Servo Motor Control System Based On Model Predictive Controller (MPC)

DR. THIDA AUNG

Professor and Head, Department of Electronic Engineering, Technological University (Lashio), Lashio, Myanmar

Abstract- It is important to get the best response in designing a motor control. This paper proposes MATLAB as a tool in designing motor control system. The Model Predictive Controller (MPC) is used for servo motor control and its important parameters are designed in MATLAB built-in function also known as MPCTOOL. The entire MPC requirement such as control horizon, prediction horizon and weight for input and output of the controller can be calculated by means of MPCTOOL. These values are used in MATLAB m-file for simulation procedure. In this paper, these parameters are set in SIMULINK block of MPC controller to control the servomotor in best stability and performance. And all of the results are shown in MATLAB/SIMULINK scopes. According to these results, the most powerful and useful values are intended to use in physical design.

Indexed Terms- Model Predictive Controller (MPC), Servo Motor Control, MPCTOOL, SIMULINK

I. INTRODUCTION

DC servomotor is widely used in industrials and other indoor applications likes as CD player and HHD of the computer. A common type of servo provides position control. Servos are commonly electrical or partially electronic in nature, using an electric motor as the primary means of creating mechanical force. Other types of servos use hydraulics, pneumatics, or magnetic principles. Servos operate on the principle of negative feedback, where the control input is compared to the actual position of the mechanical system as measured by some sort of transducer at the output. Any difference between the actual and wanted value (an error signal) is amplified and used to drive the system in the direction necessary to reduce or eliminate the error. This procedure is one widely used application of control theory.

There are various techniques to control the speed of servo motor. One such technique is implemented in this paper in order to control the speed of servo Motor using MPC controller toolbox in MATLAB.

The MPC Toolbox is a collection of functions (commands) developed for the analysis and design of MPC systems. At present, it is the most widely used multivariable control algorithm in the chemical process industries and in other areas. While MPC is suitable for almost any kind of problem, it displays its main strength when applied to problems with:

- A large number of manipulated and controlled variables
- Constraints imposed on both the manipulated and controlled variables
- Changing control objectives and/or equipment (sensor/actuator) failure
- Time delays Indeed, in its basic unconstrained form MPC is closely related to linear quadratic optimal control.

However, MPC leads to an optimization problem which is solved on-line in real time at each sampling interval. MPC takes full advantage of the power available in today's control computer hardware. This paper basically focuses on nonlinear modeling and proposes an innovative MATLAB model on MPC Toolbox to study the response of servo motors. The results of this MATLAB model shall prove to be very useful in designing the control strategy for applications involving servo motors.

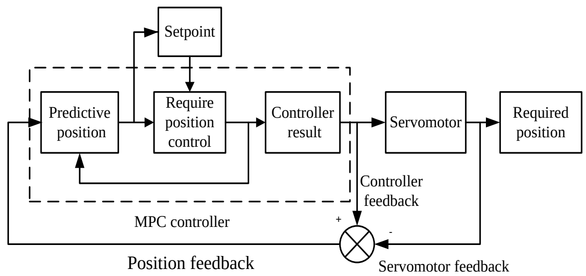

Figure 1. Block Diagram of MPC Servomotor Controller

II. IMPLEMENTATION OF THE SYSTEM

Model Predictive Controller (MPC) controls the servomotor according to the applied voltage, prediction horizon, control horizon and the multiplying with weight of the controller. The main idea of controlling Servo in right position or velocity is using the prediction and control horizon size. The weight can change the movement of the motor and prediction horizon can also affect that one. Before the controller is used, MPC recommends calculating the stable servo system.

The requirements of servo motor specification are firstly calculated. And then these specifications are changed into state space format. This state space model is suitable for MIMO system (multiple input multiple output). It has to change into the discrete system because it is an analogue system. After transforming the discrete system, it must be checked for stable. If the system is not stable, error and trial method is used until the system is stable.

Design of the MPC controller requires the complex calculation and to define the input and output variables can be difficult. Input variables are defined as the Measured Variables (MV) and outputs of the plant are known as Output Variables (OV) in MPC controller.

