53. IWK

Internationales Wissenschaftliches Kolloquium
International Scientific Colloquium

Faculty of Mechanical Engineering

....

PROSPECTS IN MECHANICAL ENGINEERING

8 - 12 September 2008

www.tu-ilmenau.de

TECHNISCHE UNIVERSITÄT ILMENAU

Home / Index:
http://www.db-thueringen.de/servlets/DocumentServlet?id=17534

Alexey Bulgakov / Firas A. Raheem

FUZZY LOGIC STRUCTURE FOR ON-LINE CONTROL OF ROBOT MANIPULATOR IN UNKNOWN ENVIRONMENT

Abstract: The goal of this research to build an autonomous robot manipulator system that is able to plan and execute robot motions in real-time. Designing an on-line Controller based on a structure of four fuzzy logic blocks is suggested to solve the problem of robot manipulator motion planning in unknown environment. This controller can apply for redundant and non-redundant robot manipulator. Unknown environment contains one or more unknown obstacles with arbitrarily shaped. Two main inputs for this structure are: the difference between the desired and the current joint configurations and the minimum distance between every link and the near obstacle. By generating an output control signal as a servo-motor command to control every robot link. A resulted structure has two stages of fuzzy logic block for every link separately. Four obstacles with different positions are used to test this controller. The results of a Computer modeling and then practical experiment (PC-Based Control) show a successful avoiding of all the obstacles even if the robot link falls between two near obstacles.

Keywords: Fuzzy logic; on-line control; robot manipulator

1. INTRODUCTION

Motion planning is one of the principal tasks of autonomous robot systems. As a consequence, robot motion planning is one of the most active research areas in robotics and in the past decades great effort has been put into the development of flexible motion planning algorithms [1].

Depending on Time Mode of Planning Robot motion can be done either off-line or on-line. Off-line Planning is a one-shot computation prior to executing any motion. It constructs the plan in advance, based on a known model of the environment (known Environment), and then hands the plan off to an executor. While On-line Planning is an ongoing activity that relies on a continuous flow of information about events occurring in the environment. In an environment that has unknown obstacles or with constantly changing and partially unpredictable environment (unknown environment), robot motion planning must be on-line. The planner receives a continuous flow of information about occurring events and generates new commands while previously planned motions are being executed. Practically, the primary distinction between off-line and on-line planning is the computation time. In on-line mode, motions must be both planned and executed fast enough in response to the environmental events. A typical on-line planner can be sensor-based, which interleaves sensing, computation and action. The trajectory is calculated on-line it means in real-time during every control cycle. Because the input values may change unpredictably [2, 3, 4, 5, 6, 7].

The main concern for path planning in a known environment is to find globally optimal path navigation algorithms. For an unknown environment do not attempt to optimize the length of the path. Therefore it is good to said that optimal control is not feasible in unknown environments. And solving the problem of motion planning in unknown environment need not be optimal but sufficiently satisfying the robot tasks [2, 8, 9, 10].

In advanced robotics, there is still a strong need for fast reactive collision avoidance computable in real-time in order to prevent the robot from collision with obstacles during motion. Handling of collision conflicts implies a sensor-based collision prediction and a fast strategy to resolve the conflict. Advances solutions do not cause simple blockades to prevent damage, but recalculate actual paths on-line and at the same time intend to reach preplanned goal positions on deviation paths without any time-out. Because of this property fuzzy logic method have been used in many

researches for motion planning of robot manipulator in both known and unknown environment. Fuzzy logic decision mechanism used especially in conditions of unknown environment where only the sensors information practically is available to produce a fast reactive motion [11, 12]. The advantages of fuzzy logic not only fast response, low cost and good real-time ability also it is not necessary to know the exact model of the object or process to be controlled when apply the fuzzy logic control and It can meet the real-time requirements for robot motion planning [6].

An obstacle avoidance controller using fuzzy logic and set of ultrasonic sensors was proposed in where the distance between the end-effetor and the obstacle is measured using these sensors. By analysis the distance, the direction of motion and the velocity as an inputs to the fuzzy controller that will produce a secure point to change the trajectory and ensure avoiding the obstacle successfully [13].

