A Gentle Introduction to Model Predictive Control (MPC) Formulations based on Discrete Linear State Space Models

Sachin C. Patwardhan
Professor, Dept. of Chemical Engineering,
I. I. T. Bombay, Powai, Mumbai 400 076
Email: sachinp@iitb.ac.in

Contents

1 Introduction

Model Predictive Control (MPC) schemes developed in late 70's, such as dynamic matrix control (DMC) or model algorithmic control (MAC), have found wide acceptance in the process industry (Qin and Badgwell, 2003). With availability of fast computers and microprocessors, MPC is increasingly finding application in many other engineering domains such as robotics, automobiles, nuclear and aerospace industries. Major strengths of MPC are abilities to handle multivariable interactions and operating constraints in systematic manner. MPC is formulated as a constrained optimization problem, which is solved on-line repeatedly by carrying out model based forecasting over a moving window of time. More importantly, MPC facilitates optimal control of *non-square* systems, i.e. systems with unequal number of manipulated inputs and measured outputs.

Unlike many other control schemes, MPC originated from the industrial practitioners with the academia stepping in later to provide backbone of theory. Excellent reviews of MPC are available in the literature such as Garcia et al. (1989), Morari and Lee (1999) and Qin and Badgwell (2003). Yu et al. (1994) provide a tutorial introduction to linear model based MPC (LMPC) formulation. The seminal contribution by Muske and Rawlings (1993) connects the LMPC with the classical state space based linear quadratic optimal control and establishes the nominal stability using Lyapunov's method. Dynamic models are at the heart of MPC formulations. While initial formulations were based on finite impulse response or step response models, a very wide variety of linear / nonlinear black box / mechanistic models are now employed in MPC formulations. Review article by Lee (1998) gives excellent exposure to diverse model forms used in MPC formulations. Books such as Camacho and Bourdon (1999) and Rawlings and Mayne (2009) provide exposure to variety of state of the art MPC formulations and related practical / theoretical issues.

This module is meant to introduce linear MPC formulation, i.e. one based on linear perturbation models, to a student of advanced control. The linear prediction model for a system can be developed either through linearization of a nonlinear mechanistic model for the system or through identification of a black box linear time series model using input-output data generated by perturbing the system. We assume that the model development exercise is already over and a linear perturbation model for the system under consideration is available with us for controller synthesis. Origins of MPC can be traced to the classical linear quadratic optimal control (LQOC) formulation. Thus, we begin with a detailed introduction to LQOC and then move on to develop the conventional and state space formulations of MPC. We restrict our discussion to the basic algorithm and practical considerations, such as offset free tracking and regulation, and only touch upon some theoretical issues like nominal stability.

2 Dynamic Model for Controller Synthesis

Let us consider a stochastic process described by the following linear discrete state space model

$$\mathbf{x}(k+1) = \mathbf{\Phi x}(k) + \mathbf{\Gamma u}(k) + \mathbf{w}(k) \tag{1}$$

$$\mathbf{y}(k) = \mathbf{C x}(k) + \mathbf{v}(k) \tag{2}$$

where $\mathbf{x} \in R^n$ represents state variables, $\mathbf{u} \in R^m$ represents manipulated inputs, $\mathbf{y} \in R^r$ represents measured outputs and $\mathbf{w} \in R^n$ and $\mathbf{v} \in R^r$ represent unmeasured disturbances (state noise) and measurement noises, respectively. Here the vectors $\mathbf{w}(k)$ and $\mathbf{v}(k)$ are assumed to be zero mean white noise sequences such that

$$\mathbf{R}_1 = E\left[\mathbf{w}(k)\mathbf{w}(k)^T\right] \tag{3}$$

$$\mathbf{R}_{12} = E\left[\mathbf{w}(k)\mathbf{v}(k)^T\right] \tag{4}$$

$$\mathbf{R}_2 = E\left[\mathbf{v}(k)\mathbf{v}(k)^T\right] \tag{5}$$

Such a model can be derived using either from linearization of a first principles (or grey box model) or state realization of a time series model developed from input-output data.

2.1 Linearization of First Principles / Grey Box Model

Linear discrete perturbation model of the form

$$\mathbf{x}(k+1) = \mathbf{\Phi x}(k) + \mathbf{\Gamma u}(k) + \mathbf{\Gamma}_d \mathbf{d}(k) \tag{6}$$

$$\mathbf{y}(k) = \mathbf{C x}(k) + \mathbf{v}(k) \tag{7}$$

can be developed in the neighborhood of an operating point starting from a nonlinear first principles (or grey box) model. Here, $\mathbf{d} \in R^d$ represents vector of (physical) unmeasured disturbances, such as fluctuations in feed concentrations, feed flows, feed temperature etc., which are assumed to be adequately represented by piecewise constant functions. If it is further assumed that $\{\mathbf{d}(k)\}$ is a zero mean white noise sequence with known covariance matrix, say $\mathbf{Q}_d$, then we can define state noise vector $\mathbf{w}(k)$ as follows

$$\mathbf{w}(k) = \mathbf{\Gamma}_d \mathbf{d}(k) \tag{8}$$

$$E\left[\mathbf{w}(k)\right] = \overline{0} \quad ; \quad Cov\left[\mathbf{w}(k)\right] = \mathbf{R}_1 = \mathbf{\Gamma}_d \mathbf{Q}_d \mathbf{\Gamma}_d^T \tag{9}$$

The state and the measurement noise are assumed to be uncorrelated in this case i.e. $\mathbf{R}_{12} = Cov\left[\mathbf{w}(k), \mathbf{v}(k)\right]$ is a null matrix of appropriate dimension.

When a state space model derived from first principles is used for inferential control, the set of controlled outputs may differ from the set of measured output. Suppose we represent the controlled outputs $\mathbf{y}_r \in R^r$ as follows

$$\mathbf{y}_r(k) = \mathbf{C}_r \mathbf{x}(k)$$

then, in general, $\mathbf{y}_r(k)$ need not be identical with $\mathbf{y}(k)$.

Example 1 *Consider a CSTR in which the state variable vector is perturbations in reactor concentration and reactor temperature*

$$\mathbf{x} = \begin{bmatrix} \delta C_A & \delta T \end{bmatrix}^T$$

If reactor temperature is measured and it is desired to control reactor concentration, then

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

and

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

However, in the development that follows, we assume that the controlled output is same as the measured output, i.e. $\mathbf{y}_r(k) \equiv \mathbf{y}(k)$. Modifications necessary when $\mathbf{y}_r(k) \neq \mathbf{y}(k)$ are relatively straightforward and are not discussed separately.

2.2 State Realization of Time Series Model

The innovations form of state space model of the form

$$\mathbf{x}(k+1) = \mathbf{\Phi x}(k) + \mathbf{\Gamma u}(k) + \mathbf{Le}(k) \tag{10}$$

$$\mathbf{y}(k) = \mathbf{Cx}(k) + \mathbf{e}(k) \tag{11}$$

can be obtained from time series models (ARX/ARMAX/BJ) models identified from input-output data. Here, the innovations (or residuals) $\{\mathbf{e}(k)\}$ are a zero mean Gaussian white noise sequence with covariance matrix $\mathbf{V}_e$ and $\mathbf{L}$ represents the corresponding steady state Kalman gain. The above model can be re-written as

$$\mathbf{x}(k+1) = \mathbf{\Phi x}(k) + \mathbf{\Gamma u}(k) + \mathbf{w}(k) \tag{12}$$

$$\mathbf{y}(k) = \mathbf{Cx}(k) + \mathbf{v}(k) \tag{13}$$

where the vectors $\mathbf{w}(k) \in R^n$and $\mathbf{v}(k) \in R^r$ are zero mean white noise sequences such that

$$\mathbf{R}_1 = E\left[\mathbf{w}(k)\mathbf{w}(k)^T\right] = \mathbf{LV}_e^T\mathbf{L} \tag{14}$$

$$\mathbf{R}_{12} = E\left[\mathbf{w}(k)\mathbf{v}(k)^T\right] = \mathbf{LV}_e \tag{15}$$

$$\mathbf{R}_2 = E\left[\mathbf{v}(k)\mathbf{v}(k)^T\right] = \mathbf{V}_e \tag{16}$$

It may be noted that, when a state space model is identified from data, we have $\mathbf{C}_r$= $\mathbf{C}$, i.e., the set of outputs that can be controlled is identical to the set of measured outputs.

3 Quadratic Optimal Control

The above model is used as a basis for design of a state feedback controller. Steps involved in design of any state feed-back controller are as follows

1. Solve regulator / controller design problem under the assumption that full state vector is available for feedback

2. Design a state estimator and implement the state feedback control law using estimated states.

For linear systems, the closed loop stability under the nominal conditions (i.e. no model plant mismatch) is guaranteed by the separation principle.

In this section, we first describe the method for designing quadratic optimal state feedback control law and later show how the control law can be implemented using an optimal state observer (Kalman predictor). The separation principle is also briefly described. While, from a practical viewpoint, it is desirable to develop a state feedback control law which can reject drifting unmeasured disturbances and track arbitrary setpoint changes, we begin the development by considering a seemingly simplistic and idealized scenario i.e. regulation at the origin of the state space using a state feedback control law. The simplifying assumptions are then removed gradually and we move towards synthesizing a controller that achieves desired tracking and regulation for arbitrary setpoint changes and arbitrary disturbances, respectively.

3.1 Linear Quadratic Optimal State Regulator Design

We first discuss the state regulator design problem, where it is desired to bring the system form non-zero initial state to zero initial state (the origin of state space). Non-zero initial state can result from impulse like disturbances, which are sufficiently spaced in time. We want to device a state feedback regulator of type

$$\mathbf{u}(k) = -\mathbf{G}\mathbf{x}(k)$$

which will bring the system to the origin in a *optimal* fashion. For designing the regulator, we only consider the deterministic part of the model, i.e.

$$\mathbf{x}(k+1) = \mathbf{\Phi}\mathbf{x}(k) + \mathbf{\Gamma}\mathbf{u}(k) \tag{17}$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) \tag{18}$$

The regulator design problem is posed as an optimization problem where it is desired to determine control sequence $\mathbf{u}(0), \mathbf{u}(1).....\mathbf{u}(N-1)$ that takes the system to the origin in some optimal manner. To achieve this goal, we need to formulate a performance objective. When origin is the desired state and the system is at $\mathbf{x}(0) \neq \overline{\mathbf{0}}$, the prime objective is to move to the origin as quickly as possible. This can be achieved by including a term in the performance objective that penalizes distance from the origin

$$\|\mathbf{x}(k)\|_2^2 = \left[\mathbf{x}(k) - \overline{\mathbf{0}}\right]^T \left[\mathbf{x}(k) - \overline{\mathbf{0}}\right] = \mathbf{x}(k)^T\mathbf{x}(k)$$

As individual elements of the states can have different ranges of magnitudes, it is often advantageous to define a weighted distance measure of the form

$$\|\mathbf{x}(k)\|_{W_x,2}^2 = \mathbf{x}(k)^T\mathbf{W_x}\mathbf{x}(k)$$

where $\mathbf{W_x}$ is a positive definite matrix. Other important concern while designing any control law is that it should not result in excessively large control moves. Also, different manipulated inputs may have different costs associated with them and it may be desirable to use some of them sparingly while manipulate the remaining in a liberal manner. This objective can be met if we include a weighted distance measure of the form

$$\|\mathbf{u}(k)\|_{W_u,2}^2 = \mathbf{u}(k)^T \mathbf{W}_{\mathbf{u}} \mathbf{u}(k)$$

where $\mathbf{W}_{\mathbf{u}}$ is a positive definite matrix, in the performance objective while designing the controller. While defining these performance measures, we have used 2 -norm for obvious reasons of analytical tractability of the resulting optimization problem. If one is not seeking a closed form analytical solution, it is possible to express these performance measures using other norms such as 1 norm or $\infty-$norm.

The performance measures mentioned above can be used to define an quadratic objective function of the form

$$J = \sum_{k=0}^{N-1} \left[\mathbf{x}(k)^T \mathbf{W}_{\mathbf{x}} \mathbf{x}(k) + \mathbf{u}(k)^T \mathbf{W}_{\mathbf{u}} \mathbf{u}(k) \right] + \mathbf{x}(N)^T \mathbf{W}_N \mathbf{x}(N) \tag{19}$$

where $N(> n)$ is some arbitrary terminal / final sampling instant. The problem of designing an optimal controller is posed as minimization of this objective function with respect to decision variable $\{\mathbf{u}(0), \mathbf{u}(1).....\mathbf{u}(N-1)\}$. Here, $\mathbf{W}_{\mathbf{x}}, \mathbf{W}_{\mathbf{u}}$ and $\mathbf{W}_N$ are symmetric positive definite matrices. This optimization problem is solved using the **Bellman's method of dynamic programming**: *solve problem at instant (k) by assuming that problem up to time (k-1) has been solved optimally.* In order to derive the optimal control law, we start optimization from time $k = N$ and work backwards in time. Let us define $\mathbf{S}(N) = \mathbf{W}_N$ and $J(k)$

$$J(k) = \min_{\mathbf{u}(k).....\mathbf{u}(N-1)} \left\{ \sum_{i=k}^{N-1} \left[\mathbf{x}(i)^T \mathbf{W}_{\mathbf{x}} \mathbf{x}(i) + \mathbf{u}(i)^T \mathbf{W}_{\mathbf{u}} \mathbf{u}(i) \right] + \mathbf{x}(N)^T \mathbf{W}_N \mathbf{x}(N) \right\} \tag{20}$$

Thus, for $k = N$, we have

$$J(N) = \mathbf{x}(N)^T \mathbf{W}_N \mathbf{x}(N) \tag{21}$$

Then, for $k = N-1$,

$$J(N-1) = \min_{\mathbf{u}(N-1)} \left\{ \mathbf{x}(N-1)^T \mathbf{W}_{\mathbf{x}} \mathbf{x}(N-1) + \mathbf{u}(N-1)^T \mathbf{W}_{\mathbf{u}} \mathbf{u}(N-1) + J(N) \right\} \tag{22}$$

Using

$$\begin{aligned} J(N) &= \mathbf{x}(N)^T \mathbf{W}_N \mathbf{x}(N) \\ &= \left[\Phi \mathbf{x}(N-1) + \Gamma \mathbf{u}(N-1)\right]^T \mathbf{S}(N) \left[\Phi \mathbf{x}(N-1) + \Gamma \mathbf{u}(N-1)\right] \end{aligned} \tag{23}$$

it follows that

$$J(N-1)=\min_{\mathbf{u}(N-1)}\left\{\begin{array}{c}\mathbf{x}(N-1)^T\left[\mathbf{W_x}+\Phi^T\mathbf{S}(N)\Phi\right]\mathbf{x}(N-1)+\mathbf{x}(N-1)^T\Phi^T\mathbf{S}(N)\Gamma\mathbf{u}(N-1)\\+\mathbf{u}(N-1)^T\Gamma^T\mathbf{S}(N)\Phi\mathbf{x}(N-1)+\mathbf{u}(N-1)^T\left[\Gamma^T\mathbf{S}(N)\Gamma+\mathbf{W_u}\right]\mathbf{u}(N-1)\end{array}\right\} \tag{24}$$

