CONTROLLERS FOR INTERACTIVE TANK PROCESS

The interactive processes are examined by PID, CDM, MPC & NMPC controllers based on all stability and phase conditions. Enhance process models from SISO into MIMO through SITO, TIFO as like TITO under various controllers.

4.1 PID CONTROLLER ACTION FOR DECOUPLED FOUR TANK PROCESS

The basic mathematical description of PID controller is explained about error $e$ tracking difference between output $y$ and reference $r$ value in equation below

$$u(t) = K_p[e(t) + \frac{1}{T_i}\int e(\tau)d\tau + T_d \frac{de(t)}{dt}]$$

(4.1)

Here error $e(t) = r(t) - y(t)$ is zero achieved by infinite time only for continuous tracking error if the system output $y(t)$ convergence to the reference signal $r(t)$ as time approaches infinity. The process behaviors depend on the proportional $K_p$, derivative $T_d$ and integrative $T_i$ these are influence individual as well as together also. These are placed before plant of closed loop process. The proportional increase the speed of response of closed loop system, it may leads toward unstable also and decrease steady state error, increase overshoot response. Integral action impose on steady state error in step response leads to stable and more effectively works together with proportional for step response. If $T_i$ is smaller closed loop will be unstable, it has to increase, and the speed of decrease tends to be lower. Derivative decreases overshoot similarly $T_i$ slower rise time and settling time when increases $T_d$ and cannot work individually.

Tuning of PID controller with under damped response system is under damped step response may be approximated by a second-order or fourth-order system transfer function. The
selection of P, I, D values, should lead to a desired closed loop response. It is desirable to have a well damped closed loop response, to reduce the number of oscillation, as well as settling time. The closed loop transfer function is 

$$G(s)H(s)\left/(1+G(s)H(s)\right.$$,

which can be expressed as 

$$N(s)\left/D(s)\right.,$$

where N(s) and D(s) are following numerator and denominator polynomials respectively. We can write D(s) as a quadratic factor in s multiplied by a first order polynomial in s, as follows

$$D(s) = \left(S^2 + 2\xi\omega_n S + \omega_n^2\right)(S + P_i)$$  \hspace{1cm} (4.2)

From Eqn. (4.2) \(P_i\) is the dominated pole with smallest real part magnitude of step response of the plant.

In order to obtain a PID controller for this system, Eqn. (4.2) can be rewritten as

$$G_c = \frac{K_p T_d}{s} \left(\frac{1}{T_d} + \frac{1}{T_i T_d}\right)$$  \hspace{1cm} (4.3)

A choice for the controller zeros may be such that the numerator polynomial \(G_c\) of Eqn. (4.3) the denominator polynomials of plant of FTP Eqn. (3.1) canceled.

$$T_d = \frac{1}{2\xi\omega_n}, T_i = \frac{2\xi}{\omega_n}, K_p = \frac{4T_i}{Kt_s}$$  \hspace{1cm} (4.4)

Where \(t_s\) is settling time and \(K\) gain of the plant are calculated from the step response.

The PID parameters can be computed as under:

Algorithm for compute P, I and D

1) Apply to the plant a step of amplitude \(A\) and record the output \(y(t)\), as shown in Figure 4.1.

2) Determine \(y(x)\) and the settling time \(t_s\) of the plant response, the first two peak values \(M_{p1}\) and \(M_{p2}\), and the corresponding time instants \(t_{p1}\) and \(t_{p2}\).

3) Compute \(d = \left(M_{p2} - y(x)\right)\left(M_{p1} - y(x)\right)\) and \(T_p = T_{p2} - T_{p1}\)

4) Compute \(\xi = \frac{1}{\sqrt{1 + \left(2\pi/\ln d\right)^2}}\) and \(\omega_n = \frac{2\pi}{\left(T_p \sqrt{1 - \xi^2}\right)}\)

5) Compute PID parameters using Eqns. (4.3) & (4.4).
Obtained P, I & D values after tune the step response of system and imposed into decoupled four tank process for minimum phase and non minimum phase under at operating point. Calculate and tune all control action for desired reaction for step input action as illustrate in figure 4.1 from PID algorithm.

![Graph](image)

Figure.4.1: System with underdamped step response used for identification of the process

4.2 CDM CONTROLLER FOR DECOUPLED FOUR TANK PROCESS

The controller must be designed under some practical limitations when a control problem is considered. The controller is desired to be of minimum degree, minimum phase (if possible) and stable. It must have a sufficiently narrow bandwidth and power rating limitations. If the controller is designed without considering these limitations, the robustness property will be very poor, although the stability and time response requirements are met. When all of these mentioned properties are considered together, the controller designed by using CDM method will have the smallest degree, the smallest bandwidth and will have the closed loop time response without an overshoot.