The controller also uses the variables, which are not like the traditional controller PID, prediction horizon, control horizon and weight of the controller. MATLAB can tune the MPC parameters and also simulate the MPC controller and plant model. Among the scenarios of the MPC GUI window, the best result of the controller variables is selected and used for this application.

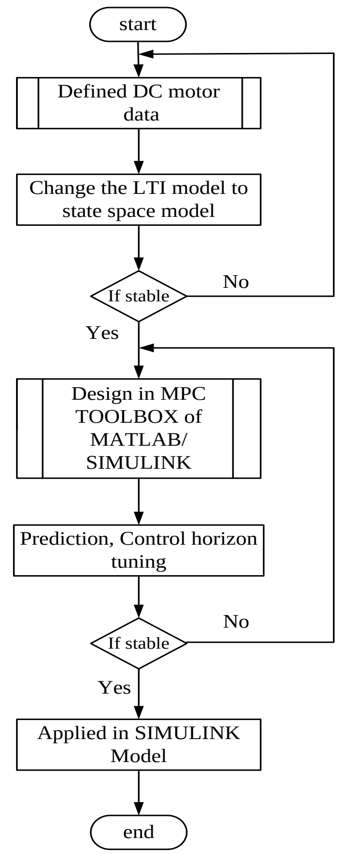

Figure 2. Overall Flow Chart of the MPC Controller

According to the MPC controlling method in MATLAB, there is no need to calculate the detail calculation of controller movement. Because MATLAB can support the MPC controlled model in MIMO system calculation. This can be done by using of power GUI or other methods. If the system using in SISO must change the GPC controlled system, then translate into MPC technique.

In this paper, Model Predictive Controller (MPC) controls the servomotor to have stable and the highest settling time. Input signal is the position command and the output signal is movement of shaft of the servomotor. The sensor can sense the output movement and adjustment of the shaft angular movement and input angular movement of the rotary variable resistor. The prediction horizon and control horizon is defined by means of MPC GUI and scenarios.

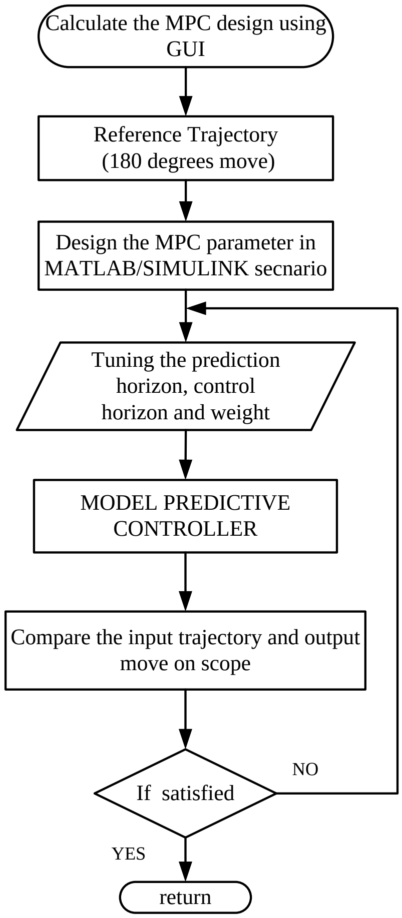

Figure 3. Flow Chart of MPC Controller Design

The simulation of the MPC controller is using the MATLAB/SIMULINK tool box. By the knowledge of operator to define the noise and dead time of the model is firstly calculated and then construct the LTI model of the servomotor. By the way, controller produces the noise and it will affect the system response. But model predictive controller can reduce the noise and other environmental noise. These noises are firstly combined to the model before the simulation start. The changing of the prediction horizon, control horizon can change the output of the plant. Lower the value of control horizon can be smooth the output of the plant (that means less damping). On the other hand, the response of the system in both lower and upper values of control horizon cannot affect the input response. Control horizon lies between the 5 and 20 or is calculated by following equation.

$$\frac{N}{3} \leq M \leq \frac{N}{2} \qquad \text{Equation 1}$$

The MATLAB scenarios can use to determine the stability and controller gain. On its scope, the important facts such as maximum overshoot, settling time and the rising time of the plant are calculated. The MPC controller design flow chart is shown in Figure 3.

