A Fuzzy Logic Positioning System for an Articulated Robot Arm

Juan Martinez, John Bowles, and Patrick Mills
Electrical and Computer Engineering
University of South Carolina
Columbia, SC 29208

Abstract

Articulated robot arms offer maximum positioning flexibility but suffer from complex kinematics. In most applications, linear motion is desirable. Calculating the kinematic equations which govern an articulated arm is straight forward; however, it is generally difficult to calculate the inverse kinematic equations that are needed to position the arm in closed form. Using a fuzzy reasoning system, it is possible to accurately position an articulated arm without explicitly solving the inverse kinematic equations.

1. Introduction

Manipulator arms which mimic the human arm are called articulated arms. Articulated arms have several important advantages over other configurations: they can reach over or under objects, cover a large work space relative to the volume of arm, and have maximum flexibility. However, these advantages come at the cost of more complex kinematic models: difficulty in controlling linear motion, and difficulty in visualizing exactly how the arm should move from one point to another [1].

The kinematic equations which govern an articulated arm convert joint coordinates into 3-dimensional world coordinates. Using basic geometric relationships these equations are generally easy to express in closed form. But the inverse kinematic equations, which convert from world coordinates to joint coordinates, often cannot be obtained in closed form. In order to position the arm, the control system must be able to convert from the world coordinates to the corresponding joint coordinates. In effect, it must solve the inverse kinematic equations, sometimes called the arm equations.

In the cases where the arm equations cannot be found or where they are too complex to calculate in real time, heuristics must be used to estimate the joint coordinates based on information about the arm configuration. A fuzzy reasoning system provides an excellent platform for dealing with the ambiguity present in such heuristics.

2. System Implementation

The experimental system implementation consists of a personal computer, a Motorola 68HC11 microprocessor, [2, 3] and the articulated arm. The personal computer provides a graphical user interface which controls the desired arm position and motion. Position is specified in spherical coordinates which are provided as inputs to the fuzzy control system. The control system gives crisp joint coordinates as outputs. The joint coordinates are then sent to the HC11 which is responsible for actually positioning the arm. The arm consists of three servo motors (waist, shoulder, and elbow) and three solid links as shown in Figure 1. The HC11 outputs the time intensive wave form necessary to position the servos accurately.

It is important to note that the arm runs "open loop." There are no position sensors or other feedback mechanisms to aid in positioning the arm. Instructions are sent to the servo motors which then move the arm to its new position. The positioning accuracy depends on

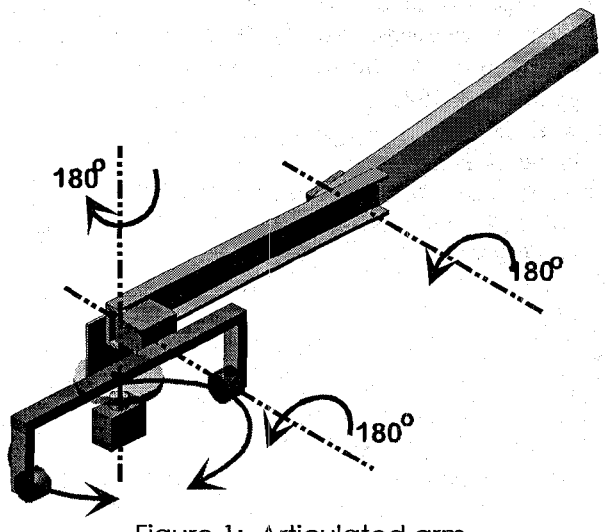


Figure 1: Articulated arm.