These properties guarantee the sufficient damping of the disturbance effects and the low economic property. CDM is a polynomial algebraic method. The advantages of the classical and
modern control techniques are integrated with the basic principles of this method which is derived by making use of the previous experience and knowledge about the controller design. This way an effective and efficient design method, namely CDM is constructed.

Without confronting with serious difficulties and necessitating much experience, CDM now makes possible to design very good controllers with less effort and relative ease when compared with the other existing methods. Many control systems have been designed successfully using CDM. It is very easy to design a controller under the conditions of stability, time domain performance and robustness. The close relation between these conditions and coefficients of the characteristic polynomial can be simply found. This means that CDM is effective for not only control system design but also for controller parameters tuning.

The basic block diagram of the CDM control system is shown in Figure 4.2. In this figure, $y$ is the output, $r$ is the reference input, $u$ is the control and $d$ is the external disturbance signal. $G(s)$ is the plant transfer matrix; $A(s)$ is the forward denominator polynomial while $F(s)$ and $B(s)$ are the reference numerator and the feedback numerator polynomials of the controller transfer matrix. Since the controller has two numerators, it resembles to a 2DOF system structure. $A(s)$ and $B(s)$ are designed as to satisfy the desired transient behavior, while pre-filter $F(s)$ is used to provide the steady-state gain. Better performance can be expected when using a 2DOF structure, because it can focus on both tracking the desired reference signal and disturbance rejection. Unstable pole-zero cancellation and use of more number of integrators are also avoided in implementations with this structure.

### 4.2.1 DESIGN PROCEDURE FOR CDM

The block diagram of CDM design for a single input-single output (SISO) system is shown in Figure 4.2. Here $y$ is the output signal, $r$ is the reference input, $d$ is the disturbance and $n$ is the measured output noise. $N(s)$ and $D(s)$ are numerator and denominator of transfer function of the plant, respectively. $A(s)$ is the denominator polynomial of the controller transfer function while $F(s)$ and $B(s)$ are called the reference numerator and the feedback numerator polynomials of the controller transfer function. Since the transfer function of the controller has two numerators, thus it is resembles to two degrees of freedom (2DOF) system structure. Better
performance can be expected when using a 2DOF structure, because it can focus on both tracking the desired reference signal and disturbance suppression.

![Block diagram of CDM control system](image)

Figure.4.2: A block diagram of CDM control system

The output of the controlled closed-loop system is

\[
y = \frac{N(s)F(s)}{P(s)} r + \frac{A(s)N(s)}{P(s)} d - \frac{N(s)B(s)}{P(s)} n
\]

(4.5)

Where \(P(s)\) is the characteristic polynomial and given by

\[
P(s) = D(s)A(s) + N(s)B(s) = \sum_{i=0}^{n} c_i s^i
\]

(4.6)

The design procedure for choosing controller parameters with design specifications may be summarized as follows:
1. The selection of proper controller polynomials, A(s), B(s) and F(s):

Generally, a linear, time invariant plant transfer function is expressed as below

\[
\frac{N(s)}{D(s)} = \frac{a_p s^p + a_{p-1} s^{p-1} + \ldots + a_1 s + a_0}{b_q s^q + b_{q-1} s^{q-1} + \ldots + b_1 s + b_0}
\]  

As it is seen from Eqn. (4.7), the degree of polynomial \(D(s)\) is \(q\) and polynomial \(N(s)\) is \(p\) \((p \leq q)\). Let's assume controller polynomials \(A(s)\) and \(B(s)\) are as follows

\[
A(s) = \sum_{i=0}^{m} a_i s^i \quad (4.8a)
\]

\[
B(s) = \sum_{i=0}^{m} b_i s^i \quad (4.8b)
\]

If there is no disturbance or noise in the system, then \(\text{deg}\{A(s)\}\) and \(\text{deg}\{B(s)\}\) are taken as \(\text{deg}\{D(s)\}-1=q-1\). If the system has disturbance or noise, \(A(s)\) and \(B(s)\) controller polynomials are chosen so that the effect of disturbance and noise is decreased to the minimum. When polynomial \(F(s)\) is chosen as

\[
F(s) = (P(s) / N(s))_{s=0} \quad (4.8c)
\]

The overall closed loop transfer function becomes Type-1 system. Therefore, a good closed-loop response can be achieved.

