UNIVERSITY OF ALBERTA

ABHI CHAUHAN

Student ID - 1811065

OPTIMAL & MODEL PREDICTIVE CONTROL

CHE 663

FACULTY OF ENGINEERING
UNIVERSITY OF ALBERTA

Project Guided by
Jinfeng Liu

Submitted on
15th December, 2023

UNIVERSITY OF ALBERTA

Table of Contents

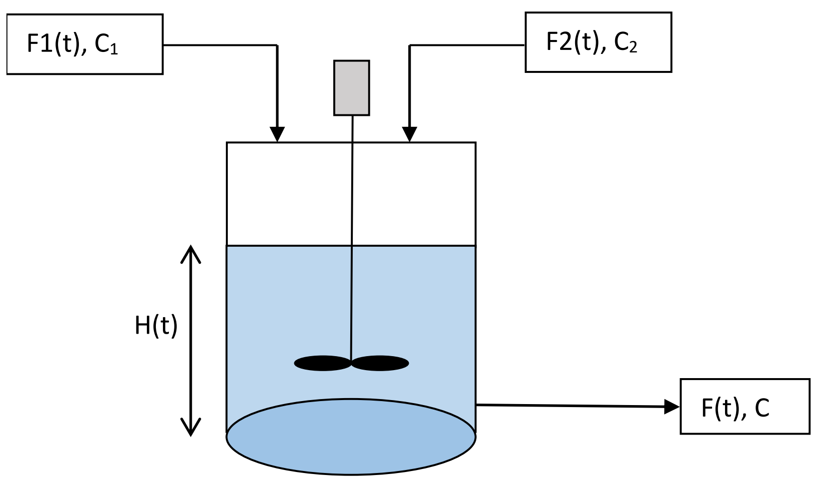
Table of Figures

1. Project Description: -

1.1 Stirred tank Mixing Process:

For the project, Continuous stirred tank mixing process with two inlet F_1 and F_2 with the different concentration C_1 and C_2 has been taken for modeling and optimization. For the system, I have taken the assumption that CSTR has well mixed as well as the outlet concentration (C) at output stream (F) is same as the tank concentration. Another, assumption is that density of the given system is constant. For given time volume of tank is V(t) and corresponding height H(t). The outlet floe of the tank is defined as

$$\boldsymbol{F = k\sqrt{H}} \tag{1.1}$$



For the given system, we have taken following variables

1. State Variable
   - Outlet concentration C(t)
   - Height of the Tank H(t)
2. Input Variable or Manipulative Variable
   - Inlet Flow F_1 (Having concentration of C_1)
   - Inlet Flow F_2 (Having concentration of C_2)

1.2 Control Objective

The aim of the project is to control the outlet concentration while maintain the height of the tank with respect to the change in the inlets flow F_1 and F_2 as well as the inlets stream concentration C_1 and C_2 respectively

To begin with let's start design the model of the system.

Applying Basic mass transfer equation on the system

Mass In – Mass out = Accumulation + ~~Mass generation~~

Here there is no generation of the mass in the system. Hence, this term become zero

$$\frac{dV}{dt} = F_1 + F_2 - F$$

$$\frac{dH}{dt} = \frac{1}{A}(F_1 + F_2 - k\sqrt{H})$$

$$\frac{d(V.C)}{dt} = F_1.C_1 + F_2.C_2 - F.C$$

$$\frac{C.dV}{dt} + \frac{V.dC}{dt} = F_1.C_1 + F_2.C_2 - k\sqrt{H}.C$$

$$\frac{dC}{dt} = \frac{1}{A.H}(F_1.(C_1 - C) + F_2.(C_2 - C)$$

From the Equation above equation State and input variables define as

X(t) = $\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} C(t) \\ H(t) \end{bmatrix}$

u(t) = $\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} F_1(t) \\ F_2(t) \end{bmatrix}$

Optimal control problem can be defined as mentioned below

$$J = \frac{1}{2}\left(X^T(t_f).P.X(t_f)\right) + \frac{1}{2}\int_t^{t_f}[X^T(t_f).Q.X(t) + u^T(t_f).R.u(t)]dt$$

To solve the control problem mentioned above Linear Quadratic Regulator (LQR) is used after linearizing the state equation at steady state condition. For non-linear state equation we used Minimum Principal method. To handle the state-constrain, Model Predictive control (MPC) used. Two different approached are used for that which are discussed in later part of the project.

Where,

$$Q = \begin{bmatrix} 0.2777 & 0 \\ 0 & 1 \end{bmatrix}$$

$$R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

2. Initial Steps

2.1 Steady State Linearization

Let's rewrite the state equation for the system.

$$\frac{dH}{dt} = \frac{1}{A}(F_1 + F_2 - k\sqrt{H})$$

$$\frac{d(V.C)}{dt} = F_1.C_1 + F_2.C_2 - F.C ==> \frac{dC}{dt} = \frac{1}{A.H}(F_1.(C_1 - C) + F_2.(C_2 - C)$$

To linearis the equation, Tylor expansion around the steady state will used.

$$f(x,u) = f(x_s, u_s) + \frac{df}{dx}|_{x_s u_s}(x_s - u_s) + \frac{df}{du}|_{x_s u_s}(x_s - u_s)$$

For that, steady state equation can be written as,

$$\frac{dH_s}{dt} = \frac{1}{A}(F_{1s} + F_{2s} - k\sqrt{H})$$

$$\frac{d(VC)_s}{dt} = F_{1s}.C_{1s} + F_{2s}.C_{2s} - F_s C_s$$

$$0 = \frac{1}{A}(F_{1s} + F_{2s} - k\sqrt{H})$$

$$0 = F_{1s}.C_{1s} + F_{2s}.C_{2s} - k.\sqrt{H}.C_s$$

These equations will give us the steady state value of the state variable for give input variables.

