement of the shoulder joint and is a function of the vertical position and velocity of the shoulder joint. Unlike $F_X$ and $F_Y$ that could be related to clear objectives and the control rules could be defined heuristically, it was more difficult to do the same with $F_S$ and therefore it was not modeled. Five triangular membership functions were used for each input variable and seven for each output variable. The membership functions were all distributed evenly with 50% overlap over the domain of the variables. The set of rules for the fuzzy controllers of $F_X$ and $F_Y$ are given in Fig. 2.

The equations of motion were derived by applying D’Alembert–Lagrange principle [27]–[29]. For the planar model in Fig. 1 to be in general equilibrium the virtual power must vanish, i.e.

$$
\sum_{i=1}^{M} \left[ \left( F_{x_i} - m_i g y_i \right) \dot{x}_i + \left( F_{y_i} - m_i g x_i \right) \dot{y}_i + \left( r_i - J_{ii} \dot{\theta}_i \right) \dot{\theta}_i \right] = 0
$$

(1)

where $M = 5$ is the number of segments, $m_i$ is the mass of the segment $i$, $x_i$, and $y_i$ are the coordinates of the center of mass of the segment $i$ in an inertial reference frame, $\theta_i$ is the generalized coordinate for the segment $i$, $F_{x_i}$, $F_{y_i}$, and $r_i$ are the external forces and moments applied to the center of mass of the segment $i$ and $J_{ii}$ is the moment of inertia about an axis passing the center of mass of the segment $i$ perpendicular to the sagittal plane.

Application of (1) results in three equations of motion. For the closed chain system there are two additional constraint equations

$$
\sum_{i=1}^{M} l_i \cos(\theta_i) = v \sum_{i=1}^{M} l_i \sin(\theta_i) = h
$$

(2)

where $l_i$ is the length of the segment $i$ and $v$ and $h$ are the vertical and horizontal distances between the wrist and the ankle joints, respectively. The equations of motion and the constraint equations must be solved for the angular accelerations, velocities and positions of the joints. In this study, the shank was fixed to simulate the effect of an ankle foot orthosis of the floor reaction type [30]. The model parameters for the simulation experiments are given in Table I.

### B. Development of the Learning Algorithms for the FLC-RL

The learning algorithms are combination of a procedure introduced by Sutton [31], known as the temporal difference (TD) procedure and the reinforcement learning (RL) procedure [32]. The combined algorithms can address the goal directed sequential decision making problems, traditionally solved by dynamic programming [33], [34] but do not require the model of the environment. The formal convergence proofs have only been obtained for the finite, stationary markovian decision process [31], [42], [43]. Although most physical processes cannot strictly meet the formal conditions for applying these techniques, many researchers have been successful in applying them [35]–[40].

Due to the changes in the body such as muscle fatigue, the problem posed here is not stationary. Further, the control actions and states are considered as continuous variables, i.e., they can assume infinite number of values. It is our main objective however, to evaluate the performance of the RL and TD in the presence of these violations of the formal conditions.

Fig. 3 illustrates the structure of the learning system. Two FLC’s represent the knee and hip joint stimulation controllers.
Fig. 3. Structure of the learning system. FLC’s are used to represent the knee and hip joint stimulation pulsewidth controllers and the value function. Parameter update unit uses the TD error and the structural information of the FLC’s to adjust their parameters. Random search unit (RSU) provides the exploration in the action selection. The detailed description of the learning system is given in the text.

and one FLC represents the value function. We call them knee FLC, hip FLC, and value FLC. The value function \( V(X) \) estimates the value of state \( X \) and is defined as the sum of the future rewards when starting from state \( X \) and following a fixed control policy to the end of the trial. The structure of the FLC function approximator is shown in Fig. 4. All the FLC’s receive the partial state information including the knee and hip joint angular positions (more on this later in discussions)

\[
X = [X_1, X_2] = [\text{knee angle, hip angle}], \quad (3)
\]

