On-Line System ID and Adaptive Optimal Control on a Linear Motor

Ruikun Yu[1] , Bo Ji[2]

[1,2]Mechanical Engineering Department, Purdue University,
610 Purdue Mall, West Lafayette, IN, US

[1]yu207@purdue.edu, [2]ji52@purdue.edu

ABSTRACT:

The paper is about using optimal control and adaptive control to control a linear motor which can follow a predefined trajectory by minimize the input power. Model Predictive Control (MPC) with constrains on the input power as the optimal control strategies to generate the minimum cost. Recursive Least Square Estimation (RLSE) works as the parameters adaption to estimate and update the system model. The combination of optimal control and adaptive control make the system more robust and efficient.

1. INTRODUCTION

The basic function of linear motor is to convert the electrical energy into the linear or rotary motions. Sometimes, the task or purpose of linear motor need to drive the end effectors to move a predefined trajectory. In order to follow the trajectory, optimal control strategies are used to generate the minimum control input to drive the linear motor to follow the trajectory accurately.

Recursive least square estimation (RLSE) is an adaptive estimation method that can recursively estimate the required parameters while minimizing the cost function. Assuming that the physical properties of the linear motor changes after long time usage. This is one of main constrains need to be overcome. Normal controller can't maintain the performances after long time usage, so RLSE works as the parameter adaption to update the new parameters of system to controller.

Model Predictive Control (MPC) is an optimal control method to help control a system or a process by meeting some predefined constrains. The biggest advantage of the MPC is that it can optimize the input and output of current timeslot by taking the future outputs into consideration. MPC is able to control the system accordingly based on the future events. An input constraint or saturation needs to be implemented on the MPC to help protect the motor.

2. ON-LINE SYSTEM IDENTIFICATION

2.1 System Modeling

Let x_1 and $x_2$represents position and velocity of the linear motor, so the governing equations of linear motor can be expressed as:

$$\dot{x_1} = x_2$$

$$\mathrm{M}\dot{x_2} = -Bx_2 + F_{sc}S_f(x_2) + d(t) + k_f u \tag{1}$$

Where M is the inertial of the motor, k_f and u stand for the input voltage and input gain, B and F_{sc} represent the viscous coefficient and coulomb coefficient,$d(t)$ represents the external disturbance and nonlinear friction force, like the cogging force.[1] In this paper, only linear optimal control systems will be studied, nonlinear parts in equation 1 are neglected. New equation 1 can be rewritten as:

$$\dot{x}_2 = -\frac{B}{M}x_2 + \frac{K_f}{M}u \tag{2}$$

So, the state space model of linear motor is shown below. Based on the purpose of linear motor mentioned in introduction, the linear motor need to drive the end effector to follow a predefined trajectory. The focused output is the position of the linear motor.

$$\begin{bmatrix}\dot{x_1}(t)\\ \dot{x_2}(t)\end{bmatrix} = \begin{bmatrix}0 & 1\\ 0 & -\frac{B}{M}\end{bmatrix}\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix} + \begin{bmatrix}0\\ \frac{K_f}{M}\end{bmatrix}u(t)$$

$$y(t) = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix} \tag{3}$$

Let's start with the numerical example of a linear motor with $J_M = 0.01kg * m^2$, $k_T = 0.01\ N * m/Amp$, and$B_M = 0.1\ N * m * s$, $K_f = 1$.so the transfer function of linear motor in continuous domain can be expressed as:

$$G_p(s) = \frac{1}{s^2+10s} \tag{4}$$

2.2 System Identification

In order to do the recursive least square estimation, the plant transfer function need to be transferred from continuous time domain to discrete time domain. Also the plant transfer function in discrete domain should be expressed in the following form:

$$G(z^{-1}) = \frac{z^{-1}B(z^{-1})}{A(z^{-1})}$$

$$G_p(z^{-1}) = \frac{z^{-1}(4.837e{-}05+4.769e{-}05z^{-1})}{1-1{,}905z^{-1}+0.9048z^{-2}} \tag{5}$$