A fuzzy controller builds for on-line planning and control of robot manipulator in unknown environment suggested by K. Althoefer, B. Krekelberg_, D. Husmeier , L. Seneviratne and P. G. Zavlangas , S. G. Tzafestas , Fu Y., Jin B., Li H. and Wang S [4, 5, 6, 14]. The structure of fuzzy controller In all these researches are the same but the design of the membership functions of both the inputs and the outputs are different. In this structure every link has a separate fuzzy control unit and two main inputs: the minimum distance between every robot link and the nearest obstacle and the difference in joint angle between the current position and the goal position. The output of every fuzzy control unit was a direct motor command signal. Planning with a direct motor command signal shows a fast response without needing of the inverse kinematics solution. Their result shows a successfully avoiding all the obstacles and an oscillations in motion when the link falls between two close obstacles.

2. PROPOSED METHOD

This paper suggests a structure of fuzzy control for solving the problem of an on-line planning and control of robot manipulator in unknown environment. The suggested fuzzy controller consists of two stages and four control blocks for a two-link robot planar as example. Each fuzzy stage controls one robot link separately.
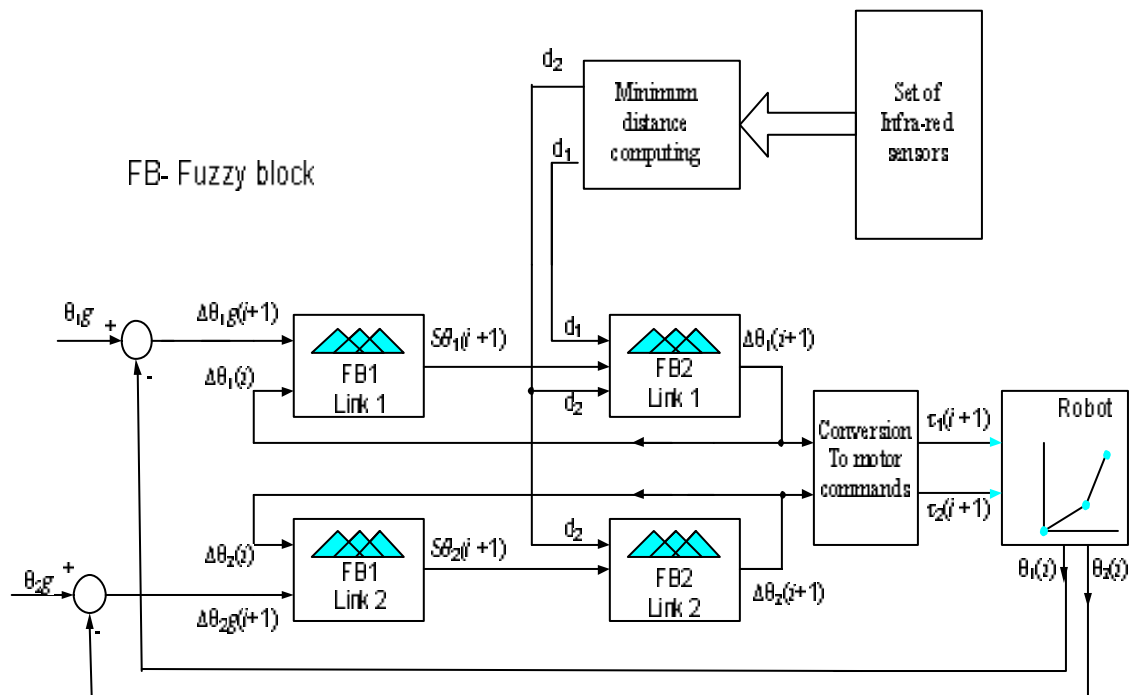

Figure (1): Fuzzy structure for on-line planning and control of a two-link robot manipulator in unknown environment.

