



Design and Implementation of Chess-Playing Robotic System

Firas Abdullah Thweny Al-Saedi¹, Ali H. Mohammed²

^{1,2}Computer Engineering Department, Al-Nahrain University,
Baghdad, Iraq

Abstract - This paper introduces a chess-playing robotic system that is designed to autonomously play board games against human opponents. The control of the robotic arm manipulator is addressed in terms of speed and position control. A complete control system is proposed to control the Lab-Volt 5150 robotic manipulator which is a five degree of freedom (DOF) robotic manipulator arm. A smart chessboard is built for tracking opponent's movement. Board representation and search techniques are provided by using the free and open source chess application the "SharpChess". It was modified under VC# environment to fulfill the project requirement. The implemented control is a networked control system (NCS) scheme, the network exchanges the necessary information between the system parts. The proposed systems with all of their parts are tested in a real time with a real chess tournament and the system gave satisfactory results.

Keywords - Smart Chessboard, Robotic Manipulator, Lab-Volt 5150

I. INTRODUCTION

On May 11, 1997, IBM's Deep Blue computer defeated the best human chess player Garry Kasparov [1]. However, when saying a computer is not entirely true. In reality, the computer only calculated the movements and a human still assisted him for executing the movement on the chessboard. Board games are rich issues for human robot cooperation research because these games have an intermediate and an easily adjustable degree of structure. Perception of the chessboard and game pieces, observation of the human, game state and thinking about the game, and manipulation of the chess pieces while playing with the human opponent was involved. Development on physical chessboard game playing systems opens the way for more general human-robot interactions systems that considered less structure. For example, this type of work could lead in the end to a manipulator capable of helping engineering as a field assistant that performs manipulation tasks in different field environments [2].

The term robot was first introduced by the Czech playwright Karel Capek in his 1920 play Rossum's Universal Robots [3]. A robot manipulator should be considered as more than as series of mechanical linkages. The mechanical arm is only one element in an aggregate robotic system which consists of several parts: the arm, end-of-arm tooling, external power source, computer interface, external and internal sensors, and control computer. Furthermore the programmed software should be accounted as an integral section of the aggregate system, since the method in which the manipulator is controlled and

programmed can have a huge impact on its performance and following range of applications [4].

In term of robot control, it can be specified by defined speed and position of the end effector. The problem has become important in both the medical and the manufacturing fields, where the robot arm must be suitably placed with respect to targets (tasks) that cannot be moved. An applicable numerical formulation is presented. While other methods are used the inverse kinematics solutions in their formulation for defining a locality for the manipulator base, this kind of solution is difficult to accomplish because of the inherent complexities in determining all inverse kinematic solutions [5].

A typical open-loop motion control system includes either a stepper motor with a programmable indexer or pulse generator and motor driver. This system does not need the feedback sensors because load position and velocity are controlled by the predetermined number and the direction of input digital pulses sent to the motor driver from the controller. Because load position is not continuously sampled by the feedback sensor, load positioning accuracy is lower and position errors (commonly called step errors) accumulate over time. For these reasons the open-loop systems are most often specified in applications where the load remains constant, load motion is simple, and low positioning speed is acceptable [6].

SharpChess is a free and an open source computer application that enables the human to play chess against the computer. SharpChess uses the chess engine communication protocol so it can play against other chess engines by using WinBoard [7].

The smart chessboard is a chessboard with switches which are used to recognize movements that act as a dedicated input device to enter movements to the chess engine.

Many chess playing systems have been proposed by many researchers, some are listed hereafter:

Nasrul et al. [8] used reed switches to detect the chess pieces that played on the chessboard. The author used the direct method to read all the 64 locations with the aid of the multiplexers which reduce the input lines from 64 to eight inputs only. This design adds an unnecessary cost and complexity to the hardware of the board but it makes the scanning procedure a lot easier.

Nandan et al. [9] presented a simple 3-DOF robotic serial manipulator capable of playing chess in real time. The chess pieces on the chessboard were detected by using standard image processing techniques. HD Logitech

webcam, a low cost custom made serial manipulator and a robust control mechanism are used.

Gurjit et al. [10] designed an artificially intelligent microcontroller based chess opponent which can compete against the user and calculate its moves. The chess pieces on the chessboard were detected with the aid of image processing techniques. The robot used consisted of a vertical section and horizontal section.

Cynthia et al. [2] developed chess playing robotic system. Their system includes a low-cost Kinect-style visual sensor, a custom 6-DOF manipulator, and a state-of-the-art learning algorithms for automatic detection and recognition of the board and objects on it. Examining the structure shows that the system is slow and expensive.

M. Santhosh [11] introduced a chess playing robotic system for people who bare darkness throughout their lifetime. The visually impaired players need to 'see' through 'touch' in the wooden chessboard and make the move using special keyboard. Speech recognition technique is used to identify the last move.

Since a robot manipulator was used, its necessary to list the research deals with robotic kinematics and control and some are listed hereafter:

Velarde et al. [12] presented a novel simulation methodology applied to a 5-DOF manipulator. The work included mathematical modeling of the direct, inverse and differential kinematics as well as the dynamics of the Catalyst-5 Robot manipulator. The method accomplished the path following in the 3D space and MATLAB-Simulink approach was used.