So the input ($u(k)$) and output ($y(k)$) relationship can be expressed as

$$u(k)\frac{B(z^{-1})}{A(z^{-1})} = y(k+1) \tag{6}$$

Where:

$$B(z^{-1}) = b_0 + b_1z^{-1} + \cdots + b_mz^{-m}$$

$$A(z^{-1}) = 1 + a_1z^{-1} + \cdots + a_nz^{-n} \tag{7}$$

So

$$y(k+1) = -\sum_{i=1}^{n} a_i y\big(k-(i-1)\big) + \sum_{i=0}^{m} b_i u(k-i) \tag{8}$$

Estimating the coefficients ($a_1, a_2 \dots a_n, b_0, b_1 \dots b_m$) of the discrete model:

$$\alpha = [-a_1, -a_2 \dots a_n, b_0, b_1 \dots b_m]^T$$

$$x(k) = [-y(k), -y(k-1), \dots - y(k-n+1), u(k), u(k-1), \dots u(k-m)]^T \tag{9}$$

So, the output can be simply demonstrated in equation 10.

$$y(k+1) = \alpha^T x(k) \tag{10}$$

The estimation of system output can be expressed as:

$$\hat{y}(k+1) = \hat{\alpha}(k)^T x(k) \tag{11}$$

The least square method should generate α to help minimize the cost function (J), which is caused by the difference between the actual output of simulated system and output of estimated system [2]. Assume time equal to K, the cost function can be expressed as:

$$J_k = \sum_{i=1}^{k} [y(i) - \hat{\alpha}(k)^T x(i-1)]^{\,2} \tag{12}$$

By letting $\frac{dJ_k}{dx(k)} = 0$, the inverse error covariance (K) can be derived as

$$K(k+1)^{-1} = K(k)^{-1} + x(k+1)^T R(k+1)^{-1} x(k+1) \tag{13}$$

And the update parameters can be derived as[2]:

$$\hat{\alpha}(k+1) = \hat{\alpha}(k) + K(k+1)x(k)^T R(k+1)^{-1}(y(k+1) - x(k+1)\hat{\alpha}(k)) \tag{14}$$

In order to compute the actual outputs of the system to help estimate the parameters of system, a square wave voltage source with an amplitude at 3 V and frequency at 1Hz, without any offset is inputted into the simulated system. Assuming that the parameters of system change after long time usage and there are some uncertainties when measuring the outputs, so the parameters of the simulated system is necessary to be estimated based on the history of inputs and outputs. During the estimation, the effects coming from the initial value of error covariance ($K(0)$) need to be considered. Three different initial values of error covariance are chosen, which are $K_i = 0.01, K_i = 0.1$ and $K_i = 1$. Figure 1 shown below are results of estimation by setting different K_i. As K_i increase, the accuracy of the estimation also increase. The estimated system with K_i at 1 and the actual simulated system almost have the same trajectory. The errors between two systems are really small.

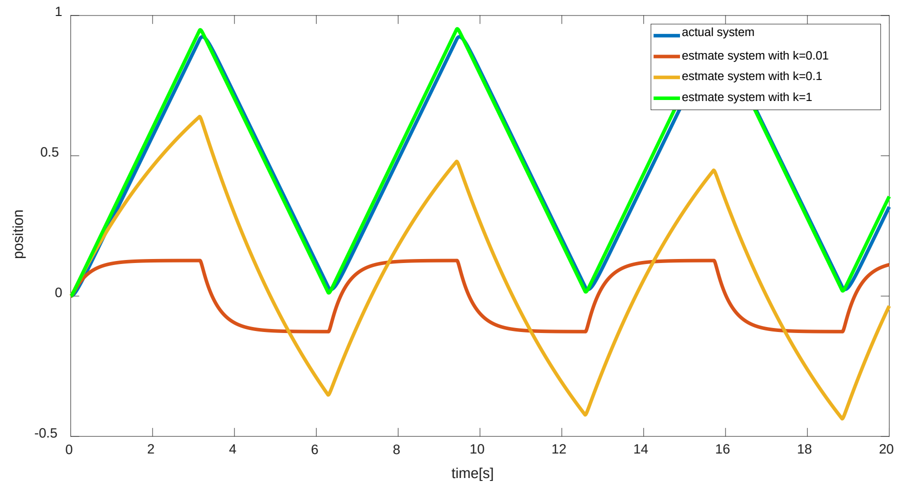

Figure 1. *The trajectories of actual simulated system and three estimated systems*

