Simulation and Implementation of Servo Motor Control with Sliding Mode Control (SMC) using Matlab and LabView

Bondhan Novandy

http://bono02.wordpress.com/
Outline

• Background
• Motivation
• AC Servo Motor
• Inverter & Controller
• Hardware Block Diagram
• Mathematical Modeling
• Sliding Mode Controller Design
• Simulation using Matlab
• Simulation & Implementation using LabView
• Several Ideas
Motivation

• Sliding Mode Control is a robust control scheme based on the concept of changing the structure of the controller in response to the changing state of the system in order to obtain a desired response
• The biggest advantage of SMC is its insensitivity to variation in system parameters, external disturbances and modeling errors
• This can be achieved by forcing the state trajectory of the plant to the desired surface and maintain the plants state trajectory on this surface for subsequent time
• Because of these factors SMC is chosen as the controller for our device
AC Servo Motor

- The difference between AC Servo Motor and DC servo motor is the design of the motor where in AC motor the permanent magnet is on the rotor. The block diagram of an AC servo motor is very similar to the block diagram of DC servo motor:

\[
R = \text{per phase resistance} \\
\quad \text{(phase to phase resistance} = 2 \ R) \\
L = \text{per phase inductance} \\
\quad \text{(phase to phase inductance} = \sqrt{3} \ L) \\
V_{UV} = \text{Phase to Phase Voltage} = \sqrt{3} \ V_{UN} \\
E_g = n_m \ K_{Eo} \quad \text{BEMF} \\
n_m = \text{motor speed} \\
T_m = \text{motor torque} \\
J_m = \text{motor moment of inertia}
\]
AC Servo Motor

- Permanent magnets on the rotor create a field vector that rotates synchronously with the rotor of the motor.
- Composite current vector is located perpendicular to the field vector at all times by locking the angular frequency of the three-phase stator currents to the properly defined rotor angle $\delta$.
- Torque is then directly proportional to the amplitude of the three-phase sinusoidal currents.

\[ T_M = K_T I_T \]

AC Servo System

- Field vector is fixed in space by the stationary permanent magnets.
- Current vector is located perpendicular to the field vector by proper location of the brushes on the commutator.
- Torque is then directly proportional to the armature current.

\[ T_M = K_T I_A \]

DC Servo System
Inverter and Controller

- Inverter is used to transform electricity from 1 single phase into 3 phase
- It works by controlling the rotational speed of an AC motor by controlling the frequency of the electrical power supplied to the motor
- Our inverter the we use the linear V/F mode

Output max voltage is equal to max input voltage = 220 V RMS

The default frequency is 50 Hz
Inverter and Controller Cont’d

- Analog input voltage 0-10 V is used to change the volt/frequency of the inverter output.
- The scale between input : output = 10 : 220 depending on the configuration of V/F.
- NI-PXI 7358 is chosen as the controller and it has DAC 0-10 output.
Hardware Block Diagram

PXI-7358 → Inverter → AC Servo Motor

- DAC 0-10 V × 22
- 0-220V/0-50 Hz

Encoder

Position, Speed, Acceleration
Mathematical Modeling

Position control:

\[ I_a \cdot R_a + L_a \cdot \frac{dI}{dt} + K_b \cdot \frac{d\theta}{dt} = V \quad \ldots (1) \]

\[ T - T_{Load} = K_t \cdot I_a - T_{Load} = T_j + T_b = J \frac{d^2 \theta}{dt^2} + B \frac{d\theta}{dt} \quad \ldots (2) \]

\[ I_a = \frac{1}{K_t} \left( J \frac{d^2 \theta}{dt^2} + B \frac{d\theta}{dt} + T_{Load} \right) \quad \ldots (3) \]
(3) → (1)

\[
\frac{Ra}{Kt} \left( J \frac{d^2\theta}{dt^2} + B \frac{d\theta}{dt} + T_{Load} \right) + \frac{La}{Kt} \left( J \frac{d^2\theta}{dt^2} + B \frac{d\theta}{dt} + T_{Load} \right) + Kb \frac{d\theta}{dt} = V
\]

\[
\frac{J * Ra}{Kt} \frac{d^2\theta}{dt^2} + \frac{B * Ra}{Kt} \frac{d\theta}{dt} + \frac{Ra}{Kt} T_{Load} + \frac{J * La}{Kt} \frac{d^3\theta}{dt^3} + \frac{B * La}{Kt} \frac{d^2\theta}{dt^2} + \frac{La}{Kt} T'_{Load} + Kb \frac{d\theta}{dt} = V
\]