Abbas [13] presented a direct kinematics modeling of 5-DOF stationary articulated robot arm. The author presented an adopted modeling method to represent and simulate the simultaneous positional coordinates for each joint of the robot while it moves from one target to another. The author used Lab-Volt 5150 simulator and robotic toolbox to test the model.

Deshpande et al. [14] derived the kinematic equations of SCORBOT-ER V plus vertical articulated robot with five revolute joints. An analytical solution for the inverse kinematics of the said robot arm is presented. MATLAB 8.0 is used to solve the mathematical model for a set of joint parameters and these solutions were found to be identical with the actual reading of the robot arm.

Yang et al. [15] described the placement of an open loop robotic manipulator in a working environment that was characterized by defining the position and the orientation of the base of the manipulator with respect to a fixed reference frame.

II. THE LAB-VOLT 5150 ROBOT MANIPULATOR

The range of movement of the base, shoulder, elbow, pitch, and roll are 338°, 181°, 198°, 185°, and 360° respectively. The robot can reach most points inside an imaginary hemisphere with a 432 mm radius that centered on the shoulder joint. The operational space of the Lab-Volt 5150 manipulator was obtained by using MATLAB robotic toolbox [16] and surface plot and it is depicted in Figure (1).

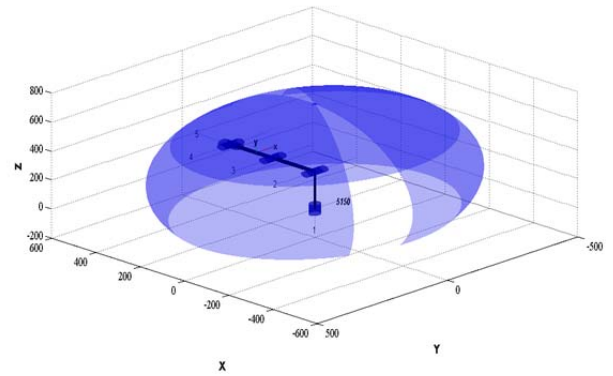


Fig. 1 Lab-Volt 5150 robot manipulator operational range.

III. LAB-VOLT 5150 ROBOT MANIPULATOR DH PARAMETERS

In order to obtain the DH parameters for the Lab-Volt 5150, the link frames must be assigned to each link as shown in Figure (2).

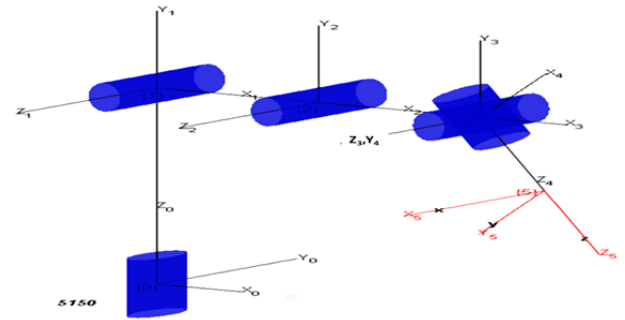


Fig. 2 Coordinate frames attached to Lab-Volt 5150 robot manipulator.

The obtained DH parameters for the Lab-Volt 5150 arm are depicted in Table (1).

Table (4.1) Lab-Volt 5150 manipulator DH parameters [13].

| Link | a_i (mm) | α_i | d_i (mm) | Θ_i |
|------|------------|------------|------------|------------|
| 1 | 0 | 90° | 255 | Θ_1 |
| 2 | 190 | 0 | 0 | Θ_2 |
| 3 | 190 | 0 | 0 | Θ_3 |
| 4 | 0 | 90° | 0 | Θ_4 |
| 5 | 0 | 0 | 115 | Θ_5 |

IV. LAB-VOLT 5150 MANIPULATOR FORWARD KINEMATICS

The position and orientation of the Lab-Volt 5150 wrist center can be calculated as [13]:

$${}^0T_3 = \begin{bmatrix} c_1c_{23} & -c_1c_{23} & s_1 & c_1(190c_2 + 190c_{23}) \\ s_1c_{23} & -s_1c_{23} & -c_1 & s_1(190c_2 + 190c_{23}) \\ s_{23} & c_{23} & 0 & 255 + 190s_2 + 190s_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

The transformation matrix from the wrist center to the end effector can be calculated as [13]:

$${}^3T_5 = \begin{bmatrix} c_4c_5 & -c_4s_5 & s_4 & 115s_4 \\ s_4c_5 & -s_4s_5 & -c_4 & -115c_4 \\ s_5 & c_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

The position and orientation of the Lab-Volt 5150 end effector can be calculated as:

$${}^0T_5 = {}^0T_3 {}^3T_5 \quad (3)$$

The first three entries of the last column which ingredients the origin o_5 in the base frame are x , y , and z are the coordinates of the end effector in the base frame [13]:

$$\begin{aligned} x &= c_1(190c_2 + 190c_{23} + 115s_{234}) \\ y &= s_1(190c_2 + 190c_{23} + 115s_{234}) \\ z &= 255 + 190s_2 + 190s_{23} - 115c_{234} \end{aligned} \quad (4)$$

The rotational part of 0T_5 gives the orientation of the origin o_5 which relates to the base frame, that is the orientation of the end effector relative to the base frame.