The outputs of the knee and hip FLC’s are the stimulation pulsewidth of the respective joints \( S_k \) and \( S_h \). The value FLC outputs the estimate of the value function \( V \). Gaussian membership functions are used to encode the input variables. Central values of the membership functions were chosen for higher concentration in the sensitive regions such as the terminal phase of standing up and the width values were chosen for approximately 50% overlap between the adjacent membership functions as depicted in Fig. 5. The firing intensities of the fuzzy rules are calculated by applying the fuzzy multiplication “AND” operator as follows:

\[
\varphi_l = \text{AND}(\mu_1, \mu_2) = \mu_1 \cdot \mu_2 \quad (4)
\]

where \( l = 1, 2, \ldots, N \) is the rule number, \( \varphi_l \) is the firing rate of the rule \( l \) and \( \mu_1 \) and \( \mu_2 \) are the membership values of the rule’s antecedents. The output of the FLC’s (value FLC for example) then can be computed as

\[
V = \sum_{l=1}^{N} \varphi_l v_l \quad (5)
\]

where \( v \) is the parameter vector of the value FLC. Similarly the parameter vectors for the knee and hip FLC’s are \( k \) and \( h \), respectively.

The variability and exploration in the action selection policies (knee and hip joint controllers) are provided by the random search unit (RSU in Fig. 3) that adds random components to \( S_k \) and \( S_h \). RSU generates random actions \( S'_k \) and \( S'_h \) with the mean values of \( S_k \) and \( S_h \) and a standard deviation that depends on the value estimate \( V \) as follows:

\[
\sigma = \sigma_{max} \cdot \left( 1 - \frac{1}{1 + e^{-5V}} \right). \quad (6)
\]

Here, \( \sigma \) is the standard deviation of the exploration and \( \sigma_{max} \) determines its maximum value. The standard deviation is higher when the value estimate is low therefore providing higher variability in actions to search for higher rewards. A higher value estimate is a sign that the optimal policy is approaching when the variability must be lowered to facilitate the convergence.

The objective of learning (parameter update box in Fig. 3) is to adjust, in each time step, the parameter vectors of the knee, hip, and value FLC’s. In case of the value function for example, a general parameter estimation rule [41] can be used to correct its value

\[
v_{t+1} = v_t + \beta e_{t+1} \varphi_t \quad (7)
\]

where \( e \) is the deviation of \( V \) from its true value, \( \beta \) is the learning rate, and \( v \) and \( \varphi \) are the parameter and feature vectors from (5), respectively. The parameter update rule for \( v \) can be obtained by replacing \( e \) with TD error, \( \eta_{t+1} = \gamma V(X_{t+1}) - V(X_t) \).
with eligibility trace $e_t$, and $\varphi$ with eligibility trace $e_{t+1}^\varphi$.

$$v_{t+1}^\varphi = v_t^\varphi + \beta_\varphi (\gamma v_{t+1} - v_{t+1}^\varphi). \quad (8)$$

Here, $\beta_\varphi$ is the learning rate, $0 \leq \gamma \leq 1$ is the discount factor, and $\varphi$ is the reward function that gives the immediate reward for taking an action $\alpha$ in state $X$. $e_{t+1}^\varphi$ is the eligibility trace for parameter $\varphi$ defined using the following recursive equation:

$$e_{t+1}^\varphi = \lambda_\varphi e_t^\varphi + \varphi_t \quad 0 < \lambda_\varphi < 1 \quad (9)$$

where $\lambda_\varphi$ is the exponential decaying factor. Equation (9) results in an exponentially decaying trace that attributes more importance to the recently activated nodes. The term “trace” refers to the ability of (9) to memorize a sequence of eligibilities (gradients here) from the past time steps. According to (9), the gradient $\varphi_t$ of an activated node $i$ at time $t$ will leave a trace in the eligibility trace of that node in the future time steps that will exponentially fade away. Although different patterns can be used, the exponential pattern is more popular because it can be implemented recursively. By memorizing the current and past gradients or activated nodes, eligibility trace makes it possible to blame the current parameters for the current and past actions. This property is useful in time delay systems especially in FES systems with considerable neural transmission delay.