For the Equation 1:

$$\frac{dH}{dt} = \frac{1}{A}(F_1 + F_2 - k\sqrt{H})$$

After applying Tylor series expansion.

UNIVERSITY OF ALBERTA

$$\frac{dH}{dt} = \frac{1}{A}[(F_1 - F_{1s}) + (F_2 - F_{2S}) - \frac{k}{2\sqrt{H_s}} . (H - H_s)]$$

$$\frac{dH'}{dt} = \frac{1}{A} . [F_1{}' + F_2{}' - \frac{k}{2\sqrt{H_s}} . (H')])$$

Where,

$$F_1 - F_{1s} = F_1{}'$$

$$F_2 - F_{2s} = F_2{}'$$

$$H - H_s = H'$$

For Equation 2

$$\frac{C.dV}{dt} + \frac{V.dC}{dt} = F_1.C_1 + F_2.C_2 - F.C$$

$$\frac{A.C.dH}{dt} + \frac{V.dC}{dt} = F_1.C_1 + F_2.C_2 - k\sqrt{H}.C$$

After applying the Taylor series expansion

$$A.C.\frac{dH}{dt}.+\frac{V.dC}{dt} = F_1.C_1 + F_2.C_2 - k\sqrt{H}.C$$

$$A.C.\frac{dH'}{dt}.+\frac{V.dC'}{dt} = (F_1 - F_{1s}).C_1 + (F_2 - F_{2s}).C_2 - k\sqrt{H}.(C - C_s) - \frac{k}{2\sqrt{H_s}} . (H - H_s)$$

$$A.C\frac{dH'}{dt} + A.H\frac{dC'}{dt} = [F_1{}' + F_2{}' - \frac{k}{2\sqrt{H_s}} . (H')] + (K.\sqrt{H}.C')$$

Let's take steady state value for the CSTR: -

$$F_{1s} = 0.05 \ {m^3}/{s}$$

$$F_{2s} = 0.1 \ {m^3}/{s}$$

$$A = 0.5 \, m^2$$

$$k = 0.125$$

$$C_1 = 0.2 \ {kg}/{m^3}$$

$$C_2 = 0.5 \ {kg}/{m^3}$$

Using above steady state value we got

$$H_s = \ 1.44\, m$$

$$C_s = \ 0.4 \ {}^{kg}/_{m^3}$$

If we put above steady state value in the equation ## and ## will get

$$\frac{dH'}{dt} = \ 2.F_1{}' + 2.F_2{}' - 0.10416.H'$$

$$\frac{dC'}{dt} = -3.2.F_1{}' - 1.6.F_2{}' - 0.8.C'$$

Above equation can be written as

$$\begin{bmatrix} \dot{H}' \\ \dot{C}' \end{bmatrix} = \begin{bmatrix} -0.10416 & 0 \\ 0 & -0.8 \end{bmatrix} . \begin{bmatrix} H' \\ C' \end{bmatrix} + \begin{bmatrix} 2 & 2 \\ -3.2 & 1.6 \end{bmatrix} . \begin{bmatrix} F_1{}' \\ F_2{}' \end{bmatrix}$$

2.2 Controllability of the System

Before moving forward into the project, it is important to check weather system is controllable. The simplest way to check controllability is to check the RANK of matrix.

$$C = [A\ BA \ldots\ldots A^{n-1}.B]$$

If C matrix has same rank as matrix A rank, then it can be said that system is controllable. For that, PYTHON code has been used. From that it was said that

rank (matrix C) = rank (matrix A) =2

3. Linear Quadratic Regulator (LQR)

As name suggest LQR is cost optimization control problem where it applicable to Linear system.

$$\begin{bmatrix} \dot{H}' \\ \dot{C}' \end{bmatrix} = \begin{bmatrix} -0.10416 & 0 \\ 0 & -0.8 \end{bmatrix} . \begin{bmatrix} H' \\ C' \end{bmatrix} + \begin{bmatrix} 2 & 2 \\ -3.2 & 1.6 \end{bmatrix} . \begin{bmatrix} F_1{}' \\ F_2{}' \end{bmatrix}$$

The performance measures for the system can be define as

$$J = \frac{1}{2}\int_{0}^{\infty} \grave{X}^{T}.Q.\grave{X} + \grave{u}^{T}.R.\grave{u}$$

$$Q = \begin{bmatrix} 0.2777 & 0 \\ 0 & 1 \end{bmatrix}$$

$$R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$x' = \begin{bmatrix} H' \\ C' \end{bmatrix}$$

$$u' = \begin{bmatrix} {F_1}' \\ {F_2}' \end{bmatrix}$$

3.1 Q and R weightage matrix selection:

The reason being the Q -matrix taken as per below mentioned methods.

$$\frac{Concentration}{Height} = \frac{0.4}{1.44} = 0.2777$$

The reason being chosen in such a manner is that our main goal of the system is the mixing of two steam which is counted in lesser than one value. To accommodate the importance of the concentration in following control methos I make weightage give to height is 0.2777 and weightage give to Concentration (state variable 2) is 1.

Hence final Q selected as

$$Q = \begin{bmatrix} 0.2777 & 0 \\ 0 & 1 \end{bmatrix}$$

For R, its input variable weightage matrix. Since we both the stream are equally important, I have given both F1 and F2 is 1 weightage.