0-7803-3645-3/96 $5.00©1996 IEEE

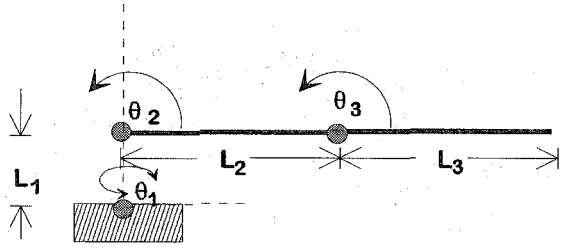


Figure 2: Manipulator arm in zero position.

the accuracy with which the joint coordinates are computed and the resolution of the servo motors (approximately .7 degrees) in moving the arm.

Figure 2 shows the articulated arm in the zero position. The direction of positive rotation for each joint is also shown. θ_1 is the angle of rotation for the waist, θ_2 for the shoulder, and θ_3 for the elbow. Each servo motor can turn through an angle of about 190°. In the implementation, each servo (joint) is limited to an angle of rotation of 180°. Each joint is able to reach 256 discrete points evenly distributed across the axis of rotation. In our implementation, the arms are all of unit length ($L_1 = L_2 = L_3 = 1$) and the arm can reach points in the (approximately) half-sphere having the cross-sectional profile shown in Figure 3. If $L_3 < L_2$, the arm would not be able to reach back to the vertical axis, and the workspace envelop would consist of two concentric half-spheres.

3. Control System Kinematic Equations

For the articulated arm presented here, the kinematic equations, in Cartesian coordinates, are:

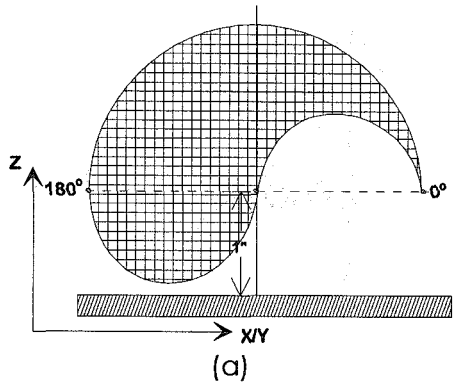
$$x = \cos(\theta_1)[L_2\cos(\theta_2) + L_3\cos(\theta_2 + \theta_3)]$$

$$y = \sin(\theta_1)[L_2\cos(\theta_2) + L_3\cos(\theta_2 + \theta_3)]$$

$$z = L_2\sin(\theta_2) + L_3\sin(\theta_2 + \theta_3) + L_1$$

As can be seen from examination of these equations, derivation of the arm equations (i.e., solving for θ_1, θ_2, and θ_3 in terms of x, y, and z) is a non-trivial endeavor. In some configurations, it may not be possible to solve the equations in closed form. In any case, solving for the joint coordinates using either a numeric procedure or a heuristic, if they are available, is a time consuming process.

To control the arm, we begin by observing (Figure 4) that the point (x, y, z) must lie in the plane defined by the vertical axis and the arm. Thus, the waist angle, θ_1 is determined by $\tan(\theta_1) = y/x$ or $\theta_1 = \arctan(y/x)$. Within the plane the point (x, y, z) can be described by the spherical coordinates ρ and φ, where ρ is the distance



(a)

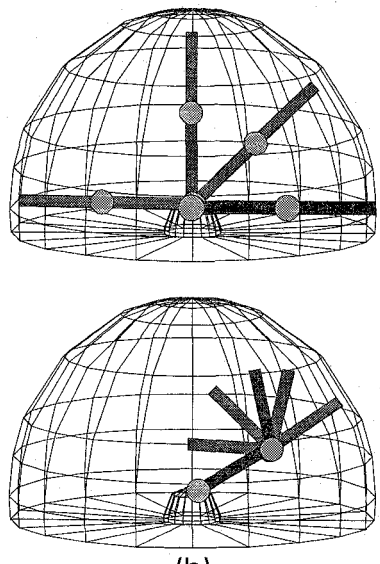

(b)

Figure 3. Workspace envelope for the articulated arm. (a) shows the envelop cross section and (b) shows the envelope in 3-dimensions.

from the origin to the end of the arm and φ is the inclination angle from the horizontal to the vector ρ.