2. Choice of the equivalent time constant \(\tau\):

In CDM, \(\tau\) is chosen as \(\tau = t_s / (2.5-3)\) where \(t_s\) is the user specified settling time.

\[
\tau = t_s / (2.5 \sim 3) \quad (4.8d)
\]

3. Selection of stability indices and stability limits:

Stability indices is given by

\[
\mu_i = 2.5, \mu_i = 2 + 2 \sim (n-1), \mu_0 = \mu_n = \infty. \quad (4.9)
\]

and stability limits are found by
4. Determining the coefficients of characteristic polynomial:

There is a relationship among the coefficients of the controlled system closed-loop characteristic polynomial:

\[ c_i = \frac{c_0 \tau^i}{(\mu_i \tau \mu_{i-2} \ldots \mu_2 \mu_1^{i-1})} \]  \hspace{1cm} (4.11)

If Eqn. (4.11) is put into Eqn. (4.6) characteristic polynomial will be expressed by \(c_0\), \(\tau\) and \(\mu_i\) as follows:

\[ P(S) = c_0 \left[ \sum_{i=2}^{n} \left( \frac{1}{\mu_i} \right)^i (\tau S)^i \right] + \tau S + 1 \]  \hspace{1cm} (4.12)

In order to determine controller parameters, by putting controller polynomials from Eqns. (4.8a) & (4.8b) into Eqn. (4.6) a polynomial of which coefficients are dependent to parameters \(m_i\) and \(l_i\) is obtained. This polynomial is compared with Eqn. (4.12). The coefficients \(m_i\), \(l_i\) and \(c_i\) are determined and then controller polynomials \(A(s)\), \(B(s)\) and closed loop characteristic polynomial \(P(s)\) are obtained.

4.2.2 TESTING THE CONTROL SIGNAL AFTER THE DESIGN

The CDM control system shown in Figure 4.3 is simulated after the controllers for both of the sub-processes are designed. Whether the each of the control signals reaches the saturation or not, the each signal is computed and drawn after the design. If one or two of control signals saturates, we return to Item 2 & 3, increase \(\tau\) sufficiently and repeat the design. On the contrary, if magnitude of the control signals is of very small size, the system time response can be accelerated by decreasing \(t\) as desired.
4.3 THE PRINCIPLE OF LINEAR AND NONLINEAR MODEL PREDICTIVE CONTROL

In general, the model predictive control problem is formulated for solving a finite horizon open-loop optimal control problem subject to system dynamics and constraints involving states and controls. Figure 4.4 shows the basic principle of model predictive control. The measurements are obtained from plant at regular interval at time $k$. The controller predicts the future dynamic behavior of the system over a predicted horizon $P$ and control horizon $M$ determines (over a control horizon $P>M$) the input such that a predetermined open-loop as well as closed-loop performance objective functional is optimized. If there were no disturbances and no model-plant mismatch, and if the optimization problem could be solved for infinite horizons, then one could apply the input function found at time $k = 0$ to the system for all times $k \geq 0$. However, this is not possible in general. Due to disturbances and model-plant mismatch, the true system behavior is different from the predicted behavior. Here we have ignored disturbance and match plant model.
In MPC the set points are typically calculated each time for MIMO process with $u$ input variables and $y$ output variables.

The current values of $u$ and $y$ as $u(k)$ and $y(k)$. The objective is to calculate the optimum set point $y_{sp}$ for the next control calculation (at $k+1$) and also to determine the corresponding steady-state value of $u$, $u_{sp}$. This value is used as the set point for $u$ for the next control calculation.

A general, linear steady-state process model can be written as
\[ \Delta y = K \Delta u \quad (4.13) \]

Where $K$ the steady-state is gain matrix and $\Delta u, \Delta y$ denotes steady-state changes in $u$ and $y$; it is convenient define $\Delta u$ and $\Delta y$ as
\[ \Delta y = y_{sp} - y_{OL}(k) \] (4.14)

Here \( y_{OL}(k) \) is steady-state value of \( y \)

\[ \Delta u = u_{sp} - u(k) \] (4.15)

To incorporate output feedback, the steady-state model Eqn. (4.13) is modified as

\[ \Delta y = K \Delta u + \left[ y(k) - y(k) \right] \] (4.16)

### 4.4 COMPONENTS OF NMPC AND MPC

NMPC has three basic elements (i) prediction, (ii) optimization problem and (iii) control law. These three elements are used in this work and described in the following sections:

#### 4.4.1 THE PREDICTION

If there is a constant input or disturbance acting on the plant any error in the steady state gain modeling will nullify offset free tracking. Thus, to handle different orders of the model and plant it is necessary to store the appropriate number of past values of the plant output in a vector. If the model order is smaller than the plant order, the vector is accordingly truncated, since the most recent plant values are already stored. The algorithm is given below:

```matlab
% Update past model inputs:
umpast = [uu(k);umpast(1:length(umpast)-1)];

% Simulation:
ym(k+1) = -denm(2:ndenm+1) * ympast + numm(2:nnumm+1) * umpast;

%%% Model Order <= Plant Order
if ndenm <= ndenp,
    % Update past model outputs:
    ympast = yppast(1:ndenm);
end
```
4.4.1.1 NMPC ALGORITHM TO HANDLE MODEL ORDER SMALLER THAN PLANT ORDER