III. STABILITY TEST FOR CONTROLLER

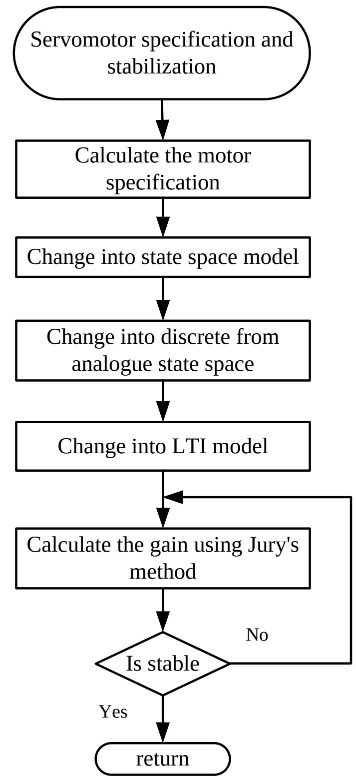

Figure 4. Flow Chart of Servomotor Calculation

MPC controller does not require the stability test, but the Jury’s can check the stability of the controller. The MATLAB can use the command is stable (sys) instead of jury’s test. If the stability test in MATLAB can use that command roots (den) or roots (num). The other method can use for calculation of MPC stability such as Bode, Nyquist, and Nichols method. Only Jury’s stability check is used in this paper. According to the testing with Jury’s table the motor is stable with the MPC controller output that is input data of the servomotor.

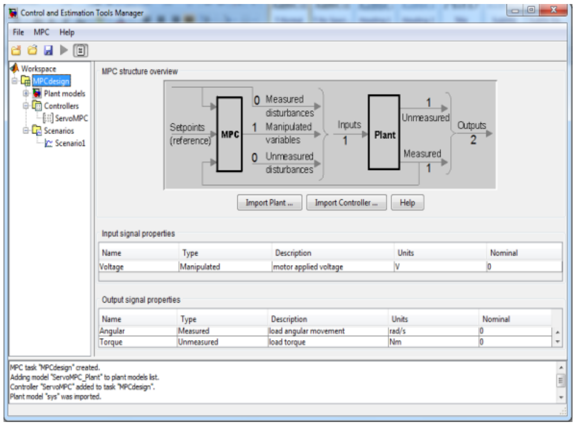

Figure 5. GUI Window of MPC Controller

The GUI model is simple and easy to understand the working of the system. And the GUI window of the Model Predictive Controller parameters can change before simulation. Using the MPC GUI, the output scenarios can tell about the system is stable or not stable under disturbance condition. MPC design is first to declare in workspace and import to the MPC GUI in SIMULINK TOOLBOX. Also the Plant under controller must be imported to the MPC GUI. This window is called by the 'mpctool' command. MATLAB produces a window on the screen and then open the MPC design node at Controller and changes the value in the tabs. The node of Controllers is opened and the controller namely ServoMPC is formed. And also the Scenarios node is opened and clicked on its. A new Scenario1 is produced. Input parameter is applied voltage and output parameters are measured signal for the load angular movement and unmeasured signal for the Torque.

After the MPC simulation with GUI of MPC TOOL BOX, selecting prediction value and control horizon value is useful or not using the SIMULINK model. In this model, prediction horizon is 10 and control horizon is 2. The output weight of the controller is 0.05 for selection form the simulation of MPC GUI.

These results can compare the scopes screen of the SIMULINK model. Double click on the present working directory that contain with SIMULINK model of the controller ending in mdl extension or using the simulation command open_system (cmpc_servo) in the working command window. After typing, the following window is appeared as shown in Figure 6.

