ABSTRACT
This paper presents the development and simulation of a fuzzy logic based learning mechanism for human motor learning. In particular, fuzzy inference was used to develop an internal model of a novel dynamic environment. The task being studied was upper limb reaching movements in the horizontal plane. A dynamic model of the human arm was developed and implemented in Simulink. A fuzzy If-Then rule structure was then created to relate trajectory movement and velocity errors to internal model update parameters. Finally, with the model and learning mechanism in place, a simulation of experiment was performed to compare the fuzzy systems performance to known results for human subjects. It is found that the dynamic model behaves as expected, and the fuzzy learning mechanism creates an internal model that is capable of opposing the environmental force field and regain a trajectory closely resembling the desired ideal.

KEY WORDS
Fuzzy Logic, Inference, Human Motor Learning, Upper Limb, Internal Model, Biomedical Engineering

1. Introduction
Recently there has been much focus on studying the learning mechanisms that humans use to control their movement. The human motor system is very adaptable and robust when it comes to experiencing new environments. One task that has been of particular interest is that of upper arm reaching movements. Researchers have used a robotic arm [Faye86] [MaBM06], or manipulandum to exert varying types of perturbing forces to a subject’s movement to try and understand the characteristics of the motor system. In their 1994 paper, [ShMu94], Shadmehr and Mussa-Ivaldi demonstrated that when adapting to a novel dynamic environment, the human central nervous system (CNS) generates an internal model of the environment. This model is then used to generate a feedforward predictive series of commands to actively compensate for the forces experienced. Since then, much other work has been performed to understand different features of this internal model. For example, [MosSh00], [ThSh00] studied how learning generalizes from one environment to another. Other researchers, [ScDM01] have attempted to determine equations that can relate a subject’s performance on a given movement trial to previously experienced errors and forces.

Fuzzy logic, initially developed by Zadeh, provides a tool that is capable of emulating human reasoning and is useful for performing control tasks in which there is incomplete or imprecise information [YeLa98]. Systems developed using a fuzzy architecture have proven to be very robust and intelligent and have the potential for out-powering human experts.

Since fuzzy logic is a mechanism that emulates human intuition and reasoning, the question arises as to whether the mechanisms in the real human brain behave in a fuzzy manner and, if so can fuzzy logic technology be used to model and/or emulate this behaviour. Kubica et al ([KuWW95]) studied the use of both conventional and fuzzy techniques for modeling balance and posture control of the human torso. With this type of application in mind, this project is intended to implement a fuzzy logic based learning system that can incrementally develop an internal model of reaching task dynamics for a simulated human arm. The goal will be to successfully implement such a system and to perform simulations to compare the fuzzy learning mechanism’s performance to that of real human test subjects.

This paper first discusses the design and implementation of a two degree of freedom planar model of the human upper limb and proceeds to introduce the fuzzy model creation and finally finishes with an application of the system to an experimental situation as reported in [ShMu94].
2. System Design

2.1 System Overview

Figure 1 presents a block diagram overview of the motor learning system. A high level planner, [Wish02], creates a desired trajectory for a given reaching task. This trajectory is passed to a lower level controller which implements a forward dynamic model of the upper arm to predict the joint torques necessary for performing the required movement and sends these commands to the simulated arm. If the dynamics environment is does applies a perturbing force to the arm’s hand, the actual trajectory will not match the desired performance and a feedback mechanism will correct for the errors and to move the hand to the desired target.

Based on the position and velocity errors present in the actual trajectory, a fuzzy learning mechanism creates an adjustment parameter for an internal model of the environment dynamics. On subsequent trials, this model is used as a predictive tool to determine the additional joint torques necessary to compensate for the environment dynamics.

2.2 Upper Limb Dynamic Model

In order to simulate the performance of the internal model formation, a model of the upper limb was created. This allowed the results of the simulations to be compared with actual performances of human subjects. In particular, a right arm as shown in Fig. 2, with the properties listed in Table 1, was simulated. The arm properties represent a typical human arm as discussed in [ShMu94].

![Simulated human arm.](image)

The values $\tau_1$ and $\tau_2$ in Fig. 2 represent the joint torques produced by the arm’s muscles about the shoulder and elbow respectively. For simplification, these torques are assumed to be supplied by ideal moments acting about the joint centres. The dynamic environment experienced
by the arm is represented by a force vector acting on the hand.