This gives the kinematic equations in spherical coordinates:

$$\theta = \theta_1 = \arctan\left(\frac{y}{x}\right) \quad (1a)$$

$$\varphi = \arctan\left(\frac{z}{\sqrt{x^2 + y^2}}\right)$$

$$= \arctan\left[\frac{[L_2\sin(\theta_2) + L_3\sin(\theta_2 + \theta_3) + L_1]}{[L_2\cos(\theta_2) + L_3\cos(\theta_2 + \theta_3)]}\right] \quad (1b)$$

$$\rho = \sqrt{x^2 + y^2 + z^2} \quad (1c)$$

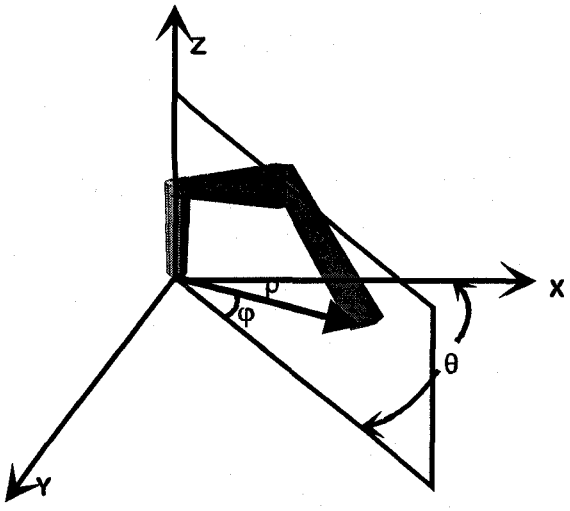


Figure 4. Spherical coordinate system for the articulated arm.

Here θ is the angle of rotation horizontally, φ is the angle of inclination, and ρ is the distance from the origin. From equation (1a) the waist angle, θ_1 is found trivially and the 3-dimensional positioning problem is reduced to a 2-dimensional problem. When the arm is in the vertical position ($\varphi = \pi/2$) the denominator in equation (1b) is 0 and the expression for arc tan is undefined. Thus, for the two dimensional positioning problem we use the alternative formulation:

$$\rho = \{[L_2\cos(\theta_2) + L_3\cos(\theta_2 + \theta_3)]^2 + [L_2\sin(\theta_2) + L_3\sin(\theta_2 + \theta_3) + L_1]^2\}^{1/2} \quad (2a)$$

$$\varphi = \arccos\left(\frac{\sqrt{x^2 + y^2}}{\rho}\right)$$

$$= \arccos\left[\frac{L_2\cos(\theta_2) + L_3\cos(\theta_2 + \theta_3)}{\rho}\right] \quad (2b)$$

Observe that ρ has its maximum value of 3 when the arm is fully extended vertically ($\theta_2 = \pi/2$, $\theta_3 = 0$).

Again, finding θ_2 and θ_3 given ρ and φ, is generally difficult. However, we can easily compute ρ and φ given θ_2 and θ_3. Figure 5a shows contours of constant θ_2, constructed by evaluating the expressions in equation 2 for ρ and φ while holding θ_2 constant and rotating θ_3 through its full range of values ($\theta_3 \in [0, \pi]$). The contours shown are for θ_2 ranging from 0° to 180° in increments of 20°. Similarly, Figure 5b shows contours of constant θ_3 from 0° to 180° in increments of 20°. Although no closed form solution exists, these contours

