EFFICIENT METHOD FOR REDUNDANCY RESOLUTION OF A 7 DOF MANIPULATOR

Mihai CRENGANIS[2], Octavian BOLOGA[1,2]

[1]Full Member of the Academy of Technical Sciences of Romania
[2] "Lucian Blaga" University of Sibiu

Abstract. In this paper we are presenting a method to solve the inverse kinematics problem of a redundant manipulator with seven degrees of freedom and a human like workspace based on mathematical equations, Fuzzy Logic implementation and Simulink models. For better visualization of the kinematics simulation a CAD model that mimics the real robotic arm was created into SolidWorks® and then the CAD parts were converted into SimMechanics model.
Keywords: redundancy circle, 7 DOF, inverse kinematics solutions.

1. INTRODUCTION

In our recent studies we have realized that many solutions to the inverse kinematics problem of seven degrees of freedom robotic arm have been proposed so far and this topic presented o lot of interest for many others [1, 2, 3, 5, 6].

A robotic arm with seven degrees of freedom (DOF) has numerous advantages because it gains a vast working space and a high mobility and therefore it can be used in the automotive, aerospace industry and medical domains [7, 8, 9].

The main advantages of robot redundancy are to improve the mobility, flexibility and versatility of a robot and to implement a collision-free and singularity avoidance motion in the workspace based on the redundant DOF.

Many solutions for the inverse kinematics problem have been presented by others, and the most significant are: the generalized inverse Jacobian matrix, the pseudoinverse $J^{\dagger}=J^{T}(J\cdot J^{T})^{-1}$ are widely applied for redundant robots [9], were J represents the Jacobian matrix. The main disadvantage of this method is that pseudoinverse often leads the robot into singularities. Another widely used method for redundancy resolution is the inertia weighted pseudoinverse $JI^{\dagger}=M^{-1}J^{T}(J\cdot M^{-1}\cdot J^{T})^{-1}$, proposed by [6] and this method is based on the fact that the energy consumption can be minimized using the inertia matrix M as the weighting matrix.

The inverse kinematics for a 7 DOF robotic arm offers an infinity of solutions because in the Euclidean 3D space there are only 6 DOF possible movements so we have an undetermined system of equations. The redundancy circle method is based on the fact that even if the end effector of the robotic arm has reached a desired position the elbow joint keeps the ability to move. Not only that but the elbow still describes a circular trajectory. Using Fuzzy logic models we are trying to limit the movement of the elbow and fix it in order to force the robotic arm to adapt a certain position in space and to be able to avoid obstacles or to have a collision free trajectory. In Fig. 1, the 7DOF robotic arm is presented.

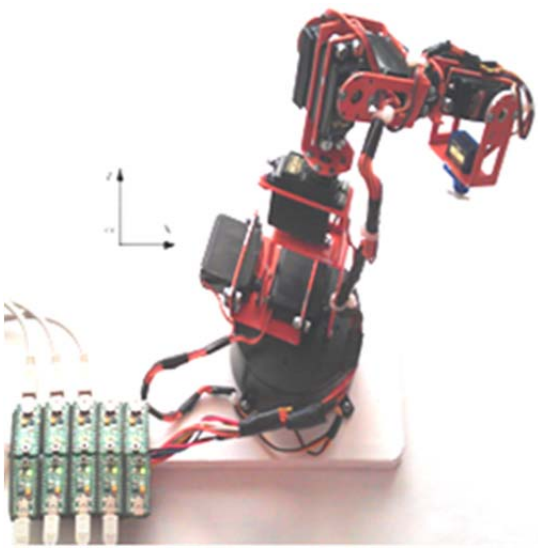

Figure. 1.The 7DOF manipulator .

Since this type of robot has a degree of mobility in excess of six possible in the 3D Euclidian space we had to impose additional conditions to solve the undetermined system of equations. The method is based on an analytical method for solving the inverse kinematics problem.

Using fuzzy logic editor form MATLAB and then the fuzzy logic toolbox for the solution of complex nonlinear systems is very useful and easy to implement. However, in some cases, the designed controller seemed to involve a weakness in the fuzzy rules.