$$R = \begin{bmatrix} c_1c_{234}c_5 + s_1s_5 & -c_1c_{234}s_5 + s_1c_5 & c_1s_{234} \\ s_1c_{234}c_5 - c_1s_5 & -s_1c_{234}s_5 - c_1c_5 & s_1s_{234} \\ s_{234}c_5 & -s_{234}s_5 & -c_{234} \end{bmatrix} \quad (5)$$

V. LAB-VOLT 5150 EULER ANGLES

Instead of using Eq. 5 to describe the orientation of the end effector in term of the joints angles, Euler angles method is used to describe orientation of the end effector in relation to the reference frame of the robot. Once the desired pitch and roll are specified, the orientation matrix is determined as:

$$R = ROT(Z, \Phi = \theta_1) * ROT(Y, \Theta) * ROT(Z, \psi)$$

$$= \begin{bmatrix} c_\Phi c_\Theta c_\psi - s_\Phi s_\psi & -c_\Phi c_\Theta s_\psi - s_\Phi c_\psi & c_\Phi s_\Theta \\ s_\Phi c_\Theta c_\psi + c_\Phi s_\psi & -s_\Phi c_\Theta s_\psi + c_\Phi c_\psi & s_\Phi s_\Theta \\ -s_\Theta c_\psi & s_\Theta s_\psi & c_\Theta \end{bmatrix} \quad (6)$$

where:

- Φ : the yaw angle which is equal to link 1 joint angle
- Θ : the pitch angle
- ψ : the roll angle

VI. LAB-VOLT 5150 INVERSE POSITION

Given the desired x , y , z , pitch and roll, it is possible to find the inverse kinematic of Lab-Volt 5150 robot by using kinematic decoupling and the geometric method. First, the wrist center can be found as:

$$o_c = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} x - 115 * c_\Phi s_\Theta \\ y - 115 * s_\Phi s_\Theta \\ z - 115 * c_\Theta \end{bmatrix} \quad (7)$$

Second, projecting o_c into the $x_0 - y_0$ plane as demonstrated in Figure (3).

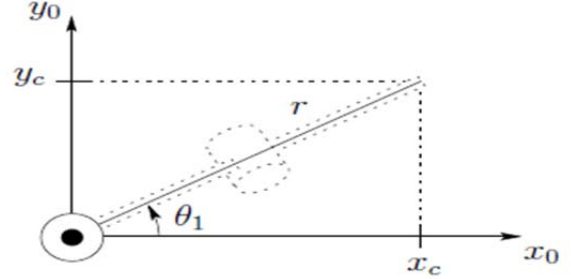


Fig. 3 Projection of the wrist center onto $x_0 - y_0$ plane [4].

gives:

$$\theta_1 = \tan^{-1} \frac{y_c}{x_c} \quad (8)$$

or:

$$\theta_1 = \pi + \tan^{-1} \frac{y_c}{x_c} \quad (9)$$

Third, considering the plane is formed by the second and the third links as demonstrated in Figure (4). Applying the law of cosines to obtain:

$$\cos \theta_3 = \frac{r^2 + s^2 - a_2^2 - a_3^2}{2a_2a_3} \quad (11)$$

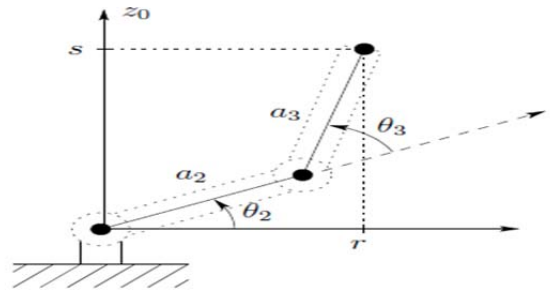


Fig. 4 Projecting into the plane formed by links 2 and 3 [4].

where:

$$r = \sqrt{x_c^2 + y_c^2}$$

and:

$$s = z_c - 255$$

gives:

$$\cos \theta_3 = \frac{\{x_c^2 + y_c^2\} + \{z_c - 255\}^2 - 190^2 - 190^2}{2 * 190 * 190} = D \quad (12)$$

since:

$$(\cos \theta_3)^2 + (\sin \theta_3)^2 = 1$$

then:

$$\sin \theta_3 = \mp \sqrt{1 - D^2} = D'$$

and:

$$\theta_3 = \tan^{-1} \frac{D'}{D} \quad (13)$$

or:

$$\theta_3 = \tan^{-1} \frac{-D'}{D} \quad (14)$$

The two solutions for θ_3 correspond to both the elbow-up position and the elbow-down position, respectively. Similarly, θ_2 is given as

$$\theta_2 = \tan^{-1} \frac{s}{r} - \tan^{-1} \frac{190 \cdot \sin \theta_3}{190 + 190 \cdot \cos \theta_3} \quad (15)$$