Practically the output variable of the suggested controller, was a direct motor command signal (t_j) for every link control where no needing to use inverse kinematics. The motor command has a bipolar-range (+/-) to drive the robot link with positive or negative changing steps. The value of the output is not produced in one stage controller but produced as a result of two stages. The first stage for every link is the stage that produce a suggested step of motion $(S?_j)$ depending of $(??g_j(i+1))$ and the last value of $(??_j(i))$. This step of motion will be as input to a second stage. In the second stage, this step will be checked with the minimum distances between every link and the nearest obstacle. The output of the second stage will be the proper change in joint angle $(??_j(i+1))$. Then the change in joint angle converted to the motor command signal (t_j) which is sent to the joint motor of the corresponding manipulator link to move it. All the controller inputs and output variables take on positive as well as negative values and therefore provide information about the magnitude as well as the sign relative to every link left or right. Figure (1) shows structure of the proposed controller for a two-link robot.

The blocks FB1 for the first link and the second link are fully identical since their task to generate a step of motion while a different blocks of FB2 for the first and second links to generate the required change in joint angle. Where: $??g_1(i+1)$, $??g_2(i+1)$; the difference between the current joint angle and the goal joint angle. $??_1(i)$, $??_2(i)$; the previous value of $(??)$ for every link. At time zero the initial conditions $(??_1(0)$, $??_2(0) = 0)$. $S?_1(i+1)$, $S?_2(i+1)$; the output of the first stage and equal a step of motion for every link. d_1, d_2; the minimum distances between the first and the second with the nearest obstacle. $??_1(i+1)$, $??_2(i+1)$; the output of the second stage and equal to the required change of joint angles $(?_1, ?_2)$. $t_1(i+1)$, $t_2(i+1)$; the final motor command signals to drive every link. For a three links and higher, we can apply this controller by increasing the number of internal blocks. The internal feedback gives the ability of producing a step of motion proportional to the goal-joint error and the last value of $(??_j(i))$. It means when the robot link near the goal point or when the robot faces an obstacle the produced step will take into account the current configuration of the robot and the situation of environment.

Fuzzy membership function design for every fuzzy block is shown in the figures (2, 3, 4). In these figures the design made for the case study in computer modeling.
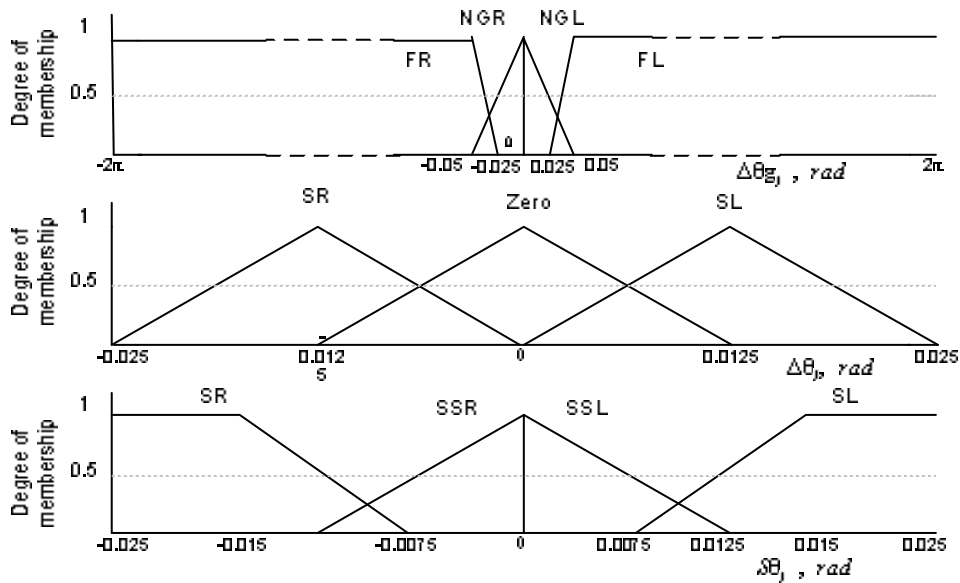