**Table 1.** Properties of the simulated human arm from [ShMu94]. The centre of mass (COM) position is defined as the distance from the proximal end of the arm segment.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forearm</td>
<td></td>
</tr>
<tr>
<td>Mass, (m_1)</td>
<td>1.93 kg</td>
</tr>
<tr>
<td>Centre of mass, (R_1)</td>
<td>0.165 m</td>
</tr>
<tr>
<td>Inertia, (I_{1z})</td>
<td>0.0141 kg.m(^2)</td>
</tr>
<tr>
<td>Length, (L_1)</td>
<td>0.33 m</td>
</tr>
<tr>
<td>Upperarm</td>
<td></td>
</tr>
<tr>
<td>Mass, (m_2)</td>
<td>1.52 kg</td>
</tr>
<tr>
<td>Centre of Mass, (R_2)</td>
<td>0.15</td>
</tr>
<tr>
<td>Inertia, (I_{2z})</td>
<td>0.0188 kg.m(^2)</td>
</tr>
<tr>
<td>Length, (L_2)</td>
<td>0.34 m</td>
</tr>
</tbody>
</table>

The model of the arm was created using Lagrangian dynamics [KhDo02]. For the multilink arm system, the Lagrangian, \(L\), is the difference between the kinetic energy \((KE)\) and the potential energy \((PE)\) of the arm segments, as shown in (1). Since this project is restricting motion to a horizontal plane, \(PE = 0\) for both arm segments. Therefore, the Lagrangian is the sum of the kinetic energies for the two limb segments.

\[
L = KE - PE
\]  

(1)

The total \(KE\) for the upperarm is shown in (2). The first term of this sum is the translational energy of the body’s COM, while the other term is the rotational energy about the COM. The segment mass (from Table 1) is \(m_1\) Table 1, while \(\sigma_1\) and \(V_1\) are respectively the angular and translational velocities of the body and \(I_1\) is its inertia tensor.

\[
KE_1 = \frac{1}{2} m_1 \|V_1\|^2 + \frac{1}{2} \sigma_1^T I_1 \sigma_1
\]  

(2)

Since there is only motion in the horizontal plane and rotation about the Z-axis, (2) simplifies to (3), where \(I_{1z}\) is the segment inertia value from Table 1. Similarly, the \(KE\) for the forearm is found to be (4).

\[
KE_1 = \frac{1}{2} m_1 \left( \dot{X}_1^2 + \dot{Y}_1^2 \right) + \frac{1}{2} I_{1z} \dot{\theta}_1^2
\]  

(3)

\[
KE_2 = \frac{1}{2} m_2 \left( \dot{X}_2^2 + \dot{Y}_2^2 \right) + \frac{1}{2} I_{2z} \left( \dot{\theta}_1 + \dot{\theta}_2 \right)^2
\]  

(4)

By making use of the COM kinematic equations in (5) and their derivatives, and by combining (3) and (4), one can build the equations of motion using (6). The generalized coordinates, [KhDo02], used for this formulation are the joint angles \(\theta_1\) and \(\theta_2\). The total moment, \(M_i\), acting about joint \(i\) is the sum of the torque from the muscles, \(\tau_i\), and the torque created by the environmental interaction force, \(\tau_{Fi}\), as shown in (7). This relationship makes use of the principle of virtual work [Tsai99], which states that the instantaneous torque created about the joints by the interaction force is described by (8), where \(J\) is the arm’s Jacobian matrix from (9).

\[
X_1 = R_1 c \theta_1
\]

(5)

\[
Y_1 = R_1 s \theta_1
\]

\[
X_2 = L_c \theta_1 + R_c \theta_2
\]

\[
Y_2 = L_s \theta_1 + R_s \theta_2
\]

\[
d \frac{\partial L}{\partial \dot{\theta}_1} - \frac{\partial L}{\partial \theta_1} = M_1
\]  

(6)

\[
M_i = \tau_i + \tau_{Fi}
\]  

(6)

\[
\tau_{Fi} = J^T F
\]  

(8)

\[
J = \begin{bmatrix}
- (L_s \theta_1 + L_s \theta_1) & -L_s \theta_1 \\
L_c \theta_1 + L_c \theta_2 & L_c \theta_2 
\end{bmatrix}
\]  

(9)