VII. LAB-VOLT 5150 INVERSE ORIENTATION

Inverse orientation is used to find the final two joints variables of the wrist. The desired orientation of the end effector R (Eq. 6) and 0R_3 (which is the rotation part of 0T_3 (Eq. 1)) are used to find the final two joints variable as:

$$R = {}^0R_3 {}^3R_5 \rightarrow {}^3R_5 = ({}^0R_3)^T R \quad (16)$$

Where 3R_5 (the rotation part of 3T_5 (Eq. 2)) contains the joints variables of the wrist. Since pitch, roll, θ_1, θ_2 , and θ_3 are known, then ${}^0R_3^T R$ can be evaluated as:

$$({}^0R_3)^T R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad (17)$$

resulting:

$$\begin{bmatrix} c_4 c_5 & -c_4 s_5 & s_4 \\ s_4 c_5 & -s_4 s_5 & -c_4 \\ s_5 & c_5 & 0 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad (18)$$

gives:

$$\theta_4 = \tan^{-1} \frac{r_{13}}{-r_{23}} \quad (19)$$

and:

$$\theta_5 = \tan^{-1} \frac{r_{31}}{r_{32}} \quad (20)$$

VIII. LAB-VOLT 5150 SPEED CONTROL

To make the robot moves in a regular fashion, all joints should reach their destination at the same time. This could be done with the aid of a speed control algorithm that takes the joints offsets as an input as adjust the speeds of joints. Figure (5) portrays the speed control algorithm. The algorithm is executed after setting the desired locations for the joints.

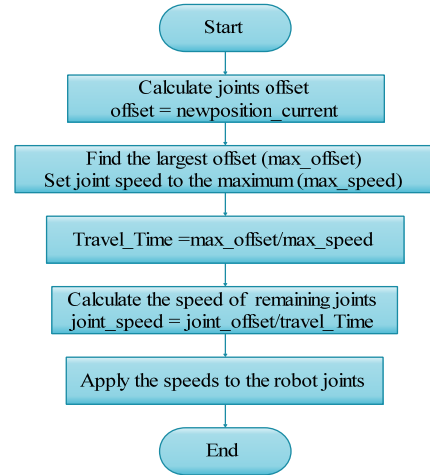


Fig. 5 Speed control algorithm.

IX. LAB-VOLT 5150 JOINTS LOCATION CONTROL

The joints are the movable components of the robot which cause relative motion between adjacent links. Due to mechanical constraints coming from the design of the robot, moving the shoulder joint affects the elbow joint and moving the elbow joint affects the wrist pitch joint as example:

- 1- If the shoulder joint angle moved from n^0 to $n+k^0$, the elbow joint would move from m^0 to $m+k^0$.
- 2- If the elbow joint angle moved from m^0 to $(m+s)^0$, the wrist pitch joint would move from c^0 to $(c+s)^0$.

In this case, it will be impossible to move the robot to the desired location, so an algorithm to correct the joint counter is developed as demonstrated in Figure (6).

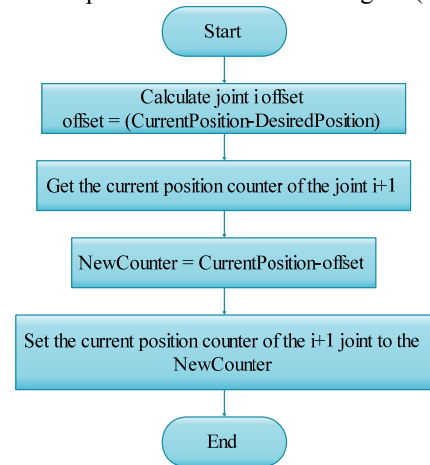


Fig. 6 Joint counter position correction algorithm.

X. THE SMART CHESSBOARD

The smart chessboard consists of a chessboard, data acquisition module, USB interface, and a computer. USB-4704 [17] is used as an acquisition module for the smart chessboard. If an 8*8 matrix is used, then there will be 64 knots in which the rows and columns are intersected. To make a smart chessboard without ghosting problem [18], the reed switch [19] in series with a LED should be connected to each knot. Smart chessboard is designed for the use of a matrix which decreases the number of pins that

are required to read all the matrix 64 reed switches to 8 input lines and 8 output lines. The block diagram of is depicted in Figure (7).

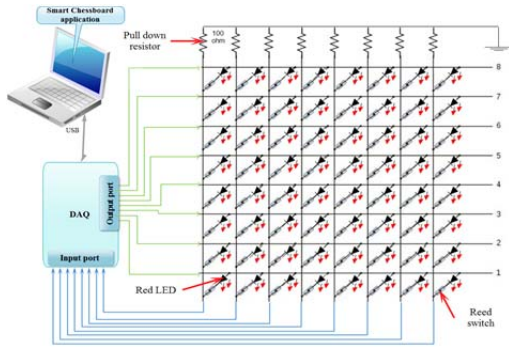


Fig. 7 Smart chessboard block diagram.

The reed blades will close when a magnet is brought toward the reed switch. When the magnet is withdrawn, the reed switch blades will open. The magnetic field is generated from the ring magnet that is installed in the bottom of all chess piece.

XI. THE SCANNING ALGORITHM

At the game start up, each piece is located in its default chess game location, so detecting the chess pieces via sensor matrix and comparing the results with last known board state and piece locations will tell the robot current chess piece locations. Movement detection is done by comparing last know piece locations with the new ones, after finding the opponent’s movement, a chess game verification algorithm will verify the movement for opponent mistakes or even cheats, if the verification fails, a warning soundtrack will be played. The application returns the "from" and "to" algebraic notation coordinate pairs of the opponent’s movement. Figure (8) shows the algorithm main program flowchart.