The parameters of the knee and hip FLC’s are updated using a RL procedure. The RL algorithms for the knee and hip FLC’s are similar, for example, the parameter update rule for the knee FLC is

$$k_{t+1}^k = k_t^k + \beta_k (\gamma v_{t+1} - v_{t+1}^k) e_{t+1}^k \quad (10)$$

where $\beta_k$ is the learning rate and $e_{t+1}^k$ is the eligibility trace of the parameter $k_t$ that is calculated using a recursive equation slightly different from (9)

$$e_{t+1}^k = \lambda_k e_t^k + \frac{(S_t^k - S_k)}{\sigma} \varphi_t \quad 0 < \lambda_k < 1. \quad (11)$$

Here, $\lambda_k$ is the exponential decaying factor. $S_k$ and $S_t^k$ are the input and output of the RSU in Fig. 3 and $\sigma$ is the standard deviation of the exploration unit. Therefore, $(S_t^k - S_k) / \sigma$ is the normalized perturbation added to the control actions, which along with the TD error determines how the parameters should be changed. For example if the positive perturbation results in higher state value (positive TD error) then it is desirable and the controller should be changed in the direction of the perturbation. On the other hand, if the resulting state value is lower, then the controller should be adjusted in the direction opposite to that of the perturbation.

C. Validating the Learning Algorithms

Formal mathematical proofs of convergence are presently only available for simple forms of RL [31], [42], [43]. The RL problem posed here is considerably different from the original RL problems, mainly because the states and actions are continuous and there are two controllers instead of one. The performance of the learning algorithms was evaluated by applying them to systems of increasing complexity. The first case was a system comprising a two-dimensional (2-D) double-integrator given by

$$x(t) = x(t-1) + \alpha_x(t)T + \frac{1}{2} a_x(t)T^2$$

$$v_x(t) = v_x(t-1) + \alpha_x(t)T$$

$$y(t) = y(t-1) + \alpha_y(t)T + \frac{1}{2} a_y(t)T^2$$

$$v_y(t) = v_y(t-1) + \alpha_y(t)T$$

where $T = 0.025$ s is the sampling interval, $x$ and $y$ represent the position of a unit mass in a 2-D state space and $v$ and $\alpha$ represent the velocity and acceleration, respectively, as indicated in Fig. 6(a). Maximum velocities on each axis are limited to 2 m/s. If these limits are attained, further acceleration will maintain the original velocities. Two FLC’s are used to control $a_x$ and $a_y$ to move the unit mass from the initial state $I$ to the goal state $G$. A third FLC represents the value function. The inputs to the FLC’s are the $x$ and $y$ coordinates of the unit mass. Each trial starts at state $I$ and ends whenever the mass touches the right or down limits ($r \rightarrow -1$), times out in 5 s ($r \rightarrow -1$) or reaches the 0.1 m neighborhood of the goal state where the reward depends on the time $t$ spent on transit

$$r = 1 - 2t/t_{\text{max}}. \quad (13)$$

Here, $t_{\text{max}}$ is the maximum time allowed for each trial equal to 5 s. The reward is zero on all the other time steps. The mass cannot penetrate the left or up limits. The objective of learning is to reach the goal state as quickly as possible. The convergence was assumed if the total time in the last ten consecutive successful trials had not varied by more than 0.1 s. Successful trials were those that ended in the goal state. By trial and error, a set of learning parameters ($\alpha_{\text{Lmax}} = 0.1, \gamma = 0.9, \lambda_\varphi = \lambda_x = \lambda_y = 0.95, \beta_\varphi = 0.1$, and $\beta_x = \beta_y = 0.001$) was found that resulted in convergence of the learning algorithms. These values were used in all the learning simulations on the double integrator problem.