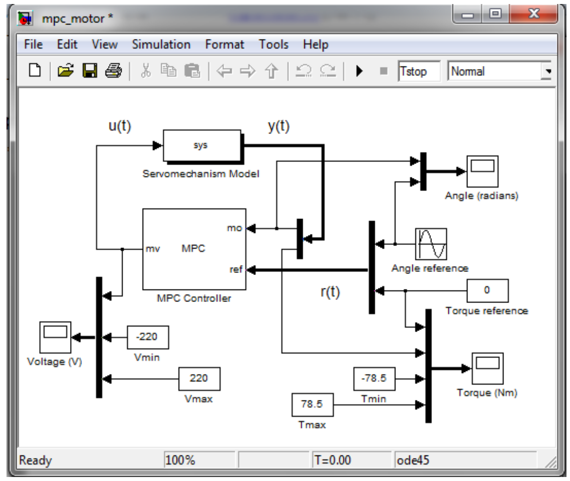

Figure 6. SIMULINK Model of MPC Servo Controller

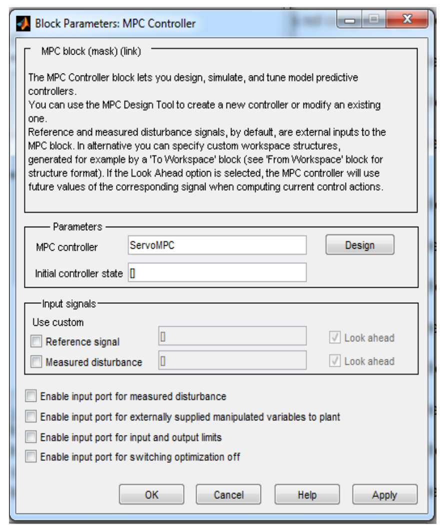

Figure 7. Block Parameters of MPC Controller

MATLAB searches the LTI model of the system in the PWD or the workspace and also search the MPC controller design in mat format or in the symbolic notation in the workspace. This controlled system is non-stiffness and does not require detail result for angular response with time so using ode45 is more suitable choosing.

This mdl model shows MPC Controller Block and LTI model of the servo mechanism. And MPC Controller Block requires exporting the Parameters used in controlled actions. There are free in all type of input signals and initial controller state. Importing by the MPC controller itself and then it becomes to be ready to simulate now. This type of window is shown in Figure 7.

IV. SIMULATION, TEST AND RESULT

Most of the servo mechanism uses simple DC motor and controller as for the input trajectory and output adjustment. Motor used in servo mechanism must be stable and faster response for all input commands. Controlled system used in servo mechanism is closed loop controller such as PID, PD or others. But, nowadays discrete controlled system is introduced and digital devices are widely used.

In this paper, the digital controller namely Model Predictive Controller is used for the motor stable and describe about the simulation of servo motor. Model Predictive Controller (MPC) is depended on the complex calculation of mathematical model. It uses control horizon for the controller movement, prediction horizon for the foretell movement of the actual movement of motor and the weight of the controller output.

Motor that assembles in servo mechanism works normally with the white noise and other disturbance such as load of unknown body that combines to the shaft of the load side. The noise is suppressed by using of MPC controller and repaired the system requirement.

A. Simulation of MPC Using Step Input

The outputs of the controller show the angular movement and torque of the motor after controlling. According to the constraint the angular movement is not over the two pi value. The plots are shown in Figure 8 and 9. These plots give the simple step response for the servomechanism controller. The response is satisfied to controller or not by comparing of the plots. According to the discrete type, the input for the controller must be stairs case but the output of the system is a simple analogue value.