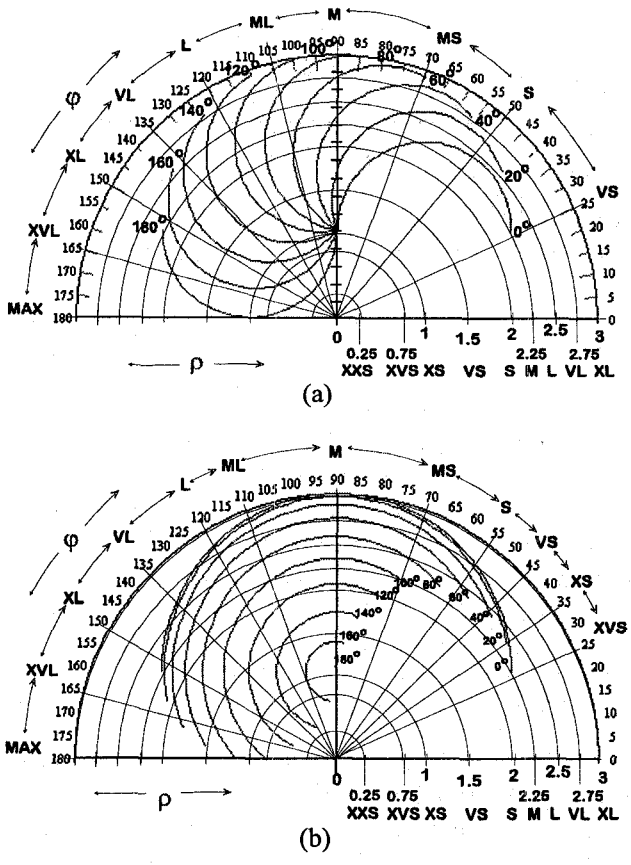


Figure 5. Contours of constant θ_2 (a) and constant θ_3 (b) for the fuzzy positioning system

represent the solution to the inverse kinematic equations for a given ρ and φ.

4. Fuzzy Positioning System

An alternative to explicitly solving the arm equations is to model the arm solution in Figure 5. Although the arm solution is highly non-linear, a good approximation can be obtained. The objective is to identify points in (ρ, φ) where the arm solution is known and use those points as markers from which the fuzzy rule base for the positioning system can be developed.

The overall structure of the fuzzy positioning system is shown in Figure 6. Input variables, provided by the

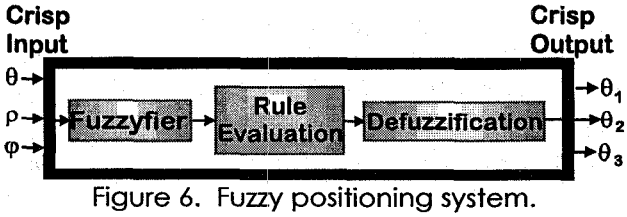


Figure 6. Fuzzy positioning system.

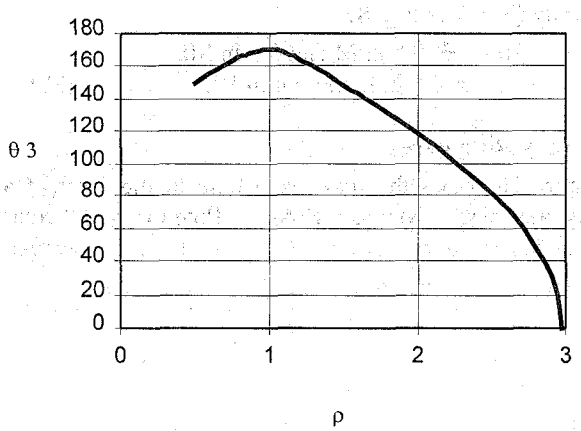


Figure 7. θ_3 versus ρ at $\varphi = 100°$.

user, give the location, in spherical coordinates, that the arm is to move to. These inputs are first fuzzified to represent the coordinates as members of a set of fuzzy input sets. The fuzzy inputs are then matched against the corresponding rules in the rule base and the results defuzzified to give the crisp outputs to the arm servo motors.

In the sequel, we illustrate the operation of the fuzzy positioning system for the inputs: $\theta = 50°$, $\varphi = 100°$, and $\rho = 2.6$. As noted above, the waist angle θ_1 is found immediately: $\theta_1 = \theta = 50°$.