Alternately, if the model order is larger than the plant order, then there are not enough past values of plant output and this requires holding additional past values of plant. The algorithm is given below.

```matlab
%% Model Order > Plant Order
if ndenm > ndenp,
    ympast(n.denp+1:ndenm) = ympast(n.denp:ndenm-1);
    ympast(1:ndenp) = yppast;
end
```

4.4.1.2 NMPC ALGORITHM TO HANDLE MODEL ORDER LARGER THAN PLANT ORDER

The prediction system shows the value of the system status P sampling times ahead, where P is the prediction horizon and can be estimated. A chosen value of ‘P’ indicates the ability of the response model (for e.g.: step-response) to cover a given value of period of the open-loop settling time. Alternately, for a given model horizon P, the value of the sampling period can be selected so as to ensure that the response model covers a given period of the open loop settling time. If it is not possible to measure the full state vector, or if the measured outputs consist of some linear combinations of the states, so that the states cannot be measured directly, then an observer can be used to estimate the state vector. For a nonlinear system dynamics, the prediction is a function instead of an explicit expression.

4.4.2 THE OPTIMIZATION PROBLEM
The optimization problem consists of minimizing the objective function, to the input sequence uF. This optimization can be constrained or unconstrained and the objective function is to evolve a system such that it follows a certain reference. This is achieved by minimizing an objective function (which is a measure of the performance) typically penalizing the tracking error. The solution of the minimization gives the adequate input signal that gives the required system output. The reference is known apriori and here referred to as xref. Due to the system dynamics and assuming that the reference is reachable, each state reference has a corresponding input reference, uref. The standard objective function can have different forms, however, a quadratic cost that penalizes the error is the most natural and simple way to chose.

4.4.2.1 PROCESS CONSTRAINTS

Since there exhibit actuator limitations as well as operational limitations, input and output constraints are often present in NMPC algorithms. For example, the feed tank level ‘h’ is constrained such that the feed tank neither runs dry nor overfills and hence ‘h’ becomes a controlled output, even though it is not to be controlled to a set point. Thus there will be two constraints on ‘h’ for each step in the prediction horizon – one minimum value and one maximum value. The additional difficulty is that even though ‘h’ is controlled within constraints it is not measured. For the four tank process, in this work, the value of ‘h’ is inferred from the measured values by using a nonlinear observer that gives sufficiently accurate estimates of ‘h’. Three types of constraints are considered in this work: input signal constraints, state constraints and terminal state constraints. These constraints are shown mathematically as,

4.4.3 THE CONTROL LAW

Since prediction control usually includes constraints, the resulting control law is usually nonlinear. The control law is chosen such that the more frequent changes of u is near the beginning of the prediction horizon; and less frequent later i.e. the input signal is assumed to remain constant over ‘blocks’ of sampling intervals. Although constrained NMPC are nonlinear, they are usually time invariant. The solution of this constrained optimization problem gives the
input sequence for the next prediction horizon. However, not the entire input sequence is applied at the next sample time, just the part of the solution that corresponds to the current time step is applied on the system. This ensures that new information in the form of most recent measurement is utilized immediately instead of being ignored for the next M sampling instants.

4.5 DESCRIPTION OF MPC FOR INTERACTIVE TANK PROCESS

In MPC applications, the output variables are also referred to as controlled variables or CV’s, while the input variables are also called as manipulated variables or MV’s. The predictions are made in two types of MPC calculations that are performed at each sampling instant: set-point calculations and control calculations. Inequality constraints as upper and lower limits can be included in either types of calculation.

The main point of this optimization problem is to compute a new control input vector, $u(k)$ to be fed to the system, and at the same time take the process constraints into consideration.

4.5.1 AN MPC ALGORITHM CONSISTS OF FOLLOWING:

- A Cost Function
- Constraints
- A Model of the Process

4.5.1.1 A COST FUNCTION FOR MPC

The main idea with MPC is that the MPC controller calculates a sequence of future control actions such that the cost function is minimized.

The cost function often used in MPC is like this (a linear quadratic function):

$$ J = \sum_{k=0}^{P} (\hat{y} - r)^T Q (\hat{y} - r) + \sum_{k=0}^{P} \Delta u^T R \Delta u $$

(4.17)

Where:
Prediction horizon

Set point

Predicted process output

Predicted change in control value, \( \Delta u_k = u_k - u_{k-1} \)

Output error weight matrix

Control weight matrix

This works for SISO (Single Input and Single Outputs) and MIMO systems (Multiple Input and Multiple Outputs) so we are dealing with vectors and matrices.