Note that the first term $\mathbf{x}(N-1)^T\left[\mathbf{W_x}+\Phi^T\mathbf{S}(N)\Phi\right]\mathbf{x}(N-1)$ in $J(N-1)$cannot be influenced by $\mathbf{u}(N-1)$. We solve the problem of minimizing the last three terms in $J(N-1)$ by method of competing squares. In order to see how this can be achieved, consider a scalar quadratic function

$$\begin{aligned}F(\mathbf{u}) &= \mathbf{u}^T\mathbf{A}\mathbf{u}+\mathbf{z}^T\mathbf{u}+\mathbf{u}^T\mathbf{z} &(25)\\ &= \mathbf{u}^T\mathbf{A}\mathbf{u}+\mathbf{z}^T\mathbf{u}+\mathbf{u}^T\mathbf{z}+\mathbf{z}^T\mathbf{A}^{-1}\mathbf{z}-\mathbf{z}^T\mathbf{A}^{-1}\mathbf{z} &(26)\\ &= \left(\mathbf{u}+\mathbf{A}^{-1}\mathbf{z}\right)^T\mathbf{A}\left(\mathbf{u}+\mathbf{A}^{-1}\mathbf{z}\right)-\mathbf{z}^T\mathbf{A}^{-1}\mathbf{z} &(27)\end{aligned}$$

where $\mathbf{A}$ is a positive definite matrix. The first term on the right hand side in the above equation is always non-negative. This implies that minimum of $F(\mathbf{c})$with respect to $\mathbf{u}$ is attained at

$$\mathbf{u}=-\mathbf{A}^{-1}\mathbf{z} \tag{28}$$

and its minimum value is

$$F_{\min}(\mathbf{u})=-\mathbf{z}^T\mathbf{A}^{-1}\mathbf{z} \tag{29}$$

In the minimization problem at hand, with some rearrangement, we have

$$\begin{aligned}\mathbf{A} &\equiv \left[\Gamma^T\mathbf{S}(N)\Gamma+\mathbf{W_u}\right] &(30)\\ \mathbf{z} &\equiv \Gamma^T\mathbf{S}(N)\Phi\mathbf{x}(N-1) &(31)\end{aligned}$$

and the optimal solution can be expressed as

$$\begin{aligned}\mathbf{u}(N-1) &= -\mathbf{G}(N-1)\mathbf{x}(N-1) &(32)\\ \mathbf{G}(N-1) &= \left(\mathbf{W_u}+\Gamma^T\mathbf{S}(N)\Gamma\right)^{-1}\Gamma^T\mathbf{S}(N)\Phi &(33)\end{aligned}$$

which gives minimum value of the loss function,

$$J(N-1)=\mathbf{x}(N-1)^T\mathbf{S}(N-1)\mathbf{x}(N-1) \tag{34}$$

where,

$$\mathbf{S}(N-1)=\Phi^T\mathbf{S}(N)\Phi+\mathbf{W_x}-\mathbf{G}(N-1)^T\left[\mathbf{W_u}+\Gamma^T\mathbf{S}(N)\Gamma\right]\mathbf{G}(N-1) \tag{35}$$

Similar arguments yield the next optimization problem

$$J(N-2)=\min_{\mathbf{u}(N-2)}\left\{\mathbf{x}(N-2)^T\mathbf{W_x}\mathbf{x}(N-2)+\mathbf{u}(N-2)^T\mathbf{W_u}\mathbf{u}(N-2)+J(N-1)\right\} \tag{36}$$

This minimization problem is similar to the earlier optimization problem, but with the time argument shifted. As a consequence, the solution can be constructed in an identical manner by repeating the procedure backward in time. Thus, at k'th instant, the optimal gain matrix can be computed as

$$\mathbf{G}(k) = \left(\mathbf{W_u} + \Gamma^T \mathbf{S}(k+1)\Gamma\right)^{-1} \Gamma^T \mathbf{S}(k+1)\Phi \tag{37}$$

and further used to compute

$$\mathbf{S}(k) = [\Phi - \Gamma \mathbf{G}(k)]^T \mathbf{S}(k+1) [\Phi - \Gamma \mathbf{G}(k)] + \mathbf{W_x} + \mathbf{G}(k)^T \mathbf{W_u} \mathbf{G}(k) \tag{38}$$

The later equations is called as the discrete time Riccati equation. The matrices $\mathbf{S}(N) = \mathbf{W}_N$ and $\mathbf{W_u}$ are assumed to be positive definite and symmetric. This implies that $\mathbf{S}(k)$ is positive definite/semi-definite and symmetric and this condition guarantees optimality at each stage. When horizon N becomes large and the system matrices obey certain regularity conditions, $\mathbf{S}(k)$ tends to a constant matrix, $\mathbf{S}(k) \rightarrow \mathbf{S}_\infty$, which can be computed by solving the algebraic Riccati equation (ARE)

$$\mathbf{G}_\infty = \left(\mathbf{W_u} + \Gamma^T \mathbf{S}_\infty \Gamma\right)^{-1} \Gamma^T \mathbf{S}_\infty \Phi \tag{39}$$

$$\mathbf{S}_\infty = [\Phi - \Gamma \mathbf{G}_\infty]^T \mathbf{S}_\infty [\Phi - \Gamma \mathbf{G}_\infty] + \mathbf{W_x} + \mathbf{G}_\infty^T \mathbf{W_u} \mathbf{G}_\infty \tag{40}$$

This ARE has several solutions. However, if (Φ, Γ) is reachable and if $(\Phi, \mathbf{\Sigma})$ is observable pair, where $\mathbf{W_u} = \mathbf{\Sigma}^T \mathbf{\Sigma}$, then there exists a unique, symmetric, non-negative definite solution to the ARE. The corresponding state feedback control law can be formulated as

$$\mathbf{u}(k) = -\mathbf{G}_\infty \mathbf{x}(k) \tag{41}$$

Further, when (Φ, Γ) is controllable and objective function is symmetric and positive definite, the linear quadratic (LQ) controller will always give asymptotically stable closed loop behavior. By selecting $\mathbf{W_x}$ and $\mathbf{W_u}$ appropriately, it is easy to compromise between speed of recovery and magnitude of control signals.

3.2 Linear Quadratic Optimal Output Regulator Design

In many situations we are only interested in controlling certain outputs of a system. Moreover, when the state space model is identified from input-output data, the states may not have a physical meaning and it is convenient to define the regulatory control problem in terms of measured outputs. In such situations, the objective function given by equation (19) can be modified as follows

$$J = E\left\{ \sum_{k=0}^{N-1} \left[\mathbf{y}(k)^T \mathbf{W_y} \mathbf{y}(k) + \mathbf{u}(k)^T \mathbf{W_u} \mathbf{u}(k)\right] + \mathbf{y}(N)^T \mathbf{W}_{\mathbf{y}N} \mathbf{y}(N) \right\} \tag{42}$$

The above modified objective function can be rearranged as follows

$$J = E\left\{\sum_{k=0}^{N-1}\left[\mathbf{x}(k)^T\left[\mathbf{C}^T\mathbf{W_y}\mathbf{C}\right]\mathbf{x}(k) + \mathbf{u}(k)^T\mathbf{W_u}\mathbf{u}(k)\right] + \mathbf{x}(N)^T\left[\mathbf{C}^T\mathbf{W_{yN}}\mathbf{C}\right]\mathbf{x}(N)\right\} \tag{43}$$

and by setting

$$\mathbf{W_x} = \left[\mathbf{C}^T\mathbf{W_y}\mathbf{C}\right] \quad ; \quad \mathbf{W_N} = \left[\mathbf{C}^T\mathbf{W_{yN}}\mathbf{C}\right]$$

we can use the Riccati equations derived above for controller design.

3.3 Stability of LQ controller

Theorem 2 ***(Stability of the closed loop system) :*** *Consider the time invariant system given by equation*

$$\mathbf{x}(k+1) = \mathbf{\Phi}\mathbf{x}(k) + \mathbf{\Gamma}\mathbf{u}(k)$$

and the loss function for the optimal control is given by equation (19). Assume that a positive-definite steady state solution $\mathbf{S}_\infty$ *exists for Riccati equation (40). Then, the steady state optimal strategy*

$$\mathbf{u}(k) = -\mathbf{G}_\infty\mathbf{x}(k) = -\left[\mathbf{W_u} + \Gamma^T\mathbf{S}_\infty\Gamma\right]^{-1}\Gamma^T\mathbf{S}_\infty\Phi\mathbf{x}(k)$$

gives an asymptotically stable closed-loop system

$$\mathbf{x}(k+1) = (\Phi - \Gamma\mathbf{G}_\infty)\mathbf{x}(k)$$

Proof. Theorem A.3 in Appendix can be used to show that the closed loop system is asymptotically stable. To prove asymptotic stability, it is sufficient to show that the function

$$V(\mathbf{x}(k)) = \mathbf{x}^T(k)\mathbf{S}_\infty\mathbf{x}(k)$$

is a Lyapunov function and $\triangle V(\mathbf{x}(k))$ is strictly -ve definite. Since $\mathbf{S}_\infty$ is +ve definite, $V(\mathbf{x}(k))$ is positive definite and ■

$$\begin{aligned}\triangle V(\mathbf{x}(k)) &= \mathbf{x}^T(k+1)\mathbf{S}_\infty\mathbf{x}(k+1) - \mathbf{x}^T(k)\mathbf{S}_\infty\mathbf{x}(k)\\ &= \mathbf{x}^T(k)(\Phi - \Gamma\mathbf{G}_\infty)^T\mathbf{S}_\infty(\Phi - \Gamma\mathbf{G}_\infty)\mathbf{x}(k) - \mathbf{x}^T(k)\mathbf{S}_\infty\mathbf{x}(k)\\ &= -\mathbf{x}^T(k)[\mathbf{W_x} + \mathbf{G}_\infty^T\mathbf{W_u}\mathbf{G}_\infty]\mathbf{x}(k)\end{aligned}$$

This follows from the ARE (i.e. equation (40)) that

$$\triangle V(\mathbf{x}(k)) = -\mathbf{x}^T(k)[\mathbf{W_x} + \mathbf{G}_\infty^T\mathbf{W_u}\mathbf{G}_\infty]\mathbf{x}(k)$$

Because $\mathbf{W_x}+\mathbf{G}_\infty^T\mathbf{W_u}\mathbf{G}_\infty$ is positive definite, $\triangle V$ is strictly negative definite. Thus, the closed loop system is asymptotically stable for **any choice of** positive definite $\mathbf{W_x}$ and positive semi-definite $\mathbf{W_u}$.

The poles of the closed loop system can be obtained in several ways. When the design is completed, the poles are obtained from

$$det(\lambda I - \Phi + \Gamma\mathbf{G}_\infty) = 0$$

It is possible to show that the poles are the n stable eigenvalues of the generalized eigenvalue problem.

$$\left\{\begin{bmatrix} I & 0 \\ \mathbf{W_x} & \Phi^T \end{bmatrix}\lambda - \begin{bmatrix} \Phi & -\Gamma\mathbf{W_u}^{-1}\Gamma \\ 0 & I \end{bmatrix}\right\} = 0$$

This equation is called the Euler equation of the LQ problem. Theorem 1 shows that LQ controller gives a stable closed loop system, i.e. all poles of $\Phi - \Gamma\mathbf{G}_\infty$ are strictly inside unit circle. Thus, a LQ controller optimal controller guarantees performance as well as asymptotic stability under the nominal conditions.

3.4 Linear Quadratic Gaussian (LQG) Control

3.4.1 State Estimation

The model (1-2) can be used to develop the optimal state predictor as follows

$$\begin{aligned} \mathbf{e}(k) &= \mathbf{y}(k) - \mathbf{C}\widehat{\mathbf{x}}(k|k-1) & (44) \\ \widehat{\mathbf{x}}(k+1|k) &= \mathbf{\Phi}\widehat{\mathbf{x}}(k|k-1) + \mathbf{\Gamma}_u\mathbf{u}(k) + \mathbf{L}_{p,\infty}\mathbf{e}(k) & (45) \end{aligned}$$

where $\mathbf{L}_p$ is the solution of the steady state Riccati equation

$$\begin{aligned} \mathbf{L}_{p,\infty} &= \left[\mathbf{\Phi}\mathbf{P}_\infty\mathbf{C^T} + \mathbf{R_{12}}\right]\left[\mathbf{C}\mathbf{P}_\infty\mathbf{C^T} + \mathbf{R}_2\right]^{-1} & (46) \\ \mathbf{P}_\infty &= \mathbf{\Phi}\mathbf{P}_\infty\mathbf{\Phi}^T + \mathbf{R}_1 - \mathbf{L}_{p,\infty}\left[\mathbf{C}\mathbf{P}_\infty\mathbf{C^T} + \mathbf{R}_2\right]\mathbf{L}_p^T & (47) \end{aligned}$$

Here, matrix $\mathbf{P}_\infty$ denotes steady state covariance of error in state estimation. It can be shown that the residual (or innovation) $\{\mathbf{e}(k)\}$ is a zero mean Gaussian white noise process with covariance matrix $\mathbf{V}_\infty = \left[\mathbf{C}\mathbf{P}_\infty\mathbf{C^T} + \mathbf{R}_2\right]$. The state feedback LQ control law designed above is implemented together with the Kalman predictor using estimated states as follows

$$\mathbf{u}(k) = -\mathbf{G}_\infty\widehat{\mathbf{x}}(k|k-1) \tag{48}$$

It may be noted that

$$\begin{aligned} \mathbf{u}(k) &= -\mathbf{G}_\infty\widehat{\mathbf{x}}(k|k) = -\mathbf{G}_\infty\left[\mathbf{x}(k) - \boldsymbol{\varepsilon}(k|k-1)\right] \\ &= -\mathbf{G}_\infty\mathbf{x}(k) + \mathbf{G}_\infty\boldsymbol{\varepsilon}(k|k-1) & (49) \end{aligned}$$

Thus, the state feedback control law (41) is now modified to account for mismatch between the true states and the estimated states. Before proceeding with the discussion about using the estimator in combination with the observer, let us recall the following results discussed in the notes on introduction to state estimation.

Theorem 3 *Assume that pair* $(\Phi, \sqrt{\mathbf{Q}})$ *is stabilizable and pair* $(\Phi, \mathbf{C})$ *is detectable. Then, the solution* $\mathbf{P}(k|k-1)$ *of the Riccati equations*

$$\begin{aligned}\mathbf{L}_p(k) &= \left[\mathbf{\Phi P}(k|k-1)\mathbf{C^T} + \mathbf{R_{12}}\right]\left[\mathbf{CP}_\infty\mathbf{C^T} + \mathbf{R}_2\right]^{-1} &(50)\\ \mathbf{P}(k+1|k) &= \mathbf{\Phi P(k|k-1)\Phi}^T + \mathbf{R}_1 - \mathbf{L}_p(k)\left[\mathbf{CP(k|k-1)C^T} + \mathbf{R}_2\right]\mathbf{L}_p(k)^T &(51)\end{aligned}$$

tends to $\mathbf{P}_\infty$ *as* $k \to \infty$.