Finally we get (4):

\[
\dot{\theta}_1 = \theta_2
\]

\[
\dot{\theta}_2 = \theta_3
\]

\[
\dot{\theta}_3 = \frac{Kt}{J * La} V - \frac{B}{J} \theta_3 - \frac{Ra}{La} \theta_3 - \frac{Ra * B}{La * J} \theta_2 - \frac{Kb * Kt}{J Kt} \theta_2 - \frac{1}{J Kt} T'_{Load} - \frac{Ra}{La * J} T_{Load}
\]
Sliding Surface

• We have 3\textsuperscript{rd} order system

\[
s = \left(\frac{d}{dt} + \lambda\right)^{n-1} \tilde{\theta}, \text{ choosing } n=3 \text{ and } \tilde{\theta} = \theta - \theta_d \text{ is tracking error}
\]

\[
= \left(\frac{d}{dt} + \lambda\right)^2 \tilde{\theta}
\]

\[
= \left(\frac{d^2 \tilde{\theta}}{dt^2} + 2\lambda \frac{d \tilde{\theta}}{dt} + \lambda^2 \tilde{\theta}\right)
\]

\[
= \ddot{\theta} + 2\lambda \dot{\theta} + \lambda^2 \tilde{\theta}
\]

\[
= (\ddot{\theta} - \ddot{\theta}_d) + 2\lambda (\dot{\theta} - \dot{\theta}_d) + \lambda^2 (\theta - \theta_d)
\]
Equivalent Control

• We try to force the state trajectory to slide on our surface so that:  
\[ s = 0 \]
\[ \dot{s} = 0 \]

\[ \dot{s} = (\ddot{\theta} - \ddot{\theta}_d) + 2\lambda(\dot{\theta} - \dot{\theta}_d) + \lambda^2(\dot{\theta} - \dot{\theta}_d) \]

\[ 0 = \frac{Kt}{J \cdot La} V - \frac{B}{J} \theta_3 - \frac{Ra}{La} \theta_3 - \frac{Ra \cdot B}{La \cdot J} \theta_2 - \frac{Kb \cdot Kt}{La \cdot J} \theta_2 - \frac{1}{J \cdot Kt} T'_{Load} \]

\[ -\frac{Ra}{La \cdot J} T_{Load} - \ddot{\theta}_d + 2\lambda(\theta_3 - \ddot{\theta}_d) + \lambda^2(\theta_2 - \dot{\theta}_d) \]

\[ V_{eq} = \frac{B \cdot La}{Kt} \theta_3 + \frac{La}{Kt} T'_{Load} + \frac{Ra \cdot J}{Kt} \theta_3 + \frac{Ra \cdot B}{Kt} \theta_2 + \frac{Ra}{Kt} T_{Load} + Kb \cdot \theta_2 + \frac{La \cdot J}{Kt} \ddot{\theta}_d \]

\[ -2\lambda \frac{La \cdot J}{Kt} (\theta_3 - \ddot{\theta}_d) - \lambda^2 \frac{La \cdot J}{Kt} (\theta_2 - \dot{\theta}_d) \]
SMC Controller

\[ V = V_{eq} + V_{\text{switching}}, \text{where } V_{\text{switching}} = -K \ast \text{sat}(s / \phi) \]

and sat is the function:

\[ \text{sign}(s) \quad \text{if } \text{abs}(s) > \phi \]

\[ s / \phi \quad \text{if } \text{abs}(s) < \phi \]

\[ V_{eq} = \frac{B \ast La}{Kt} \theta_3 + \frac{La}{Kt} T_{\text{load}}' + \frac{Ra \ast J}{Kt} \theta_3 + \frac{Ra \ast B}{Kt} \theta_2 + \frac{Ra}{Kt} T_{\text{load}} + Kb \ast \theta_2 + \frac{La \ast J}{Kt} \dot{\theta}_d \]

\[ -2 \lambda \frac{La \ast J}{Kt} (\theta_3 - \dot{\theta}_d) - \lambda^2 \frac{La \ast J}{Kt} (\theta_2 - \dot{\theta}_d) - K \ast \text{sat}(s / \phi) \]
Lyapunov Function