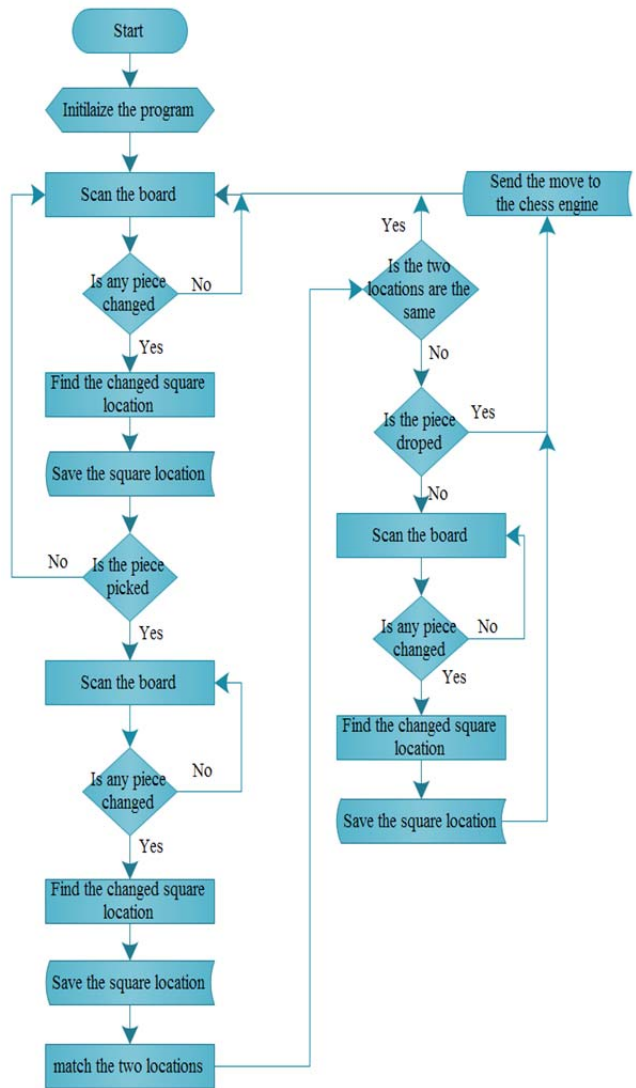


Fig. 8 Smart chessboard algorithm.

XII. THE PROPOSED SMART CHESSBOARD

The implemented smart chessboard is depicted in Figure (9). The dimensions of the surface of the chessboard are 45 cm * 35.5 cm and the height is 6.6 cm. The board contains two side pads where captured pieces are inserted. The board is inserted in front of the robot where the chess squares are 105 mm away from the robot reference frame

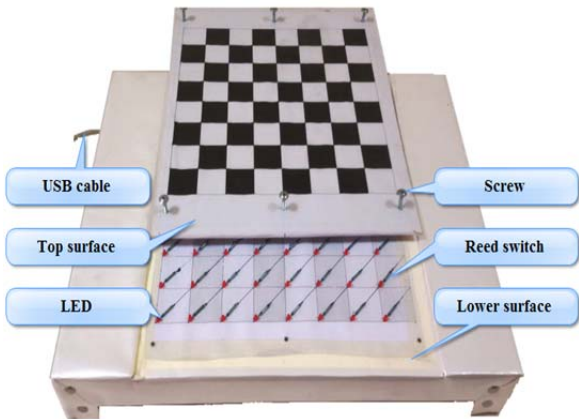


Figure (9) Smart chess board.

XIII. MOVING THE PIECES

The execution of the specific task requires a robot to follow a pre-planned path which is a major problem of motion planning and motion control. The robot moves the piece according to pre-defined path calculated from the start and end coordinates of the moving piece. In the case of application such as pick and place operation as demonstrated in Figure (10) for piece relocation, task is specified as initial and final end-effector location.

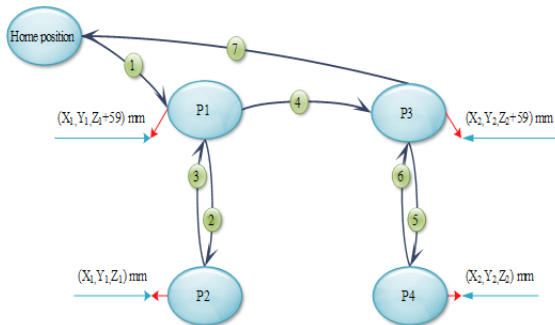


Fig. 10 Piece relocation paths.

This is called point to point-to-point motion. The path is divided into set of points with specific path between them. The points are chosen to create the collision free paths. Figure (11) portrays the path control algorithm.