The average number of trials for convergence in ten consecutive simulations was 154 (ranging from 75 to 224). For a learning experiment that took 95 trials to converge, various trajectories corresponding to the different stages of learning and the learning rate are shown in Fig. 6(a) and (b), respectively. To evaluate the effect on learning efficiency of typical complexities seen in FES systems the above simple double-integrator system was modified to include coupling between the two degrees of freedom and gravitational terms. Before applying to the unit mass, the outputs of the two controllers ($a_x$ and $a_y$) were modified as follows:

$$a_x = a_x + C_x a_y + G_x$$

$$a_y = a_y + C_y a_x + G_y$$

where $G_x$ and $G_y$ were the $x$ and $y$ components of the gravitational force. $C_x$ and $C_y$ were the coupling indexes representing how the acceleration in one axis affects the acceleration on the other axis. Positive $C_x$ and negative $C_y$ tends to move the unit mass in Fig. 6(a) toward $I$ and away from the goal state $G$. This is similar to the effect of the gravity in moving the subject toward the sitting position and away
from the goal state which is the standing position. Coupling indices represent the effects that each joint’s motion has on the others and depending on the configuration could be negative or positive. Although the gravitational forces and the coupling indices in standing up have more complex forms, only constant values of $C$ and $G$ are simulated here for simplicity. When only coupling was implemented, the performance depended on the sign of the coupling indices. For example, in ten consecutive simulations, with $C_x = C_y = 0.1$, the average number of trials required to reach convergence increased to 178 (range from 60 to 493). Whereas with $C_x = C_y = -0.1$ the average number of trials required for convergence decreased to 93 (range from 37 to 150). The addition of the gravitational term always degraded the learning rate because it tends to move the mass away from the goal state. For example, setting the gravity terms to $G_x = -G_y = 2 \text{ m/s}^2$ and $G_x = -G_y = 3 \text{ m/s}^2$ on average required 189 and 332 trials for convergence, respectively.

D. Computer Implementation

The model of the voluntary arm forces was developed using MATLAB’s Fuzzy Logic Toolbox (The MathWorks Inc., USA) and then exported to a C program. The equations of motion, the three FLC’s of the learning system and the RL algorithms were all programmed in C. These programs along with the C code of the voluntary arm forces were implemented in the LabWindows/CVI environment (National Instrument, USA). The latter provides the investigator with a useful “virtual Instrument Panel” for changing the simulation parameters. Fourth-order variable step Runge–Kutta numerical integration method was used to integrate the equations of motion. A 3-D animation facility was also developed to help in visualizing the simulated motions.

III. SIMULATION RESULTS

A. Applying the FLC-RL to the Control of Standing Up

In addition to complications such as couplings and gravitational effects, the dynamics of arm-assisted standing up in paraplegia is highly nonlinear and there are neural transmission delays associated with the muscle actuators. Furthermore, control actions that may compromise patient safety are unacceptable and a large number of learning trials to reach a solution would be impractical. In arm-assisted standing up, the arms are used for balance and body weight support regardless of the FES control strategy. The role of the RL could be to learn control strategies that would improve the quality of standing up. For example, by reducing the required arm forces or reducing the terminal velocity of the knee joint. The following simulation experiments explore such possibilities. The learning parameters for the following simulations were $\gamma = 0.15$, $\lambda = 0.9$, $\lambda_k = 0.95$, $\beta_k = 0.1$ and $\beta_h = 0.0025$.

B. Learning to Compensate for Weak Arm Forces

The maximum value of the vertical arm force $F_Y$ was usually assumed to equal the subject’s weight. However, in this simulation, this maximum was reduced to 76% of body weight so that the subject was unable to stand-up as depicted in Fig. 7 by the thin lines. Starting with this setting the goal of the FLC-RL was to find FES controllers for the knee and hip joints that were successful in standing up. Standing up was considered as successful when the absolute values of the knee and hip joint angles were less that 5° and the absolute value of the knee and hip joint velocities were less that 5°/s. The reward was always zero except for the end of trial (failure, timeout, or reaching to standing position) where it was defined as $\rho = 1 - 2d/d_{\text{max}}$. Here, $d$ is the Euclidean distance between the end state and the standing position and $d_{\text{max}}$ is the highest possible value of $d$ in the state space. By this definition, the control actions that moved the subject closer to the standing position were rewarded, the closer the end state was to the standing position, the higher the reward. Convergence was assumed when the last ten trials all successfully ended in the standing position. Convergence was achieved in 29 trials and Fig. 7 shows the performance of the FES controllers at different stages of learning.