Substituting the model parameters and combining (7) and (8) and (9), the final arm model is defined in matrix form as:

\[
\begin{bmatrix}
0.2852 + 0.1505 c \theta_2 & 0.0872 + 0.1505 c \theta_2 \\
0.053 + 0.0752 c \theta_2 & 0.053
\end{bmatrix}
\begin{bmatrix}
\dot{\theta}_1 \\
\dot{\theta}_2
\end{bmatrix}
+ \begin{bmatrix}
0.1505 s \theta_2 & 0.0752 s \theta_2 \\
0.1505 s \theta_2 & 0
\end{bmatrix}
\begin{bmatrix}
\tau_1 \\
\tau_2
\end{bmatrix}
+ \begin{bmatrix}
-0.33 s \theta_1 + 0.34 s \theta_2 \\
-0.34 s \theta_2
\end{bmatrix}
\begin{bmatrix}
F_x \\
F_y
\end{bmatrix}
\]  

(10)

\[F = -BV\]  

(11)

\[B = \begin{bmatrix}
B_{11} & B_{12} \\
B_{21} & B_{22}
\end{bmatrix} N \text{sec/m}
\]  

(12)

As mentioned previously, in [ShMu94] Shadmehr and Mussa-Ivaldi showed that when learning to reach in a
novel environment (an unknown $B$ value), humans create an internal model of the task dynamics to compensate for the viscous field. Traditionally this force field is generated by a robotic arm or manipulandum [Faye86] [MaBM06]. In order to simplify the model in this project, the dynamics of the manipulandum were not considered. The environmental force field was assumed to be an ideal Cartesian force generator and was created by directly implementing the relationship in (11).

In this project, as in previous work such as [ShMu94] and [MaBM06], the velocity-based environmental force implies a velocity-based internal model. This model must generate a force to oppose that in (11), which is performed by generating the joint torques in (13).

$$\tau_M = J^T (-F) \quad \text{(13)}$$

Since the ideal trajectories are defined in terms of the joint trajectories the torque from must be described in the joint space. This is done by using the velocity relationship defined by the Jacobian equation in (14), and the results from (11) and (13). The final equation relating compensation torque to joint velocities is shown in (15).

$$V = J \dot{\theta} \quad \text{(14)}$$

$$\tau_M = J^T BJ \dot{\theta} \quad \text{(15)}$$

The $J^T BJ$ combination provides the ideal model for a given $B$ matrix. Using the Jacobian from (9) and the viscosity matrix from (12), the internal model, $IM$ is defined as shown in (16). Note that, for simplification of the fuzzy learning algorithm, the numerical values in (16) have been rounded to two significant figures.

$$IM = \begin{bmatrix} 0.16B_{11} & 0.08B_{11} + 0.11B_{12} \\ 0.08B_{11} + 0.11B_{21} & 0.43B_{11} + 0.06B_{12} + 0.6B_{21} + 0.07B_{22} \end{bmatrix} \quad \text{(16)}$$

Previous studies, [ShMu94] [ThSh00], have shown that the internal model of the environment dynamics is direction dependent. This means that for different points in the velocity state-space, the elements of IM from (16) will be different. Since this project is considering eight possible movement directions, as seen in Fig. 3, the simulation must form eight different IM matrices.

### 2.4 Simulation

The simulation of the system in Fig. 1 was implemented using the models from Sections 2.2 and 2.3 through the use of Simulink. The block diagrams for this system and its components are provided in Appendix A. The Matlab scripts and functions for running the simulation and executing the experiments discussed later are provided in Appendix B.

*Figure 3. Movement directions. During the simulation, each movement starts from a common central location and moves outward.*

For a given reaching task, a desired ideal Cartesian minimum jerk trajectory [ShMu94] is defined in terms of the required joint trajectories using the inverse kinematic model of the simulated arm. A series of experiment-specific target directions and viscosity values are then created. The simulated arm is then directed to make the required movements within the unknown viscous field from (11). The system then attempts to adapt to the field using the fuzzy learning system discussed in the following section.

Together with the known ideal arm dynamics, the internal model is used to predict the torques required to follow the desired trajectory to the target. However, there may be errors in this feedforward prediction. Therefore, a corrective feedback controller was added to the simulation, as shown in Fig. 1. This controller generates corrective torques based on errors between the actual and desired trajectories using an impedance control approach. This impedance controller is implemented in with respect to the joint trajectories using the joint stiffness, $K_J$ and joint viscosity, $B_J$ as discussed in [ShMu94] and reproduced in (17).

$$K_J = \begin{bmatrix} 15 & 6 \\ 6 & 16 \end{bmatrix} \text{ N sec/ rad} \quad \text{(17)}$$

$$B_J = \begin{bmatrix} 2.3 & 0.9 \\ 0.9 & 2.4 \end{bmatrix} \text{ N sec/ rad}$$
3. Fuzzy Learning Implementation

3.1 Overview

The purpose of the fuzzy logic-based learning mechanism in this project is to make adjustments to the internal model of the environment based on performance errors experienced during previous movements. The fuzzy system does not operate in a controller role during a given movement, but rather it operates in between reaching tasks to adjust the internal model to improve performance on the subsequent trials.