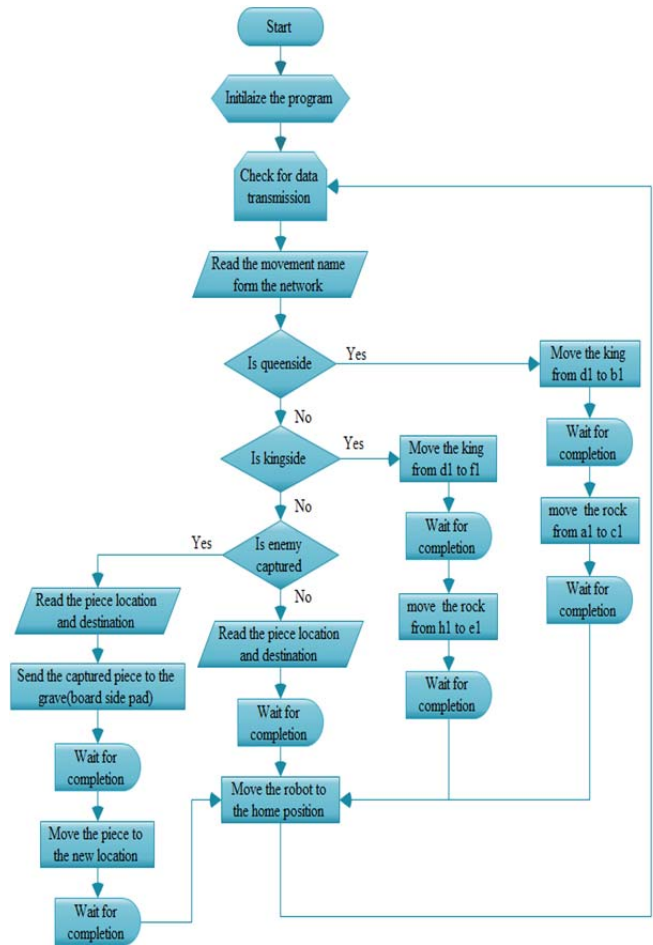


Fig. 11 Path control algorithm.

XIV. THE PROPOSED CHESS-PLAYING SYSTEM

This system introduces a robotic arm manipulator system that is designed to autonomously play board games against human opponent. Figure (12) portrays the proposed system block diagram.

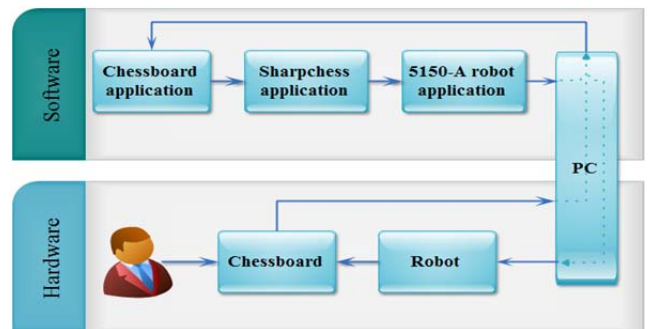


Fig. 12 Human to robot chess-playing system diagram.

The proposed system consists of the following elements:

- Lab-Volt 5150 robot: 5-DOF stepper system robotic arm manipulator system that can play physical board game.
- Chessboard: the smart chess board introduced in Section XII is used for detecting the opponent's movement.

- Applications PC: This PC is connected to one Lab-Volt 5150 robot and one chessboard through USB ports. It contains the robot application, the smart chessboard application, and the chess application which they exchange control data via a TCP/IP localhost connection. Figure (13) portrays human to robot chess-playing system.

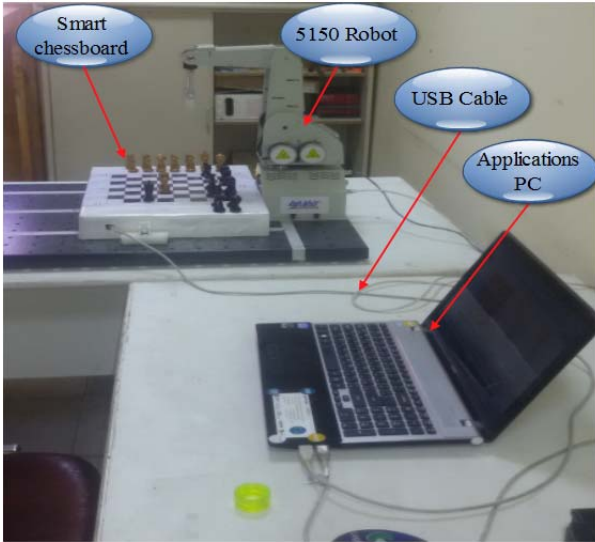


Fig. 13 Human to robot chess-playing system.

XV. RESULTS

1- Joints Accuracy

In order to measure the accuracy for the robot joints, the following tools were used:

- Samsung Galaxy Grand 2 smartphone.
- The android application "Clinometer" used as gyroscope data logger.

After attaching the smart phone to the robot links and calibrating the robot and "Clinometer" the following results were obtained.

A- Shoulder Joint

The results for this joint were obtained by attaching the smartphone to the upper. Figure (14) shows the results obtained for the following joints.

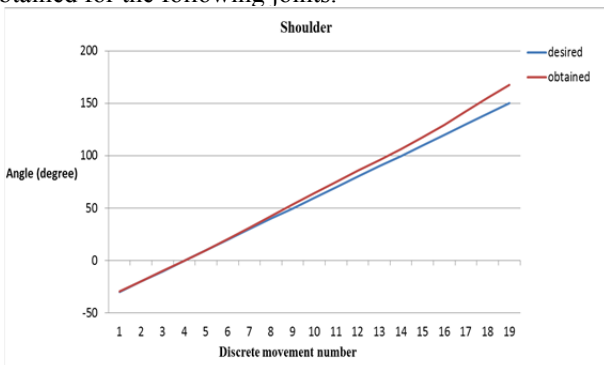


Fig. 14 Shoulder joint test.