If there exist Lyapunov function, so that

1) \( V(x) > 0 \) \quad \forall |x| < r

2) \( \dot{V}(x) = \nabla V^T(x) f(x) \leq 0 \)

for all \( |x| < r \)

It is stable in the sense of Lyapunov
Stability

We ensure the stability of our system choosing $K$ to be large enough so that stable in the sense of Lyapunov.

Lyapunov candidate:

$$V = \frac{1}{2} s^2 \quad > 0$$

$$\dot{V} = s \ddot{s} = s\left(\frac{Kt}{J \cdot La} V - \frac{B}{J} \dot{\theta}_3 - \frac{Ra}{La} \theta_3 - \frac{Ra \cdot B}{La \cdot J} \theta_2 - \frac{Kb \cdot Kt}{La \cdot J} \theta_2 - \frac{1}{J \cdot Kt} T_{Load}' \right)$$

$$- \frac{Ra}{La \cdot J} \ddot{T}_{Load} - \ddot{\theta}_d + 2\lambda (\dot{\theta}_3 - \dot{\theta}_d) + \lambda^2 (\dot{\theta}_2 - \dot{\theta}_3) - K \cdot \text{sat}(s / \phi))$$

$$\dot{V} = s(f - K \cdot \text{sat}(s / \phi))$$

$$= f \cdot s - K |s| \quad \leq 0$$
Simulation Using Matlab

Plant parameter:

\[ j=0.01; \]
\[ b=0.1; \]
\[ Kt=0.01; \]
\[ Kb=0.29098; \]
\[ Ra=1; \]
\[ La=0.5; \]

The unity feedback transfer function is:

\[ \frac{2}{s^2 + 12s + 22.58} \]
Transient Response Without Controller

Stability Analysis (Linear System)
Controllability:
% Number of uncontrollable states
>> unco = length(A)-rank(ctrb(A,B))
unco =
0
Observability:
% Number of unobservable states
unob = length(A)-rank(Ob)
unob =
0
Thus the system is stable.
Simulation Result With SMC Controller
Step Response

$K = 50; \quad \text{Lambda} = 10; \quad \text{delta} = 0.5;$
Step Response
\[ K = 100; \quad \text{Lambda} = 10; \quad \text{delta} = 0.9; \]
Trajectory Following

The trajectory is defined so that it will not produce shock while moving because of discontinuity.

Quintic polynomial:

\[ q_0 = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5 \]

where:

- \( q_0 \) = start position
- \( q_f \) = final position
- \( v_0 \) = start velocity
- \( v_f \) = final velocity
- \( \alpha_0 \) = start acceleration
- \( \alpha_f \) = final acceleration
Quintic polynomial trajectory for position, velocity and acceleration
Trajectory Following Response
K = 80; Lambda = 10; delta = 0.9;
Simulation Using LabView

Trajectory Generator
Simulation Using LabView
Step Response
Implementation

• Implementation in NI Real-Time can be done by replacing the runge-kutta ODE solver and array composition with the real motor which is the SERVO Motor

• The feedback of the motor must be available (encoder)
Problems and Discussion

• Real-time clock generation -> Sampling time
• Motor parameters
• The feedback of the system are:
  – Position
  – Velocity
  – Acceleration
*) In the simulation when acceleration feedback was defined as 0, the error is still small

• Integral sliding mode control
\[ s = (\frac{d}{dt} + \lambda)^{n-1} \int_{0}^{t} \tilde{\theta}, \text{ choosing } n=4 \text{ and } \tilde{\theta} = \theta - \theta_d \text{ is tracking error} \]