4.1 Fuzzification

The first step in designing the rule base is to create membership sets which describe the inputs. In this case, there are 2 input variables: ρ and φ. ($\theta_1 = \theta$ is assumed to be already known.) The degree of accuracy of the model will be strongly influenced by these set descriptions.

The variable φ describes an arc of approximately 25° to 180° for the given arm configuration. Observe from Figure 5b that the contours of constant θ_3 are closest together, indicating a rapid change with respect to ρ, for large values of ρ. This is shown in Figure 7 which shows θ_3 versus ρ for a cross section of Figure 5b at $\varphi = 100°$. To approximate this type of surface with triangular fuzzy sets (which gives a piecewise linear approximation to the surface), it is best to have fuzzy sets that are more closely spaced where the change in θ_3 is varying and more widely spaced where θ_3 is changing linearly. As shown in Figure 7 the rapid change in ρ starts at about ρ=2.7 at φ =165°, ρ starts to change rapidly at about ρ=1.8. Hence we approximate ρ with the fuzzy sets shown in Figure 8a.

Also observe in Figure 5b, for circles of constant ρ, the value of θ_3 for φ near 90° changes relatively little with respect to changes in φ. Near the ends of the range, θ_3

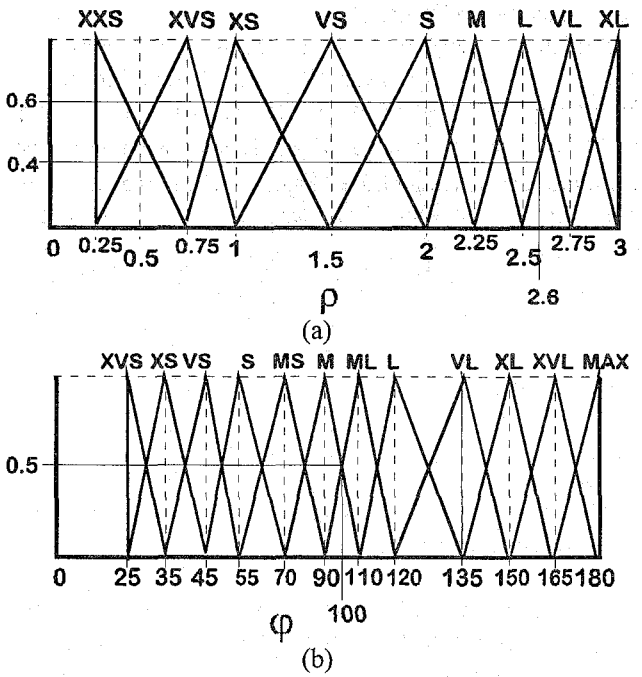


Figure 8. Fuzzy sets representing φ used to approximate θ_3.

changes rather quickly for changes in φ. Thus we chose to represent φ with fuzzy sets that are more closely spaced at the extremes than near the 90^o point. These

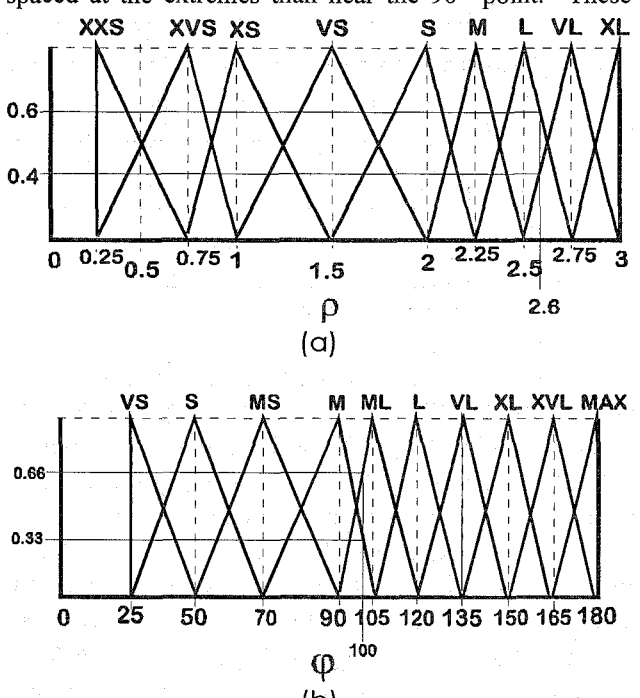


Figure 9. Fuzzy sets representing φ (a) and ρ (b) used to approximate θ_2.

sets are shown in Figure 8b.