B- Elbow Joint

The results for this joint were obtained by attaching the smartphone to the forearm. Figure (15) shows the results obtained.

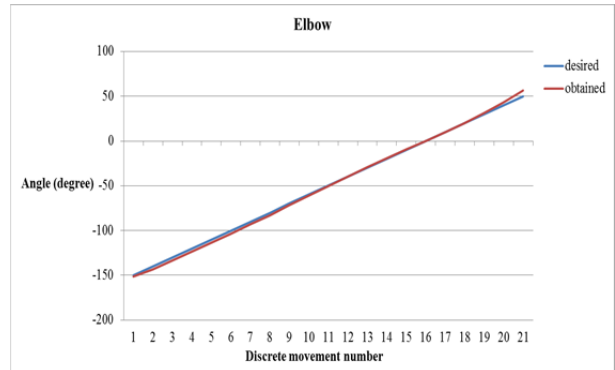


Fig. 15 Elbow joint tests.

C- Wr-Pitch Joint

The results for this joint were obtained by attaching the smartphone to the gripper. Figure (16) shows the results obtained.

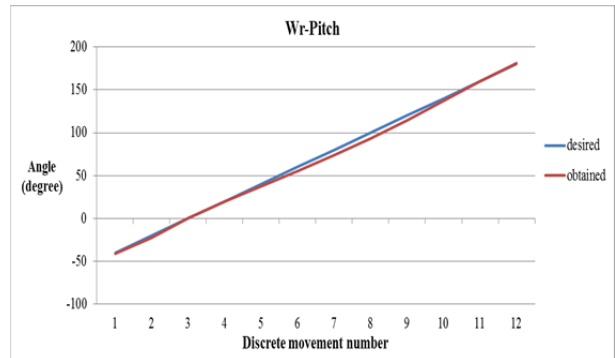


Fig. 16 Wr-Pitch joint tests.

D- Wr-Roll Joint

The results for this joint were obtained by attaching the smartphone by the gripper. Figure (17) shows the results obtained.

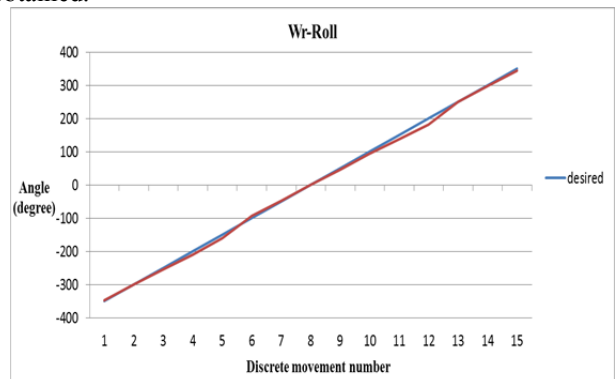


Fig. 17 Wr-Roll joint tests.

The degradation of the robot accuracy is notified during playing, for example, consider Figure (18) that shows the robot dropping the queen on the chess board.



Fig. 18 Robot dropping the queen.

After 4 moves, the robot is ordered to pick the queen but the robot could not return to the same position and it hits the queen as depicted in Figure (19).



Fig. 19 Robot trying to pick the queen.

2- Path Tests

In these tests, the end effector was moved from a point to a point. The orientation of the end effector was set to fixed values (Pitch=180°, Roll=0°) during the movement's testing procedures.

1- Moving along the XYZ space: In this test, the end effector is moved from the point (x=190, y=0, z=331) to the point (x=217, y=120, z=185). The resulting path is shown in Figure (20).

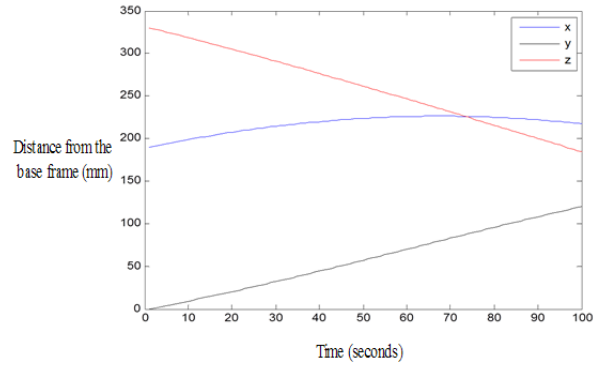


Fig. 20 Moving in XYZ space.

2- Moving in the XY plane: The end effector is moved from the point (x=121, y=120, z=185) to the point (x=346, y=-120, z=185). The resulting path is shown in Figure (21).

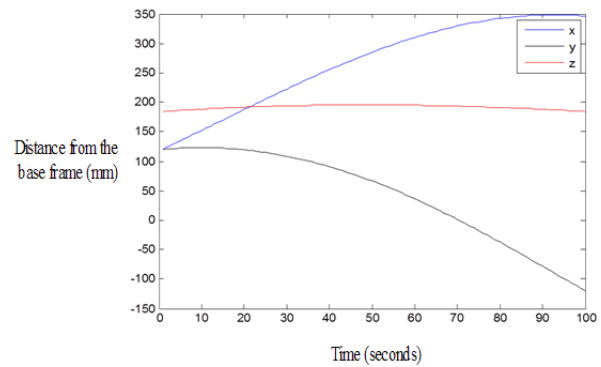


Fig. 21 Moving in the XY plane.