C. Learning to Minimize Arm Forces

At the start of the learning phase, the model subject was forced to use only her arms to stand-up because the stimulus
intensity outputs of the FES controllers were set to zero (the weights of the FLC’s were initially set to zero). The control actions leading to failure, i.e., not being able to assume standing position within 4 s were punished by setting $r = -1$. During standing up, the reward for all other time steps was zero except for successful trials where the reward was inversely related to the integral of the arm forces. Therefore, to maximize its reward, the FLC-RL must learn control strategies for the knee and hip joints that reduce the arm forces. The integral of the arm forces was used as a convenient measure representing, approximately, the energy expended by the musculature of the upper body. Convergence was assumed if the last ten trials were all successful and the knee end velocity in these trials did not change by more than 1 $\text{cm/s}$. Unlike, the previous learning experiments, the convergence here was more sensitive to the learning parameters. Even for the fixed set of the learning parameters, not all the simulations converged to a solution. Fig. 9 shows one of the successful simulations that converged in 238 trials.

**E. Effects of System Changes and Disturbances**

The optimal fuzzy logic controllers that minimized the arm forces were robust to sudden changes (duration 200 ms in the

---

**Fig. 7.** The performance of the RL when compensating for the inadequate arm forces in standing up. The results are shown for trial 0 (thin), trial 10 (medium) and after convergence in trial 29 (thick). Continuous and dashed lines represent the values corresponding to the hip and knee joints, respectively. The stimulation to the knee and hip joint extensors are increased just enough to compensate for the weak arm forces. Note that the stimulation pulsewidth applied to the joints is always positive. The sign of the stimulation in the figures determines whether it is applied to the extensor or flexor muscles of the joint as explained in the text.

**Fig. 8.** The performance of the RL when minimizing the arm forces. The results are shown for trial 0 (thin), 100 (medium), and after convergence in trial 276 (thick). Continuous and dashed lines represent the values corresponding to the hip and knee joints, respectively. Since the graphs are drawn from the seat-off moment, the vertical arm force ($F_v$) is high at the beginning. The value of $F_v$ increases after achieving the standing position, which is due to the modeling assumptions requiring the vertical arm force to maintain a minimum vertical velocity.

**D. Learning to Minimize the Terminal Velocity of the Knee and Arm Forces Simultaneously**

Reducing both arm forces and the knee end velocity was achieved as follows. Instead of setting the initial controller weights to zero, we set them to produce the maximum stimulus intensity everywhere in the state space (this can be easily achieved by setting all the weights to a constant value) causing the knee end velocity to be high and the arm forces to be low. To reduce the knee end velocity, the control actions leading to successful trials were rewarded inversely with the magnitude of the knee end velocity. The reward was $-1$ for failure or timeout and zero at all other time steps. Convergence was assumed if the last ten trials were successful and the knee end velocity in these trials did not change by more than $1\text{cm/s}$.

---

The learning converged after 276 trials and the FES control strategies at different stages of learning are shown in Fig. 8. The FLC-RL controller was able to reduce the total arm force by 78% compared to the case of arm only standing up.
middle of the sit-stand maneuver) of more than ±100% in the arm forces or the knee joint strength due to the spasticity. Like any other adaptive system, RL cannot adapt to these kinds of sudden disturbances with no regular pattern or association to the state of the system.

The RL can adapt to the sustained disturbances with sudden onset. For example, when an optimized controller is trained on a model subject then switched to another subject with significantly different physical parameters. The adaptation of the RL to the permanent changes in the system parameters was typically much faster than the initial learning. In the experiment to compensate for the weak arm forces, it took 29 learning trials to tune the controllers so that they could reliably perform successful standing up maneuvers. At this point, we increased the subject’s mass by 10% whilst learning was in progress. As a result the controller failed to perform standing up successfully, however, it took only seven additional trials to recover and perform satisfactorily thereafter. Similarly when we increased the subject’s mass by 14% it took 11 additional learning trials to recover. As another example, the learning to minimize the arm forces was converged in 276 trials. At this point we increased the subjects’ mass by 5% whilst learning was in progress. The learning was converged in only 11 additional trials. Increase of another 5% to the mass at this point, required another 13 trials to converge.