Base on figure 2 shown below, the larger value of K_i can make the estimation converge faster, which helps estimate more accurately. While, the system with K_i equals 0.01 and 0.1 can't estimate system accurately, because the convergence speed of estimation can be fast enough to generate right parameters. In figure 2, when K_i equal to 1, the convergence process finish at 200 samples. While when K_i equals to 0.1 and 0.01, the convergence process finishes at 400 and 1000 samples. Figure 3 demonstrates the convergence process of the four parameters with steady value. The estimated parameters converge to $[-1.65, 0.65, 2.587e-04, 9.4043e-05]^T$ correspond to$[a_1, a_2, b_0, b_1]^T$.

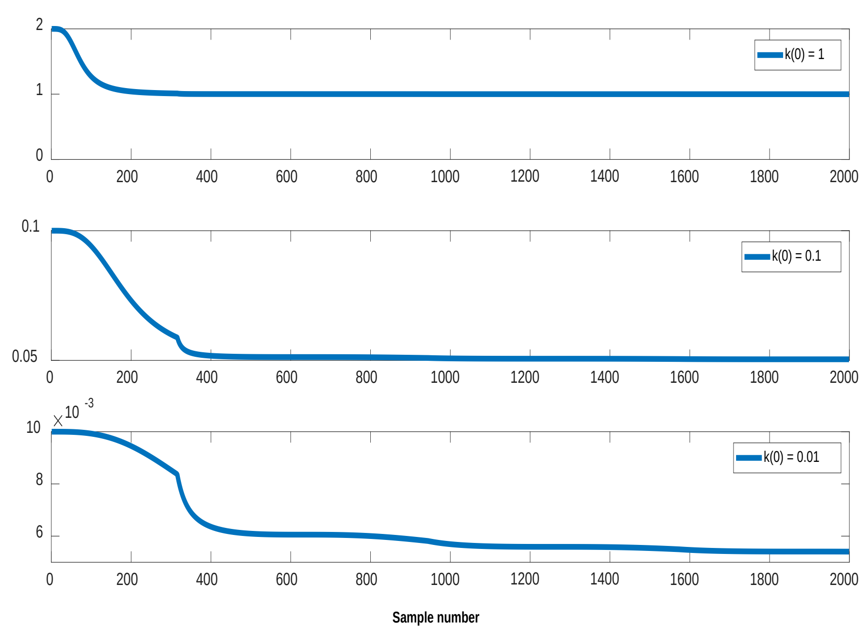

Figure 2. *The convergence speed of error covariance with different initial error covariance*