XVI. DISCUSSION

The robot accuracy decreases as a function of the discrete movement it makes. Figure (14) and Figure (15) show that the accuracy of the joint decreases gradually as a function of the movements stops the robot made. The stepper motors of the joint drifts each time the joints stops. Since the overshoot is proportional to the mass, the base suffers the largest drift, then the shoulder, then the elbow. The wrist joints have too much loose as shown in Figure (16) and Figure (17). The robot is very sensitive to the temperature, belts tightness and the lubrication.

XVII. CONCLUSIONS AND RECOMMENDATIONS

- Lab-Volt 5150 is not reliable and loses its accuracy over the time. The controller of the robot may stop responding and returns an inaccurate reading if the disconnected the reconnected to the USB port in a regular fashion (using the provided commands).
- During the calibration, it is recommended to reset the point saved in the tech pendant, then save the new calibration points. If the procedure of the calibration followed without resetting the points, the arm of the

robot will hit the workspace if the robot is ordered to go to the calibration position.

- Open loop stepping systems of the Lab-Volt 5150 robot loses the accuracy gradually over the time no matter how the stepper motor accuracy was due to the physical properties of the said robot.
- The proposed chess and robot software can be used to control other robot manipulators by modifying the kinematics class, also it can be used to control any picks and drops operations.
- It is recommended to calibrate the of the Lab-Volt 5150 robot on regular times according to its working plan.

XVIII. SUGGESTIONS FOR FUTURE WORK

Design and implement the following:

- A trajectory tracking control scheme, taking into account the robot dynamic model in order to perform effective trajectory tracking capabilities like welding, cutting, and speed controlled tasks.
- Image recognition algorithm to detect the chess pieces on the chessboard.
- A gripper for the robot specialized for picking and dropping the chess pieces.
- A distributed system algorithm to increase chess engine strength.
- A virtual reality based control scheme for the Lab-Volt 5150, in order to move the pieces on the chessboard.

REFERENCES

- [1] M. Newborn, *"Beyond Deep Blue: Chess in the Stratosphere"*, Springer, 2011.
- [2] C. Matuszek, B. Mayton, R. Aimi, M. P. Deisenroth, L. Bo, R. Chu, M. Kung, L. LeGrand, J. R. Smith and D. Fox, *"Gambit: An Autonomous Chess-Playing Robotic System"*, 2011 IEEE International Conference on Robotics and Automation (ICRA), 2011.
- [3] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *"Robot Dynamics and Control, Second Edition"*, John Wiley and Sons, 2004.
- [4] A. H. Mohammed, *"Design and Implementation of PSO-BASED PID Controller for MA2000 Robotic Manipulator"*, M.Sc. Thesis, University of Al-Nahrain, 2012.
- [5] J. J. Yang, W. Yu, J. Kim, and K. Abdel-Malek, *"On the Placement of Open-Loop Robotic Manipulators for Reachability"*, Mechanism and Machine Theory, 2009.
- [6] P. E. Sandin, *"Robot Mechanisms and Mechanical Devices Illustrated"*, McGraw-Hill, 2003.
- [7] Sharpchess, "SharpChess C# Chess Game". Available at: <http://sharpchess.com/>.
- [8] N. H. Mahmood and C. K. M. Long, *"Smart Electronic Chess Board Using Reed Switch"*, penerbit UTM Press, 2011.
- [9] N. Banerjee, D. Saha, A. Singh and G. Sanyal, *"A Simple Autonomous Robotic Manipulator for Playing Chess Against Any Opponent in Real Time"*, Proceedings of the International Conference on Computational Vision and Robotics, 2011.
- [10] G. Kaur, A. K. Yadav and V. Anand, *"Design and Implementation of Artificially Intelligent Microcontroller based Chess Opponent"*, Proceedings of the World Congress on Engineering, 2010.
- [11] M. Santhosh, *"A Robotic Arm Based Chessboard for Visually Challenged"*, IIIYEAR CSE, 2010.
- [12] V. Sanchez, R. Gutierrez, G. Valdovinos and P. Ortega, *"5-DOF Manipulator Simulation Based on MATLAB Simulink Methodology"*, Communications and Computer (CONIELECOMP), 2010.
- [13] T. F. Abbas, *"Forward Kinematics Modeling of 5-DOF Stationary Articulated Robots"*, University of Technology, 2013.
- [14] V. A. Deshpande and P. M. George, *"Analytical Solution for Inverse Kinematics of SCORBOT-ER-Vplus Robot"*, International Journal of Emerging Technology and Advanced Engineering, 2012.
- [15] J. J. Yang, W. Yu, J. Kim, and K. Abdel-Malek, *"On the Placement of Open-Loop Robotic Manipulators for Reachability"*, Mechanism and Machine Theory, 2009.
- [16] P. I. Corke, *"Robotics Toolbox 9 for MATLAB"*, MathWorks Inc, 2011.
- [17] USB-4704, *"USB-4704 User Manual"*, Advantix Ltd., 2010.
- [18] Dribin, *"Keyboard Matrix Help"*. Available at: http://www.dribin.org/dave/keyboard/one_html/.
- [19] COM-08642, "Reed Switch Data Sheet", Hamlin Inc. Available at: <https://www.sparkfun.com/products/8642>.