There are various performance metrics which could be used as inputs to this system such as Cartesian movement/velocity errors, joint movement/velocity errors, interaction forces experienced and joint torques experienced. For the current work, two inputs were selected based on the Cartesian performance. First, the movement error (ME) is defined as the maximum perpendicular displacement from the straight line joining the starting and ending points of a movement. With reference to Fig. 3, positive ME is defined as moving to the left of the desired direction of travel (counter-clockwise). As will be seen later, this direction dependent error definition has a significant impact on the designed fuzzy rule base. Second, the velocity error (VE) is defined as the error in peak parallel to the desired direction of motion. For example, if the desired path is in direction 1, the VE calculation will only consider the X component of the velocity vector. Once again, positive VE is defined along the direction of motion and is therefore also direction dependent.

3.2 Data Acquisition and Fuzzification

After the execution of each reaching movement trial, the resulting position and velocity trajectories (both Cartesian and angular) are exported to the Matlab workspace. The software, as seen in Appendix B, then calculates the ME and VE values by comparing the actual performance with the ideal minimum jerk trajectory. These values are then passed to the Fuzzy Inference System (FIS) corresponding to the trial’s target direction. The fuzzification stage is identical for all of the direction dependent FIS configurations. Each error value is fuzzified using the linguistic variables defined in Fig. 4. Gaussian membership functions were selected to allow for a smooth transition between regions. Also, it has been

**Figure 4.** Membership functions for fuzzification of input variables. Membership functions: LN – Large Negative, MN – Medium Negative, SN – Small Negative, Z – Zero, SP – Small Positive, MP – Medium Positive, LP – Large Positive. **A:** Movement error, **B:** Velocity error.
shown that other aspects of the learning process such as learning generalization have evidence of underlying Gaussian basis functions [DoSh02].

### 3.3 Rule-Based System

The next stage of the learning process is to pass the fuzzified performance metrics to the knowledge base in order to determine how to adjust the IM. The knowledge base consists of a series of fuzzy rules relating the input and output variables. The direction dependent nature of the error definitions has a significant impact on the nature of the rules for each direction. The following highlights the details of the rule matrix derivation for movement directions 1 and 5 and direction 2 and 6. Only the final result is shown for the other directions, as the derivation procedure is similar. The final rule matrices that form the knowledge base are presented in Appendix C.

The key to defining the fuzzy rules for the learning system is to find a relationship between the measured errors and the IM parameters. As was previously shown in (16), there is a direct relationship between the internal IM and the environment viscosity matrix terms. With this in mind, if we can define a relationship between the experienced interaction force and the error inputs, we will be able to construct a chain linking the errors and the IM adjustment parameter. For directions 1 and 5 (Fig. 3), there is a simple relationship between the Cartesian force components and the ME and VE values. Clearly, any force in the Y direction is going to result in a deviation from the straight line path and therefore will affect the value of ME. Similarly, only the X component of the force vector is going to alter the velocity along the direction of motion.

When ME > 0 for direction 1, one can intuitively see that the Y component of the interaction force is too high and therefore adjustments must be made to IM to compensate for this. Particularly, we want to adjust the model such that the Y force is decreased. Examination of (11) and (12) shows that only the elements in the second row of $B$ affect the Y component of the force and that:

$$ F_Y = B_{21}V_X + B_{22}V_Y $$

(18)

Ideally, for direction 1, $V_X > 0$ and $V_Y = 0$, if we assume these values and substitute them into the above equation, it is clear that a decrease in $F_Y$ is achieved by decreasing $B_{21}$. The opposite situation occurs when ME < 0. Using a similar approach, we can see that decreasing $B_{21}$ has the affect of lowering $F_Y$ and thus decreasing VE. The analysis results in the following viscosity adjustment matrix:

$$ \Delta B_1 = \begin{bmatrix} -C_1 \cdot VE & 0 \\ -C_2 \cdot ME & 0 \end{bmatrix} $$

(19)

The above equation states that the required adjustments to $B_{11}$ and $B_{21}$ are opposite in sign and proportional to their associated error values. Assuming that we can change the gain of the adjustment at a later time, we can set $C_1=C_2=1$. Substituting (19) into the expression for IM in (16), we arrive at the IM adjustment matrix for direction 1 expressed in terms of the input error values:

$$ \Delta IM_1 = \begin{bmatrix} 0.16VE & -0.08VE \\ -0.08VE -0.114ME & -0.43VE -0.06ME \end{bmatrix} $$

(20)

A similar analysis for direction 5 shows that the adjustment matrices for opposing directions are identical.