2. KINEMATIC ANALYSIS

In Fig. 2 the structural scheme of the seven DOF robotic arm is shown. To resolve the inverse kinematics problem for a 7 DOF manipulator we have modeled a robot manipulator in SolidWorks® and then based on this structure we have developed the kinematical equations. The robotic arm was then manufactured and assembled based on the CAD parts and drawings.

If we attach to each element *i, (i = 0...7)* of the robotic arm, one fixed coordinate system *ki*, then we can express the homogeneous transfer matrices *Ai* which characterize the relative movements between each element of the mechanic structure.

If we know the relative parameters *θi* and the homogeneous transfer matrix form between two elements, or the homogeneous transfer matrix between the coordinate systems attached to each element, we can determine the total transfer matrix between the system of coordinate *k7* and *k0*.

$$H_{07}=A_1\cdot A_2\cdot A_3\cdot A_4\cdot A_5\cdot A_6\cdot A_7 \quad (1)$$

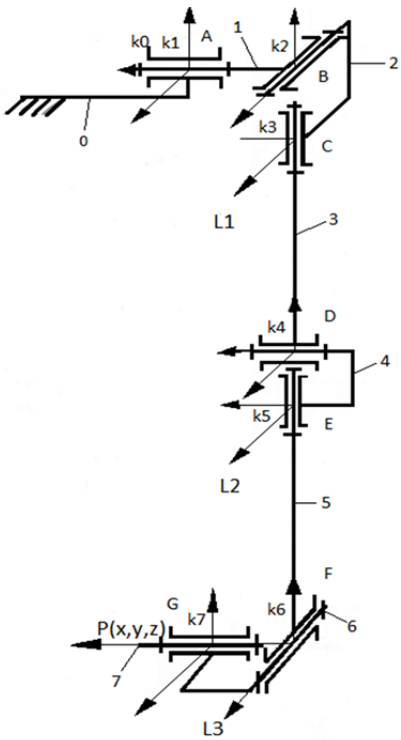

Figure. 2. Structural scheme of the redundant manipulator

where: $A_1=R_y(\theta_1)$,represents the rotation between the base and the first element of the robotic arm or between *k0* and *k1* coordinate systems, $A_2=R_x(\theta_2)$ represents the rotation between the first and the second element of the robotic arm or between *k1* and *k2* coordinate systems, $A_3=R_z(\theta_3)\cdot T_z(L1)$ represents the rotation between the second and the third element of the robotic arm or between *k2* and *k3*, $A_4=R_y(\theta_4)\cdot T_z(L2)$ represents the rotation between the third and the forth element of the robotic arm or between *k3* and *k4*, $A_5=R_z(\theta_5)$ represents the rotation between the forth and the fifth element of the robotic arm or between *k4* and *k5*, $A_6=R_x(\theta_6)$ represents the rotation between the fifth and the sixth element of the robotic arm or between *k5* and *k6*, $A_7=R_y(\theta_7)\cdot T_y(L3)$ represents the rotation between the sixth and the seventh element of the robotic arm or between *k6* and *k7*.

We will note the elements of H_{07}, $(a_{i,j})$.To resolve the inverse kinematics problem we need to start from the fact that we know the position and the orientation of the end effector *X*, *Y*, *Z*, *φx*, *φy*, *φz*, in reference to the fixed coordinate system *k0* and we need to determine the relative positions between robot's elements.

Therefore we wish to determine the relative parameters *θi* between elements which represent the rotation in the kinematical couplings. In this case *i = (0 ... 7)* [7, 8].

Since this type of robot has a degree of mobility in excess of six possible in the 3D Euclidian space we impose additional conditions to solve the system of equations. We will consider the angle *θi* known. In our recent studies this angle was characterized by a fixed value and so the simulations were limited.

The main steps in our research to solve the inverse kinematics problem are:

Creating the transfer matrix which characterizes the end effector position and orientation in reference to a fixed coordinate system *k0*, based on the absolute parameters *X*, *Y*, *Z*, *φx*, *φy*, *φz*.