Figure (2): fuzzy membership functions FB1 for the first and second robot links.
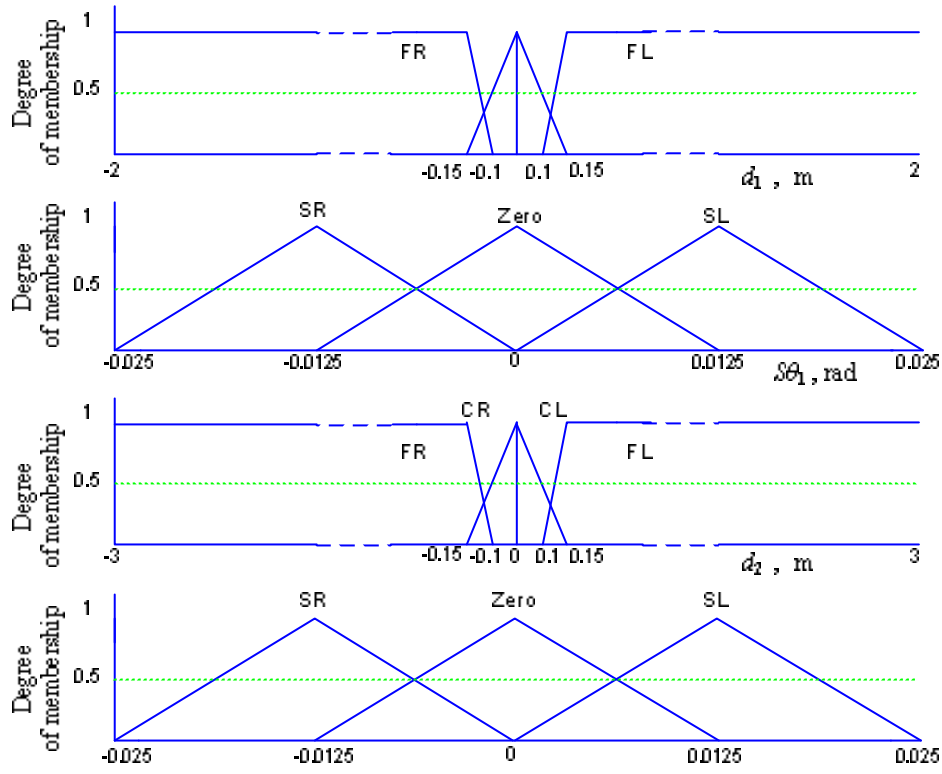
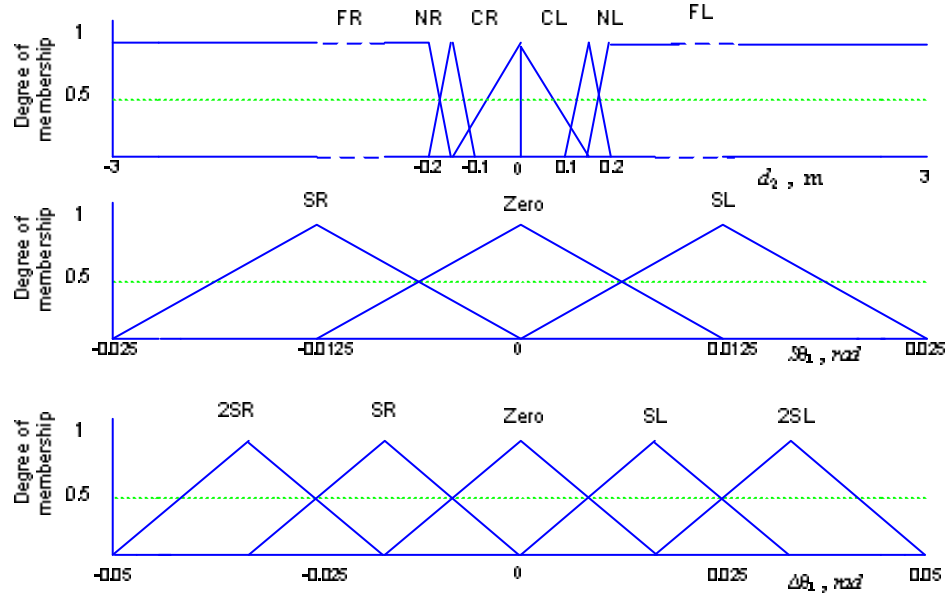Figure (3): fuzzy membership function FB2 for the first robot link.