4.5.1.2 CONSTRAINTS

All physical systems have constraints. In generally, physical constraints like actuator and valve limits, etc and performance constraints like overshoot, settling time, etc. In MPC one normally defines these constraints for minimize inequalities.

Constraints for Tank Model: The constraints considered, besides the model dynamics, are input signal constraints, output constraints and terminal state constraints. The input constraints are considered to account for the recommended input flow from the pump specifications. The output state constraints are due to the height of the tanks. The minimum height is zero for all tanks.

4.5.1.3 MODEL OF PROCESS

The main drawback with MPC is that a model for the process, i.e., a model which describes the input to output behavior of the process, is needed. Mechanistic models derived from conservation laws can be used. Usually, however in practice simply data-driven linear models are used.

In MPC it is assumed that the model represents a state-space model of the form:
\[ x(k+1) = Ax(k) + Bu(k), \]
\[ y(k) = Cx(k) + Du(k). \]

(4.18)

4.6 DESCRIPTION OF NMPC FOR INTERACTIVE TANK PROCESS

The NMPC control scheme is shown in the flowchart of figure 4.5. Even though an input variable is unavailable for control, it can serve as a disturbance variable if it is still measurable.

As the control structure changes from one control execution time to another, the subsequent control calculations can become ill-conditioned. It is important to identify and correct the situations before executing the NMPC calculations in steps 5 and 6. Another approach is based on singular value and by omitting a small singular value; the process model can be adjusted so that it is no longer ill conditioned. An important advantage of this ill conditioning removal approach is that none of the output variable is removed. A disadvantage is that the SVA approach depends on how the input and outputs are scaled. Alternately, ill conditioning can also removed by adjusting an NMPC design parameter R (the suppression matrix).
4.6.1 NMPC ALGORITHM CONSISTS OF FOLLOWING:

- A Cost Function or optimization
- Constraints
- A Model of the Process

4.6.1.1 NONLINEAR MODEL FOR LINEAR AND NONLINEAR PROCESS

Consider the stabilization problem for a class of systems described by the following nonlinear differential equations. Consider the nonlinear differential equation for stabilizing the problem.

\[ \dot{x}(t) = f(x(t), u(t)) \quad x(t_0) = x(0) \quad (4.19) \]
\[ y(t) = g(x(t), u(t)) \quad (4.20) \]
\[ u(t) \in u, \forall t \geq 0, x(t) \in x, \forall t \geq 0, y(t) \in y, \forall t \geq 0 \quad (4.21) \]
Where $x(t) \in \mathbb{R}^n$ and $u(t) \in \mathbb{R}^m$ determine the vector of states and inputs. Denotes $x$ and $u$ are feasible set of inputs and states and $y$ is estimated or measured output.

4.6.1.2 CONSTRAINS FOR NMPC

We assume a set of feasible assumptions i.e., $x$, $u$ and $y$ as follows:

Assume 1: In its simplest form, $u$ and $x$ are given by constraints of the form

Constraints in the outputs:

$$y_{\text{min}} \leq y \leq y_{\text{max}} \quad (4.22a)$$

Constraints in the inputs:

$$\Delta u_{\text{min}} \leq \Delta u \leq \Delta u_{\text{max}} \quad (4.22b)$$

$$u_{\text{min}} \leq u \leq u_{\text{max}} \quad (4.22c)$$

Note: $\Delta u_k = u_k - u_{k-1}$

$$x_{\text{min}} \leq x \leq x_{\text{max}} \quad (4.22d)$$

Assumption 2: The vector field $f(x(t), u(t))$ is continuous and satisfies $f(0,0) = 0$ at initial condition.

Assumption 3: Eqn. (4.19) has a unique continuous solution for any initial condition in the region of interest and continuous manipulated input function $u(t) : [0,M] \rightarrow u$ and continuous predicted state function $x(t) : [0,P] \rightarrow x$

Real systems and models are mainly used for predicting the future values within the limits selected by the controller.

The finite horizontal open loop described above is mathematically formulated from Eqn. (4.19) as follows. $\overline{u}(t)$ is represented as internal controller.
\[
\min_{\bar{u}(t)} J(x(t), \bar{u}(t); M, P)
\]

\[
J(x(t), \bar{u}(t); P, M) := \int_{t}^{t+P} f(x(t), \bar{u}(t)) dt + \int_{t}^{t+M} g(\bar{u}(t)) dt
\]  \hspace{1cm} (4.23)

Subject to:

\[
\bar{x}(t) = f(\bar{x}(t), \bar{u}(\tau)), \bar{x}(t) = x(t)
\]  \hspace{1cm} (4.24a)

\[
\bar{x}(t) \in X, \forall t \in [t, t+P]
\]  \hspace{1cm} (4.24b)

\[
\bar{u}(t) \in U, \forall t \in [t, t+M]
\]  \hspace{1cm} (4.24c)

\[
\bar{u}(\tau) = \bar{u}(\tau + P), \forall t \in [t+M, t+P]
\]  \hspace{1cm} (4.24d)

### 4.6.1.3 OPTIMIZED THEORY FOR NMPC ALGORITHM

The designer needs to optimize control algorithm to minimize cost and maximize performance measure. These depend on the system variables, which are states \( x \), output \( y \), tracking error \( e \) and control \( u \).