Theorem 4 *Assume that pair* $(\Phi, \sqrt{\mathbf{Q}})$ *is reachable and* $\mathbf{R}_2$ *is non-singular. Then all eigenvalues of* $(\mathbf{\Phi} - \mathbf{L}_{p,\infty}\mathbf{C})$ *are inside the unit circle.*

Alternatively, model (1-2) can be used to develop the optimal current state estimator as follows

$$\begin{aligned}\mathbf{e}(k) &= \mathbf{y}(k) - \mathbf{C}\widehat{\mathbf{x}}(k|k-1) &(52)\\ \widehat{\mathbf{x}}(k|k-1) &= \mathbf{\Phi}\widehat{\mathbf{x}}(k-|k-1) + \mathbf{\Gamma}_u\mathbf{u}(k-1) &(53)\\ \widehat{\mathbf{x}}(k|k) &= \widehat{\mathbf{x}}(k|k-1) + \mathbf{L}_c\mathbf{e}(k) &(54)\end{aligned}$$

where $\mathbf{L}_c$ is the solution of the steady state Riccati equation

$$\begin{aligned}\mathbf{P}_{1,\infty} &= \mathbf{\Phi P}_{0,\infty}\mathbf{\Phi}^T + \mathbf{R}_1 &(55)\\ \mathbf{L}_c &= \left[\mathbf{\Phi P}_{1,\infty}\mathbf{C^T} + \mathbf{R_{12}}\right]\left[\mathbf{CP}_{1,\infty}\mathbf{C^T} + \mathbf{R}_2\right]^{-1} &(56)\\ \mathbf{P}_{0,\infty} &= \left[\mathbf{I} - \mathbf{L}_c\mathbf{C}\right]\mathbf{P}_{1,\infty} &(57)\end{aligned}$$

We can then use Kalman filer (current state estimator) to implement control law as follows

$$\mathbf{u}(k) = -\mathbf{G}_\infty\widehat{\mathbf{x}}(k|k) \tag{58}$$

Combination of LQ controller and Kalman filter is referred to as linear quadratic Gaussian (LQG) controller.

It may be noted that, if the state space model has been derived a realization of ARMAX / BJ type time series model, then the identified model (10-11) can be directly used to arrive at the observer of the form (44-45).

3.4.2 Separation Principle and Nominal Closed Loop Stability

To assess the nominal stability of the closed loop generated by observer - regulator pair, the observer and the plant dynamics has be considered together. For example, the closed loop system with Kalman predictor can be described by following set of equations

$$\mathbf{x}(k+1) = \mathbf{\Phi x}(k) + \mathbf{\Gamma u}(k) + \mathbf{w}(k) \tag{59}$$

$$\mathbf{y}(k) = \mathbf{Cx}(k) + \mathbf{v}(k) \tag{60}$$

$$\widehat{\mathbf{x}}(k+1|k) = \mathbf{\Phi}\widehat{\mathbf{x}}(k|k-1) + \mathbf{\Gamma u}(k) + \mathbf{L}\left[\mathbf{y}(k) - \mathbf{C}\widehat{\mathbf{x}}(k|k-1)\right] \tag{61}$$

Let us define state estimation error

$$\boldsymbol{\varepsilon}(k|k-1) = \mathbf{x}(k) - \widehat{\mathbf{x}}(k|k-1) \tag{62}$$

It is easy to show that the dynamics of the estimation error is governed by

$$\boldsymbol{\varepsilon}(k+1|k) = [\mathbf{\Phi} - \mathbf{LC}]\boldsymbol{\varepsilon}(k|k-1) + \mathbf{w}(k) - \mathbf{Lv}(k)$$

Combining the error dynamics with system dynamics and rearranging, we arrive at the following equation governing the *closed loop dynamics*

$$\begin{bmatrix} \mathbf{x}(k+1) \\ \boldsymbol{\varepsilon}(k+1|k) \end{bmatrix} = \begin{bmatrix} [\mathbf{\Phi} - \mathbf{\Gamma G}_\infty] & \mathbf{\Gamma G}_\infty \\ [0] & [\mathbf{\Phi} - \mathbf{LC}] \end{bmatrix} \begin{bmatrix} \mathbf{x}(k) \\ \boldsymbol{\varepsilon}(k|k-1) \end{bmatrix} + \begin{bmatrix} \mathbf{I} \\ \mathbf{I} \end{bmatrix} \mathbf{w}(k) + \begin{bmatrix} [\mathbf{0}] \\ -\mathbf{L} \end{bmatrix} \mathbf{v}(k) \tag{63}$$

Thus, the unforced dynamics of LQG controller is determined by dynamics of LQ controller is governed by eigenvalues of matrix

$$\mathbf{\Phi}_c = \begin{bmatrix} [\mathbf{\Phi} - \mathbf{\Gamma G}_\infty] & \mathbf{\Gamma G}_\infty \\ [\mathbf{0}] & [\mathbf{\Phi} - \mathbf{LC}] \end{bmatrix} \tag{64}$$

Since

$$\det\left[\lambda\mathbf{I} - \mathbf{\Phi}_c\right] = \det\left[\lambda\mathbf{I} - (\mathbf{\Phi} - \mathbf{\Gamma G}_\infty)\right]\det\left[\lambda\mathbf{I} - (\mathbf{\Phi} - \mathbf{LC})\right] \tag{65}$$

it implies that the eigen values of matrices $[\mathbf{\Phi} - \mathbf{\Gamma G}_\infty]$ and $[\mathbf{\Phi} - \mathbf{LC}]$ determine the stability characteristics of the nominal closed loop system. Thus, designing state feedback controller and state observer to be individually stable ensures stability of the closed loop (*separation principle*). As a consequence, if the system under consideration is observable and controllable, then the resulting closed loop is guaranteed to be stable if weighting matrices in LQ formulation are chosen to be positive definite / semi-definite and the Kalman predictor (or filter) is used for state estimation.

It may be noted that the separation principle does not restrict the type of observer used for state estimation or the method used for designing the state feedback control law. For example, the observer can be a Luenberger observer designed using the pole placement approach such that

$$\rho[\mathbf{\Phi} - \mathbf{LC}] < \mathbf{1}$$

Also, the feedback gain matrix $\mathbf{G}$ can be obtained using pole placement approach such that

$$\rho[\mathbf{\Phi} - \mathbf{\Gamma G}] < 1$$

The nominal closed loop stability follows from the equation

$$\det\left[\lambda\mathbf{I} - \mathbf{\Phi}_c\right] = \det\left[\lambda\mathbf{I} - (\mathbf{\Phi} - \mathbf{\Gamma G})\right]\det\left[\lambda\mathbf{I} - (\mathbf{\Phi} - \mathbf{LC})\right] \tag{66}$$

which implies that designing a stable state feedback controller and a stable state observer ensures stability of the closed loop. When the system under consideration is open loop stable, one can also use the open loop observer

$$\widehat{\mathbf{x}}(k+1|k) = \mathbf{\Phi}\widehat{\mathbf{x}}(k|k-1) + \mathbf{\Gamma}_u\mathbf{u}(k) \tag{67}$$

and implement the state feedback control law while ensuring the nominal closed loop stability. The main advantage of using LQ approach for designing the control law and Kalman predictor / filter for state estimation is that nominal stability and performance are ensured simultaneously.

3.5 Tracking and Regulation using Quadratic Optimal Controller

Linear quadratic regulator designed above can generate an offset if (a) the unmeasured disturbances are non-stationary, i.e. they have slowly drifting behavior (b) mismatch exists between the plant and the model. Thus, it becomes necessary to introduce integral action in the control to deal with plant-model mismatch and reject the drifting unmeasured disturbances. Also, the regulator designed above only solves the restricted problem of moving the system from any initial state to the origin. If it is desired to move the system from any initial condition to an arbitrary setpoint, the state feedback control laws has to be modified. The problem of regulation in the face of unknown disturbances / plant-model mismatch and tracking an arbitrary setpoint trajectory is solved by modifying the regulatory control law as follows

$$\mathbf{u}(k) - \mathbf{u}_s = -\mathbf{G}_\infty\left[\mathbf{x}(k) - \mathbf{x}_s\right] \tag{68}$$

$$\mathbf{u}(k) = \mathbf{u}_s - \mathbf{G}_\infty\left[\mathbf{x}(k) - \mathbf{x}_s\right] \tag{69}$$

where $\mathbf{x}_s$ represent the *steady state target* corresponding to the setpoint, say $\mathbf{r}$, and $\mathbf{u}_s$ represents the *steady state input* necessary to reach this steady state target. This is equivalent to the change of origin. In this section, we discuss various approaches to incorporate regulation in the face of of drifting disturbances and tracking arbitrary setpoint changes.

3.5.1 Model Transformation for Output Regulation and Tracking

Before we move to handle unknown drifting disturbances and tracking arbitrary setpoint trajectories, let us consider a simplified scenario. Consider the problem of designing a LQ controller for a system

$$\mathbf{x}(k+1) = \mathbf{\Phi x}(k) + \mathbf{\Gamma u}(k) + \mathbf{\Gamma}_\beta \boldsymbol{\beta}_s \tag{70}$$

$$\mathbf{y}(k) = \mathbf{Cx}(k) + \mathbf{C}_\eta \boldsymbol{\eta}_s \tag{71}$$

where $\boldsymbol{\beta}_s$represents input disturbance vector and $\boldsymbol{\eta}_s$ represents additive output disturbance vector. It is further assumed matrices $(\mathbf{\Gamma}_\beta, \mathbf{C}_\eta)$ and vectors $(\boldsymbol{\beta}_s, \boldsymbol{\eta}_s)$ are known *apriori*. It is desired to control this system at an arbitrarily specified setpoint $\mathbf{r}\left(\neq \overline{\mathbf{0}}\right)$, i.e., $\mathbf{y}(k) = \mathbf{r}$ as $k \to \infty$. The steady state behavior of this system is given by

$$\mathbf{x}_s = \mathbf{\Phi x}_s + \mathbf{\Gamma u}_s + \mathbf{\Gamma}_\beta \boldsymbol{\beta}_s \tag{72}$$

or

$$\mathbf{x}_s = [\mathbf{I} - \mathbf{\Phi}]^{-1} [\mathbf{\Gamma u}_s + \mathbf{\Gamma}_\beta \boldsymbol{\beta}_s] \tag{73}$$

Since we require $\mathbf{y}(k) = \mathbf{r}$ as $k \to \infty$, it follows that at the steady state

$$\mathbf{r} = \mathbf{Cx}_s + \mathbf{C}_\eta \boldsymbol{\eta}_s \tag{74}$$

$$\mathbf{r} = \mathbf{C}[\mathbf{I} - \mathbf{\Phi}]^{-1} [\mathbf{\Gamma u}_s + \mathbf{\Gamma}_\beta \boldsymbol{\beta}_s] + \mathbf{C}_\eta \boldsymbol{\eta}_s \tag{75}$$

$$= \mathbf{K}_u \mathbf{u}_s + \mathbf{K}_\beta \boldsymbol{\beta}_s + \mathbf{C}_\eta \boldsymbol{\eta}_s \tag{76}$$

where

$$\mathbf{K}_u = \mathbf{C}[\mathbf{I} - \mathbf{\Phi}]^{-1} \mathbf{\Gamma} \quad \text{and} \quad \mathbf{K}_\beta = \mathbf{C}[\mathbf{I} - \mathbf{\Phi}]^{-1} \mathbf{\Gamma}_\beta$$

represent the steady state gain matrices. When the number of manipulated inputs equals the number of controlled outputs and $\mathbf{K}_u$ is invertible, it follows that

$$\mathbf{u}_s = \mathbf{K}_u^{-1} [\mathbf{r} - \mathbf{K}_\beta \boldsymbol{\beta}_s - \mathbf{C}_\eta \boldsymbol{\eta}_s] \tag{77}$$

$$\mathbf{x}_s = [\mathbf{I} - \mathbf{\Phi}]^{-1} \left(\mathbf{K}_u^{-1} [\mathbf{r} - \mathbf{K}_\beta \boldsymbol{\beta}_s - \mathbf{C}_\eta \boldsymbol{\eta}_s] + \mathbf{K}_\beta \boldsymbol{\beta}_s \right) \tag{78}$$

Now, subtracting the steady state model equations (72-74) from the dynamic model equation (70-71), we have

$$\begin{aligned} \mathbf{x}(k+1)-\mathbf{x}_s &= \mathbf{\Phi}\left[\mathbf{x}(k)-\mathbf{x}_s\right]+\mathbf{\Gamma}\left[\mathbf{u}(k)-\mathbf{u}_s\right] && (79)\\ \left[\mathbf{y}(k)-\mathbf{r}\right] &= \mathbf{C}\left[\mathbf{x}(k)-\mathbf{x}_s\right] && (80) \end{aligned}$$

Defining new perturbation variables $\Delta\mathbf{x}(\mathbf{k})=\left[\mathbf{x}(k)-\mathbf{x}_s\right]$, $\mathbf{\Delta u}(k)=\left[\mathbf{u}(k)-\mathbf{u}_s\right]$ and $\Delta\mathbf{y}(k)=\left[\mathbf{y}(k)-\mathbf{r}\right]$, we have a transformed system

$$\begin{aligned} \Delta\mathbf{x}(k+1) &= \mathbf{\Phi}\mathbf{\Delta x}(k)+\mathbf{\Gamma}\mathbf{\Delta u}(k) && (81)\\ \Delta\mathbf{y}(k) &= \mathbf{C}\mathbf{\Delta x}(k) && (82) \end{aligned}$$

We can now develop a LQ controller for the transformed system and arrive at a control law of the form

$$\mathbf{\Delta u}(k)=-\mathbf{G}_{\infty}\mathbf{\Delta x}(k) \tag{83}$$

which regulates the transformed system at the origin, $\mathbf{\Delta x}=\overline{\mathbf{0}}$, which is equivalent to achieving offset free behavior for the original system for the specified stepoint and for the given disturbance level. The LQ control law that solves servo and regulatory problem simultaneously can be expressed as follows

$$\mathbf{u}(k)=\mathbf{u}_s-\mathbf{G}_{\infty}\left[\mathbf{x}(k)-\mathbf{x}_s\right] \tag{84}$$

The situation where $(\boldsymbol{\beta}_s,\boldsymbol{\eta}_s)$ are slowly time varying and / or the setpoint is $\mathbf{r}$ is time varying van be handled by introducing *time varying target steady states* and *time varying target steady state inputs* as follows

$$\mathbf{u}_s(k)=\mathbf{K}_u^{-1}\left[\mathbf{r}(k)-\mathbf{K}_{\beta}\boldsymbol{\beta}_s(k)-\mathbf{C}_{\eta}\boldsymbol{\eta}_s(k)\right] \tag{85}$$

$$\mathbf{x}_s(k)=\left[\mathbf{I}-\mathbf{\Phi}\right]^{-1}\left(\mathbf{K}_u^{-1}\left[\mathbf{r}(\mathbf{k})-\mathbf{K}_{\beta}\boldsymbol{\beta}_s(k)-\mathbf{C}_{\eta}\boldsymbol{\eta}_s(k)\right]+\mathbf{K}_{\beta}\boldsymbol{\beta}_s(k)\right) \tag{86}$$

and modifying the control law as follows

$$\mathbf{u}(k)=\mathbf{u}_s(k)-\mathbf{G}_{\infty}\left[\mathbf{x}(k)-\mathbf{x}_s(k)\right] \tag{87}$$

Note that, the above modification implicitly assumes that unmeasured disturbances remain constant in future i.e.

$$\begin{aligned} \boldsymbol{\beta}_s(k+j+1) &= \boldsymbol{\beta}_s(k+j)\\ \boldsymbol{\eta}_s(k+j+1) &= \boldsymbol{\eta}_s(k+j) \end{aligned}$$

for $j=0,1,2,.....\infty$. This modification facilitates regulation in the face of drifting measured disturbances and time varying setpoints.