Figure (4): fuzzy membership function FB2 for the second robot link.

A set of rules used for every fuzzy block to produce the proper output using the Mamdani method. For example the rules for fuzzy block FB2 for the second link will be as:

if (d_2 is CR) and ($S?_2$ is SR) then (? $?_2$ is 2SL).

if (d_2 is CL) and (S_2 is Z) then (? $?_2$ is SR).

The most popular methods to calculate the fuzzy intersection (fuzzy-AND) operation according the fuzzy rules are the minimum and product operators. The final output of every fuzzy block can be computed using the center of gravity (COG) defuzzification method over all rules.

$$\Delta\theta^{crisp} = \frac{\sum_k b_k \int \mu(k)}{\sum_k \int \mu(k)}$$

Where b_k is the center of membership function of the consequent of rule (k). $\int \mu(k)$ denote the area under the membership function.

For the calculation of the minimum distance, the only obstacles considered are those which fall into a bounded area surrounding each link and moving along with it. In this implementation, each such area is chosen to be of cylindrical volume around each link. The area is as long as the link and reaches up to a predefined horizon. This area can be seen as a simplified model for the space scanned by ultrasonic or infra-red sensors attached to both sides of a link. In this paper we use infra-red sensors. The reading of the full set of these infra-red sensors will be input to a separate stage function of minimum distance computation between every link and nearest obstacle. In fact this stage it's not the main subject of this paper.

3. COMPUTER MODELING AND RESULTS

Computer modeling and simulation has been done to test the overall system of figure (1). The signals used to model the robot motion were ($??_1(i+1)$, $??_2(i+1)$). Four unknown obstacles are taken into account with different positions. A two-link robot arm (1 meter, 1 meter) used for this model. The minimum distance was measured by a mathematical algorithm similar to using infra-red sensors.

The results of the computer modeling is shown in figure (5) where the robot has to move form the start configuration ($?_1=60°$, $?_2=25°$) to the goal configuration ($?_1= 180°$, $?_2= 60°$). The error in reaching the goal after (554) program iterations was (0.017853° for $?_2$). The four obstacles were successfully avoided and the first link was successfully stops moving before collision with obstacle number four. Modeling shows that there are an oscillations in motion of the second link occurs when the this link starts moving from two near obstacles till the link success to pass this situation towards the goal point. Figures (6) show the graphs of d_1, d_2, $S?_1$, $S?_2$, $??_1$, $??_2$, $?_1$ and $?_2$.