Describe the process state Eqns. (4.19) & (4.20) of nonlinear time invariant:

Performance function:

\[
J(x(t), u(t), y(t)) = \int_{t_0}^{t_f} w_{xyu}(x, y, u) dt
\]  \hspace{1cm} (4.25)

Is minimized to the dynamic system which is represented as maximized performance measure for determining control law with penalty term \( k(x(t, f)) \).
\[ J = h(x(t)) + \int_{t_0}^{t_f} f(x(t), u(t)) \, dt \]  
(4.26)

\[ t_f = \text{final time}, \quad t_0 = \text{initial time}; \quad t_0 \leq t \leq t_f \]

Optimal solution to optimized problem is denoted \( u^*(t) \) and repeatedly solved at sampling instants \( t= k\delta \), \( K=0, 1, 2... \) for open loop control problem. Admissible optimal control law is defined for closed loop control for Eqn. (4.19) at sampling instants.

\[ u^*(t) = \bar{u}^*(\tau, x(t), P, M), \quad \tau \in [t, \delta] \]  
(4.27)

The optimal value of NMPC open loop optimal control as a function of the state will be denoted in the following as value function.

\[ V(x, P, M) = J\left(x(t), \bar{u}^*(t), P, M\right) \]  
(4.28)

In a similar method, obtain performance measure form, from Eqns. (4.26) to (4.28)

\[ J\left(x(t), u(t), P, M\right) = h(x(P)) + \int_{t_0}^{t_0+P} f(x(t), u(t)) \, dt + \int_{t_0}^{t_0+M} g(u(t)) \, dt \]  
(4.29)

The admissible controls are constrained to lie in a set \( U \); i.e. \( u \in U \). First approximate the continuous operation of Eqn. (4.19) by a discrete system.

\[ \frac{x(t + \Delta t) - x(t)}{\Delta t} \approx f(x(t), u(t)) \]  
(4.30)

\[ x(t + \Delta t) = x(t) + \Delta f(x(t), u(t)) \]  
(4.31)

Shortening the above notion
\[ x(k+1) = x(k) + \hat{f}(x(k), u(k)) \]

\[ x(k+1) \approx \hat{f}(x(k), u(k)) \]  \hspace{1cm} (4.32)

In a similar manner, get performance measure form as:

\[
J = h(x(k)) + \sum_{k=0}^{N-1} (f(x(k), u(k)) + g(u(k)))
\]  \hspace{1cm} (4.33)

To minimize the deviation of the final state of system from its desired values, there are more analytical squared terms much more analytically solvable than other types. Because positive & negative deviations are equally undesirable, so absolute value could be used in quadratic form.

Using matrix notation:

\[
J = x^T(k)Hx(k) + \sum_{k=0}^{N-1} \left( x^T(k)Qx(k) + u^T(k)Ru(k) \right)
\]  \hspace{1cm} (4.34)

\[
J(x(k), u(k), P, M) = \min_{0} \left( \sum_{k=0}^{P} x^T(k)Qx(k) + \sum_{k=0}^{M} u^T(k)Ru(k) + x^T(k)Hx(k) \right)
\]  \hspace{1cm} (4.35)

Optimized solution for Eqn. (4.35) with number of intervals
Here if, k is present, k-1 is past, if k-1 is present then k is future. Here k is described as discrete or continuous function Q, H; R is real symmetric positive semi-definite $n \times n$ matrix. Q is output weighted matrix and R input weighted matrix. H is solution of Ricatti equation from linear standard state space referring Eqn. (4.18)

$$H = A^T HA - A^T HB \left( B^T HB + R \right)^{-1} BHA + Q$$

(4.37)

4.7 STABILITY FOR NMPC AND MPC

Comparing the predicted result of open and closed loop behavior is always different. An NMPC strategy that achieves stability independent of the choice of performance measure, cost function and constraints of model is desirable.