The wrist position *xim*, *yim*, *zim* will then be determined. After we have computed the wrist position and orientation we will determine the relative parameters *θi*. We will consider these notations*:* *P* (*X*, *Y*, and *Z*) is the end effector position in reference to the fixed coordinate system; the *k7* coordinate system is attached to this section. *L1* is the arm length, *L2* the length of the forearm, *L3* the length from *P (X, Y, Z)* to the wrist.

The transfer matrix which expresses the position and orientation of the end effector in reference to a fixed coordinate system *k0* is composed. The transfer matrix is denoted *H07* and consists of the product of all homogeneous transfer matrices that characterize the end effectors position and orientation in 3D space in reference to a fixed system of coordinates,

$$H_{07} = T_x \cdot T_y \cdot T_z \cdot R_{\varphi x} \cdot R_{\varphi y} \cdot R_{\varphi z} \quad (2)$$

where: *Tx* represents the translation with *X* dimension along the *OX* axis of the *k0* coordinate system, *Ty* represents the translation with *Y* dimension along the *OY* axis of the *k0* coordinate system, *Tz* represents the translation with *Z* dimension along the *OZ* axis of the *k0* coordinate system, *Rφx* represents the rotation with *φx* degrees around the *OX* axis of the *k0* coordinate system, *Rφy* represents the rotation with *φy* degrees around the *OY* axis of the *k0* coordinate system, *Rφz* represents the rotation with *φz* degrees around the *OZ* axis of the *k0* coordinate system.

The matrix that determines the position of the hand wrist is shown below:

$$H_{im} = H_{07} \cdot (0 \;\; -L3 \;\; 0 \;\; 1)^T \quad (3)$$

We will note:

$$(a_{14} \cdot a_{24} \cdot a_{34} \cdot a_{44})^T = H_{im} \cdot (0 \;\; 0 \;\; 0 \;\; 1)^T \quad (4)$$

The distance between the hand wrist position and *k0* is:

$$l = \sqrt{a_{24}^2 + a_{24}^2 + a_{34}^2} \quad (5)$$

The angle θ_4 is determined using the expression:

$$\theta_4 = \pi \pm \arccos\left(\frac{L1^2 + L2^2 - p^2}{2 \cdot L1 \cdot L2}\right) \quad (6)$$

To determine the other angles, we have used the redundancy circle method. We know by fact that even if the hand wrist is fixed the elbow joint manages to describe a circular trajectory around the line segment from *O* to *B*, like shown in Fig. 3. This is the most efficient method for solving the inverse kinematics problem ofa serial and redundant robotic arm.

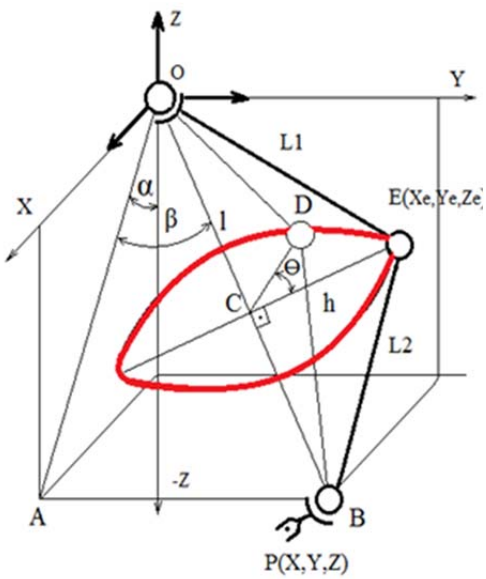

Fig. 3. Redundancy circle of a 7 DOF robotic arm.

The next objective is to determine the position and orientation of the elbow joint. To do that we need to use the *Roll*, *Pitch* and *Yaw* angles starting from *k0* coordinate system. In this case we have used *α*, *β*, *θ*. The last one is the most important because this angle will be used to resolve the entire inverse kinematics of the robotic arm. To be short, the input data to the inverse kinematics equations are the position and orientation of the end effector *X*, *Y*, *Z*, *Rφx, Rφy, Rφz* and the *θ*angle. The *θ* angle is zero when the *OBE* triangle is perpendicular onto *YOZ* plane.