4. PC- BASED CONTROL EXPERIMENT

The same analysis above was used. Lynx 5 robot arm was used in this experiment as laboratory robot manipulator. Only two links from this robot are used here. A conversion to a motor command signal was needed in order to move the robot. Infra-red sensors cover two sides from the arm area as shown in figure (7). The infra-red sensors used here (GP2D120XJ00F) that can measure the distance in the range (4-30 cm). So practically the problem was when the obstacle near the robot (d < 4 cm) the sensor measure this distance more than (d > 4 cm) which is not real. Another problem in this sensor was the measurement time relatively big (38.3±9.6 ms). It means that approximately half second loses every iteration only in distance measurement. This time delay was the reason of slowly moving.
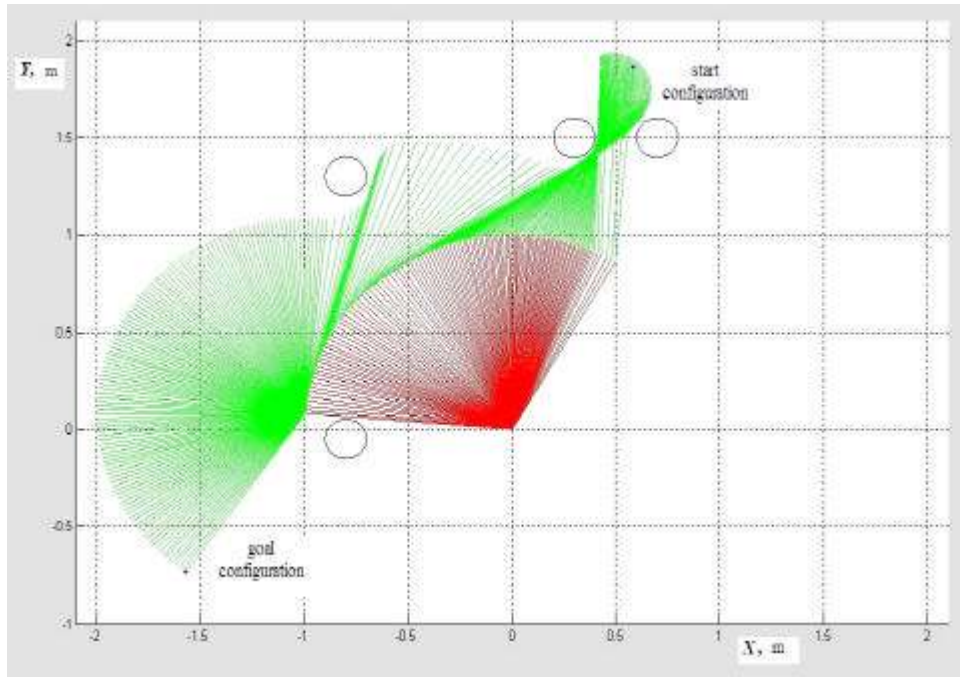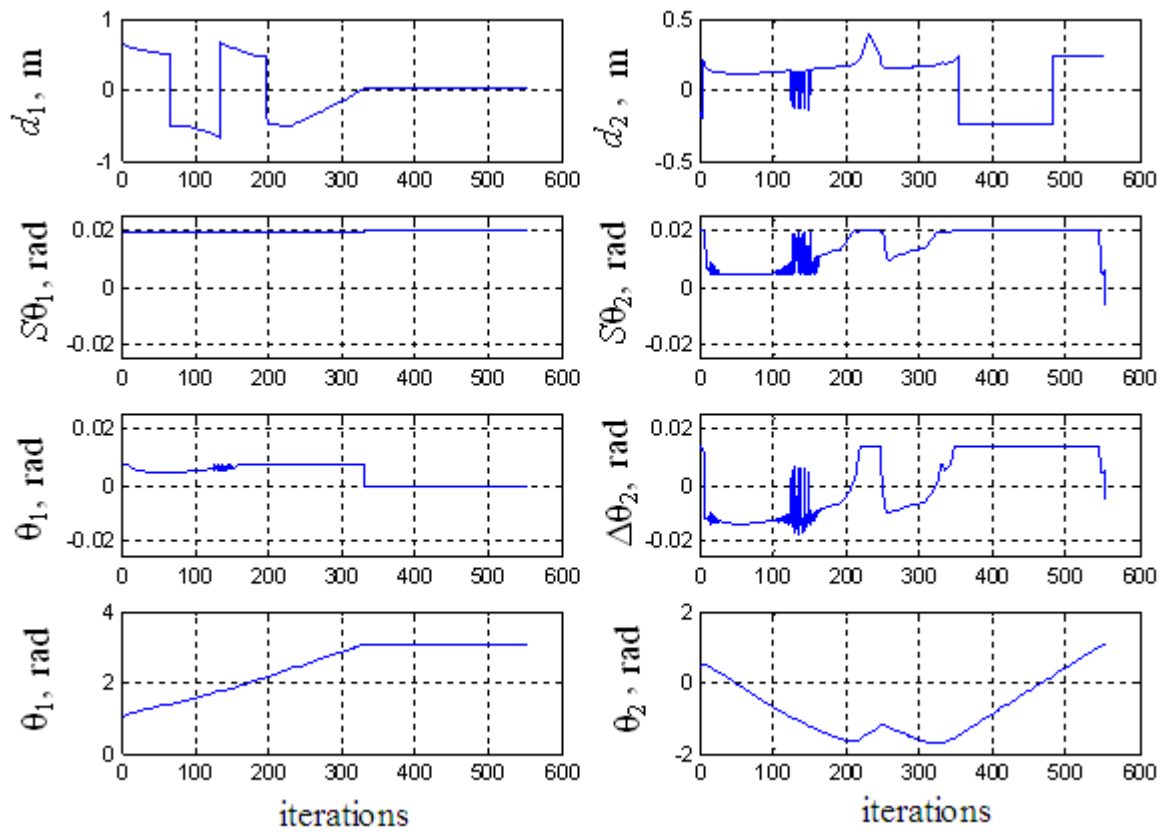
Figure (5): modeling results
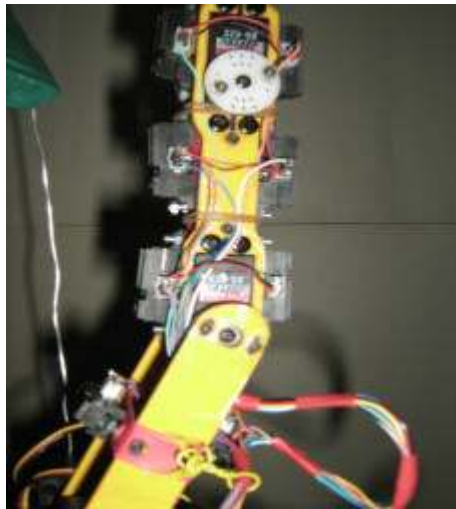