3.5.2 Dealing with Unmeasured Disturbances and Model Plant Mismatch

The formulation presented in the previous section caters to the case where the unmeasured disturbances are measured. In practice, however, systems are invariably subjected to (additional) drifting unmeasured disturbances and model plant mismatch. Let us assume that a Kalman predictor is developed using the nominal set of model parameters as follows

$$\mathbf{e}(k) = \mathbf{y}(k) - \mathbf{C}\widehat{\mathbf{x}}(k|k-1) \tag{88}$$

$$\widehat{\mathbf{x}}(k+1|k) = \mathbf{\Phi}\widehat{\mathbf{x}}(k|k-1) + \mathbf{\Gamma}\mathbf{u}(k) + \mathbf{L}\mathbf{e}(k) \tag{89}$$

In the absence of model plant mismatch or drifting unmeasured disturbances, it can be shown that the innovation signal $\mathbf{e}(k)$ is a zero mean Gaussian process. The innovation sequence, however, is no longer a zero mean Gaussian process in the presence of following scenarios, which are referred to as Model Plant Mismatch (MPM) in the rest of the text.

- **Plant-model parameter mismatch:** Plant dynamics evolves according to

$$\widetilde{\mathbf{x}}(k+1) = \overline{\mathbf{\Phi}}\widetilde{\mathbf{x}}(k) + \overline{\mathbf{\Gamma}}\mathbf{u}(k) + \mathbf{w}(k) \tag{90}$$

$$\mathbf{y}(k) = \overline{\mathbf{C}}\widetilde{\mathbf{x}}(k) + \mathbf{v}(k) \tag{91}$$

  where $(\overline{\mathbf{\Phi}}, \overline{\mathbf{\Gamma}}, \overline{\mathbf{C}})$ are different from $(\mathbf{\Phi}, \mathbf{\Gamma}, \mathbf{C})$ used in the model

- **Plant-model structural mismatch:** Models for controller synthesis are often low order approximations of the plant dynamics in the desired operating range. This results in a structural mismatch between the plant model. Thus, the system under consideration could be evolving as

$$\boldsymbol{\xi}(k+1) = \widetilde{\mathbf{\Phi}}\boldsymbol{\xi}(k) + \widetilde{\mathbf{\Gamma}}\mathbf{u}(k) + \widetilde{\mathbf{w}}(k) \tag{92}$$

$$\mathbf{y}(k) = \widetilde{\mathbf{C}}\boldsymbol{\xi}(k) + \mathbf{v}(k) \tag{93}$$

  while the controller synthesis is based on (1-2). Here, dimensions of the plant state vector $(\boldsymbol{\xi})$ are different from the dimensions of the model state vector $(\mathbf{x})$. As a consequence, not just the entrees but the dimensions or $(\widetilde{\mathbf{\Phi}}, \widetilde{\mathbf{\Gamma}}, \widetilde{\mathbf{C}})$ are different from $(\mathbf{\Phi}, \mathbf{\Gamma}, \mathbf{C})$.

- **Unmeasured drifting (colored) disturbances:** Plant dynamics is affected by some unknown drifting colored disturbance $\mathbf{d}(k)$

$$\widetilde{\mathbf{x}}(k+1) = \mathbf{\Phi}\widetilde{\mathbf{x}}(k) + \mathbf{\Gamma}\mathbf{u}(k) + \mathbf{\Gamma}_d\mathbf{d}(k) + \mathbf{w}(k) \tag{94}$$

$$\mathbf{y}(k) = \mathbf{C}\widetilde{\mathbf{x}}(k) + \mathbf{v}(k) \tag{95}$$

  which is an autocorrelated (colored) stochastic process and which has not been accounted for in the model.

- **Nonlinear plant dynamics:** Dynamics of most real systems are typically nonlinear and this leads to errors due to approximation of dynamics using a local linear model

It is possible to modify the formulation presented in the previous section to deal with these scenarios. There are two approaches to estimate the moving steady state targets and implement the control law using a state observer:

- Innovation bias approach: Filtered innovation signal $\mathbf{e}(k) = \mathbf{y}(k) - \widehat{\mathbf{y}}(k|k-1)$ is used as a proxy for the unmeasured disturbances

- State augmentation approach : By this approach disturbance vectors $(\boldsymbol{\beta}_s(k), \boldsymbol{\eta}_s(k))$ are estimated together with the states

We describe these approaches in detail.

Innovation Bias Approach In the presence of MPM, the sequence $\{\mathbf{e}(k)\}$ becomes colored and has significant power at the control relevant low frequency region. The low frequency drifting mean of $\{\mathbf{e}(k)\}$ can be estimated using a simple unity gain first order filter of the form

$$\mathbf{e}_f(k) = \boldsymbol{\Phi}_e \; \mathbf{e}_f(k-1) + [\mathbf{I} - \boldsymbol{\Phi}_e]\mathbf{e}(k) \tag{96}$$

$$\boldsymbol{\Phi}_e = diag \begin{bmatrix} \alpha_1 & \alpha_2 & ... & \alpha_r \end{bmatrix}$$

$$0 \leq \alpha_i < 1 \quad \text{for} \quad i = 1, 2, ..., r \text{ are tuning parameters}$$

where $0 \leq \alpha_i < 1$ are tuning parameters. This filtered signal can be taken as a proxy for the low frequency unmeasured disturbances / model plant mismatch. Thus, we assume

- unmeasured disturbance in the state dynamics, $\boldsymbol{\beta}_s(k) \equiv \mathbf{e}_f(k)$ and $\boldsymbol{\Gamma}_\beta \equiv \mathbf{L}$

- unmeasured disturbance in the outputs, $\boldsymbol{\eta}_s(k) \equiv \mathbf{e}_f(k)$ and $\mathbf{C}_\eta \equiv \mathbf{I}_r$

and use control law (87) together with equations (86) and (85). It is also assumed that these disturbances remain constant over the future, i.e. these disturbances behave according the following linear difference equations

$$\begin{aligned} \boldsymbol{\beta}(k+j+1) &= \boldsymbol{\beta}(k+j) \quad \text{for} \quad j = 1, 2, 3, ... & (97) \\ \boldsymbol{\beta}(k) &= \mathbf{e}_f(k) & (98) \end{aligned}$$

and

$$\begin{aligned} \boldsymbol{\eta}(k+j+1) &= \boldsymbol{\eta}(k+j) \quad \text{for} \quad j = 1, 2, 3, ... & (99) \\ \boldsymbol{\eta}(k) &= \mathbf{e}_f(k) & (100) \end{aligned}$$

The choice of this parameters α_i influences the regulatory behavior of the LQG controller and incorporate robustness against the MPM. Typical range of values for α_i is between 0.8 to 0.99. Thus, the LQG control law is modified as follows

$$\mathbf{u}(k) = \mathbf{u}_s(k) - \mathbf{G}\ [\widehat{\mathbf{x}}(k|k-1) - \mathbf{x}_s(k)]$$

where $\mathbf{x}_s(k), \mathbf{u}_s(k)$ are computed by solving the following set of equations

$$\begin{aligned} \mathbf{x}_s(k) &= \mathbf{\Phi}\mathbf{x}_s(k) + \mathbf{\Gamma}_u\mathbf{u}_s(k) + \mathbf{L}\mathbf{e}_f(k) && (101) \\ \mathbf{r}(k) &= \mathbf{C}\mathbf{x}_s(k) + \mathbf{e}_f(k) && (102) \end{aligned}$$

When $\mathbf{\Phi}$ has no poles on the unit circle, the above equation reduce to

$$\begin{aligned} \mathbf{u}_s(k) &= \mathbf{K}_u^{-1}\left[\mathbf{r}(k) - \mathbf{K}_e\,\mathbf{e}_f(k)\right] && (103) \\ \mathbf{x}_s(k) &= (\mathbf{I} - \mathbf{\Phi})^{-1}\ \left[\mathbf{\Gamma}_u\mathbf{u}_s(k) + \mathbf{L}\mathbf{e}_f(k)\right] && (104) \end{aligned}$$

where

$$\mathbf{K}_u = \mathbf{C}\left(\mathbf{I} - \mathbf{\Phi}\right)^{-1}\Gamma_u \quad ; \qquad \mathbf{K}_e = \mathbf{C}\left(\mathbf{I} - \mathbf{\Phi}\right)^{-1}\mathbf{L} + \mathbf{I}$$

This control law can be used for setpoint tracking as well as unknown disturbance rejection. This approach for dealing with unmeasured disturbances is referred to as *innovation bias formulation* in these notes.

While we started the development by assuming the state estimator to me Kalman predictor, innovation bias formulation can be used to achieve offset free regulation using any stable state estimator. The approach can also work when the system under consideration is open loop stable and open loop observer of the form (67) is used for state estimation, i.e. $\mathbf{L}$ is a null matrix.

Remark 5 *Consider the scenario when (a) there is no measurement and state noise and (b) unfiltered innovations are used for computing the target steady states, i.e.*

$$\begin{aligned} \mathbf{u}_s(k) &= \mathbf{K}_u^{-1}\left[\mathbf{r}(k) - \mathbf{K}_e\,\mathbf{e}(k)\right] = \mathbf{K}_u^{-1}\left[\mathbf{r}(k) - \mathbf{K}_e\ (\mathbf{y}(k) - \mathbf{C}\widehat{\mathbf{x}}(k|k-1))\right] \\ &= \mathbf{K}_u^{-1}\left[\mathbf{r}(k) - \mathbf{K}_e\,\mathbf{C}\boldsymbol{\varepsilon}(k|k-1)\right] = \mathbf{K}_u^{-1}\mathbf{r}(k) - \mathbf{K}_{u\varepsilon}\boldsymbol{\varepsilon}(k|k-1) && (105) \\ \mathbf{x}_s(k) &= (\mathbf{I} - \mathbf{\Phi})^{-1}\ \left[\mathbf{\Gamma}_u\mathbf{K}_u^{-1}\left[\mathbf{r}(k) - \mathbf{K}_e\,\mathbf{C}\boldsymbol{\varepsilon}(k|k-1)\right] + \mathbf{L}\mathbf{C}\boldsymbol{\varepsilon}(k|k-1)\right] \\ &= (\mathbf{I} - \mathbf{\Phi})^{-1}\ \left[\mathbf{\Gamma}_u\mathbf{K}_u^{-1}\mathbf{r}(k) + \left[\mathbf{L} - \mathbf{\Gamma_u}\mathbf{K_u^{-1}}\mathbf{K}_e\right]\mathbf{C}\boldsymbol{\varepsilon}(k|k-1)\right] && (106) \\ &= \mathbf{K}_r\mathbf{r}(k) + \mathbf{K}_{x\varepsilon}\boldsymbol{\varepsilon}(k|k-1) && (107) \end{aligned}$$

Now, the control law can be expressed as follows

$$\begin{aligned} \mathbf{u}(k) &= \mathbf{u}_s(k) - \mathbf{G}\ [\mathbf{x}(k) - \boldsymbol{\varepsilon}(k|k-1) - \mathbf{x}_s(k)] \\ &= \mathbf{K}_u^{-1}\mathbf{r}(k) - \mathbf{K}_{u\varepsilon}\boldsymbol{\varepsilon}(k|k-1) \\ &\quad - \mathbf{G}\left[\mathbf{x}(k) - \boldsymbol{\varepsilon}(k|k-1) - \mathbf{K}_r\mathbf{r}(k) - \mathbf{K}_{x\varepsilon}\boldsymbol{\varepsilon}(k|k-1)\right] \\ &= \left[\mathbf{K}_u^{-1} + \mathbf{K}_r\mathbf{G}\right]\mathbf{r}(k) - \left[\mathbf{K}_{u\varepsilon} - \mathbf{G}(\mathbf{I} + \mathbf{K}_{x\varepsilon})\right]\boldsymbol{\varepsilon}(k|k-1) - \mathbf{G}\mathbf{x}(k) && (108) \end{aligned}$$

State Augmentation Approach By this approach, the state space model (1-2) is augmented with extra artificial states as follows

$$\mathbf{x}(k+1) = \mathbf{\Phi}\mathbf{x}(k) + \mathbf{\Gamma}_u\mathbf{u}(k) + \mathbf{\Gamma}_\beta\,\boldsymbol{\beta}(k) + \mathbf{w}(k) \tag{109}$$

$$\boldsymbol{\beta}(k+1) = \boldsymbol{\beta}(k) + \mathbf{w}_\beta(k) \tag{110}$$

$$\boldsymbol{\eta}(k+1) = \boldsymbol{\eta}(k) + \mathbf{w}_\eta(k) \tag{111}$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{C}_\eta\boldsymbol{\eta}(k) + \mathbf{v}(k) \tag{112}$$

where $\boldsymbol{\beta} \in \mathbf{R}^s$ and $\boldsymbol{\eta} \in \mathbf{R}^t$ are input and output disturbance vectors while vectors $\mathbf{w}_\beta \in \mathbf{R}^s$and $\mathbf{w}_\eta \in \mathbf{R}^t$ are zero mean white noise sequences with covariances $\mathbf{Q}_\beta$ and $\mathbf{Q}_\eta$, respectively. The model coefficient matrices ($\mathbf{\Gamma}_\beta$, $\mathbf{C}_\eta$) and noise covariances matrices ($\mathbf{Q}_\beta$, $\mathbf{Q}_\eta$) are treated as tuning parameters, which can be chosen to achieve the desired closed loop disturbance rejection characteristics. Note that the total number of extra states cannot exceed the number of measurements due to the requirement that the additional states should be observable. Typical choices of the artificial state variables and the corresponding coupling matrices are as follows

- **Output bias formulation:** A simple approach is to view the drifting disturbances as causing a bias in the measured outputs, i.e., we can choose

  $$\mathbf{\Gamma}_\beta = [0] \;\; ; \mathbf{Q}_\beta = [0] \;\; ; \;\; \mathbf{C}_\eta = I \;\; ; \quad \mathbf{Q}_\eta = \sigma^2 I$$

- **Input Bias Formulation:** The elements of vector $\boldsymbol{\beta}$ can be viewed as bias in r manipulated inputs. When the number of manipulated inputs equals the number of measured outputs ($r = m$), then we can choose

  $$\mathbf{\Gamma}_\beta = \mathbf{\Gamma}_u \;\; ; \mathbf{Q}_\beta = \sigma^2 I \;\; ; \;\; \mathbf{C}_\eta = [0] \;\; ; \quad \mathbf{Q}_\eta = [0]$$

  If number of manipulated inputs exceeds the number of measurements, then r linearly independent columns of $\mathbf{\Gamma}_u$ can be selected as $\mathbf{\Gamma}_\beta$.