After we have determined the distance from the wrist to the shoulder *l*, we can compute the area *Ar* and height *h* of the *OBE* triangle, using the Heron's formula. Starting from *k0* we can reach the elbow position using some simple homogenous transformations like:

$$He = R_x(\alpha) \cdot R_y(\beta) \cdot T_z(a) \cdot R_z(\theta) \cdot T_y(h) \quad (7)$$

where: *He*, represents the entire homogenous transfer matrix that characterizes the position and orientation of the elbow joint; α, β, θ, represent the *Roll*, *Pitch* and *Yaw* angles, the orientation of the elbow joint; *a,* represents the projection of *L1* onto *l* and it is determined using Pythagora's formula in the *OCE* triangle; *h*, is the height of the *OCE* triangle and *OBE* triangle.

From *He* matrix we can determine:

$$\theta_1 = atan2(He(1,1), He(3,1)). \tag{8}$$

Knowing that:

$$He = A_1 \cdot A_2 \cdot A_3 \cdot A_4, \tag{9}$$

We have the following system of equations:

$$A_1 \cdot A_2 \cdot A_3 \cdot A_4 \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} a_{i,4} \\ 1 \end{pmatrix}. \tag{10}$$

If we multiply to the right with the inverse matrix of *A1* it results the following system of equations:

$$A_2 \cdot A_3 \cdot A_4 \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} b_{i,4} \\ 1 \end{pmatrix}, \tag{11}$$

Because $\theta 4$ is known it results from the first equation of the system (11):

$$\theta_3 = arccos\left(\frac{b_{14}}{L2 \cdot sin(\theta_4)}\right). \tag{12}$$

$\theta 2$ results from the second equation of the system.

To determine the angles $\theta 5$, $\theta 6$, $\theta 7$, we consider the following system:

$$A_5 \cdot A_6 \cdot A_7 = A_4^{-1} \cdot A_3^{-1} \cdot A_2^{-1} \cdot A_1^{-1} \cdot H_{im} \tag{13}$$

Also we will note:

$$A_4^{-1} \cdot A_3^{-1} \cdot A_2^{-1} \cdot A_1^{-1} \cdot H_{im} = (m_{i,j}) \tag{14}$$

From the equality of the two matrices the following results:

$$\begin{cases} \theta_6 = arcsin(m_{32}) \\ \theta_7 = -arctan\,2(m_{31}, m_{33}) \\ \theta_5 = -arctan\,2(m_{12}, m_{22}) \end{cases} \tag{15}$$

Testing the mathematical equations using MATLAB-Simulink

To obtain numerical results for the kinematics analysis, the equations presented above needed to be tested, that for were written in MATLAB and then the code was implemented in Simulink. The implementation was done in a MATLAB Function block, see Fig. 4.The proposed equations were tested using the direct kinematics of the robotic arm and the reached positions and orientations were the same as the ones used as input for the inverse kinematics.

Fuzzy logic approach for redundancy resolution

We have used fuzzy logic because it is conceptually easy to understand and easy to implement. The mathematical concepts behind the controller are very simple and have a more intuitive approach without a very high complexity. First of all by using fuzzy logic we can benefit of its great flexibility. To benefit of the redundancy of the robotic arm we have imposed that the end effector of the robot should follow some linear trajectories on the back of a given panel. After we have tested the mathematical equations of the inverse kinematics with a constant value of the angle θ we have noticed

that the most important inputs for redundancy resolution, are the variation of the Y coordinate and Z coordinate of the position of the end effector. In Fig. 5. a, the end effector reaches a desired position using redundancy resolution with fuzzy logic and in Fig. 5. b, the position is reached without using the fuzzy logic. If we are manipulating the robotic arm without fronting any obstacle in the imposed path, the redundancy resolution is not necessary.