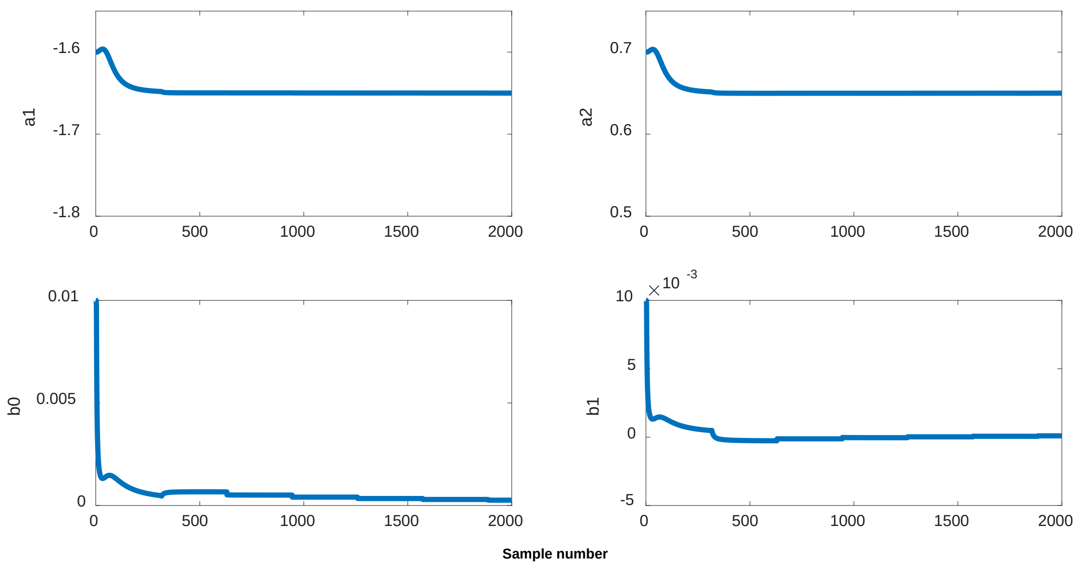

Figure 3. *The convergence of four parameters when K(0)=1*

3. CONTROL STRATEGIES

The overall control scheme is consisted of two major parts: on-line parameter adaptation and MPC with optimization under constraints. The following block diagram shows the control system structure.

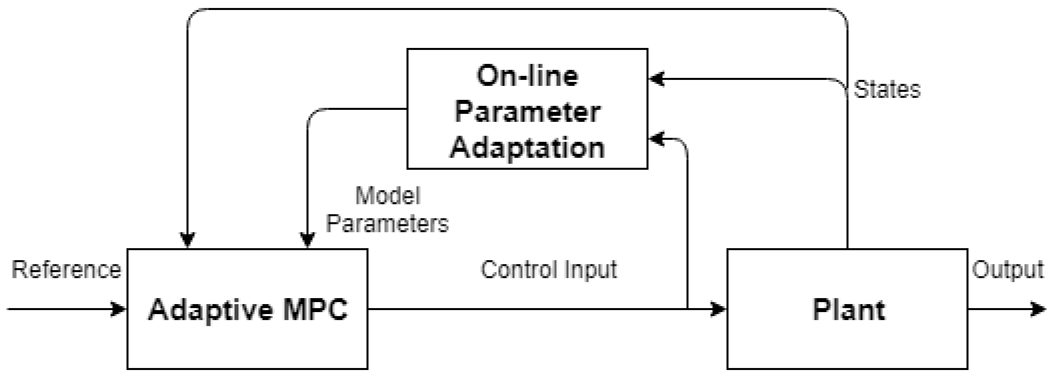

Figure 4. *Block diagram of the overall system*

As discussed in previous section, the on-line parameter adaptation is done by a Recursive Least Square Estimator. The estimator keeps taking in control input and states measurements, and outputting adapted model parameters based on the knowledge of the system model structure [3]. Then the adapted model parameters are passed into the adaptive MPC module, which is depicted in the following block diagram.