For the diagonal directions in Fig. 3, we can perform an identical analysis in a reference frame aligned with those directions (rotated $+45^\circ$ from the base frame) and then apply the appropriate rotations to define the adjustment matrix in the base frame. In the rotated coordinate frame, direction 8 corresponds to direction 1, resulting in the same form of viscosity matrix adjustment.

$$ \Delta B_8' = \begin{bmatrix} -C_s \cdot VE & 0 \\ -C_s \cdot ME & 0 \end{bmatrix} $$

(21)

The analysis in (22) shows how this matrix can be transformed to be expressed in terms of the base coordinate frame, with the final result presented in (23).

$$ R = Rot(Z, -45^\circ) $$

$$ V' = RV = RJ_\theta $$

$$ F' = B'V' = RF $$

$$ F = R^{-1}B'RJ_\theta $$

$$ \tau = J^T F = J^T R^{-1}B'RJ_\theta $$

$$ B = R^{-1}B'R $$

(22)

Substituting (20) into (23) for $B'$, and calculating $\Delta IM$ as in (20), we arrive at the model adjustment for direction 8 (and opposite direction 4):

$$ \Delta IM_8 = \begin{bmatrix} -0.08ME +0.08VE & -0.095ME +0.095VE \\ 0.095VE +0.015ME & -0.31VE +.18ME \end{bmatrix} $$

(24)

The final stage is to turn the relative adjustments from (20) and (24) into the linguistic variables that will be the outputs of the rule base. If we assume that ME and VE are both equally affected (relative to their own ranges) by the environment force, then each individual element in the $\Delta IM$ matrix has a maximum adjustment value which corresponds to the sum of the absolute value of the ME and VE terms. Each of these matrix elements is normalized by this maximum value to determine the relative effect of ME and VE on that particular element. The result for $\Delta IM_1$ is shown in (25).
\[ \Delta IM_1 = \begin{bmatrix} -VE & -VE \\ -0.42VE - 0.58ME & -0.88VE - 0.12ME \end{bmatrix} \] (25)

Finally, the following demonstrates the determination of an appropriate output membership function for the element in the second row and column of (25) assuming that ME and VE have been respectively fuzzified to be SP and MN.

\[ \Delta IM_{22} = -0.88VE - 0.12ME \]

ME = SP

\[ SP \implies Mean = + \text{ range}/3 \]

VE = MN

\[ MN \implies Mean = -2*\text{ range}/3 \]

\[ \Delta IM_{22} = -0.88(-2/3)-0.12(1/3) = 0.5467 \] (26)

This numerical result can then be related to the linguistic variable value of MP using the output membership functions defined in Fig. 5. Each entry in the rule matrices of Appendix C represents a fuzzy If-Then rule with the ME and VE variables being ANDed in the antecedent and the four output adjustment parameters being set to their respective values in the consequent.

### 3.4 Defuzzification

Once the rules for a given direction are applied for a certain ME and VE, several rules will produce output membership values. The combination of these multiple fuzzy outputs into a single crisp adjustment value is called defuzzification. For this project, the centroid method was employed. A typical defuzzification is visually presented in Fig. 6. Each of the input variables displays partial membership in two adjacent membership functions. The two rules that produce output membership values are displayed in Fig. 6. For each rule, the input variables in the antecedent are combined using an ‘AND’ operation, which is defined as passing the minimum of the two inputs. The output of each AND block is used to trim the membership function of its associated consequent (shaded regions). These two trimmed membership functions are combined as shown and the centroid is calculated as the centre of area (COA) as in (27), where \( \mu(x) \) is the combined output membership function and \( x \) is the output variable.

\[ COA = \frac{\int x \cdot \mu(x)dx}{\int \mu(x)dx} \] (27)

### 4. Experimental Procedure

#### 4.1 Experimental Setup

The purpose of this experiment was to reproduce the results from the experiment performed in [ShMu94]. The simulated arm was exposed to a viscous field of:

\[ B = \begin{bmatrix} 10.1 & 11.2 \\ 11.2 & 11.1 \end{bmatrix} \text{ N sec/ m} \] (28)

There was a total of 216 trials (27 in each direction) with the first and last movement in each direction being performed in the null field (\( B=0 \)). The first trial represented the ideal behaviour of the simulation. This served to demonstrate that the arm dynamics and the forward dynamic model were correct. The next 25 trials in each direction represent the adaptation phase, during which the simulated arm experienced the field in (28) on every trial. The final trial was a catch trial to determine the after-effects, if any, of adaptation.

