PhD THESIS

Contributions to the Simulation and Control of the Serial Industrial Robots

ABSTRACT

Thesis advisor:
Prof.PhD.Eng. Gheorghe LAZEA
The PhD Thesis contains six major chapters, a chapter of bibliographic references and a chapter with additional materials referenced as appendices.

Chapter 1 - Introduction, presents a short introduction within the fields of computer simulation and industrial robots simulation. A system formed by a Fanuc M-6iB/2HS articulated robot, a Fanuc R-J3iB robot controller and the Fanuc Roboguide simulation software is analyzed. The author sets the objectives of the research based on the analysis results. The main objective of the thesis is the development of a control and 3D simulation software for the Fanuc M-6iB/2HS robot.

Chapter 2 - Mathematical models of robot structures, presents the direct kinematics equations used to obtain the position and the orientation of the robot end effector. In order to obtain these equations I used an analytical method represented by the 2nd variant of the Denavit-Hartenberg method. This method was applied to obtain two variants of the kinematic model, one for fixed coordinates and one for mobile coordinates. The two sets of equations were used later to develop the algorithms for joint interpolated motion and for linear interpolated motion. In the same chapter, I determined the equations of the inverse kinematics model, used to obtain the robot joints’ rotation angles function of the position and the orientation of the end effector. In order to simplify the model, I applied a kinematic decoupling algorithm, due to the fact that the mechanical structure of the robot contains three consecutive joints which intersect in the same point.

The direct and inverse equations of the velocities and accelerations were obtained based on the equations of the kinematics model. The direct equations were obtained using the Newton-Euler iterative method. The joints’ velocities and accelerations equations were obtained using the Jacobi matrix method. In order to simplify the calculus of the inverse Jacobi matrix I applied the Singular Value Decomposition algorithm. Based on the kinematics model I obtained, in a symbolic form, the dynamic model’s equations.

Chapter 3 - The simulation of the industrial robots, presents the 3D graphical simulation program of the Fanuc M-6iB/2HS robot. The first section of this chapter presents an analysis of the programming languages which can be used to create a portable and open-source application for simulation of the industrial robots. Based on the analysis results, the
The Trolltech Qt programming language and OpenGL libraries were chosen. The 3D components of the robot were created based on the Fanuc M-6iB/2HS robot datasheet.

The simulation program contains a graphical interface and a 3D simulation scene and implements the equations of the direct and inverse kinematics models. In addition, the program implements the algorithms for joint interpolated motion and linear interpolated motion.

The simulation scene implements a series of functions used to control the simulated model: rotate, zoom, add/remove components, show/hide trajectory and coordinate frames etc. In order to improve the image quality and the scene illumination I developed and implemented an algorithm which divides the planar surfaces. In this way, any rectangular of triangular surface will contain a predefined number of normal vectors and the resulting surface is generated in a more realistic way. Within the simulation scene the user has the option to select any 3D component of the robot. The selection mechanism is based on an algorithm which parses the depth buffer of the current frame and searches the intersections with the depth buffer planes with a normal vector which has the origin in the selection point.

The trajectory is generated dynamically using OpenGL lists and contains a list of points updated at every scene redraw. In order to avoid the main graphical interface freezing, the scene redraw is based on an algorithm which uses timers. Thus, the 3D graphical scene is redrawn when the timer emits the timeout signal, allowing the program to run other functions from the queue. The communication between classes is based on the signal-slot software mechanism, which is implemented in the Qt programming language.

Chapter 4 - The Fanuc M-6iB/2HS robot control and monitor, presents the functions developed to control and monitor the industrial robot. These functions were implemented both in the simulation program and in the robot controller. The robot control method is based on a TCP/IP client-server application. The client is represented by a thread implemented in the simulation program while the server is represented by a series of programs written in the Karel programming language and installed in the robot controller. The data exchange between the client and the server is based on text messages commands.

The server implements the motion tasks used to control the robot in joint interpolated motion and in linear interpolated motion. The motion of the simulated model is made in parallel with the real robot motion.
This chapter also presents the end effector control scheme and series of text commands which can be used to control the robot.

Chapter 5 - **Testing and applications**, presents a series of tests on the simulation program. The purpose of these tests was to identify the internal and external factors which can affect the simulation and the robot control. Therefore, the simulation program was tested in order to experimentally determine the scene redraw time during simulation, the ping reply time and the motion time during simulation. The tests results shown that the trajectory representation increases the scene redraw time and consequently, the number of trajectory points’ decrease. In addition, any user action in the 3D scene during simulation increases the redraw time up to ten times. Therefore, for safety reasons, some of the 3D scene control functions were disabled during simulation. The ping reply time is small and remains constant. The scene redraw time is also influenced by other applications which run in parallel with the simulation program. A series of tests on different graphical cards shown that the performances of these devices affects the scene redraw time.

This chapter presents two applications developed to control both the simulated model and the real robot using computer peripherals and serial video cameras. The first application uses a joystick to control individually each robot joint. The communication between the joystick and the simulation program was made using the Simple Directmedia Library functions, which are portable, free and open-source. The joystick buttons are linked with the OpenGL object selection functions while the joystick axes are used to start and stop the selected joint motion. The application can be used both in *Offline* and *Online* mode.

The second application uses a serially connected CMUCam3 video camera. The camera is used to identify the objects in working space of the robot by scanning three different colors: red, blue and green. The positioning cycle contains 5 predefined points and 6 automatically determined points. The process is controlled by a class implemented in the simulation program which associates a dialog window. In order to eliminate the position errors, the camera performs three consecutive scans. Due to the mechanical configuration of the end effector, a positioning error of ±5 [mm] is admitted.
Chapter 6 - **Conclusions**, presents the final conclusions of the thesis, the main contributions brought within the robots simulation field and the future research directions. The major original contributions proposed by the present research are:

- the mathematical models of the robot, represented by the kinematics and dynamics direct and inverse equations;
- the *portable and open-source* control and 3D simulation program;
- the algorithms for:
  - simulation control based on timers;
  - scene illumination based on planar surfaces segmentation;
  - joint interpolated and linear interpolated motion simulation;
  - dynamic representation of the trajectories in a 3D scene.
- the TCP/IP communication functions between the robot controller and the simulation program;
- the robot monitor and control algorithms;
- the robot end effector control scheme;
- the tests made to identify the internal and external factors which can affect the robot control and simulations;
- a control application using joysticks;
- a control application using serially connected video cameras.

The bibliographical references cover 90 books and papers used for documentation during the research. A number of 43 articles and books were published between 2005 and 2010.