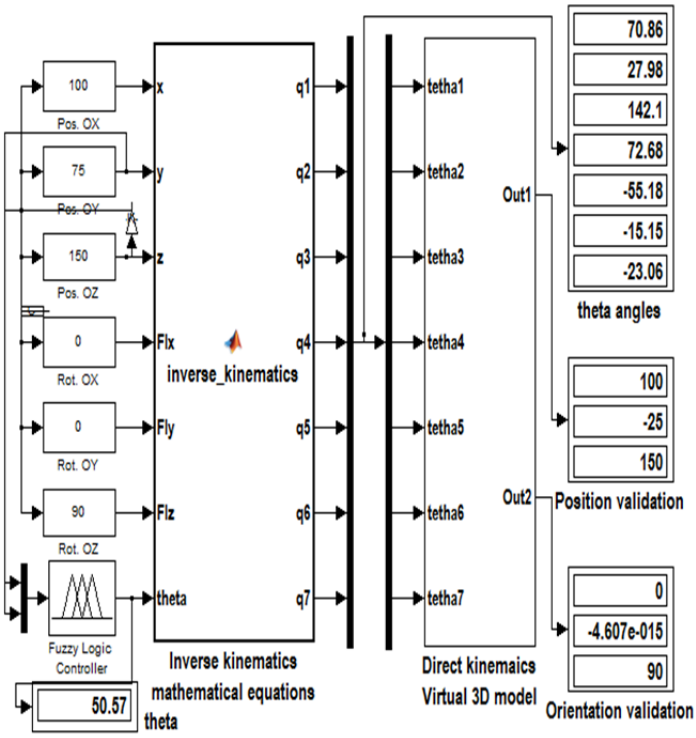

Fig. 4. Simulink model of the inverse kinematics control.

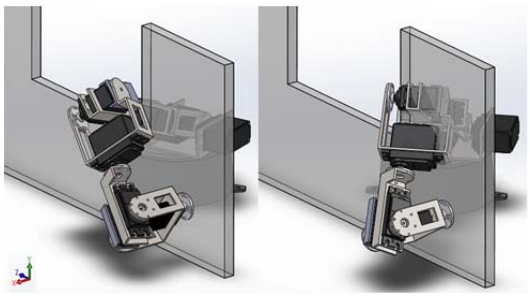

Fig. 5. Robot reching a desired position
a. using fuzzy logic b. without fuzzy logic.

The real problem of redundancy resolution is when one needs to avoid an obstacle or a collision with an object. This is why the robot needs to change the elbow path and θ angle in order to fully complete a given task. To implement fuzzy logic in redundancy resolution we imposed some conditions is such way that we should be able to create the fuzzy rules. So we decided that the θ should vary so the elbow of the robot not collide with the obstacle. As an example for fuzzy logic implementation in redundancy resolution we have chosen a certain trajectory like the one presented in Fig. 6. with red color. To create the fuzzy model we have divided the area of interest in equal small parts, each of this areas is characterized by a fuzzy rule. Next the paper presents the fuzzy model setup. For Z coordinate variation, we have chosen triangular shape membership functions (mf), also we have set the range of variation from -150 to 150 [mm]. For the Y coordinate variation we have also used triangular shape membership functions but the range of variation is from 0 to 150 [mm]. In Fig 7. Z position variation mf are presented.

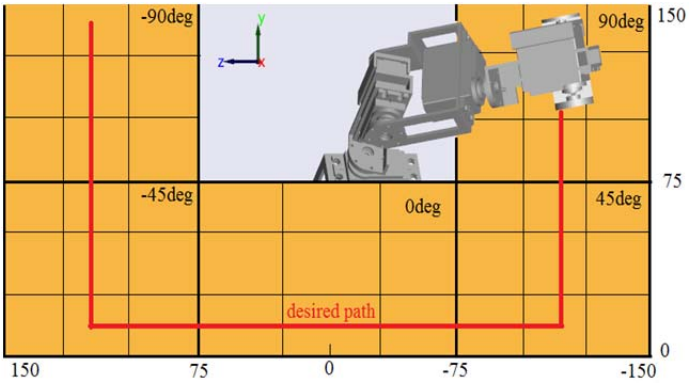

Fig. 6. Robot arm following a certain trajectory.