The commonly used technique for ensuring stability with MPC are terminal constrain MPC, terminal penalty constrains and terminal control MPC. The three terminal parameters stabilize the system tuning of control horizon M. The terminal penalty is employing to guarantee stabilize for closed loop system. Other terminal constrain stabilize nonlinear discreet system. Tuning of control horizon significantly play key role for tracking of prediction and closed loop system response. Small value of M, significantly mismatch between prediction and closed loop system response. The performance level enhanced for infinite horizon optimal with finite value M.
Assume that the prediction horizon and control horizon if set such that \( P \neq M \cdot \) and \( P < M \) will result in instability. The one way to achieve stability is the use of an infinite horizon cost function, i.e., \( P \) in Eqns. (4.23) and (4.36) is set at. Practically as well as theoretically this may not determine the response. More appropriate and feasible condition is \( P > M \). Similarly the model is examined under non feasible conditions, where \( P = M, P < M \). Whereas \( P = M \) exhibits somewhat admissible response, \( P < M \) exhibits a more aggressive response.

The input computed as the solution of NMPC optimization problem is equal to the closed loop trajectory of non-linear system at any given instance of time. Basic steps for infinite horizon proof are based on use of value functions. Feasibility at one sampling instance does impel for next sampling instance for the normal case.

### 4.7.1 STABILITY WITH TERMINAL STATE CONDITIONS

To ensure stability of the nonlinear tank system a terminal state constraint is added to the constraints and a terminal cost can be added to the cost function. However, to guarantee stability these two terms have to fulfill the following conditions: (i) The terminal set \( \Omega \) is an admissible positive invariant set for the system controlled by \( u_t = h(x_t) \). (ii) The terminal cost \( V_f(x_t = P) \) is a Lyapunov function such that the local control law is associated with the Lyapunov function as

\[
V(f(x_t, h(x_t))) - V(x_t) \leq -L(x_t, h(x_t))
\]

for all \( x_t \in \Omega \). This makes the nonlinear system asymptotically stable using the local control law.

### 4.8 TUNING METHODOLOGY OF NMPC

NMPC optimization is a function of input \( u \) and state variable \( x \); these two parameters are tuned externally by \( P, M \) and internally by \( Q, \lambda, R \) as per Eqn. (4.36). All these parameters are however described underneath in the following sections.
4.8.1 PREDICTION HORIZON P

Different outputs will be obtained because of the input values of P as settling time, rise time is quite different. Increasing the value of P tends to minimize controller aggressiveness. This section provides various techniques to tune the prediction horizon by References. The final horizon is set to be finite or infinite to ensure stability. In this case, the final horizon is described based on tuning result for closed loop stability of control system or process. List of guide lines are

\[ P = N + M - 1 \] \hspace{1cm} (4.39)

\[ P = t_{60} + t_{95} / T_s - 1 \] \hspace{1cm} (4.40)

\[ P = t_{80} + t_{90} / 2 / T_s \] \hspace{1cm} (4.41)

\[ P = t_{95} / T_s \] \hspace{1cm} (4.42)

\[ \eta < P < t_r / T_s \] \hspace{1cm} (4.43)

\[ P > M + t_d / T_s \] \hspace{1cm} (4.44)

All the parameters of any SISO or MIMO process are based upon the poles and zeros of transfer function and no of outputs as well as inputs. Stability, of course is a function of the transfer function of process. Tuning the value of P affects the stability value so as to make the system reach the steady state of the reference value, based on values of settling time and rise time. Normally the number of outputs and order are constant while the response stands different for different locations of poles and zeros. Proposed new tuning methods for NMPC of P are elaborated below.

\[ P = k\eta + t_r / \eta T_s \] \hspace{1cm} (4.45)

\[ P \geq \text{int}(M + C\eta k) + 1 \] \hspace{1cm} (4.46)

\[ t_r < P < T_p / k(or)\eta \] \hspace{1cm} (4.47)
By default $P = 10$ is probable value of objective function, as per stability criterion $P$ is tuned from the various parameter, like, settling time $t_s$, rise time $t_r$, no of outputs $k$, higher order of process $\eta$, no of controller intervals $C$, model horizon $N$, process response time $t_p$, sampling time $T_s$, delay time $t_d$ and response of rise time 60,80,90,95 w.r.t $t_p$. $P$ value is calculated as average of number of outputs.

The system performance and precision is analyzed by varying the prediction horizon, $P$, between 5 and 10 seconds. The effect of variation in $P$ on the system performance is discussed.

### 4.8.2 CONTROL HORIZON $M$

Evaluating the value of $M$, if it increases in value, it tends to become more aggressive over the prediction horizon ($M > P$). This is to monitor and control the response of data from output by adjusting the manipulated variable. This leads to a trade-off between increasing performance and robustness of formulation of control law, as a default control horizon is equal to 1. Formulate control horizon without more aggressiveness and existing robustness of permissible computation load. Collecting tuning methods are from reference and exhibit the result of the model. Implement those collected tunings are listed below.

\[
M = \frac{t_{60}}{T_s} \quad (4.48)
\]

\[
M = \text{int}\left(\frac{P}{4}\right) \quad (4.49)
\]

$M$ is different from $P$ and oppositely working. Response of process depends on rise time and settling time. The value of $M$ effectively tunes inputs or manipulated variable of Eqn. (4.36) and is inversely proportional to the rise time and settling time and is dependent on $P$ as well. A high value of $P$ minimizes the effect of $M$ on the response.