Hence final R selected as

$$R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$X'\ and\ u'$ are the deviation variable to the steady state which we define earlier in Linearization part of the model.

Optimal Control Law can be defined as

$$u^{*\prime} = -R^{-1}.B^{T}.p^{*}(t).{x^{*}}'(t) = -Kx'^{*}(t)$$

To find the optimal control we have to find the $p^*(t)$. The LQR performance measure is infinite time duration, Hence, Riccati equation can be written as follow

$$Q + A^T P + PA - PBR^{-1}B^T P = 0$$

To solve the equation, PYTHON is used. Inbuilt function ***solve_continuous_are*** and ***np.linalg.inv*** are used to solve the above equation.

For Non -linear System: -

$$\frac{dH}{dt} = \frac{1}{0.5}(F_1 + F_2 - 0.125.\sqrt{H})$$

$$\frac{dC}{dt} = \frac{1}{0.5\,.\,H}[F_1.(0.2 - C) + F_2.(0.5 - C)]$$

To express in terms of the nonlinear, rather than using deviation variable, we can use it directly as per following manner. Hence, optimal control u can be expressed in following manner.

$$u - u_s = -K(x - x_s)$$

$$u = u_s - K(x - x_s)$$

Matrix P:

$$P = \begin{bmatrix} 0.35757398 & 0.17969369 \\ 0.17969369 & 0.32360044 \end{bmatrix}$$

Gain matrix K:

$$K = \begin{bmatrix} 0.14012813 & -0.67613401 \\ 0.42763804 & -0.15837331 \end{bmatrix}$$