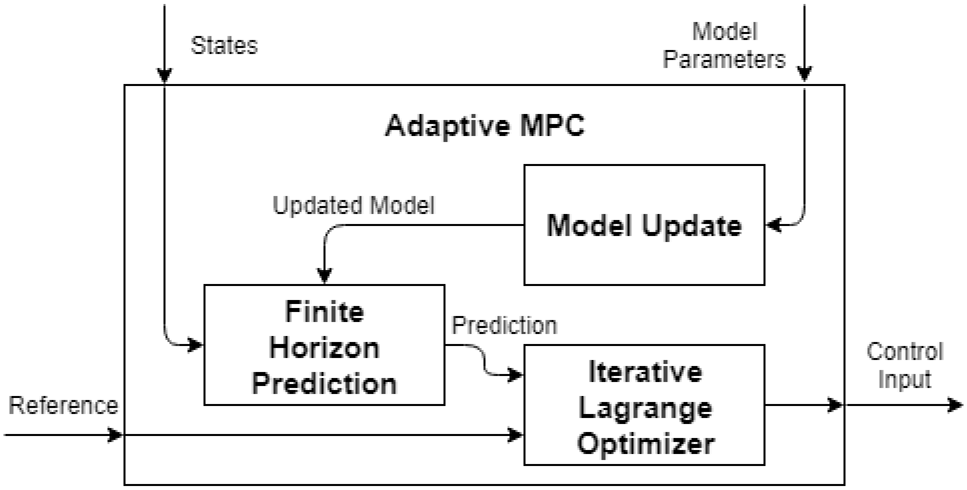

Figure 5. *Block diagram of the adaptive MPC module*

4.1 Model Update

With the newly updated model parameters, we will get an updated model for each control iteration through the Model Update module:

$$\alpha(k) \Rightarrow \hat{A}, \hat{B}, \hat{C}$$

$$x(k+1) = \hat{A}_{n\times n}x(k) + \hat{B}_{n\times m}u(k) \tag{15}$$

$$y(k) = \hat{C}_{p\times n}x(k) \tag{16}$$

4.2 Finite Horizon Prediction

Prediction will be made based on the updated model in the adaptive MPC:

$$Y = Wx(k) + ZU \tag{17}$$

Where:

$$Y = finite\ horizon\ (N)\ prediction = \begin{bmatrix} y(k+1|k) \\ y(k+2|k) \\ \vdots \\ y(k+N|k) \end{bmatrix}$$

$$U = predicted\ control\ input\ sequence = \begin{bmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k+N-1) \end{bmatrix}$$

$$W = \begin{bmatrix} \hat{C}\hat{A} \\ \hat{C}\hat{A}^2 \\ \vdots \\ \hat{C}\hat{A}^N \end{bmatrix}, Z = \begin{bmatrix} \hat{C}\hat{B} & 0 & \cdots & 0 \\ \hat{C}\hat{A}\hat{B} & \hat{C}\hat{B} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \hat{C}\hat{A}^{N-1}\hat{B} & \hat{C}\hat{A}^{N-2}\hat{B} & \cdots & \hat{C}\hat{B} \end{bmatrix}$$

By constructing the control cost function and applying the FONC:

$$J(U) = \frac{1}{2}\left(ref_{k,N} - Y\right)^T \bar{Q}\left(ref_{k,N} - Y\right) + \frac{1}{2}U^T \bar{R} U \tag{18}$$

$$\frac{\partial J_N(U)}{\partial U} = -\left(ref_{k,N} - Wx(k) - ZU\right)^T \bar{Q} Z + U^T \bar{R} = O \tag{19}$$

The optimal control sequence is obtained:

$$U^* = (R + Z^T \bar{Q} Z)^{-1} Z^T \bar{Q}(ref_{k,N} - Wx(k)) \tag{20}$$

Where:

$$ref_{k,N} = \begin{bmatrix} ref(k+1) \\ ref(k+2) \\ \vdots \\ ref(k+N) \end{bmatrix}, \bar{Q} = QI_{N\times p}, \bar{R} = RI_{N\times m}$$

4.3 Iterative Lagrange Optimization

Then this optimal control sequence is passed into the Iterative Lagrange Optimizer for optimization under constraints:

$$U_{min} < U < U_{max}$$

$$\begin{bmatrix} -U \\ U \end{bmatrix} < \begin{bmatrix} -U_{min} \\ U_{max} \end{bmatrix}$$

$$\begin{bmatrix} -I_m & O & \cdots & O \\ I_m & O & \cdots & O \\ O & -I_m & \cdots & O \\ O & I_m & \cdots & O \\ \vdots & \vdots & \ddots & \vdots \\ O & O & \cdots & -I_m \\ O & O & \cdots & I_m \end{bmatrix} U - \begin{bmatrix} -u_{min} \\ u_{max} \\ -u_{min} \\ u_{max} \\ \vdots \\ -u_{min} \\ u_{max} \end{bmatrix} \leq 0$$

$$g(U) = A_{con} U - B_{con} \leq 0$$

Thus we have formed the optimization problem:

$$\min_U J(U)\,, subject\ to\ g(U) \leq 0 \tag{21}$$

Although there are other numerical optimization methods such as Gradient Descent and Newton’s Method, here we choose to use the Iterative Lagrange Optimization [4].