Apply new tuning methods based on the above equations, which are designed mainly based on parameter settling time $t_s$, rise time $t_r$, number of outputs $k$, higher order of process $\eta$, sampling time $T_s$.

\[
M = \min\left(\text{int}(t_s/2),\text{int}(P/4)\right) + 1 \quad (4.50)
\]
\[ M = \text{int} \left( kn/t_s \right) \]  \hspace{1cm} (4.51)

\[ M \propto k/t_r \]  \hspace{1cm} (4.52)

\[ M \propto \eta/t_s \]  \hspace{1cm} (4.53)

**4.8.3 OUTPUT WEIGHTED MATRIX IS REPRESENTED BY Q**

The output variables are relatively weighted according to their significance in the process model. It provides individual significance relative to output variable, with the most important variable having a larger weight compared to others. Increasing linearly the weight on the upper limit of output to achieve a smooth response till the desired output is obtained. The elements of Q that correspond to corrected error have nonzero weight to help in relative prediction. Derived expression for the output weight for minimum phase also works for non-minimum phase for the closed loop; the bandwidth is made “small” enough as explained in the reference [17] below

\[ Q = C^T C \]  \hspace{1cm} (4.54)

Based on above expression and output weights also consider smoothness, expression for both non-minimum and minimum phase will be

\[ Q < 1 \]  \hspace{1cm} (4.55)

\[ Q \leq \det |C^T C| \]  \hspace{1cm} (4.56)

Here C is output matrix of linear state space equation.

**4.8.4 WEIGHTS ON THE MAGNITUDE OF THE INPUTS R**

In similar fashion, R allows to be weighted for input variable according to their relative importance. R is normally considered as diagonal matrix with diagonal elements of \( rM \times rM \) matrix. It is referred as input weighting matrix or move suppression matrix. It is more convenient for tuning parameters based on parameter of \( r_i \) as suppression factor.

The weighting matrices ‘Q’ and ‘R’ are tuned until desired performance is achieved. This is a tradeoff between a smooth signal and a fast system performance. If a smooth signal is desirable then the ratio Q/R is to be low and if one wants a fast system then the ratio should be
greater.

As only state 1 and state 2 are to be controlled the matrix $Q$ is chosen as a diagonal matrix with nonzero values on the positions that correspond to state 1 and 2, $R$ is a diagonal 2 x 2 matrix.

$$Q = \lambda Q C^T C, \quad R = \lambda R I$$

(4.57)

4.8.5 WEIGHTS ON THE RATE OF CHANGE OF INPUTS $\lambda$

This section discusses existing and new approaches for tuning the weights on the rate of change of inputs. Penalizing the rate of change produces a more robust controller but at the cost of the controller becoming more sluggish. Small value adjustments yield a more aggressive controller. We consider tuning guideline from reference as follows.

$$\lambda < 1/mP$$

(4.58)

Based on the above approach and analyzing various guidelines, even a small change exhibits overshoots, but decrease in rise time and settling time. This is compensated by output weights.

$$\lambda < 1/\eta P$$

(4.59)

4.8.6 REFERENCE TRAJECTORY PARAMETERS $\beta$

In MPC application, reference trajectory provides the necessary path to reach final desired set point. It can be specified in several different ways. It is designed between initial value and final value between $0 \leq \beta_j < 1$, $j=1...P$.

$$\beta_j = \text{closedloopt, openloopt}$$

(4.60)

$$0 < \beta_j < 1$$

(4.61)

4.8.6.1 REFERENCE MANAGEMENT

The reference $x_{ref}$ is chosen apriority and calculated throughout the corresponding final input reference $u_{ref}$. For every sampling time the objective function tries to reach a smooth
temporary input target wt. This temporary input target is the final input reference, $u_{ref}$, run through a filter. The filter dynamics are chosen as $\frac{1}{\beta_{s+1}}$. In discrete time, the parameter $\beta$ determines how fast the filter reaches the constant reference. A small $\beta$ gives a fast filter and a big $\beta$ gives a slow filter.

$$w_{t+1} = \beta w_t + (1 - \beta) u_{ref}$$

$$\beta = e^{-(T_s/50)} \quad \ldots (4.62)$$

The function of the input reference run through the filter is shown in Figure 4.6. Therefore, a reference filter has been used to avoid it.

![Figure 4.6: Step Function of the Input Signal Reference Run through the Filter](image-url)