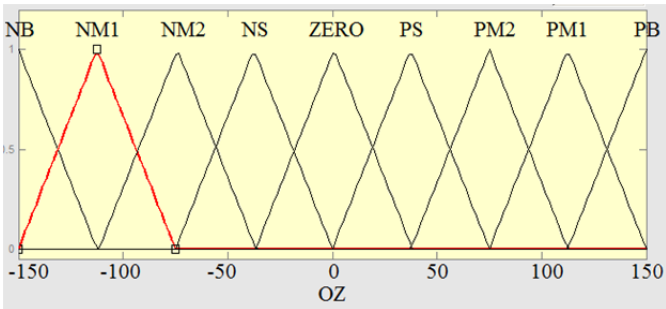

Fig. 7. MF for OZ pozition variation.

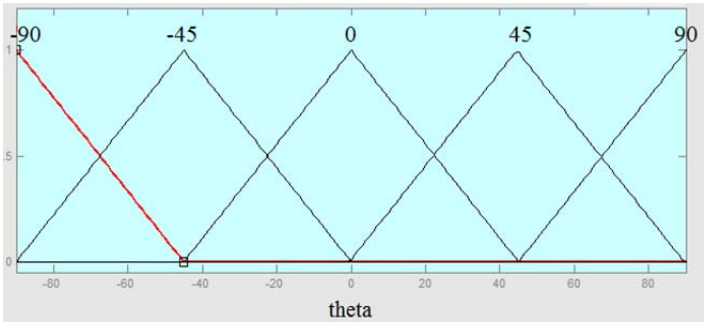

Fig. 8. MF for theta variation.

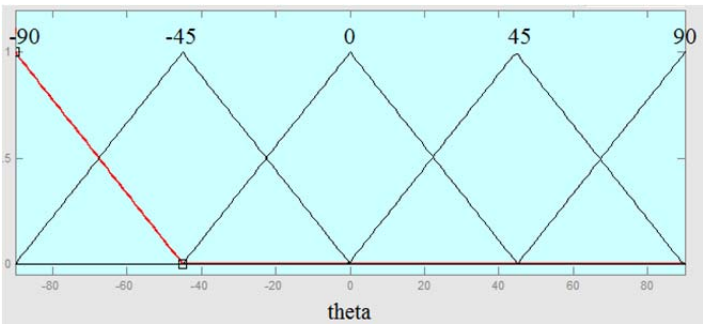

Fig. 8. MF for theta variation.

The two inputs are divided into nine mf for Z position variation and six membership functions for Y position variation. Each combination of Z mf and Y mf characterize a small area from the panel. Each area from the panel corresponds to a well defined θ angle. θ 's angle interval of variation for this example is from –90 to 90 [°].The mf are presented in Fig 8. The fuzzy logic model created for redundancy resolution is presented in Fig. 9. For the fuzzy inference system we have used the

Mamdani type, this represents the way the output θ is determined. Mamdani's fuzzy inference method is the most commonly seen fuzzy methodology [4, 10].

For the defuzzification we have used the centroid method. Perhaps the most popular defuzzification method is the centroid calculation, which returns the center of area under the curve [4, 10]. In Fig. 10 some of the fuzzy rules are presented.

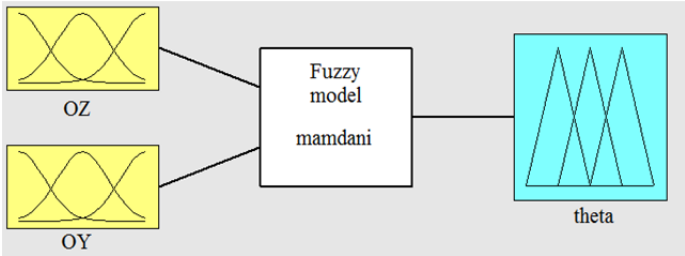

Fig. 9. Entire fuzzy logic model.