$$U_{n+1}^* = U_n^* - \alpha[\nabla J(U_n^*) + Dg(U_n^*)^T \mu_n] \tag{22}$$

$$\mu_{n+1} = \mu_n + \beta g(U_n^*) \tag{23}$$

As the optimizer keeps iterating, the control sequence will approach the optimal value while satisfying the control input constraints.

$$u(k) = [I_m \quad 0 \quad \cdots \quad 0]_N U_{final}^* \tag{24}$$

4. RESULTS AND EVALUATION

The performance of three control strategies has been evaluated via simulation. Various assumptions and limitations have been established in order to carry out the results.

- In all three cases, the system is tracking a same sinusoidal trajectory
- Full state feedback, but with measurement noise (white noise)
- A [-5,5] control input saturation is imposed

In order to illustrate the performance, the trajectory following, tracking error, and control input are plotted for each control strategy.

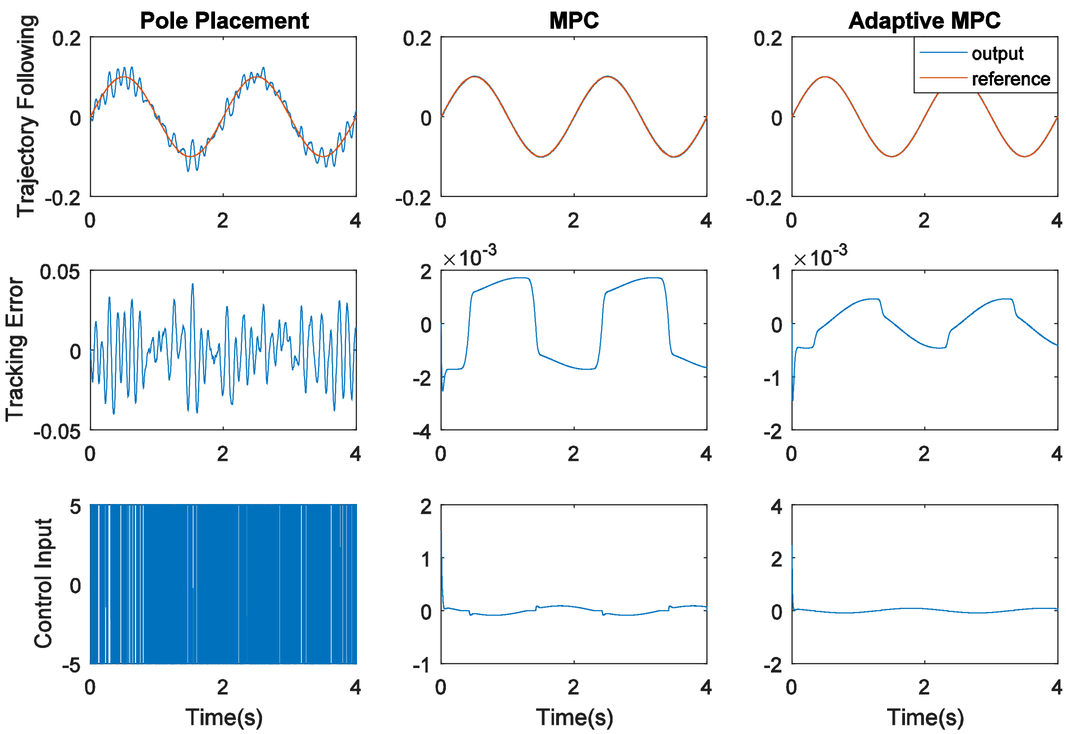

Figure 6. *Plots of trajectory following, tracking error, and control input for 3 control strategies*

From above plots, it can be easily observed that both MPC strategies have a better trajectory following and a much cleaner control input profile under saturation. In addition, the adaptive MPC has a slightly smaller tracking error range, meaning that it has the best trajectory tracking performance out of the 3 strategies.