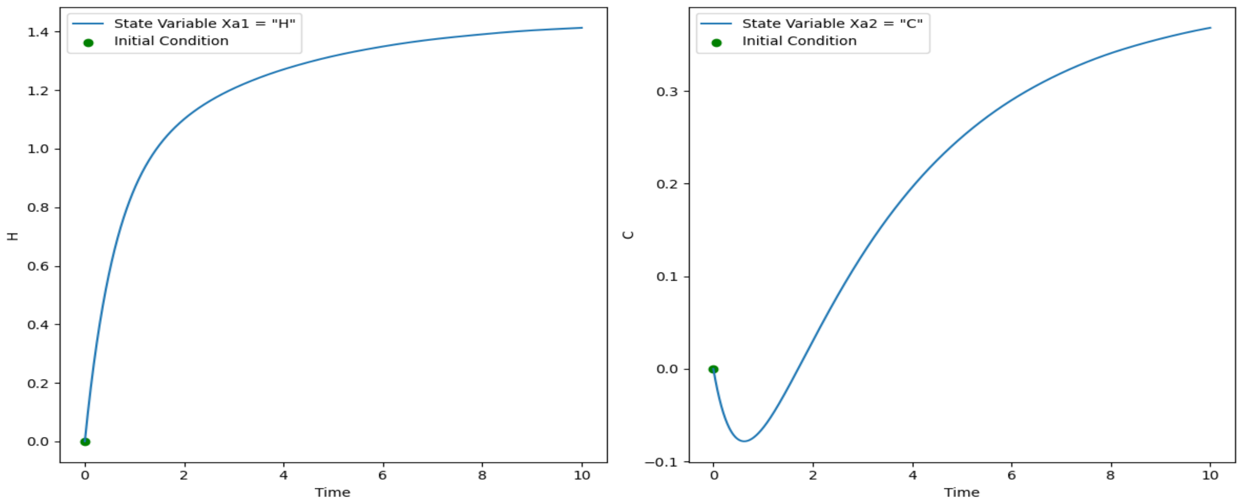


Figure 1 Linear LQR plot of State Variable H and C

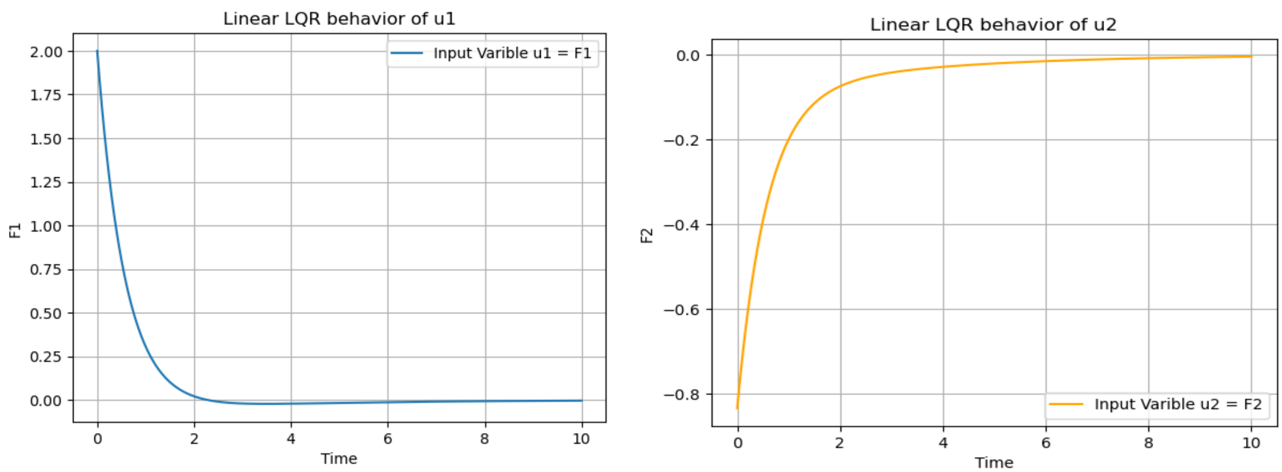


Figure 2 Linear LQR plot of Input Variable F1 and F2

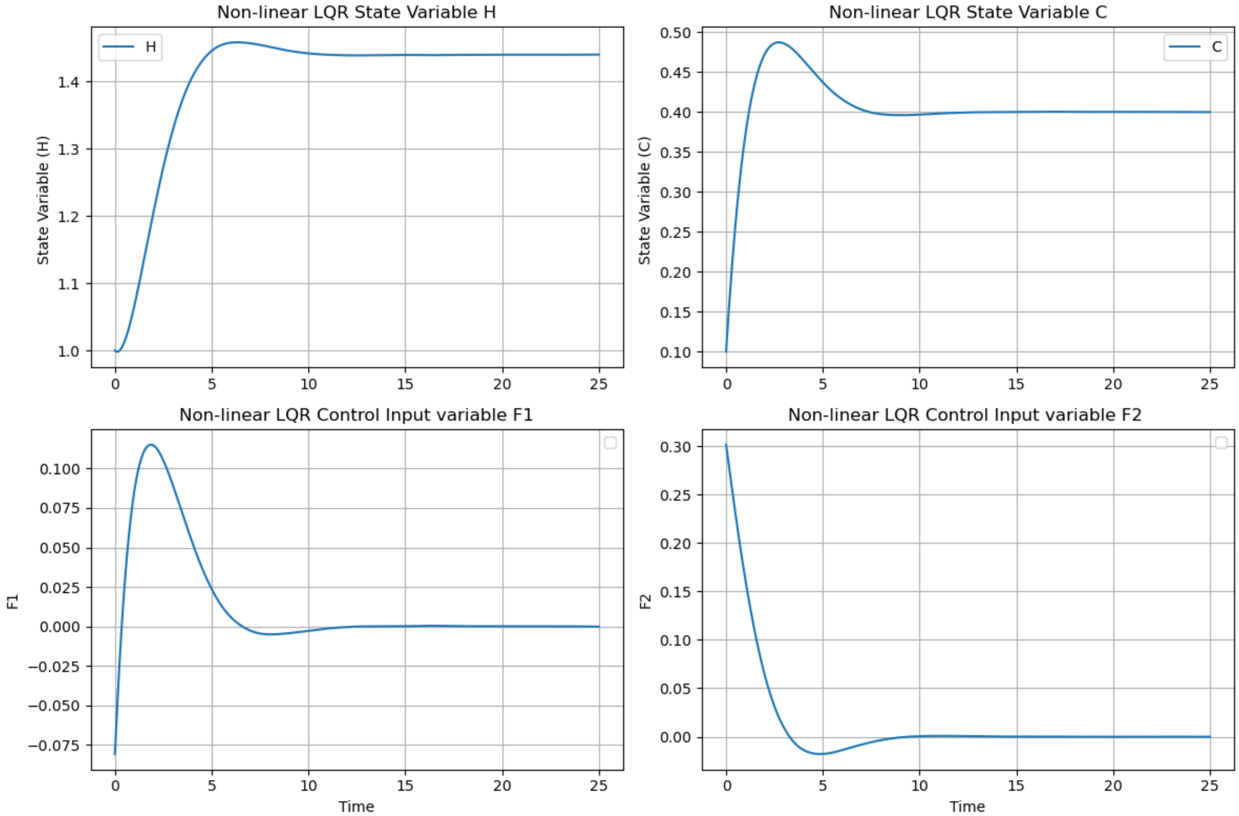


Figure 3 Non-Linear LQR plot of State and Input Variable behavior

4. MPC (Model Predictive control) (Linear and Non-linear)

As the name indicates, "Model Predictive Control" solves a problem related to optimal control using the system's current state as a starting state and forecasts the set of control actions for a finite amount of time. Stated differently, it generates the control action necessary to move the system from its current state to the intended one. During the sequence, the initial control measure is implemented on the system is measured, along with the associated system state. The procedure is repeated till the define sampling interval.[1]

Compare to the other control low where the control system only acts on the current state of the system (e.g. u = $-k.x$) MPC update every time with change in the state variables. Another advantage of using MPC that, it can handle the constraints which is failed to handle by the convention control such as LQR.

The project discussion includes the both linear and nonlinear MPC problem.