Figure (6): modeling results

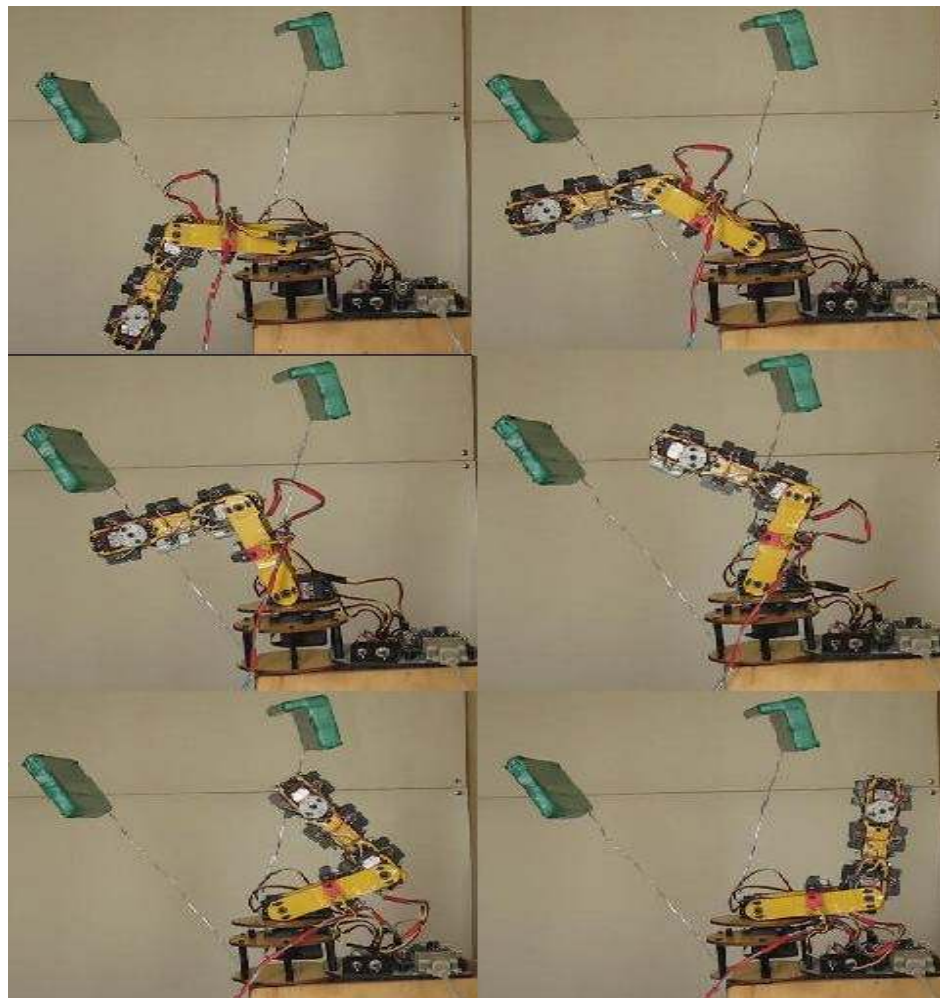Figure (7): two links from Lynx 5 robot arm with infra-red sensors

Figure (8): practical experiment results.