![Figure 5. Fuzzy membership functions for output variables. Membership functions: LN – Large Negative, MN – Medium Negative, SN – Small Negative, Z – Zero, SP – Small Positive, MP – Medium Positive, LP – Large Positive. The same membership functions are used for all of the IM adjustment parameters.](image-url)
5. Experimental Results and Discussion

5.1 Results

Figure 7 shows the null field performance of the simulation. Clearly the desired straight path trajectory is being followed correctly. The initial performance of the simulation when exposed to the force field is shown in Fig. 8. Comparing this with the simulated result reported in [ShMu94], as reproduced in Fig. 9, one can see that the two simulations performed similarly. The human subjects in the [ShMu94] experiment had average initial exposure trajectories as seen Fig. 10.

Figure 11 shows the progression of the movement trajectories, with sub figures A-C presenting the performance after 5, 15 and 25 trials in each direction. Clearly, one can see the improvement in performance resulting from the adaptation process. Finally, with evidence of adaptation, the presence of the internal model was determined using the after-effect trial in each direction. These trajectories are presented in Fig. 12, and when compared with the simulated (Fig. 13) and actual (Fig. 14) after-effect trials from [ShMu94], we can again see a favourable correlation in trajectories.

Figure 6. Defuzzification (fuzzy inference) example. If-Then rules are applied to the input variables and the value of the output variable is inferred by calculating the centre of area.

Figure 7. Simulated performance in the null environment.

Figure 8. Movement trajectory for the first simulation adaptation trial.
Figure 9. Simulated initial performance as presented in [ShMu94].

Figure 10. Average human subject performance during early exposure to the force field [ShMu94].

Figure 11. Progression through adaptation trials. A: 5th movement in each direction, B: 15th movement, C: 25th movement.
5.4 Discussion

The null field performance from Fig. 7 demonstrates that the dynamic model of the arm was working as expected. The simulation was able to follow the desired trajectory exactly. This is further supported by the initial force field exposure trials, since the resulting trajectories match closely with previous results.

It is clear from the adaptation process outlined in Fig. 11 that the fuzzy learning strategy was successful in incrementally updating an internal model of the environment dynamics. For all but one direction the adapted trajectory approached the ideal minimum jerk trajectory originally seen in the null field. However, there appears to be an issue with the fuzzy adaptation procedure for direction 8. Even by the 25th movement, the trajectory has not approached the ideal trajectory, although it has improved over the initial trajectory. The system did form an internal model based on the errors experienced in this direction, as evidenced by the after-effect seen in Fig. 12. Examination of Fig. 11C suggests that the error lies in the ME dependent terms of the rule matrix, since there is still an ME of approximately 2 cm. The opposite direction, 4, did not show errors even though it made use of the exact same FIS for updating its model. However, since each direction is operating in a different region of the potential velocity state space, there may be some additional direction dependence for which the current model does not compensate.

6. Conclusions and Recommendations

Overall, this work supports the idea concept of using fuzzy-logic to understand and model the learning and control mechanisms of the human brain. A relatively simple fuzzy model was implemented and verified with a simulated human arm and the resulting performance closely matched both previous simulations and human experiments. Using just two performance metrics, a four element model was constructed that was able to compensate for the unknown environment dynamics. Naturally, by including more inputs into the system there is potential for improvements in performance and model accuracy.

In future, the discrepancy in the direction 8 performance must be addressed. This will involve varying the values of the $B$ matrix and performing trials in both direction 4 and direction 8. Once this problem has been successfully addressed, there is potential to fine-tune the model to more closely represent human subject performance and to move beyond constant viscosity matrices. This would potentially include using additional input parameters such as interaction force and joint based trajectory errors. Also, there would be an opportunity to study learning generalization by using additional fuzzy inference systems to determine how adaptation in one
direction/environment may affect performance in another. Finally, with the close relationship between fuzzy-systems and neural networks, there is potential to take any improved model and implement it using a neuro-fuzzy approach.

References