- **Disturbance bias formulation:** When the state space model is derived from first principles, it is possible to choose

  $$\mathbf{\Gamma}_\beta = \Gamma_d \; ; \mathbf{Q}_\beta = \sigma^2 I$$

  provided number of disturbance variables $(d) = r$. If $d > r$, then r linearly independent columns of Γ_d can be chosen as Γ_β.

In all the above cases, σ^2 is treated as a tuning parameter. The above set of equations can be combined into an augmented state space model of the form

$$\mathbf{x}_a(k+1) = \mathbf{\Phi}_a\mathbf{x}_a(k) + \mathbf{\Gamma}_{ua}\mathbf{u}(k) + \mathbf{w}_a(k) \tag{113}$$

$$\mathbf{y}(k) = \mathbf{C}_a\mathbf{x}_a(k) + \mathbf{v}(k) \tag{114}$$

where

$$
\begin{aligned}
\mathbf{x}_a(k) &= \begin{bmatrix} \mathbf{x}(k) \\ \boldsymbol{\beta}(k) \\ \boldsymbol{\eta}k) \end{bmatrix} \quad ; \quad \mathbf{w}_a(k) = \begin{bmatrix} \mathbf{w}(k) \\ \mathbf{w}_\beta(k) \\ \mathbf{w}_\eta k) \end{bmatrix} \\
\boldsymbol{\Phi}_a &= \begin{bmatrix} \boldsymbol{\Phi} & \boldsymbol{\Gamma}_\beta & [\mathbf{0}] \\ [\mathbf{0}] & \mathbf{I}_\beta & [\mathbf{0}] \\ [\mathbf{0}] & [\mathbf{0}] & \mathbf{I}_\eta \end{bmatrix} ; \qquad \boldsymbol{\Gamma}_{ua} = \begin{bmatrix} \boldsymbol{\Gamma}_u \\ \mathbf{0} \end{bmatrix} \\
\mathbf{C}_a &= \begin{bmatrix} \mathbf{C} & [\mathbf{0}] & \mathbf{C}_\eta \end{bmatrix}
\end{aligned}
$$

$$
\begin{aligned}
\mathbf{R}_{1a} &= E\left[\mathbf{w}_a(k)\mathbf{w}_a(k)^T\right] = \begin{bmatrix} \mathbf{R}_1 & [\mathbf{0}] & [\mathbf{0}] \\ [\mathbf{0}] & \mathbf{Q}_\beta & [\mathbf{0}] \\ [\mathbf{0}] & [\mathbf{0}] & \mathbf{Q}_\eta \end{bmatrix} \\
\mathbf{R}_{12a} &= E\left[\mathbf{w}_a(k)\mathbf{v}(k)^T\right] = \begin{bmatrix} \mathbf{R}_{12} \\ [\mathbf{0}] \end{bmatrix} \\
\mathbf{R}_{2a} &= E\left[\mathbf{v}(k)\mathbf{v}(k)^T\right] = \mathbf{R}_2
\end{aligned}
$$

This augmented model can be used for developing a Kalman predictor of the form

$$
\begin{aligned}
\mathbf{e}_a(k) &= \mathbf{y}(k) - \mathbf{C}\widehat{\mathbf{x}}_a(k|k-1) && (115) \\
\widehat{\mathbf{x}}_a(k+1|k) &= \boldsymbol{\Phi}_a\widehat{\mathbf{x}}_a(k|k-1) + \boldsymbol{\Gamma}_{ua}\mathbf{u}(k-1) + \mathbf{L}_a\mathbf{e}_a(k) && (116)
\end{aligned}
$$

where the steady state Kalman gain is obtained by solving the corresponding steady state Riccati equations

$$
\begin{aligned}
\mathbf{L}_a &= \left[\boldsymbol{\Phi}_a\mathbf{P}_{a\infty}\mathbf{C}_\mathbf{a}^\mathbf{T} + \mathbf{R}_{\mathbf{12a}}\right]\left[\mathbf{C}_a\mathbf{P}_{a\infty}\mathbf{C}_\mathbf{a}^\mathbf{T} + \mathbf{R}_2\right]^{-1} && (117) \\
\mathbf{P}_{a\infty} &= \boldsymbol{\Phi}_a\mathbf{P}_{a\infty}\boldsymbol{\Phi}_a^T + \mathbf{R}_1 - \mathbf{L}_a\left[\mathbf{C}_a\mathbf{P}_{a\infty}\mathbf{C}_\mathbf{a}^\mathbf{T} + \mathbf{R}_2\right]\mathbf{L}_a^T && (118)
\end{aligned}
$$

In order to maintain the observability of the artificially introduced states, the number of additional states introduced in the augmented model should not exceed the number of measured outputs . When the state space model (10-11) is observable and stable with no integrating modes, the augmented state space model will be observable (detectable) in most of the cases.

The control law (87) is now implemented using the estimates of $\boldsymbol{\beta}(k)$ and $\boldsymbol{\eta}(k)$. Thus, when number of inputs (m) equals the number of controlled outputs (r), the target states are estimated as follows

$$
\begin{aligned}
\mathbf{u}_s(k) &= \mathbf{K}_u^{-1}\left[\mathbf{r}(k) - \mathbf{K}_\beta\widehat{\boldsymbol{\beta}}(k|k-1) - \mathbf{C}_\eta\widehat{\boldsymbol{\eta}}(k|k-1)\right] && (119) \\
\mathbf{x}_s(k) &= (\mathbf{I}-\boldsymbol{\Phi})^{-1}\left[\Gamma_u\mathbf{K}_u^{-1}\left[\mathbf{r}(k) - \mathbf{C}_\eta\widehat{\boldsymbol{\eta}}(k|k-1)\right] + \left(\Gamma_\beta - \Gamma_u\mathbf{K}_u^{-1}\mathbf{K}_\beta\right)\widehat{\boldsymbol{\beta}}(k|k-1)\right] && (120)
\end{aligned}
$$

For the case $m = r$, two special cases of quadratic optimal tracking control law are

- **Output bias formulation:** In this case we have $\mathbf{\Gamma}_\beta = [0]$, $\mathbf{C}_\eta = I$, which implies that $\mathbf{K}_\beta = [0]$ and computation for $\mathbf{x}_s(k), \mathbf{u}_s(k)$ reduces to

$$\mathbf{u}_s(k) = \mathbf{K}_u^{-1}\left[\mathbf{r}(k) - \widehat{\boldsymbol{\eta}}(k|k-1)\right] \tag{121}$$

$$\mathbf{x}_s(k) = (\mathbf{I} - \mathbf{\Phi})^{-1}\,\Gamma_u \mathbf{K}_u^{-1}\left[\mathbf{r}(k) \ - \widehat{\boldsymbol{\eta}}(k|k-1)\right] \tag{122}$$

- **Input Bias Formulation:** In this case we have $\mathbf{\Gamma}_\beta = \mathbf{\Gamma}_u$, $\mathbf{C}_\eta = [0]$, which implies that $\mathbf{K}_\beta = \mathbf{K}_u$ and computation for $\mathbf{x}_s(k), \mathbf{u}_s(k)$ reduces to

$$\mathbf{u}_s(k) = \mathbf{K}_u^{-1}\mathbf{r}(k) - \widehat{\boldsymbol{\beta}}(k|k-1) \tag{123}$$

$$\mathbf{x}_s(k) = (\mathbf{I} - \mathbf{\Phi})^{-1}\,\Gamma_u \mathbf{K}_u^{-1}\mathbf{r}(k) \tag{124}$$

When number of the manipulated inputs (m) is not equal to number of controlled outputs (r), matrix $\mathbf{K}_u^{-1}$ in the above expression should be replaced by $\mathbf{K}_u^{\dagger}$, i.e., pseudo-inverse of the steady state gain matrix $\mathbf{K}_u$.

4 Model Predictive Control

LQG formulation described above provides a systematic approach to designing a control law for linear multi-variable systems. However, main difficulty associated with the classical LQG formulation is inability to handle operating constraints explicitly. Operating constraints, such as limits on manipulated inputs or on their rates of change, limits on controlled outputs arising out of product quality or safety considerations, are commonly encountered in any control application. Model Predictive Control (MPC) refers to a class of control algorithms originally developed in the process industry for dealing with operating constraints and multi-variable interactions. MPC can be viewed as modified versions of LQ (or LQG) formulation, which can deal with the operating constraints in a systematic manner. This approach was first proposed independently by two industrial groups

- Dynamic Matrix Control (DMC): Proposed by Cutler and Ramaker from Shell, USA in 1978.
- Model Algorithmic Control (MAC): proposed by Richalet (1978, France)

An MPC formulation is based on the following premise: given a reasonably accurate model for system dynamics (and a sufficiently powerful computer), possible consequences of the current and future manipulated input moves on the future plant behavior (such as possible constraint violations in future etc.) can be forecasted on-line and used while deciding the input moves in some optimal manner. To facilitate computational tractability, on-line forecasting is carried out over a moving window of time. In this section, we develop a version of MPC formulation based on the Kalman predictor.

4.1 Model Based Prediction of Future Behavior

At the centre of an MPC formulation is the state estimator, which is used to carry out on-line forecasting. Let us assume that we have developed a Kalman predictor using model equations (1-2) and used it to predict the current state

$$\widehat{\mathbf{x}}(k|k-1)=\mathbf{\Phi}\widehat{\mathbf{x}}(k-1|k-2)+\mathbf{\Gamma}\mathbf{u}(k-1)+\mathbf{L}\mathbf{e}(k-1) \tag{125}$$

and innovation

$$\mathbf{e}(k)=\mathbf{y}(k)-\mathbf{C}\widehat{\mathbf{x}}(k|k-1)$$

At each sampling instant, the Kalman predictor is used for predicting future behavior of the plant over a finite future time horizon of length p (called as the *prediction horizon)* starting from the current time instant k. Let us assume that at any instant k, we are free to choose p future manipulated input moves, which are denoted as follows

$$\{\mathbf{u}(k|k),\mathbf{u}(k+1|k)......\mathbf{u}(k+p-1|k)\}$$

Since, in the absence of model plant mismatch (MPM), $\{\mathbf{e}(k)\}$, is a whine noise sequence, expected values of the future innovations is zero i.e. $E\left[\mathbf{e}(k+i)\right]=\overline{\mathbf{0}}$ for $i=1,2,....$. Thus, in the absence of MPM (here abbreviation MPM includes the unmeasured drifting disturbances), the observer can be used for forecasting over the future as follows

$$\begin{aligned}
\widehat{\mathbf{x}}(k+1|k) &= \mathbf{\Phi}\widehat{\mathbf{x}}(k|k-1)+\mathbf{\Gamma}_u\mathbf{u}(k|k)+\mathbf{L}\mathbf{e}(k) && (126)\\
\widehat{\mathbf{x}}(k+j+1|k) &= \mathbf{\Phi}\widehat{\mathbf{x}}(k+j|k)+\mathbf{\Gamma}_u\mathbf{u}(k+j|k) && (127)\\
(\text{for } j &= 1....p-1)
\end{aligned}$$

Such an ideal scenario, i.e. absence of MPM, rarely exists in practice and, as mentioned in the previous section, the innovation sequence is a colored noise, which carries signatures of MPM and drifting unmeasured disturbances. As a consequence, it becomes necessary to compensate the future predictions for the MPM. Here, we describe two approaches to deal with this problem: (a) innovation bias approach and (b) state augmentation approach.

4.1.1 Innovation Bias Approach

One possibility is to use filtered innovations (given by equation 96) as a proxy for the unknown component and employ it to correct the future predictions by employing unmeasured disturbance models (97-100). We introduce a different notation here, $\widehat{\mathbf{z}}(k)$, to denote the future predictions with corrections for MPM. At the beginning of the prediction, $\widehat{\mathbf{z}}(k)$ is initialized as

$$\widehat{\mathbf{z}}(k) = \widehat{\mathbf{x}}(k|k-1)$$

where $\widehat{\mathbf{x}}(k|k-1)$ is computed using equation (125). Now, MPM compensated future predictions are generated as follows

- At future instant $(k+1)$

$$\begin{aligned} \widehat{\mathbf{z}}(k+1) &= \mathbf{\Phi}\widehat{\mathbf{z}}(k) + \mathbf{\Gamma u}(k|k) + \mathbf{L}\mathbf{e}_f(k) && (128) \\ \widehat{\mathbf{y}}(k+1|k) &= \mathbf{C}\widehat{\mathbf{z}}(k+1) + \mathbf{e}_f(k) && (129) \end{aligned}$$

- At future instant $(k+2)$

$$\begin{aligned} \widehat{\mathbf{z}}(k+2) &= \mathbf{\Phi}\widehat{\mathbf{z}}(k+1) + \mathbf{\Gamma u}(k+1|k) + \mathbf{L}\mathbf{e}_f(k) \\ &= \mathbf{\Phi}^2\widehat{\mathbf{z}}(k) + \mathbf{\Phi\Gamma u}(k|k) + \mathbf{\Phi\Gamma u}(k+1|k) \\ &\quad + (\mathbf{\Phi} + \mathbf{I})\mathbf{L}\mathbf{e}_f(k) && (130) \\ \widehat{\mathbf{y}}(k+2|k) &= \mathbf{C}\widehat{\mathbf{z}}(k+2) + \mathbf{e}_f(k) && (131) \end{aligned}$$

- In general, at future instant $(k+j)$

$$\begin{aligned} \widehat{\mathbf{z}}(k+j) &= \mathbf{\Phi}\widehat{\mathbf{z}}(k+j-1) + \mathbf{\Gamma u}(k+j-1|k) + \mathbf{L}\mathbf{e}_f(k) \\ &= \mathbf{\Phi}^j\widehat{\mathbf{z}}(k) + \mathbf{\Phi}^{j-1}\mathbf{\Gamma u}(k|k) + \mathbf{\Phi}^{j-2}\mathbf{\Gamma u}(k+1|k) + ... \\ &\quad + \mathbf{\Gamma u}(k+j-1|k) + (\mathbf{\Phi}^{j-1} + \mathbf{\Phi}^{j-2} + ... + \mathbf{I})\mathbf{L}\mathbf{e}_f(k) \\ \widehat{\mathbf{y}}(k+j|k) &= \mathbf{C}\widehat{\mathbf{z}}(k+j) + \mathbf{e}_f(k) && (132) \end{aligned}$$

Before we proceed with the rest of the development, it is instructive to examine the output prediction equation at the j'th instant

$$\begin{aligned} \widehat{\mathbf{y}}(k+j|k) &= \mathbf{C\Phi}^j\widehat{\mathbf{x}}(k|k-1) \\ &\quad + \left\{\mathbf{C\Phi}^{j-1}\mathbf{\Gamma u}(\mathbf{k}|\mathbf{k}) + \mathbf{C\Phi}^{j-2}\mathbf{\Gamma u}(k+1|k) + ... + \mathbf{C\Gamma u}(k+j-1|k)\right\} \\ &\quad + \left[\mathbf{C}(\mathbf{\Phi}^{j-1} + \mathbf{\Phi}^{j-2} + ... + \mathbf{I})\mathbf{L} + \mathbf{I}\right]\mathbf{e}_f(k) && (133) \end{aligned}$$

The prediction consists of three terms

- Term $\mathbf{C}\mathbf{\Phi}^j\widehat{\mathbf{x}}(k|k-1)$ signifies effect of the current state estimate over the future dynamics.
- Term $\left\{\mathbf{C}\mathbf{\Phi}^{j-1}\mathbf{\Gamma}\mathbf{u}(k|k)+....+\mathbf{C}\mathbf{\Gamma}\mathbf{u}(k+j-1|k)\right\}$ quantifies effects of current and future input moves over the predictions
- Term $\left[(\mathbf{\Phi}^{j-1}+...+\mathbf{I})\mathbf{L}+\mathbf{I}\right]\mathbf{e}_f(k)$ attempts to quantify effect or MPM over the predictions

It may be noted that, at a given instant k, the first and the third term are invariant. We can only choose the current and future manipulated input moves, $\{\mathbf{u}(k|k),...,\mathbf{u}(k+p-1|k)\}$, to influence the future dynamic behavior.