To further evaluate the advantages and disadvantages of the adaptive MPC, the control cost and computation time per sample have been plotted.

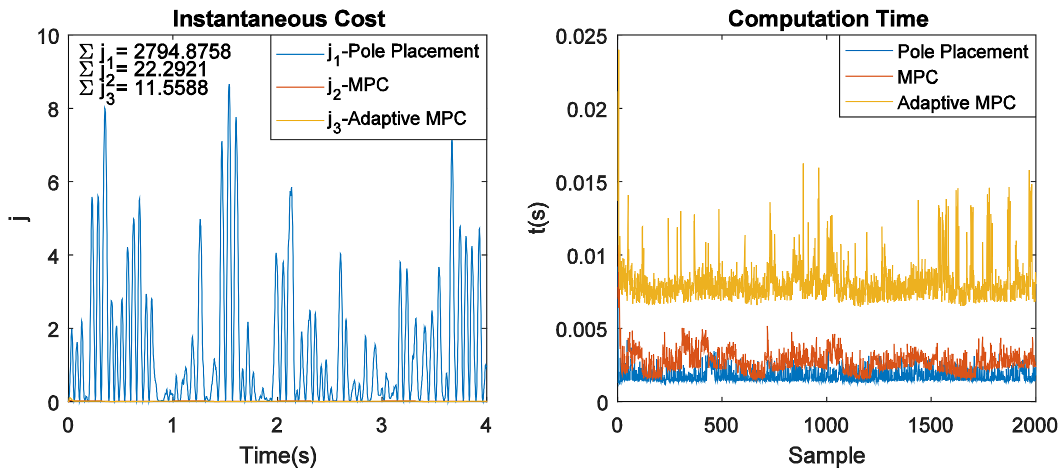

Figure 7. *Plots of trajectory following, tracking error, and control input for 3 control strategies*

It can be seen that the control cost of pole placement strategy in this simulation is significantly larger than that of other two approaches. In fact, summing up the instantaneous control cost for each control will give a more direct view of the difference.

$$J_{Pole\ Placement} \approx 2795.6$$

$$J_{MPC} \approx 22.3$$

$$J_{Adaptive\ MPC} \approx 11.6$$

The computation time per each control iteration for all three strategies are plotted in the Computation Time plot. It is very obvious that the complex algorithms in adaptive MPC brings a significant amount of computation power consumption. While the computation time per sample for pole placement and MPC are fairly low and close to each other, that for adaptive MPC is almost three time larger at a average level.

5. CONCLUSION

In this project, the adaptive MPC have been formulated for controlling a linear motor with time-varying model parameters. The controller adapts changing parameter and optimizes its control input under constraints. Other two control strategies have been formulated as contrasts to the proposed strategy. The performance of the adaptive MPC has been evaluated in simulation. Cross comparison between the performances of different controls have been made and advantages and disadvantages of the proposed adaptive MPC have been concluded. The adaptation of the adaptive MPC makes sure that the controller follows the model closely and the MPC optimization ensures that the control input is the optimum under constraints. However, due to the algorithm complexity of both adaptation and iterative optimization, the adaptive MPC consumes a great amount of computation power for every control iteration. Overall, it can be concluded that the adaptive MPC is a powerful but costly control strategy.

6. REFERENCES

[1] Yun Hong, Bin Yao, 2007, *A Globally Stable High-Performance Adaptive Robust Control Algorithm With Input Saturation for Precision Motion Control of Linear Motor Driver Systems*, IEEE/ASME Transactions on Mechatronics, VOL 12, No 2.

[2] Saher Arshad, Saima Qamar,Tayba Jabbar, Ahsan Malik, 2010, *Parameter Estimation of a DC Motor using Ordinary Least Square and Recursive Least Square Algorithms*.

[3] Åström, K. J., & Wittenmark, B. (2008). *Adaptive control*. Mineola, NY: Dover.

[4] Żak, S. (2002). *Systems and Control*. The Oxford Series in Electrical and Computer Engineering.