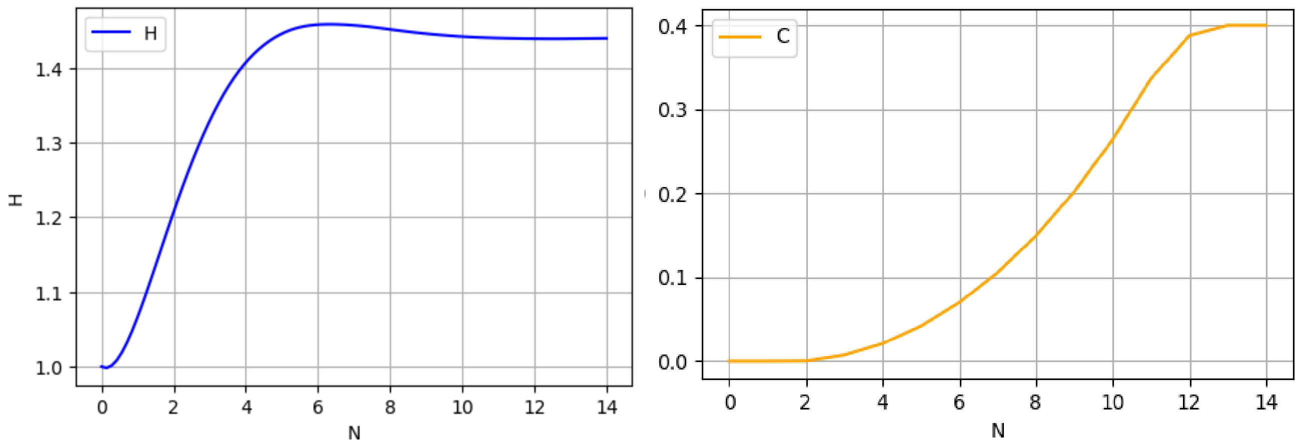


Figure 4 Linear MPC plot of State Variable H and C

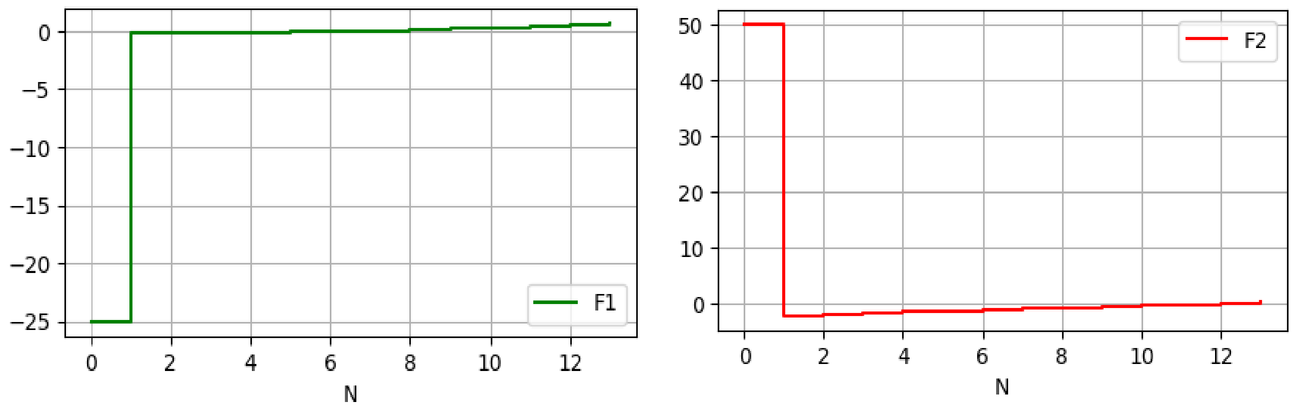


Figure 5 Linear MPC behavior plot of input variable F1 and F2

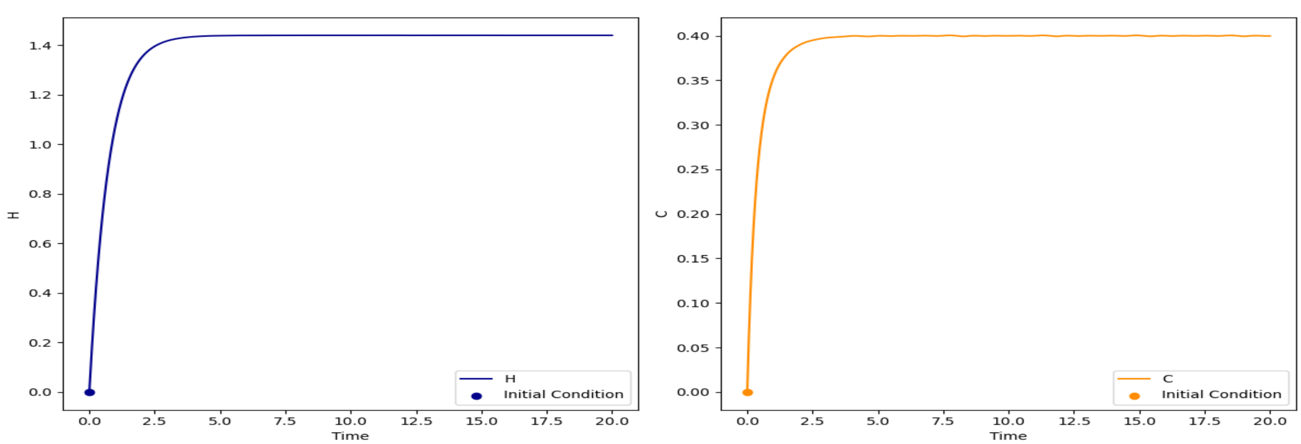


Figure 6 Non-liear MPC behavior of State Variable H and C

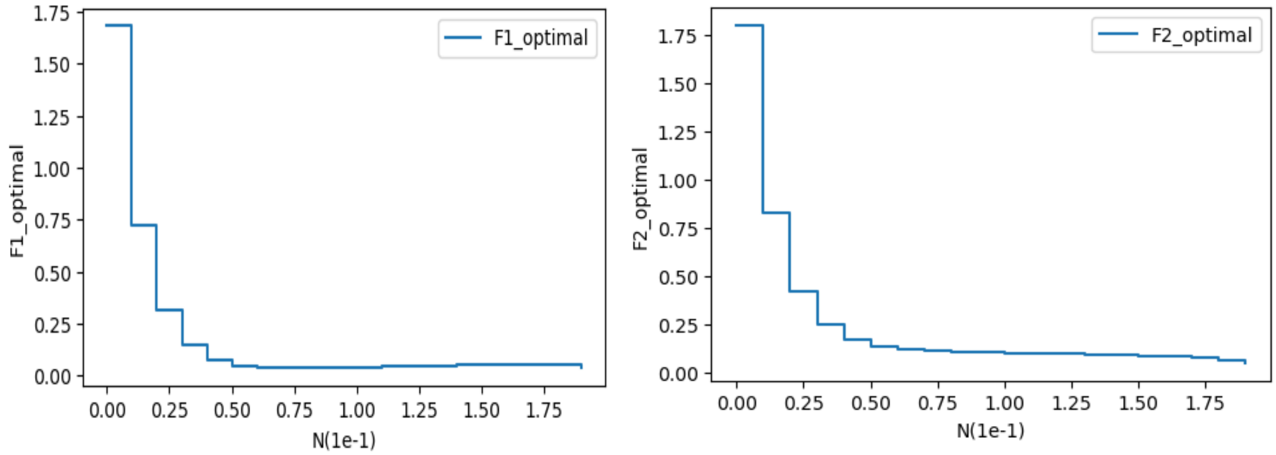


Figure 7 Non-Linear MPC plot of Input variable F1 and F2.

4.2 Exploration of the MPC Properties.

From the Figure 10 we can say that as we increase the constrain value of state variable of system H_max and C_max out cost function increases steadily. Hence for optimum performance cost for the state variable should be as less as possible.