Remark 6 *For an open loop stable system, predictions can be carried out using open loop observer of the form*

$$\widehat{\mathbf{x}}(k|k-1)=\mathbf{\Phi}\widehat{\mathbf{x}}(k-1|k-2)+\mathbf{\Gamma}\mathbf{u}(k-1) \tag{134}$$

The model predictions in such scenario are carried out as follows

$$\begin{aligned}
\widehat{\mathbf{z}}(k+j) &= \mathbf{\Phi}\widehat{\mathbf{z}}(k+j-1)+\mathbf{\Gamma}\mathbf{u}(k+j-1|k) && (135)\\
\widehat{\mathbf{y}}(k+j|k) &= \mathbf{C}\widehat{\mathbf{z}}(k+j)+\mathbf{e}_f(k) && (136)\\
\widehat{\mathbf{z}}(k) &= \widehat{\mathbf{x}}(k|k-1) && (137)
\end{aligned}$$

for $j=1,2,...p$, where $\mathbf{e}_f(k)$ represnts residuals

$$\mathbf{e}(k)=\mathbf{y}(k)-\mathbf{C}\widehat{\mathbf{x}}(k|k-1)$$

filtered using equation (96). The resulting predictions are qualitatively similar to the DMC formulation.

Remark 7 *The initial MPC formulations, such as DMC or MAC, employed* finite impulse response (FIR) *or step response models for dynamic modeling and predictions. Many industrial versions of MPC still use these model forms for modeling and predictions. The predictions carried out using the state space model (125) are not different from the predictions carried out using an FIR model. To see the connection, consider the state space model given by equation (17)-(18) subjected to impulse input (with zero order hold) at $k=0$, i.e.*

$$\begin{aligned}
\mathbf{u}(0) &= \begin{bmatrix} 1 & 1 & .... & 1 \end{bmatrix}^T\\
\mathbf{u}(k) &= \overline{\mathbf{0}} \quad for \quad k>0
\end{aligned}$$

with $\mathbf{x}(0)=\overline{\mathbf{0}}$ .The output of the system is given by

$$\mathbf{y}(k)=\mathbf{H}(k)\mathbf{u}(0) \ \ where \ \ \mathbf{H}(k)=\mathbf{C}\mathbf{\Phi}^{k-1}\Gamma \quad for \quad k>0 \tag{138}$$

where $\mathbf{H}(k) : k = 1, 2, ...$ represent the impulse response coefficients. Thus, assuming $\mathbf{L} = [\mathbf{0}]$, the output prediction equation (133) can be expressed as follows

$$\begin{aligned}\hat{\mathbf{y}}(k+j|k) &= \mathbf{C}\mathbf{\Phi}^j\widehat{\mathbf{x}}(k|k-1) + \mathbf{e}_f(k) \\ &\quad + \{\mathbf{H}(j)\mathbf{u}(k|k) + \mathbf{H}(j-1)\mathbf{u}(k+1|k) + ... + \mathbf{H}(1)\mathbf{u}(k+j-1|k)\} \quad (139)\end{aligned}$$

which is similar to the prediction equation used in industrial MPC formulations.

4.1.2 State Augmentation Approach

Alternate approach to account for MPM is to augment the state space model with artificially introduced input and / or output disturbance variables, which behave as integrated white noise sequences, as given by equations (109-112). Typical choices of the artificial state variables and the corresponding coupling matrices have been discussed in Section 2.3. This augmented model can be used for developing a Kalman predictor of the form

$$\widehat{\mathbf{x}}_a(k|k-1) = \mathbf{\Phi}_a\widehat{\mathbf{x}}_a(k-1|k-2) + \mathbf{\Gamma}_{ua}\mathbf{u}(k-1) + \mathbf{L}_a\mathbf{e}_a(k-1) \quad (140)$$

$$\mathbf{e}_a(k) = \mathbf{y}(k) - \mathbf{C}\widehat{\mathbf{x}}_a(k|k-1) \quad (141)$$

where $\mathbf{L}_a$ represents the steady state Kalman gain is obtained by solving the corresponding steady state Riccati equations. Since the augmented observer accounts for MPM through explicit estimation of drifting disturbance terms, it may be expected that the $\{\mathbf{e}_a(k)\}$ is zero mean white noise process. The optimal predictions of the states based on the augmented state space model can be generated as follows

$$\widehat{\mathbf{x}}_a(k+1|k) = \mathbf{\Phi}_a\widehat{\mathbf{x}}_a(k|k-1) + \mathbf{\Gamma}_{ua}\mathbf{u}(k|k) + \mathbf{L}_a\mathbf{e}_a(k) \quad (142)$$

$$\widehat{\mathbf{x}}_a(k+j+1|k) = \mathbf{\Phi}_a\widehat{\mathbf{x}}_a(k+j|k) + \mathbf{\Gamma}_{ua}\mathbf{u}(k+j|k) \quad (143)$$

$$(\text{for } j = 0, 1....p-1) \quad (144)$$

$$\widehat{\mathbf{y}}(k+j|k) = \mathbf{C}_a\widehat{\mathbf{x}}_a(k+j|k) \qquad (\text{for } j = 1....p) \quad (145)$$

Remark 8 *A common modification to the above formulation is to restrict the future degrees of freedom to q future manipulated input moves*

$$\{\mathbf{u}(k|k), \mathbf{u}(k+1|k)......\mathbf{u}(k+q-1|k)\}$$

and impose constraints on the remaining of the future input moves as follows

$$\mathbf{u}(k+q|k) = \mathbf{u}(k+q+1|k) = ... = \mathbf{u}(k+p-1|k) = \mathbf{u}(k+q-1|k) \quad (146)$$

Here, q is called as the 'control horizon'*. Alternatively, q degrees of freedom over the future are spread over the horizon using the concept of* 'input blocking' *as follows*

$$\begin{aligned}
\mathbf{u}(k+j|k) &= \mathbf{u}(k|k) \text{ :}\mathit{for} \quad j = m_0+1, ...m_1-1 && (147)\\
\mathbf{u}(k+j|k) &= \mathbf{u}(k+m_1|k) \text{ :}\mathit{for} \quad j = m_1+1, m_2-1 && (148)\\
&\ldots\ldots \\
\mathbf{u}(k+j|k) &= \mathbf{u}(k+m_i|k) \text{ :}\mathit{for} \quad j = m_i+1, m_{i+1}-1 && (149)\\
&\ldots\ldots\ldots\ldots \\
\mathbf{u}(k+j|k) &= \mathbf{u}(k+m_{q-1}|k) \text{ :}\mathit{for} \quad j = m_{q-1}+1, m_q-1 && (150)
\end{aligned}$$

where m_j are selected such that

$$m_0 = 0 < m_1 < m_2 < .... < m_{q-1} < m_q = p \tag{151}$$

4.2 Conventional Formulation of MPC

The model used for developing an industrial MPC scheme is often developed from the operating data and the state vector may not have any physical meaning. In such cases, MPC is formulated as an output control scheme as follows. During this development, we assume that only q future manipulated input moves (i.e. equal to the control horizon) are to be decided.

Let us assume that $\{\mathbf{y}_r(k+j|k) : j = 1, 2, ...p\}$ denotes future desired setpoint trajectory at instant k.Given the setpoint trajectory, the model predictive control problem at the sampling instant k is defined as a constrained optimization problem whereby the future manipulated input moves $\mathbf{u}(k|k), \mathbf{u}(k+1|k)......\mathbf{u}(k+q-1|k)$ are determined by minimizing an objective function J defined as follows

$$\begin{aligned}
J &= \mathcal{E}(k+p|k)^T \mathbf{w}_\infty \mathcal{E}(k+p|k) + \sum_{j=1}^{p} \mathcal{E}(k+j|k)^T \mathbf{w}_e \mathcal{E}(k+j|k) \\
&\quad + \sum_{j=0}^{q-1} \Delta\mathbf{u}(k+j|k)^T \mathbf{w}_{\Delta u} \Delta\mathbf{u}(k+j|k) && (152)
\end{aligned}$$

where

$$\mathcal{E}(k+j|k) = \mathbf{y}_r(k+j|k) - \widehat{\mathbf{y}}(k+j|k) \quad \text{for} \quad j = 1, 2, ...p \tag{153}$$

represents predicted control error and

$$\begin{aligned}
\Delta\mathbf{u}(k+j|k) &= \mathbf{u}(k+j|k) - \mathbf{u}(k+j-1|k) && (154)\\
j &= 1, ...q-1 \\
\Delta\mathbf{u}(k|k) &= \mathbf{u}(k|k) - \mathbf{u}(k-1) && (155)
\end{aligned}$$

represents changes in the manipulated input moves. Here, $\mathbf{w}_e$ represents positive definite the error weighting matrix, $\mathbf{w}_{\Delta u}$ represents positive semi-definite the input move weighting matrix and $\mathbf{w}_\infty$ represents the terminal weighting matrix. The above minimization problem is subject to following operating constraints

- Model prediction equations described in the previous section
- Manipulated input constraints

$$\mathbf{u}(k+q|k) = \mathbf{u}(k+q+1|k) = ... = \mathbf{u}(k+p-1|k) = \mathbf{u}(k+q-1|k) \tag{156}$$

$$\begin{aligned} \mathbf{u}^L &\leq \mathbf{u}(k+j|k) \leq \mathbf{u}^H && (157) \\ \Delta\mathbf{u}^L &\leq \Delta\mathbf{u}(k+j|k) \leq \Delta\mathbf{u}^H && (158) \\ j &= 0,1,2,....q-1 \end{aligned}$$

- Output quality constraints

$$\begin{aligned} \mathbf{y}^L &\leq \widehat{\mathbf{y}}_c(k+j|k) \leq \mathbf{y}^H && (159) \\ j &= p_1, p_1+1, ...p \end{aligned}$$

  where p_1 is referred to as *constraint horizon.*

The resulting constrained optimization problem can be solved using any standard nonlinear programming method such as SQP (Sequential Quadratic Programming). The controller is implemented in a moving horizon frame work. Thus, after solving the optimization problem, only the first move $\mathbf{u}_{opt}(k|k)$ is implemented on the plant, i.e.

$$\mathbf{u}(k) = \mathbf{u}_{opt}(k|k)$$

and the optimization problem is reformulated at the next sampling instant based on the updated information from the plant. A schematic representation of MPC scheme is shown in Figure 1 while Figure 2 shows a schematic representation of the moving window formulation.

Remark 9 *If input blocking constraints are employed, then input constraints get modified as follows*

$$\mathbf{u}^L \leq \mathbf{u}(k+m_j|k) \leq \mathbf{u}^H \quad for \quad j = 0,1,2,....q-1$$

$$\Delta\mathbf{u}^L \leq \Delta\mathbf{u}(k+m_j|k) \leq \Delta\mathbf{u}^H \;\; for \quad j = 0,1,2,....q-1$$

$$\begin{aligned} \Delta\mathbf{u}(k+m_j|k) &= \mathbf{u}(k+m_j|k) - \mathbf{u}(k+m_{j-1}|k) && (160) \\ for \quad j &= 1,2,....q-1 && (161) \\ \Delta\mathbf{u}(k+m_0|k) &= \mathbf{u}(k|k) - \mathbf{u}(k-1) && (162) \end{aligned}$$

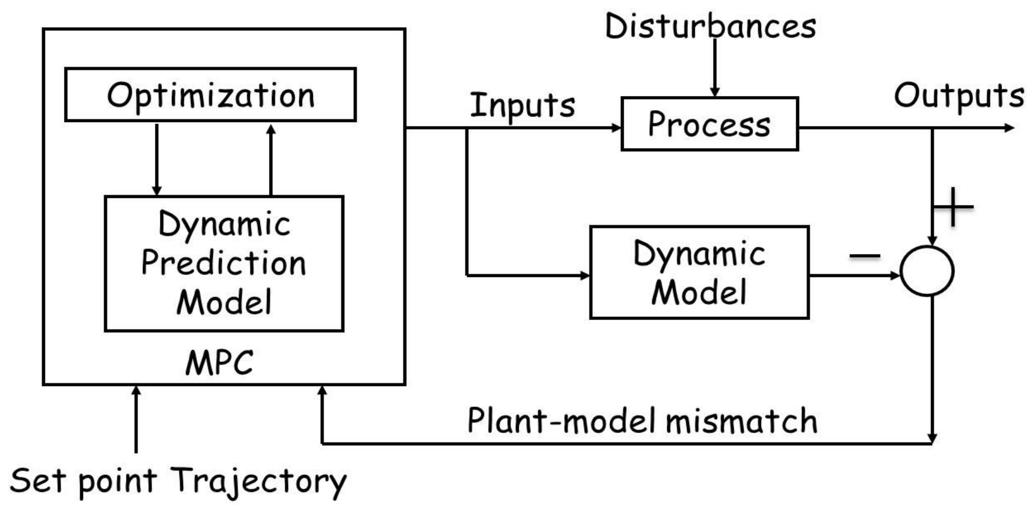

Figure 1: Schematic Representation of MPC Scheme

Remark 10 *When the number of controlled outputs exceeds the number of manipulated inputs, then it is possible to specify setpoints only for the number of outputs equal to or less than the number of manipulated inputs. For the remaining controlled outputs, it is possible to define only ranges or zones in which they should lie by imposing bounds on the predicted outputs. Such outputs are often referred to as zone control variables in the MPC parlance.*

Remark 11 *When the number of manipulated inputs (m) exceeds the number of controlled output (r), then it is possible to specify ($m - r$) inputs through optimization and leave only m degrees of freedom to MPC. Many commercial MPC implementations solve a separate linear programming (LP) problem, which employes linear steady state model with a suitable economic objective function, online to independently decide these ($m - r$) inputs.*

4.2.1 Tuning Parameters

The predictive control formulation has many parameters that need to be judiciously chosen by the control system designer.