IV. DISCUSSIONS

A. Validating the RL Algorithms

At the time of writing the authors are unaware of any formal proofs of convergence for a problem with more than one controller and continuous state and action spaces. However, our RL algorithm converged in all simulations involving the double integrator test system even when coupling and gravitational factors similar to those found in the dynamics of FES standing up were included. However, the learning rate became slower with the increased complexity.

B. Applying RL to the Control of Standing Up

In our modeling, we have assumed that the use of the arms would avoid falling and that the RL would intervene to improve some measures of the reinforcement signal. In general for RL problems, failure constitutes an important part of the learning process since it is always possible to find the goal state by visiting all the failure states and avoiding them. However, we chose to constrain the system to operate within a safe region. The trained controllers will work as long as the state of the system remains within or close to that region of the state space. In our simulations there were no defined control actions for states outside the region. In practice a simple but safe control scheme could be used to return the system back to the operating region. For example, a simple rule based scheme could be hand crafted based on the common sense and incorporated in the knee and hip FLC’s. Such a control system would have two sets of rules, one designed by the RL for the normal operating region and the other designed heuristically for the remainder of the state space that might be visited under abnormal conditions such as a fault or an unusually high disturbance.

It was shown that the RL was able to compensate for weak arm forces by identifying the system very quickly and raising the stimulation intensity just high enough to allow the maneuver to be successfully completed. In the experiment to minimize arm forces, the FLC-RL learned to increase the stimulus intensity to the hip and knee extensor muscles in order to reduce the arm forces, resulting in shorter and faster maneuvers (Fig. 8). Since, the graphs are drawn after the seat-off moment, the arm force is high at the beginning of the maneuver. RL minimizes the integral of the arm force from the seat-off to the standing position. The minimization is achieved by reducing the average arm force during the standing up sequence and reducing the total time of the maneuver. The arm force increases again once the standing position is reached. This increase is because of the assumptions made in modeling the arm forces where we assumed that the vertical arm force should maintain a minimum upward velocity of 0.35 m/sec for the shoulder joint. This high arm force after achieving the standing position that will account for the lower arm forces and weight bearing by the legs. Because of such limitations in the model of the arm force, used in this study, and the differences...
in the voluntary control strategies among paraplegic subjects, we are currently developing the neural network models of the arm forces based on the data measured from the paraplegic subjects. These models will help us to individualize the general model used in this study and minimize the arm forces not only during standing up but also during standing. The final control solution shown in Fig. 8 is very similar to the open-loop maximal stimulation of the knee and hip extensor muscles. RL was able to reason these simple goal-seeking tasks out without any information about the system using only that obtained from its own exploratory interactions and a scalar reinforcement signal. Although these are simple examples they serve to illustrate the potential of the FLC-RL to deal with far less obvious situations.

In the experiment to minimize the knee terminal velocity, RL found that it could reduce the terminal knee velocity by not only modifying stimulation to the knee joint but also the stimulation to the hip joint, taking advantage of the dynamic coupling between the joints. Stimulation has shifted from the extensors to the flexors at just the right time to reduce the knee terminal velocity. This is a delicate task since the dynamics of the final stages of the standing up motion is very sensitive to joint moments. Furthermore, because of the kinematic constraints, minimization of the knee terminal velocity also resulted in minimization of the hip terminal velocity. As indicated in Fig. 9, the arm forces remained almost unchanged in a maneuver with minimal arm forces and minimal terminal velocities.

C. Heuristically Chosen Parameters and Dimension of the State Space

The choice of the learning parameters affects both the rate of convergence and the quality of the final solution. Some values could result in complete failure to learn. By trial and error, we found a set of learning parameters that resulted in convergence. However, we did not try to find the optimal parameters so further adjustment may provide faster convergence.