Similarly, observe from Figure 5a that for values of φ greater than approximately 80° the contours of constant θ_2 are approximately evenly spaced for constant ρ. For φ < 80° the rate of change of θ_2 is somewhat less (the contours are spaced farther apart). Thus, we chose to represent θ_2 with the fuzzy sets shown in Figure 9b. (More can be used for greater accuracy.) Although the effect is not as pronounced as in Figure 5b, we again observed that θ_2 changes most rapidly for large and small values of ρ. Thus, again we use fuzzy sets that are closely spaced for small and large values of ρ and relatively widely spaced for ρ in the middle of its range to best approximate this curve behavior. These sets are shown in Figure 9a.

Example: Matching the values given for φ and ρ against the fuzzy sets in Figures 8 and 9 we find the corresponding fuzzy inputs:

For θ_2 (Fig. 9):

$\varphi = 100°$ → 0.33 in M and 0.66 in ML;
$\rho = 2.6$ → 0.6 in L and 0.4 in VL (3a)

For θ_3 (Fig. 7 & Fig. 8):

$\varphi = 100°$ → 0.5 in M and 0.5 in ML;
$\rho = 2.6$ → 0.6 in L and 0.4 in VL (3b)

4.2 Fuzzy Rule Base

Figure 10 shows the fuzzy rule base in the form of a Fuzzy Associative Memory (FAM). Potentially, 90 rules are needed to control θ_2 and 108 rules for θ_3; however, due to physical constraints only 40 rules are actually required for θ_2 and 42 rules for θ_3. Observe that the rule output is a crisp number rather than a linguistic term referring to a fuzzy set (often singleton valued). This simplifies the rules and enables the rule base to be constructed directly from Figure 5. For example, rule at the FAM location (L,M) in Figure 10a is read as:

if φ = Medium and ρ = Large, then $\theta_2 = 49°$

is derived by examining Figure 5a for $\varphi = 90^0$ and ρ = 2.5. Note that these are the respective nominal values (when viewed as fuzzy numbers) of the fuzzy sets M for φ in Figure 9a and L for ρ in Figure 9b. If linguistic terms and singleton fuzzy output sets had been used, tuning the

φ

ρ	VS	S	MS	M	ML	L	VL	XL	XVL	MAX
XL				90						
VL			35	61	89	120				
L		10	20	49	75	100	135			
M			10	35	65	95	125	160		
S				25	40	90	120	160		
VS				10	60	90	120	145	165	
XS				0	115	120	135	150	165	180
XVS					145	145	150	160	170	
XXS										

θ_2 (b)

φ

ρ	XVS	XS	VS	S	MS	M	ML	L	VL	XL	XVL	MAX
XL						0						
VL				20	50	58	45	30				
L			35	60	80	83	74	70	40			
M					100	105	100	90	70	30		
S						125	115	105	90	70		
VS						155	150	135	115	100	80	
XS						180	160	150	140	120	110	90
XVS							155	150	145	130	115	
XXS												

θ_3 (b)

Figure 10. Fuzzy Associative Memory (FAM) for θ_2 and θ_3 as functions of φ and ρ.

controller to get the most accurate positioning of the arm would have resulted in the linguistic output for the previous rule being assigned the numerical value 45°. Hence, by using the contours in Figure 5 to construct the rule base directly, considerable tuning of the controller and adjusting of the fuzzy sets was avoided.