- **Prediction Horizon** (p) and **Control Horizon** (q) : The closed loop stability and the desired closed loop performance can be achieved by judiciously selecting the prediction horizon p, control horizon q. Typically, prediction horizon is selected close to the open loop settling time while control horizon (q) is chosen significantly smaller (say between 1 to 5). Since the moving horizon formulation implies that a constrained optimization problem has to be solved on-line at each sampling instant, time required for solving the

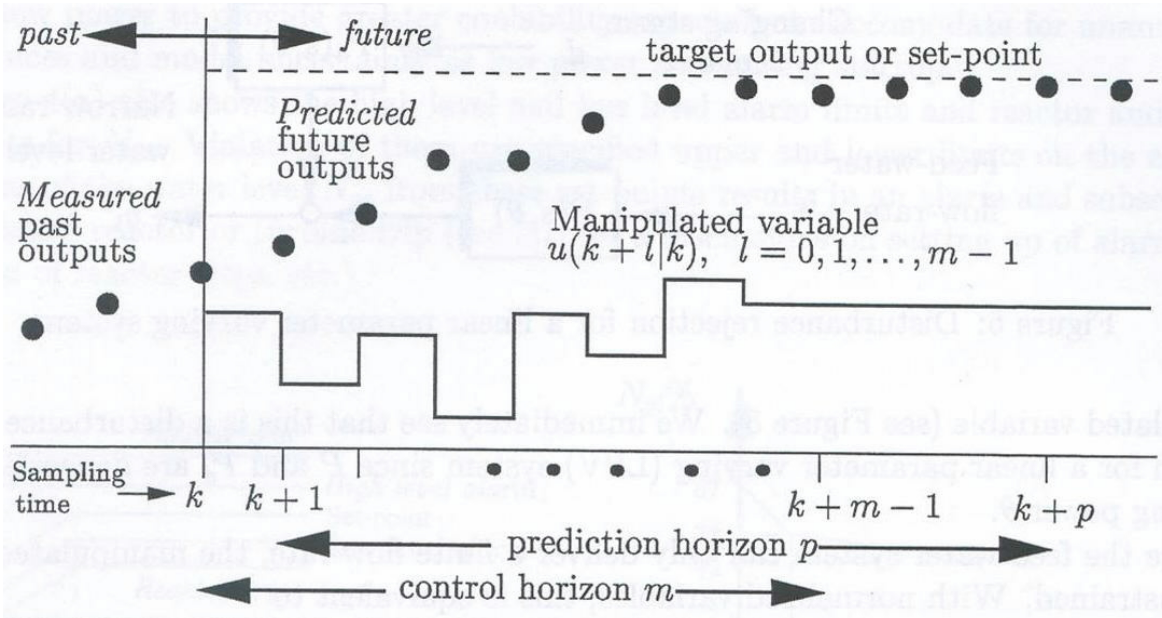

Figure 2: Schematic Representation of Moving Window Formulation (Kothare et al., 2000)

optimization problem is a major concern while developing an MPC scheme. The choice of relatively smaller control horizon reduces the dimension of the optimization problem at each sampling instant and thereby reduces the time for on-line computations.

- **Weighting matrices:** The weighting matrices $\mathbf{w}_e$ and $\mathbf{w}_{\Delta u}$ are typically chosen to be diagonal and can be effectively used to specify relative importance of elements of control error vectors and elements of manipulated input vectors, respectively.

- **Future Setpoint Trajectory:** In addition to predicting the future output trajectory, at each instant, a filtered future setpoint trajectory is generated using a reference system of the form

$$\mathbf{x}_r(k+j+1|k) = \mathbf{\Phi}_r \mathbf{x}_r(k+j|k) + [\mathbf{I} - \mathbf{\Phi}_r][\mathbf{r}(k) - \mathbf{y}(k)] \tag{163}$$

$$\mathbf{y}_r(k+j+1|k) = \mathbf{y}(k) + \mathbf{x}_r(k+j+1|k) \tag{164}$$

$$\text{for } j = 0, 1, ...p-1$$

$$\mathbf{\Phi}_r = diag\begin{bmatrix} \gamma_1 & \gamma_2 & ... & \gamma_r \end{bmatrix}$$

$$0 \leq \gamma_i < 1 \quad \text{for} \quad i = 1, 2, ..., r \text{ are tuning parameters}$$

with initial condition $\mathbf{x}_r(k|k) = \overline{\mathbf{0}}$. Here, $\mathbf{r}(k) \in R^r$ represents the setpoint vector. The coefficient matrices of the reference system are tuning parameters which can be selected to achieve the desired closed loop tracking performance. Typical range of values for γ_i is between 0.8 to 0.99. In order to ensure the free servo responses for step changes in the setpoint, the coefficient matrices of the reference system should be selected such that the steady state gain of the reference system is equal to the identity matrix, i.e.

$$\mathbf{C}_{rf}(\mathbf{I}-\mathbf{\Phi}_r)^{-1}\mathbf{\Gamma}_r = \mathbf{I}$$

Typically, the reference system is selected such that its transfer function matrix is diagonal with unit gain first (or higher) order low pass filters on the main diagonal.

- **Robustness against MPM:** If the innovation bias formulation is employed for generating the output predictions, then the diagonal elements of matrix $\mathbf{\Phi}_e$ in equation (96) can be chosen to incorporate robustness against MPM. Typical range of values for α_i is between 0.8 to 0.99. In the state augmentation based formulation approach, the choice of covariance matrices $\mathbf{Q}_\beta$ and $\mathbf{Q}_\eta$ decides the quality of the estimates of the artificially introduced disturbance variables and thereby influences the quality of the regulatory responses.

4.2.2 Unconstrained MPC

To see connection between the conventional MPC formulation and state feedback controller, we derive unconstrained MPC control law using innovation bias formulation. To begin with, we assume that the control horizon (q) is equal to the prediction horizon (p) and later show how to modify the formulation for ($q < p$) and input blocking. Defining the future input vector $\mathbf{U}_p(k)$ and the predicted output vector $\widehat{\mathbf{Y}}(k)$ over the future horizon as

$$\mathbf{U}_p(k) = \begin{bmatrix} \mathbf{u}(k|k)^T & \mathbf{u}(k+1|k)^T & .... & \mathbf{u}(k+p-1|k)^T \end{bmatrix}^T \tag{165}$$

$$\mathbf{U}_s(k) = \begin{bmatrix} \mathbf{u}_s(k)^T & \mathbf{u}_s(k)^T & .... & \mathbf{u}_s(k)^T \end{bmatrix}^T \tag{166}$$

$$\widehat{\mathbf{Y}}(k) = \begin{bmatrix} \widehat{\mathbf{y}}(k+1|k)^T & \widehat{\mathbf{y}}(k+2|k)^T & ..... & \widehat{\mathbf{y}}(k+p|k)^T \end{bmatrix}^T \tag{167}$$

the prediction model

$$\begin{aligned} \hat{\mathbf{y}}(k+j|k) &= \mathbf{C}\mathbf{\Phi}^j\widehat{\mathbf{x}}(k|k-1) \\ &+ \left\{\mathbf{C}\mathbf{\Phi}^{j-1}\mathbf{\Gamma}\mathbf{u}(k|k) + \mathbf{C}\mathbf{\Phi}^{j-2}\mathbf{\Gamma}\mathbf{u}(k+1|k) + ... + \mathbf{C}\mathbf{\Gamma}\mathbf{u}(k+j-1|k)\right\} \\ &+ \left[\mathbf{C}(\mathbf{\Phi}^{j-1} + \mathbf{\Phi}^{j-2} + ...+\mathbf{I})\mathbf{L} + \mathbf{I}\right]\mathbf{e}_f(k) \end{aligned} \tag{168}$$

for $j = 1, 2, ...p$ can be expressed as a single vector equation

$$\widehat{\mathbf{Y}}(k) = \mathbf{S}_x\widehat{\mathbf{x}}(k|k-1) + \mathbf{S}_u\mathbf{U}_p(k) + \mathbf{S}_e\mathbf{e}_f(k) \tag{169}$$

where

$$\mathbf{S}_x = \begin{bmatrix} \mathbf{C\Phi} \\ \mathbf{C\Phi}^2 \\ ...... \\ \mathbf{C\Phi}^p \end{bmatrix} \quad ; \quad \mathbf{S}_e = \begin{bmatrix} \mathbf{CL} + \mathbf{I}_r \\ \mathbf{C}(\mathbf{\Phi} + \mathbf{I}_n)\mathbf{L} + \mathbf{I}_r \\ ...... \\ \mathbf{C}(\mathbf{\Phi}^{p-1} + \mathbf{\Phi}^{p-2} + ...+\mathbf{I}_n)\mathbf{L} + \mathbf{I}_r \end{bmatrix} \tag{170}$$

$$\begin{aligned} \mathbf{S}_u &= \begin{bmatrix} \mathbf{C\Gamma}_u & [0] & [0] & .... & [0] \\ \mathbf{C\Phi\Gamma}_u & \mathbf{C\Gamma}_u & [0] & .... & [0] \\ ..... & ..... & .... & ... & [0] \\ \mathbf{C\Phi}^{p-1}\mathbf{\Gamma}_u & \mathbf{C\Phi}^{p-2}\mathbf{\Gamma}_u & .... & ... & \mathbf{C\Gamma}_u \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{H}(1) & [0] & [0] & .... & [0] \\ \mathbf{H}(2) & \mathbf{H}(1) & [0] & .... & [0] \\ ..... & ..... & .... & ... & [0] \\ \mathbf{H}(p) & \mathbf{H}(p-1) & .... & ... & \mathbf{H}(1) \end{bmatrix} \end{aligned} \tag{171}$$

where $\mathbf{H}(j)$ represent the impulse response coefficients as defined by equation (138). Here, matrix $\mathbf{S}_u$ is often referred to as *dynamic matrix* of the system.

If it is desired to implement input blocking constraints, then $\mathbf{S}_u$ matrix needs to be modified as follows. Consider matrices $\{\mathbf{I}_{m_i} : i = 1, 2, ..., q\}$, each of dimension $(m_i - m_{i-1} + 1)\, m \times m$ and defined as follows

$$\mathbf{I}_{m_i} = \begin{bmatrix} \mathbf{I}_m \\ \mathbf{I}_m \\ .... \\ \mathbf{I}_m \end{bmatrix}_{(m_i - m_{i-1}+1)m\times m}$$

Now, define a matrix

$$\mathbf{\Psi}_{pq} = block\ diag \begin{bmatrix} \mathbf{I}_{m_1} & \mathbf{I}_{m_2} & .... & \mathbf{I}_{m_q} \end{bmatrix}_{(pm\times pm)}$$

Now, let us define a vector for future q manipulated input moves

$$\mathbf{U}_q(k) = \begin{bmatrix} \mathbf{u}(k|k)^T & \mathbf{u}(k+m_1|k)^T & .... & \mathbf{u}(k+m_{q-1}|k)^T \end{bmatrix}^T \tag{172}$$

Then, it is easy to construct vector $\mathbf{U}_p(k)$ defined by equation (165) starting from $\mathbf{U}_q(k)$ as follows

$$\mathbf{U}_p(k) = \mathbf{\Psi}_{pq}\mathbf{U}_q(k)$$

$$\mathbf{U}_p(k) = \begin{bmatrix} \mathbf{u}(k|k)^T & ... & \mathbf{u}(k|k)^T & ... & \mathbf{u}(k+m_{q-1}|k)^T & ... & \mathbf{u}(k+m_{q-1}|k)^T \end{bmatrix}^T \tag{173}$$

The vector form of prediction equation (165) is now modified as follows

$$\widehat{\mathbf{Y}}(k) = \mathbf{S}_x\widehat{\mathbf{x}}(k|k-1) + \mathbf{S}_u\mathbf{U}_q(k) + \mathbf{S}_e\mathbf{e}_f(k) \tag{174}$$

where

$$\mathbf{S}_u = \begin{bmatrix} \mathbf{C}\mathbf{\Gamma}_u & [0] & [0] & .... & [0] \\ \mathbf{C}\mathbf{\Phi}\mathbf{\Gamma}_u & \mathbf{C}\mathbf{\Gamma}_u & [0] & .... & [0] \\ ..... & ..... & .... & ... & [0] \\ \mathbf{C}\mathbf{\Phi}^{p-1}\mathbf{\Gamma}_u & \mathbf{C}\mathbf{\Phi}^{p-2}\mathbf{\Gamma}_u & .... & ... & \mathbf{C}\mathbf{\Gamma}_u \end{bmatrix} \mathbf{\Psi}_{pq} \tag{175}$$

Defining the future reference trajectory vector $\mathbf{R}(k)$ as

$$\mathbf{R}(k) = \begin{bmatrix} \mathbf{y}_r(k+1|k)^T & \mathbf{y}_r(k+2|k)^T & ..... & \mathbf{y}_r(k+p|k)^T \end{bmatrix}^T \tag{176}$$

the predicted control error vector $\mathcal{E}(k)$ at instant k can be defined as follows

$$\mathcal{E}(k) = \mathbf{R}(k) - \widehat{\mathbf{Y}}(k) \tag{177}$$

The unconstrained version of the MPC control problem can be re-cast as follows

$$\min_{\mathbf{U}_f(k)} \left\{ \begin{array}{l} \mathcal{E}(k)^T\mathbf{W}_E\mathcal{E}(k) + \Delta\mathbf{U}_p(k)^T\,\mathbf{W}_{\Delta U}\,\Delta\mathbf{U}_p(k) \\ +\left[\mathbf{U}_q(k) - \mathbf{U}_s(k)\right]^T\ \mathbf{W}_U\ \left[\mathbf{U}_q(k) - \mathbf{U}_s(k)\right] \end{array} \right\} \tag{178}$$

where $\mathbf{W}_E$ and $\mathbf{W}_U$ represents error weighting and input move weighting matrices, respectively, and are defined as

$$\mathbf{W}_E = \mathit{block\ diag}\begin{bmatrix} \mathbf{w}_e & \mathbf{w}_e & .... & \mathbf{w}_\infty \end{bmatrix}_{(pn\times pn)} \tag{179}$$

$$\mathbf{W}_{\Delta U} = \mathit{block\ diag}\begin{bmatrix} \mathbf{w}_{\Delta u} & \mathbf{w}_{\Delta u} & .... & \mathbf{w}_{\Delta u} \end{bmatrix}_{(qm\times qm)} \tag{180}$$

$$\mathbf{W}_U = \mathit{block\ diag}\begin{bmatrix} \mathbf{w}_u & \mathbf{w}_u & .... & \mathbf{w}_u \end{bmatrix}_{(qm\times qm)} \tag{181}$$