In practice, we envisage that training sessions will be needed when fitting the neural prosthesis. It will be important to limit the number of learning trials required to converge to an initial useful control strategy. One possibility lies in the proper choice of the learning system structure including the learning rates of the RL algorithms, the number of the membership functions of the FLC’s and the reinforcement signal. For example, at higher learning rates the learning process may become unstable. The higher the number of the membership functions the higher the number of the tuning parameters and therefore slower the learning. On the other hand, fewer membership functions could compromise the resolution and result in poor control. Therefore, the skill to properly choose the structure of the learning system is very important for speeding-up the learning and usually comes with practice.

As the dimension of the state space increases the number of learning trials required to form the value and action functions and the computational cost increase exponentially. In the experiment to reduce the terminal knee velocity it seemed reasonable to include joint velocities as inputs to the value and action functions. This modification increased the dimension of the state space from two to four which in turn increased the number of the tuning parameters by 266 times. As an example, if each dimension used ten membership functions then the number of the network weights for two, four, and six dimensions would be $10^2$, $10^4$, and $10^6$, respectively. Learning rate for the case with four inputs was very slow, e.g., the velocity was reduced by 20% in 600 trials. Therefore, we did not include the knee and hip joint velocities as the input to the value and action functions. With the reduced set of the input variables that included only the angular positions of the knee and hip joints, it was shown that the RL could successfully learn the control strategies. One explanation is that the RL builds internal representations of the excluded or hidden state variables [44]. On the other hand, one remedy to the “curse of dimensionality” in problems with many inputs is the use of multilayered neural networks that are able to handle many input variables [42]. Another is to partition the state space into tilings each including only some of the dimensions as explained in [44], [45] rather than use our simple grid partitioning scheme. With FLC’s however, it is straightforward to incorporate previous knowledge in the form of the rules and they have a better local representation than the neural networks.

D. Optimality of the Controllers

In the simple double integrator, even the trajectories in the earlier stages of the learning process were close to the optimal solution although many more trials were needed for convergence to the optimal solution [Fig. 6(a)]. In more complex situations the number of trials could be very high, however, in practice even suboptimal controllers could give satisfactory performance. The RL uses a method similar to the gradient following procedure that could be trapped in local optima. Although this suboptimal solution may be considered as practically sufficient, there are ways to improve the chance of finding the global optima in multimodal search spaces. The RSU in Fig. 3 is a mechanism that could avoid the local optima by selecting different exploratory actions. However, it is not a complete solution because the size of the exploratory actions must be limited to facilitate convergence and improve safety which, means it can help to escape only smaller hills in the multimodal search space. Another more general approach is to repeat the learning experiment with different starting parameters.

E. Suggestions for Further Work and Future Practical Implementation

The simulations demonstrated good recovery from transient disturbances such as those that may be expected due to spasticity or sudden actions of the upper body and good accommodation to slow changes in the system dynamics. These changes may be expected from FES induced muscle fatigue, slow changes in the subject’s physical parameters such as weight loss/gain or slow changes in the voluntary control strategy due to the patient becoming more skilled in the maneuver.
The capability to adapt to the changes in the voluntary control strategy may be one of the strongest aspects of the RL opening up the possibility for mutual learning. This may provide a better “cybernetic interface” in which both the subject and the RL controllers learn to cooperate to perform the maneuver better since neither the FLC-RL algorithm or the human require explicit a priori models for learning. Interaction and reinforcement is all that is needed for the mutual learning to proceed.