The "degree of truth" of the rule output is the minimum of the degrees of truth of the rule antecedents. For example, with the given inputs, the output of the above rule will have the degree of truth min(0.33, 0.6) = 0.33.

Example: For the given inputs, equation (3), the following rules are activated; the degree of truth of the output is also shown:

For θ_2: (4a)

if φ = Medium and ρ = Large, then θ_2 = 49°;
degree of truth = 0.33
if φ = Medium and ρ = VeryLarge, then θ_2 = 61°;
degree of truth = 0.33
if φ = MLarge and ρ = Large, then θ_2 = 75°;
degree of truth = 0.6
if φ = MLarge and ρ = VeryLarge, then θ_2 = 89°;
degree of truth = 0.4

For θ_3: (4b)

if φ = Medium and ρ = Large, then θ_3 = 83°;
degree of truth = 0.5
if φ = Medium and ρ = VeryLarge, then θ_3 = 58°;
degree of truth = 0.4
if φ = MLarge and ρ = Large, then θ_3 = 74°;
degree of truth = 0.5
if φ = MLarge and ρ = VeryLarge, then θ_3 = 45°;
degree of truth = 0.4

4.3 Defuzzification

The fuzzy outputs are defuzzified using the "weighted averages" technique commonly used for singleton output sets (This is the same as the center of gravity technique applied to singleton fuzzy sets [4, p. 336]) :

$$Z = \frac{\sum_{i=1}^{n} \mu_i z_i}{\sum_{i=1}^{n} \mu_i} \qquad (5)$$

where z_i is the output of the ith rule and μ_i is its degree of truth. The resulting crisp output matches the control surface exactly at each of the marker points and provides a linear interpolation of the control surface between these points.

Example: The outputs in equation (4), defuzzified according to equation (5) yield:

$\theta_2 = 70.42°$
$\theta_3 = 66.5°$

Hence, for the input: $\theta = 50°$, $\varphi = 100°$, and $\rho = 2.6$ the solution to the inverse kinematic equations, found by the fuzzy positioning system is:

$\theta_1 = 50°$
$\theta_2 = 70.42°$
$\theta_3 = 66.5°$

To verify the accuracy of the system, we can substitute these values into equations (1a), (1b), and (1c), yielding:

$\theta = 50°$
$\varphi = 98.6°$
$\rho = 2.65.$

The arm position is quite sensitive to small errors in θ_2 and θ_3; the exact solution, to two decimal places, is θ_2 = 70.44 and θ_3 = 71.33.

Thus, the fuzzy positioning system provides an effective solution for controlling the articulated arm. The accuracy of the result depends critically on the marker point values in the FAM, Figures 10a and 10b, and only marginally on the definition of the input fuzzy sets. At these points the solution should be exact. For other points the solution is found by interpolating between the values at the marker points. If the fuzzy membership functions are changed, the accuracy of the interpolation may vary, but generally not by a great deal; if the values at the marker points are in error, gross errors in the positioning of the arm may occur.

5. Conclusion

Articulated arms allow maximum flexibility approaching that of the human arm. The major problem with articulated arms in the past has been the difficulty in calculating the solution to the inverse kinematic equations. However, the fuzzy positioning system developed here shows that a simple representation can be used in some applications.

The ability to control linear arm motion without solving for the inverse kinematic equations allows more complex articulated arms to be utilized. In a test model, the fuzzy positioning system proved to be extremely fast, flexible, and easy to maintain. Changes in the arm configuration require changes in the rule base, but such changes are easily made off-line.

References

[1] P. J. McKerrow, *Introduction to Robotics*, Addison-Wesley, 1991.

[2] Motorola, *68HC11 Reference Manual*, Prentice-Hall Inc.

[3] Motorola, *MC68HC11EVB2 Evaluation Board User's Manual*, Motorola.

[4] G. J. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic Theory and Applications*, Prentice-Hall, 1995.