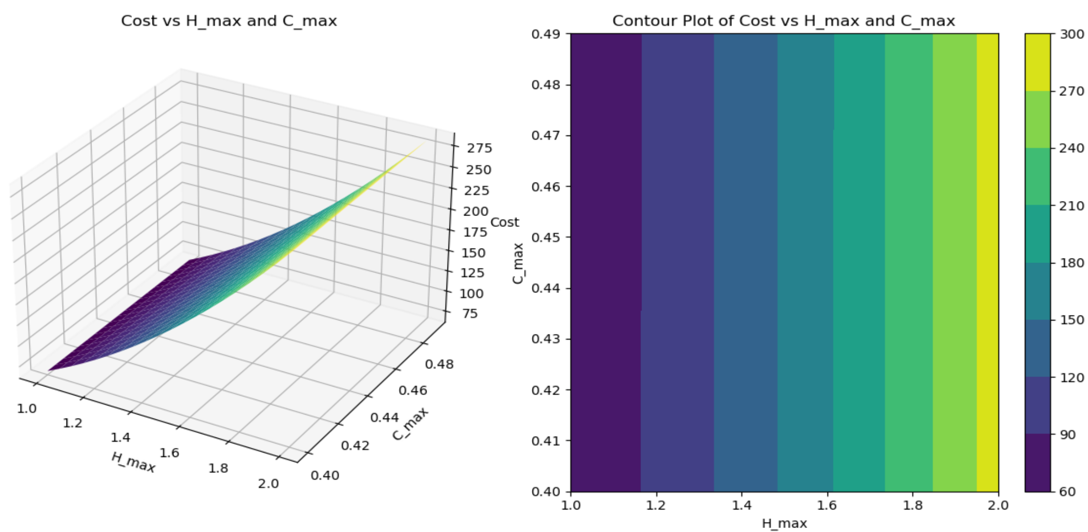


Figure 8 MPC Exploration: Effect H max and C max on the cost function.

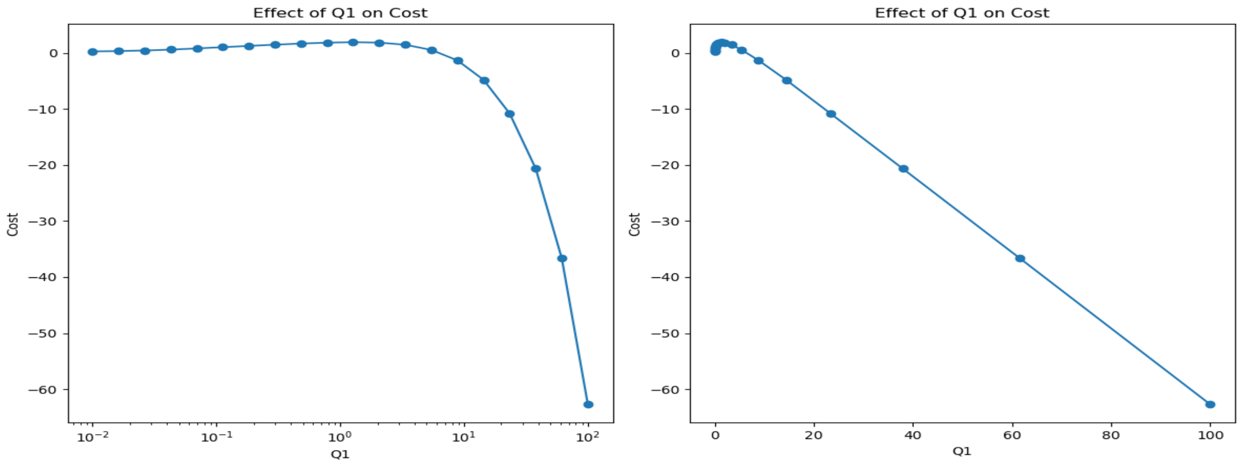
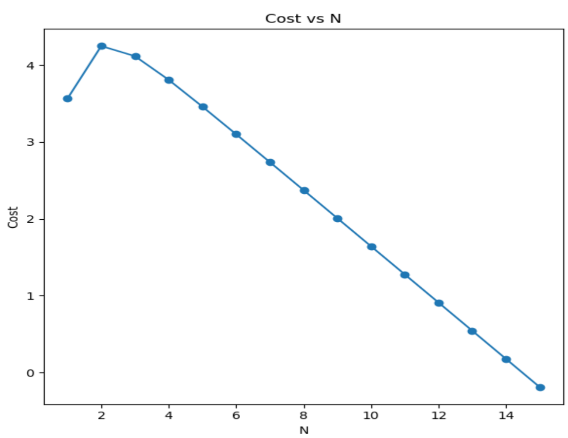


Figure 9 MPC exploration: Effect of Q(1st Diagonal) on the cost function

Figure 11 illustrate the effect of the Q (weight matrix of state variable) on the cost function. As the 1st diagonal component of the Q that is weightage put on the H increases cost function it decreases. However, the cost function is steady from 0.01 to 10 and after that cost drastically decreases.