In more conventional closed-loop FES controllers specialized sensors are chosen that monitor specific state variables and must be accurately aligned with the anatomy and precisely positioned in specific locations, e.g., goniometers across lower limb joints. This may be inconvenient, particularly if the sensors are to be surgically implanted. RL however, has no such requirement and may use any available set of sensory signals that are sufficiently rich in information about the state of the system and the reinforcement signal. These could include sets of miniature artificial sensors located in convenient external places or implanted [46]–[48] or in combination with natural sources such as EMG or ENG using electrodes in the periphery or microelectrodes in the central nervous system. In such arrangements, the reinforcement signal, such as the knee velocity if the knee end velocity is to be minimized, may not be directly available. In such case it must be derived either intuitively using handcrafted rules or indirectly using supervised machine learning techniques as described in [47]. The FLC-RL may provide similar flexibility in terms of the stimulation sites. For example, as an alternative to stimulating highly differentiated peripheral nerves it may be desirable to stimulate the spinal cord or the spinal roots and establish control despite the more complex responses. For example, stimulating the lumbar anterior sacral roots produces multiple muscle contractions that affect multiple joints in more than one degree of freedom [49], [50]. The flexibility in the choice of inputs and the outputs comes from the fact that the RL process essentially learns associations between situations and actions. A further consequence of this feature may offer a measure of fault tolerance if there is a redundancy in the sensors and stimulus sites. Should a sensor or stimulus site suddenly or progressively fail to provide consistent signals or responses the FLC-RL may progressively learn to discount them from its control strategy.

To ensure patient safety during the initial training phase and subsequent use of the RL controller, we envisage an initial FLC-RL controller based on reliable handcrafted rules, for example, those commonly used in clinical practice [51], [52]. This “training wheels” possibility is suggested by the results of the simulation to minimize both the arm forces and the terminal velocity of the knee joint. Our intuitive understanding of the system was used to handcraft the initial controllers so that it was possible to minimize both criteria. The results of investigating the effects of system changes suggest an alternative in which an initial controller could be pretrained by applying the RL algorithm to a dynamical model approximately scaled to the individual patient. Of course, the latter would be feasible only if a model was available that also included the sensors. The latter may not be possible for natural sensors or when sensor alignment and position is uncertain. The subsequent learning process which, needs smaller number of trials to converge, could then continue to fine-tune the controller without compromising the patient safety.

V. Conclusions

The classic RL algorithms can be extended to the continuous state and action spaces using function approximation techniques. These algorithms were validated and performed well in continuous space, multicontroller problems and in the presence of the simulated complexities normally encountered in the FES control systems such as dynamic coupling. The RL was able to learn appropriate strategies to compensate for the weak arm forces and was able to simultaneously reduce arm force requirement and the terminal velocity of the knee joint. The FLC-RL was able to recover from simulated disturbances approximating those encountered in FES assisted standing up in paraplegia. It may be possible to include a priori heuristic rule based knowledge in the learning system structure, which may accelerate the initial learning rate and provide safety. Although the method appears to be promising only the theoretical feasibility has been demonstrated, further work is required to demonstrate clinical feasibility.

REFERENCES


Rahman Davoodi received the B.S. degree in mechanical engineering and the M.S. degree in biomechanical engineering both from Sharif University of Technology, Tehran, Iran, in 1988 and 1991, respectively. He is currently pursuing the Ph.D. degree in biomedical engineering at the University of Alberta, Edmonton, Alta., Canada. He has worked as a Lecturer in the University of Air Science and Technology, Iran, as a Research Engineer in the aerospace industry, and as a Practicing Engineer in the design of air conditioning systems.

He has used the finite element method in biomechanical problems and he is currently interested in control of FES systems, especially man-machine interactions, natural motor control and learning, neurofuzzy control, genetic algorithms, and reinforcement learning.

Brian J. Andrews received degrees in cybernetics, control systems, and bioengineering from the Universities of Reading, Sheffield, and Strathclyde, U.K.

His interests focus on the clinical application of control systems technology to assist individuals with spinal injury. He has held clinical and academic appointments at the University Hospital of Wales, Cardiff, U.K., the University of Strathclyde, Glasgow, Scotland, and the Case Western Reserve University, Cleveland, OH. He is presently Director of Rehabilitation Technology at the Glenrose Hospital, Edmonton, Alta., Canada, Professor of Biomedical Engineering at the University of Alberta, and Visiting Professor in the Department of Cybernetics at Reading University.