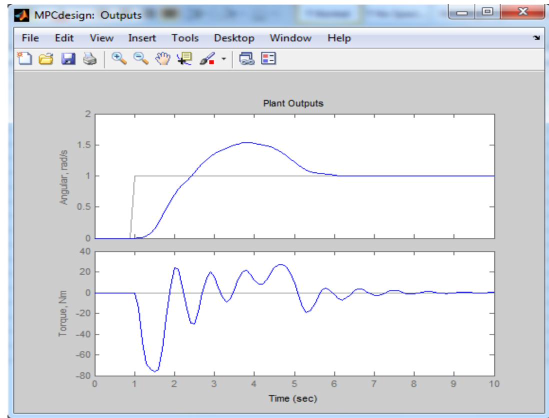

Figure 8. Step Response of Servomotor using MPC Controller

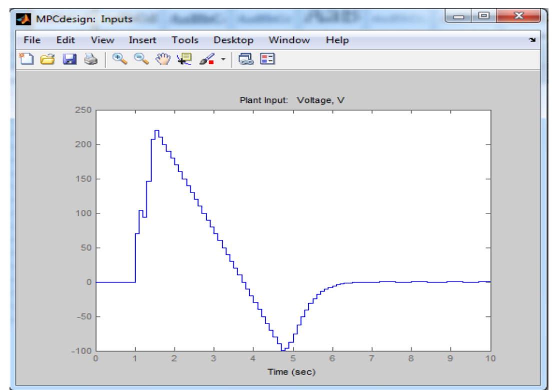

Figure 9. Input Step Response for Servomotor by MPC Controller

B. Simulation of MPC Using Ramp Input

The step response is used in the controller but other type of input methods for simulation of the controller is also required. The step response is compared with ramp response of the servo controller.

The results of the scenario are shown in Figure 10 and 11. The response compares the controller with step and ramp response is much difference. In step response the motor moves within two seconds and the setting time is shorter than the ramp. In ramp signal, the motor moves with long range and not to have the correct angular movement in this case. Also the torque of the motor is more satisfied to compare with step response signal. So the system is designed more suitable for step response. Because this system under controlled is aimed only the angular movement of the motor shaft in correct position in few seconds. If servo motor is designed to use as torque controlled like in CNC machine, this response is more suitable for the servo.

Figure 10 and 11 show the ramp response of the servo motor and input ramp signal for the servomechanism with white noise. The white noises are produced form the integrators of the MPC controller.

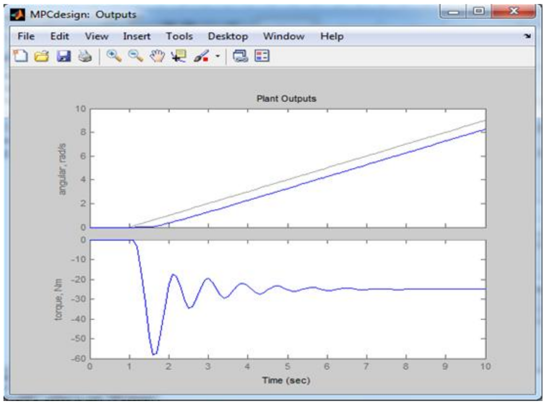

Figure 10. Ramp Response of Servomotor

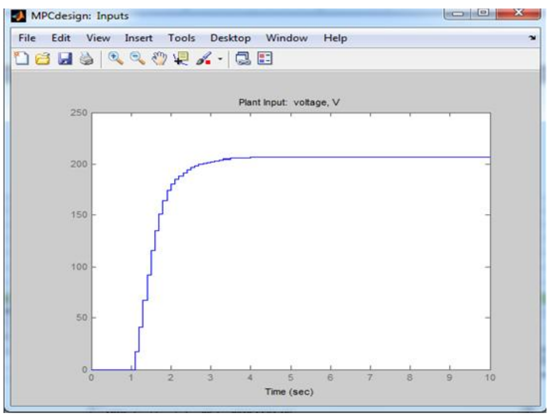

Figure 11. Input Ramp Response of Servomotor by MPC Controller

C. Test Responses of Servo Motor

The scopes of the applied voltage to the motor, angular movement and torque of the motor after controlling is as shown in Figure 12, 13 and 14

In Figure 12 shows about the set point that applied to the controller and the actual movement of the servo shaft. The two lines represent delay time in the controller output to the reference signal. The output of the controller is little to response to the step response but this time is not to be counted in this case. This dead time is acceptable for high power motor with load.