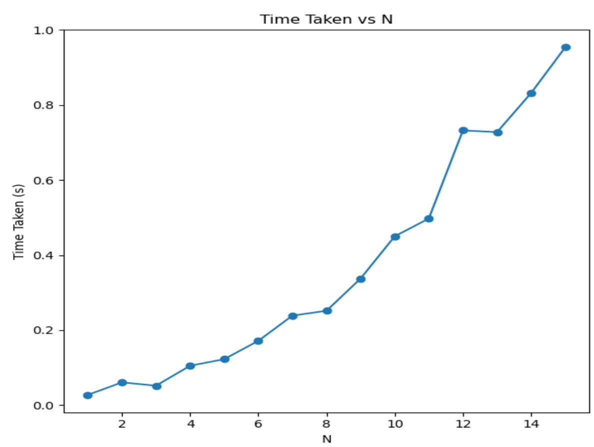


Figure 11 MPC exploration: Effect of incremental predicted horizon N on the cost function

Figure 11 MPC Exploration: Time taken to calculate the cost of MPC function with every incremental horizon N

From the Figure 8, it can conclude that predicted horizon N = 15 gives us minimal cost. (Further increased in N does not change much effect as N=15 cover the most of the dynamic behavior. The plot of the Time taken vs N clearly state that the as N increase, time taken by the model to evaluate the cost function increases. The simple-minded system like used in project this seem has no affect but as the model complexity and system complexity increases this parameter plays an important role as it directly related to computation cost to the user.

5. Minimum Principal Method.

Unlike the LQR, which can only handle the linear system, MP method can handle the non-linear system. Minimum principal method can also handle the constrain. However, for the project we did not include the constrain for simpler mathematics.

Let's write the equation first

$$\dot{x} = f(x^*(t), u^*(t))$$

$$\dot{p} = \frac{dH}{dx}\left(x^*(t), u^*(t)\right)$$

$$\frac{dH}{du} = 0$$

$$\frac{dH}{dt} = \frac{1}{0.5}(F_1 + F_2 - 0.125.\sqrt{H})$$

$$\frac{dC}{dt} = \frac{1}{0.5\,.H}[F_1.(0.2 - C) + F_2.(0.5 - C)]$$

$$J = \frac{1}{2}\left((x_f - x_s)^T.Q.(x_f - x_s)\right) + \frac{1}{2}\int_t^{t_f}[(x_f - x_s)^T.Q.(x_f - x_s) + (u_f - u_s)^T.R.(u_f - u_s)]dt$$

$$J = \int_0^{10}[x_1 - 1.44 \quad x_2 - 0.4].\begin{bmatrix}0.2777 & 0\\ 0 & 1\end{bmatrix}.\begin{bmatrix}x_1 - 1.44\\ x_2 - 0.4\end{bmatrix} +$$

$$[u_1 - 0.05 \quad u_2 - 0.1]\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}.\begin{bmatrix}u_1 - 0.05\\ u_2 - 0.1\end{bmatrix}$$

Where we taken

$$\mathrm{x(t)} = \begin{bmatrix}x_1\\ x_2\end{bmatrix} = \begin{bmatrix}C(t)\\ H(t)\end{bmatrix}$$

$$\mathrm{u(t)} = \begin{bmatrix}u_1\\ u_2\end{bmatrix} = \begin{bmatrix}F_1(t)\\ F_2(t)\end{bmatrix}$$

$$H = g + p^T f(x,u)$$

$$H = 0.277x_1^2 - 2.0143x_1 + 0.57427 + x_2^2 - 0.8x_2 + 0.7452 + u_1^2 - 0.1u_1 + u_2^2 - 0.2u_2$$

$$+ p_1\left(2u_1 + 2u_2 - 0.25\sqrt{x_1}\right) + p_{2.}\left(\frac{2u_1}{x_1}.(0.2 - x_2) + \frac{2u_2}{x_2}.(0.5 - x_2)\right)$$

Deriving the costate Equation:

$$\dot{p}_1 = -\frac{dH}{dx_1}$$

$$\dot{p}_1 = -[0.544x_1 - 0.4 - \frac{0.125}{\sqrt{x_1}}p1 - \frac{p_2.u_1}{x_1^2}.(0.2 - x_2) \ - \frac{p_2.u_2}{x_1^2}.(0.5 - x_2)\,]$$

$$\dot{p}_2 = -\frac{dH}{dx_2}$$

$$\dot{p}_2 = -(2x_2 - 0.8 - \frac{p_2.u_1}{x_1} - \frac{p_2.u_2}{x_1})$$

$$\frac{dH}{du_1} = 0$$

$$0 = 2u_1 - 0.1 + 2p_1 + \frac{2p_2.(0.2 - x_2)}{x_1}$$

$$u_1 = \frac{1}{2}(0.1 - 2p_1 - \frac{2p_2.(0.2 - x_2)}{x_1})$$

$$\frac{dH}{du_2} = 0$$

$$0 = 2u_2 - 0.1 + 2p_1 + \frac{2p_2.(0.5 - x_2)}{x_1}$$

$$u_1 = \frac{1}{2}(0.2 - 2p_1 - \frac{2p_2.(0.5 - x_2)}{x_1})$$

Finally write the state and co-state equation:

$$\dot{H} = 0.3 - 4p_1 - 4p_2 + 0.25\sqrt{x_1} - \frac{4p_2}{x_1}\ (0.7 - 2x_2)$$

$$C = \frac{2}{x_1}[\frac{(0.2 - x_2)}{2} . \left(0.1 - 2p_1 - \frac{2p_2.(0.2 - x_2)}{x_1}\right) + \frac{(0.5 - x_2)}{2} . \left(0.2 - 2p_1 - \frac{2p_2.(0.5 - x_2)}{x_1}\right)]$$

$$\dot{p_1} = -(0..544x_1 - 0.4 - \frac{0.125}{\sqrt{x_1}} + \frac{p_{2.}}{x_1^2}.(0.1 - 2p_1 - \frac{2p_2.(0.2 - x_2)}{x_1}) - \frac{p_{2.}}{x_1^2}.(0.5 - x_2).(0.2 - 2p_1 - \frac{2p_2.(0.5 - x_2)}{x_1})\ )$$

$$\dot{p_2} = -(2x_2 - 0.8 - \frac{p_2}{x_1}(0.1 - 2p_1 - \frac{2p_2.(0.2 - x_2)}{x_1}) - \frac{p_{2.}}{x_1}\ (0.2 - 2p_1 - \frac{2p_2.(0.5 - x_2)}{x_1}))$$

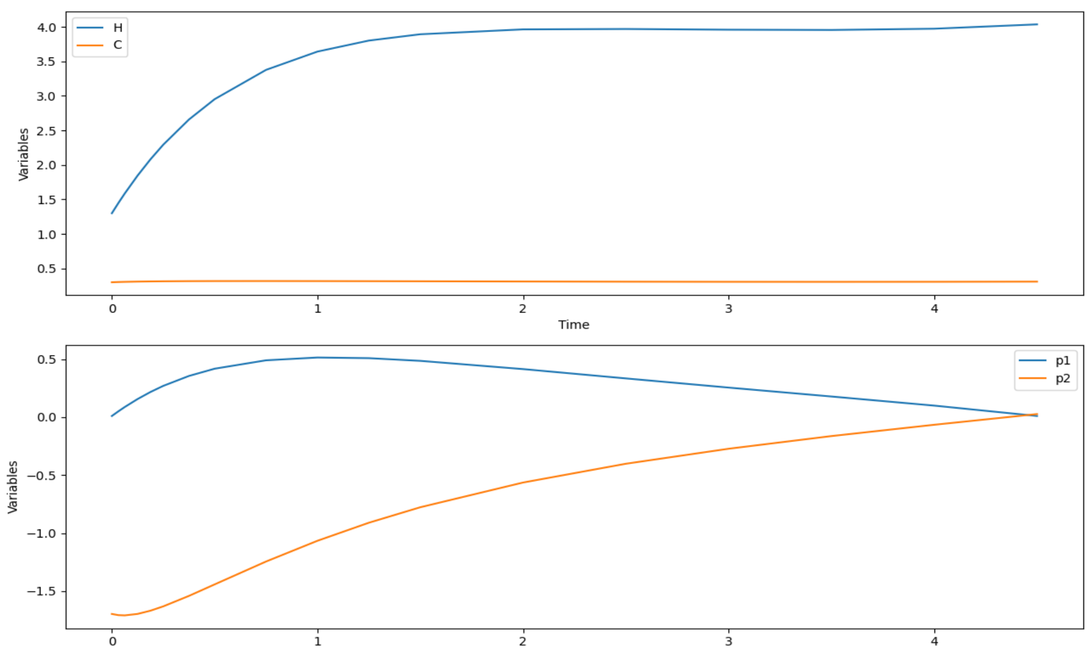


Figure 12 TPBVP plot

Boundary Condition

For this project we are not putting any boundary condition. Hence,

$$\left[\frac{dh}{dx} - p\right].(t_f)\delta t_f = 0$$

$$p^*(t_f) = 0$$

For Two-point boundary value problem (TPBVP), initial value is taken below

$$\begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} = \begin{bmatrix} 1 \\ 0.3 \end{bmatrix}$$

$$\begin{bmatrix} {P_1}^*(10) \\ {P_2}^*(10) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

From TPBVP plot it is clear that H variable is not going to converge its value. However, I double check with the internet. Stil H is not able to converge to its steady state. However, from C, P_1 and P_2 it is clear that TPBVP code is right. I try with all possible change with the initial guess and SS as per your recommendation still it's not working.

6. Result and Discussion

In this project, I have taken two state variable and two input variable system non-linear system. After linearization of the system, we have got the steady state value of the $H_{ss} = 1.44$ m and $C_{ss} = 0.4$.

We have taken the deviation variable of linearized system and apply that to LQR system. I calculate the gain matrix K and P matrix. Non-liear LQR is calculated based on the deviation variables. The next we did is the model predictive control. We have taken two methods; one is linear method and another is non-linear method. Both results are good as shown in above figures. We also calculate the effect of the C_max and H_max constrain on the cost function (Approach 2 effects on the cost). To understand the depth of the model we plot the graph of cost vs predicted horizon. Computational time is always is significant parameter for MPC specially when system is complex. Hence, we also plot the time taken for every incremental of N. For minimum principal we have derive the state and costate equation and apply TPBVP

7. Appendix