5. CONCLUSIONS

Motion control of a robot manipulator in unknown environment needs fast reactive collision avoidance computable in real-time in order to prevent the robot from collision with obstacles during motion. Fuzzy logic suitable for using as a fast reactive stage because of fuzzy advantages: fast response, low cost, good real-time ability and it is not necessary to know the exact model of the object or process to be controlled when apply the fuzzy logic control.

Fuzzy logic structure for on-line control consists of two stages and four fuzzy blocks. The modeling results and the practical results shows that the robot was successfully avoid all the unknown obstacles. Choosing the type of the sensors is very important since it affects the real-time ability and the accuracy of measurement means better avoiding any unknown obstacles.

REFFERENCES

1. *Gupta K., Pobil A.P.* Practical Motion Planning in Robotics. John Wiley & Sons, Inc. 1998.
2. *W. Xiaojun* (2007). Development of On-Line Motion Planning For Industrial Robot In Dynamic Environments. School of Mechanical and Aerospace Engineering, Nanyang Technological University.
3. *S. Macfarlane* (2001). On-Line Smooth Trajectory Planning for Manipulators. Department of Mechanical Engineering, University Of British Columbia.
4. *Althoefer K.* (1996). Neuro-Fuzzy Motion Planning for Robotic Manipulators. King's College London, University of London.
5. *Zavlangas P.G., Tzafestas S.G.* (2000). Industrial Robot Navigation and Obstacles Avoidance Employing Fuzzy Logic. http://springerlink.metapress.com.
6. *Fu Y., Jin B., Li H., Wang S.* (2006). A robot fuzzy motion planning approach in unknown environments. http://springerlink.metapress.com.
7. *T. Kröger, A. Tomiczek,* and F. M. Wahl, (2006). Towards On-Line Trajectory Computation, http://ieeexplore. ieee.org.
8. *Tsoularis A., Kambhampati C.* (1998). On-line Planning for Collision Avoidance on the Nominal Path. http://springerlink.metapress.com.
9. *E. Mumm,* (2002). Behavior-Based Control for Cooperative Robotic Planetary Cliff Descent. The Graduate College at the University of Nebraska.
10. *V. Lumelsky,* (1987). Effect of Kinematics on Motion Planning for Planar Robot Arms Moving Amidst Unknown Obstacles. http://ieeexplore. ieee.org.
11. *S. Boonphoapichart, S. Komada and T. Hod,* (2002). Robot's Motion Decision-Making System in Unknown Environment and its Application to a Mobile Robot. http://ieeexplore. ieee.org.
12. *Bischo and M. Gerke* (2000). Fuzzy Collision Avoidance for Industrial Robots. http://prt.fernuni-hagen.de/~bischoff/research/bischoff_jb2000_2.pdf
13. *Llata, J.R., Sarabia, E.G., Arce, J., and Oria, J.P.* (1998). Fuzzy Controller for Obstacle Avoidance in Robotic Manipulators Using Ultrasonic Sensors. http://ieeexplore. ieee.org.
14. *Althoefer K., Krekelberg B., Husmeier,D., Seneviratne L.* (2001). Reinforcement Learning in Rule-based Navigator for Robotic Manipulators. http://vision.rutgers.edu.

Autorenangabe:
Professor Alexey Bulgakov
Dipl.-Ing. Firas A. Raheem
132 Proswestschenija st.
346428 Nowotcherkassk / Russia
Tel: 007-86352 2 4568
Fax: 007-86352 2 4568
e-mail: a.bulgakow@gmx.de