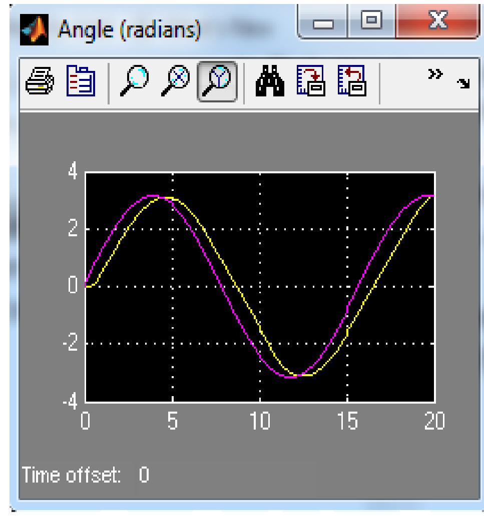

Figure 12. Load Angular Movement Scope of Servomotor

In Figure 13 shows the applied voltage of the controller and that values are bounded by -220V and 220V. But the motor uses actual voltage of about 200 volt for maximum voltage and lowest value for -200 voltages. Assuming the motor is running in full load condition and drawing high current form the main line. Motor spent all of the voltage in initial condition so, the protector is required in real assembling. Under the MPC controller the input is changed into the stairs voltage and then fed to the servomotor. In this thesis, the DC motor is higher voltage for the heavy application and the feeding of HV is as a pulse to the system. All the HVDC applied to the motor will be cost the electrical power and the stair or PWM is the right choice.

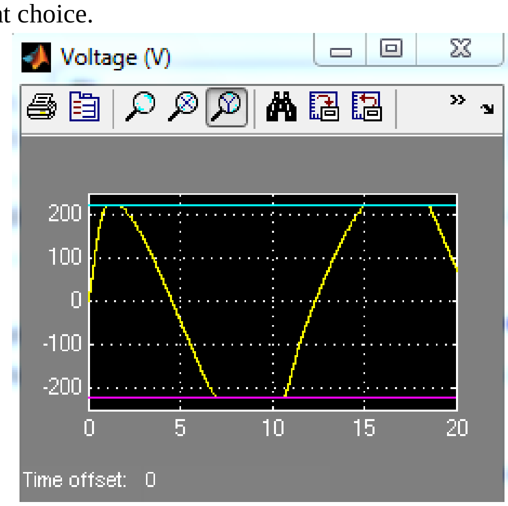

Figure 13. Applied Voltage Scope of Servomotor

.

Figure 14. Torque Scope of Servomotor

In Figure 14 shows torque of the servo motor with controller. Motor is running to the applied voltage that is a step response for the motor and does not require defining the torque in its case. Torque of the motor is required only in the curve filling, carrying object and grinding machine but is not suitable for HVDC servomotor.

V. CONCLUSION

The model based predictive controlled method is used for servo mechanism and it is very strange to compare with traditional controlled methods like PI, PD and PID. The calculation of the MPC is very complex than traditional controller but today computer processing power is very great and can calculate the complex one.

Using the MPC visualization, the servomotor movement can be observed under the correct trajectory of the plant input or not. Simulation of the servomotor is very simple with the help of MATLAB. The state space matrix value must be imported in the present working directory and then calling the MPCTOOL for finding the prediction horizon, control horizon and output weigh of the controller first. The target movement is obtained or not by using the observation of that GUI window. If the result is satisfied, the control horizon, prediction horizon and weight tuning values are set in the MATLAB workspace. Then these values are changed into the m-file for MPC calculation. This m-file is used by THE SIMULINK block of the MPCTOOL BOX.

ACKNOWLEDGMENT

The author would like to thank all persons who involved towards the successful completion for this research work.

REFERENCES

[1] www.wikipedia, Model Predictive Controller history and other relative data for position control

[2] Riazollah Firoozian, Servomotor and Controlled Theory, 2009.

[3] Eduardo F. Camacho and Carlos Bordons, Model Predictive Controlled Advance Text book, 1999.

[4] Roland S. Burns, Advance Controlled Engineering, 2001.

[5] Katsuhiko Ogata, Modern Controlled Engineering Third Edition, 2000.

[6] Dimitry Gorinevsky, Controlled Engineering EE392m course, algorithm and analysis of the system.

[7] James R. Carstens, Automatic Controlled System and Components, 1990.

[8] P.N.P Paraskevopoulos, Modern Control Engineering, 2002.

[9] Model Predictive Controller Algorithm in MATLAB by Department of Electronic and Communication Engineering, National Institutes of Technology.

[10] www.mark.cannon@eng.ox.cc.uk for MPC controller.

[11] http://www.mathworks.inc, Model Predictive Controller guide line for the servo control method.