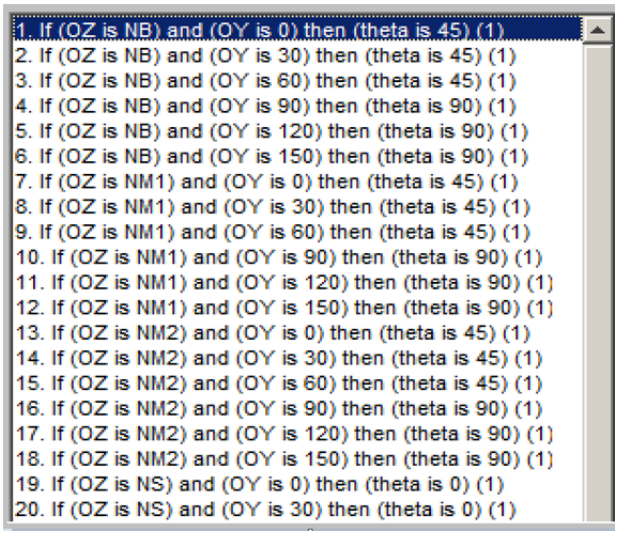

Fig. 10. Fuzzy rules.

3. CONCLUSIONS

The chosen analytical method for solving inverse kinematics problem was effective, accurate and easy to implement.The mathematical equations based on the redundancy circle described by the 7 DOF robotic arm elbow joint led to stability in movement and no impossible to reach positions for the end effector. Also the fuzzy models were correct.

In our future research we are googing to tune the Fuzzy Logic controller, using different range divisions for the inputs and output variation and different number of membership functions, also different types of membership functions to reach some better results.

REFERENCES

[1] J. Craig, Introduction to robotics, Reading, MA: Addison-Wesley, 1989.
[2] J. Angeles, Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms, 3rd Edition, Springer-Verlag, New York, 2006.
[3] G. Antonelli, G. Indiveri and S. Chiaverini, Prioritized closed loop inverse kinematic algorithms for redundant robotic systems with velocity saturations, Proceedings of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 5892–5897, St. Louis, USA, Oct 2009.

[4] D. Howerd, Application of Fuzzy Logic for the Solution of Inverse Kinematics and Hierarchical Controls of Robotic Manipulators, Journal of Intelligent and Robotic Systems 23: 217–247, 1998.
[5] C. Klein, C. H. Hung, Review of pseudoinverse control for use with kinematically redundant manipulators, IEEE Trans. on Systems, Man, and Cybernetics, Vol. 13, 245–250,
[6] B. Siciliano and G. Bastin, Theory of Robot Control, Springer Verlag, London, U.K., 1996.
[7] L. Sciavicco and B. Siciliano, A solution algorithm to the inverse kinematic problem for redundant manipulators, IEEE J. of Robotics and Automation, Vol. 4, No. 4, 303–310, 1998.
[8] C. Carignan, Trajectory optimization for kinematically redundant arms, Journal of Robotic Systems, Vol. 6, No. 5, 221–248, 1993.
[9] T. Deepak, Real-Time Inverse Kinematics of the Human Arm, Presence. Vol. 5. No.4., 393-401 by the Massachusetts Institute of Technology, 1996.
[10] MathWorks Suport-Fuzzy Logic, http://www.mathworks.com/help/html.

O METODA EFICIENTA PENTRU REZOLVAREA REDUNDANTEI UNUI MANIPULATOR CU 7 DOF

Mihai CRENGANIS[2], Octavian BOLOGA[1,2]

[1] Membru titular al Academiei de Stiinte Tehnice din Romania
[2] Universitatea "Lucian Blaga" din Sibiu

Rezumat: În această lucrare este prezentata o metodă pentru a rezolva problema cinematicii inverse a unui manipulator redundant cu șapte grade de libertate și cu spațiu de lucru asemanator bratului uman. Totodata sunt prezentate ecuațiile matematice ce stau la baza rezolvarii redundantei manipulatorului. Redundanta este rezolvata deasemenea prin punerea în aplicare a logicii Fuzzy și prin modele bloc Simulink . Pentru o mai bună vizualizare a simularii cinematicii un model CAD care imita bratul robotic real a fost creat în SolidWorks și apoi piesele CAD au fost transformate în modelul SimMechanics .