Here, $\Delta\mathbf{U}_p(k)$ is defined as

$$\Delta\mathbf{U}_q(k) = \begin{bmatrix} \mathbf{u}(k|k) - \mathbf{u}(k-1) \\ \mathbf{u}(k+m_1|k) - \mathbf{u}(k|k) \\ ...... \\ \mathbf{u}(k+m_{q-1}|k) - \mathbf{u}(k+m_{q-2}|k) \end{bmatrix} = \mathbf{\Psi}\,\mathbf{U}_q(k) - \mathbf{\Psi}_0\mathbf{u}(k-1) \tag{182}$$

where

$$\mathbf{\Psi} = \begin{bmatrix} \mathbf{I}_m & [0] & [0] & [0] \\ -\mathbf{I}_m & \mathbf{I}_m & [0] & [0] \\ ... & ... & ... & ... \\ [0] & .... & -\mathbf{I}_m & \mathbf{I}_m \end{bmatrix}_{(qm\times qm)} \quad ; \ \mathbf{\Psi}_0 = \begin{bmatrix} \mathbf{I}_m \\ [0] \\ ... \\ [0] \end{bmatrix}_{(qm\times m)} \tag{183}$$

Here, $\mathbf{I}_m$ represents $m \times m$ identity matrix. Now, let us examine the first term in the objective function

$$\mathcal{E}(k)^T \mathbf{W}_E \mathcal{E}(k) = \left[\mathbf{R}(k) - \widehat{\mathbf{Y}}(k)\right]^T \mathbf{W}_E \left[\mathbf{R}(k) - \widehat{\mathbf{Y}}(k)\right]$$

Defining *estimation error compensated setpoint trajectory*

$$\boldsymbol{\xi}(k) = \left[\mathbf{R}(k) - \mathbf{S}_x \widehat{\mathbf{x}}(k|k-1) - \mathbf{S}_e \mathbf{e}_f(k)\right]$$

we have

$$\begin{aligned} \mathbf{R}(k) - \widehat{\mathbf{Y}}(k) &= \boldsymbol{\xi}(k) - \mathbf{S}_u \mathbf{U}_p(k) \\ \mathcal{E}(k)^T \mathbf{W}_E \mathcal{E}(k) &= \boldsymbol{\xi}^T(k) \mathbf{W}_E \boldsymbol{\xi}(k) + \mathbf{U}_q^T(k) \mathbf{S}_u^T \mathbf{W}_E \mathbf{S}_u \mathbf{U}_q(k) - 2\mathbf{U}_q^T(k) \mathbf{S}_u^T \mathbf{W}_E \boldsymbol{\xi}(k) \end{aligned} \tag{184}$$

Similarly, considering term

$$\begin{aligned} \Delta\mathbf{U}_q(k)^T \mathbf{W}_{\Delta U} \Delta\mathbf{U}_q(k) &= \left[\boldsymbol{\Psi} \mathbf{U}_p(k) - \boldsymbol{\Psi}_0 \mathbf{u}(k-1)\right]^T \mathbf{W}_{\Delta U} \left[\boldsymbol{\Psi} \mathbf{U}_p(k) - \boldsymbol{\Psi}_0 \mathbf{u}(k-1)\right] \\ &= \mathbf{U}_q^T(k) \left[\boldsymbol{\Psi}^T \mathbf{W}_{\Delta U} \boldsymbol{\Psi}\right] \mathbf{U}_q(k) - 2\mathbf{U}_q^T(k) \left[\boldsymbol{\Psi}^T \mathbf{W}_{\Delta U} \boldsymbol{\Psi}_0\right] \mathbf{u}(k-1) \\ &\quad + \mathbf{u}^T(k-1) \left[\boldsymbol{\Psi}_0^T \mathbf{W}_{\Delta U} \boldsymbol{\Psi}_0\right] \mathbf{u}(k-1) \end{aligned} \tag{185}$$

Similarly, we have

$$\begin{aligned} \left[\mathbf{U}_q(k) - \mathbf{U}_s(k)\right]^T \mathbf{W}_U \left[\mathbf{U}_q(k) - \mathbf{U}_s(k)\right] &= \mathbf{U}_q^T(k) \mathbf{W}_U \mathbf{U}_q(k) - 2\mathbf{U}_q^T(k) \mathbf{W}_U \mathbf{U}_s(k) \\ &\quad + \mathbf{U}_s^T(k) \mathbf{W}_U \mathbf{U}_s(k) \end{aligned}$$

Defining matrix

$$\boldsymbol{\Psi}_u = \begin{bmatrix} \mathbf{I}_m \\ \mathbf{I}_m \\ \dots \\ \mathbf{I}_m \end{bmatrix}_{(qm \times m)} \qquad \mathbf{U}_s(k) = \boldsymbol{\Psi}_u \mathbf{u}_s(k)$$

which consists of $\mathbf{I}_m$ stacked q times, this term reduces to

$$\begin{aligned} \left[\mathbf{U}_q(k) - \mathbf{U}_s(k)\right]^T \mathbf{W}_U \left[\mathbf{U}_q(k) - \mathbf{U}_s(k)\right] &= \mathbf{U}_q^T(k) \mathbf{W}_U \mathbf{U}_q(k) - 2\mathbf{U}_q^T(k) \mathbf{W}_U \boldsymbol{\Psi}_u \mathbf{u}_s(k) \\ &\quad + \mathbf{u}_s^T(k) \boldsymbol{\Psi}_u^T \mathbf{W}_U \boldsymbol{\Psi}_u \mathbf{u}_s(k) \end{aligned} \tag{186}$$

Using equations (184), (185) and (186), the unconstrained optimization problem (178) can be reformulated as minimization of a *quadratic objective function* as follows

$$\min_{\mathbf{U}_q(k)} \frac{1}{2} \mathbf{U}_q(k)^T \mathbf{H} \mathbf{U}_q(k) + F(k)^T \mathbf{U}_q(k) \tag{187}$$

where

$$\begin{aligned}
\mathbf{H} &= 2(\mathbf{S}_u^T\mathbf{W}_E\mathbf{S}_u + \mathbf{\Psi}^T\mathbf{W}_{\Delta U}\mathbf{\Psi} + \mathbf{W}_{\mathbf{U}}) && (188)\\
\digamma(k) &= -2\left[\left(\mathbf{S}_u^T\mathbf{W}_E\right)\boldsymbol{\xi}(k) + \left(\mathbf{\Psi}^T\mathbf{W}_{\Delta U}\mathbf{\Psi}_0\right)\mathbf{u}(k-1) + \mathbf{W}_U\mathbf{\Psi}_u\mathbf{u}_s(k)\right] && (189)
\end{aligned}$$

This unconstrained problem can be solved analytically to compute a closed form control law. The least square solution to above minimization problem is

$$[\mathbf{U}_q(k)]_{opt} = -\mathbf{H}^{-1}\digamma(k) \tag{190}$$

Since only the first input move is implemented on the process

$$\mathbf{u}_{opt}(k|k) = \mathbf{\Psi}_0^T\,[\mathbf{U}_q(k)]_{opt} = -\mathbf{\Psi}_0^T\mathbf{H}^{-1}\digamma(k) \tag{191}$$

With some algebraic manipulations, control law (191) can be rearranged as follows

$$\mathbf{u}_{opt}(k|k) = -\mathbf{G}_x\widehat{\mathbf{x}}(k|k-1) + \mathbf{G}_u\mathbf{u}(k-1) + \mathbf{G}_{us}\mathbf{u}_s(k) + \mathbf{G}_e\mathbf{e}_f(k) + \mathbf{G}_r\mathbf{R}(k) \tag{192}$$

where $(\mathbf{G}_u, \mathbf{G}_{us}, \mathbf{G}_x, \mathbf{G}_e, \mathbf{G}_r)$ are matrices of appropriate dimension derived by combining equations (188-189) with equation (191). From equation (191), it is easy to see that the unconstrained MPC control law can be viewed as a type of state feedback controller.

4.2.3 QP Formulation of MPC

The conventional MPC formulation with constraints on manipulated inputs and predicted outputs can now be re-cast as follows:

$$\min_{\mathbf{U}_q(k)} \left\{ \begin{array}{c} \mathcal{E}(k)^T\mathbf{W}_E\mathcal{E}(k) + \Delta\mathbf{U}_q(k)^T\,\mathbf{W}_{\Delta U}\,\Delta\mathbf{U}_q(k) \\ +\left[\mathbf{U}_q(k) - \mathbf{U}_s(k)\right]^T\ \mathbf{W}_U\ \left[\mathbf{U}_q(k) - \mathbf{U}_s(k)\right] \end{array} \right\} \tag{193}$$

subject to the following constraints

$$\widehat{\mathbf{Y}}(k) = \mathbf{S}_x\widehat{\mathbf{x}}(k|k-1) + \mathbf{S}_u\mathbf{U}_p(k) + \mathbf{S}_e\mathbf{e}_f(k) \tag{194}$$

$$\begin{aligned}
\mathbf{Y}^L &\le \widehat{\mathbf{Y}}(k) \le \mathbf{Y}^H && (195)\\
\mathbf{\Psi}_u\mathbf{u}_L &\le \mathbf{U}_q(k) \le \mathbf{\Psi}_u\mathbf{u}_H && (196)\\
\mathbf{\Psi}_u\Delta\mathbf{u}_L &\le \Delta\mathbf{U}_q(k) \le \mathbf{\Psi}_u\Delta\mathbf{u}_H && (197)
\end{aligned}$$

Note that above formulation has a quadratic objective function and linear constraints. Thus, for improving the computational efficiency, the above problem can be transformed into an equivalent *quadratic programming (QP) formulation* as follows

$$\min_{\mathbf{U}_q(k)} \frac{1}{2}\mathbf{U}_q(k)^T\mathbf{H}\mathbf{U}_q(k) + +\digamma(k)^T\mathbf{U}_q(k) \tag{198}$$

$$\text{Subject to } \mathcal{A}\mathbf{U}_p(k) \leq \mathcal{B} \tag{199}$$

where

$$\mathcal{A}_p = \begin{bmatrix} \mathbf{I}_{qm} \\ -\mathbf{I}_{qm} \\ \mathbf{\Psi} \\ -\mathbf{\Psi} \\ \mathbf{S}_u \\ -\mathbf{S}_u \end{bmatrix} \quad ; \quad \mathcal{B}_p = \begin{bmatrix} \mathbf{\Psi}_u \mathbf{u}_H \\ -\mathbf{\Psi}_u \mathbf{u}_L \\ \mathbf{\Psi}_u \Delta\mathbf{u}_H + \mathbf{\Psi}_0 \mathbf{u}_{k-1} \\ -\mathbf{\Psi}_u \Delta\mathbf{u}_L - \mathbf{\Psi}_0 \mathbf{u}_{k-1} \\ \mathbf{Y}^H - \mathbf{S}_x\widehat{\mathbf{x}}(k|k-1) - \mathbf{S}_e\mathbf{e}_f(k) \\ \mathbf{S}_x\widehat{\mathbf{x}}(k|k-1) + \mathbf{S}_e\mathbf{e}_f(k) - \mathbf{Y}^L \end{bmatrix}$$

Here, $\mathbf{I}_{pm}$ represents identity matrix of dimension $p \times m$.

The QP formulation is more suitable for on-line implementation as efficient algorithms exist for solving QP in polynomial time.

4.3 State Space Formulation of MPC

In this section, we present a state space formulation of MPC controller. For the sake of brevity, a formulation is based on the input bias approach for future trajectory predictions is developed. It is possible to develop a similar formulation based on state augmentation approach. This is left as an exercise to the reader.

Thus, it is assumed that state predictions are generated as follows

$$\begin{aligned} \widehat{\mathbf{z}}(k+j) &= \mathbf{\Phi}\widehat{\mathbf{z}}(k+j-1) + \mathbf{\Gamma}\mathbf{u}(k+j-1|k) + \mathbf{L}\mathbf{e}_f(k) && (200) \\ \widehat{\mathbf{z}}(k) &= \widehat{\mathbf{x}}(k|k-1) && (201) \\ \text{for } j &= 1, 2, ...p \end{aligned}$$

where $\widehat{\mathbf{x}}(k|k-1)$ represents estimate of the state generated using a suitable prediction estimator.

4.3.1 Generation of Target States

First task is to generate target states, $\mathbf{z}_s(k)$, and target inputs, $\mathbf{u}_s(k)$. It may be noted that MPC formulation takes into account bounds on the manipulated inputs. Thus, unlike the LQG case, target state computation is carried out by solving a constrained minimization problem. If the number of controlled outputs exceeds or is equal to the number of manipulated inputs, the target state can be computed by solving the following optimization problem

$$\underset{\mathbf{u}_s(k)}{Min} \; [\mathbf{r}(k) - \mathbf{y}_s(k)]\, \mathbf{w}_e \,[\mathbf{r}(k) - \mathbf{y}_s(k)]$$

Subject to

$$\begin{aligned} \left[\mathbf{I}-\mathbf{\Phi})\right]\mathbf{z}_s(k) &= \mathbf{\Gamma}_u\mathbf{u}_s(k)+\mathbf{L}\mathbf{e}_f(k) && (202)\\ \mathbf{y}_s(k) &= \mathbf{C}\mathbf{z}_s(k)+\mathbf{e}_f(k) && (203)\\ \mathbf{u}_L &\leq \mathbf{u}_s(k)\leq \mathbf{u}_H && (204) \end{aligned}$$

If the number of manipulated inputs is more than the number of controlled outputs, then time-varying target state can be computed by solving the following optimization problem

$$\min_{\mathbf{u_s}(k)} \mathbf{u_s}(k)^T\mathbf{w}_U\mathbf{u_s}(k) \tag{205}$$

subject to,

$$\begin{aligned} \left[\mathbf{I}-\mathbf{\Phi})\right]\mathbf{z}_s(k) &= \mathbf{\Gamma}_u\mathbf{u}_s(k)+\mathbf{L}\mathbf{e}_f(k) && (206)\\ \mathbf{r}(k) &= \mathbf{C}\mathbf{z}_s(k)+\mathbf{e}_f(k) && (207)\\ \mathbf{u}_L &\leq \mathbf{u}_s(k)\leq \mathbf{u}_H && (208) \end{aligned}$$

4.4 State Space Formulation of MPC

Given the target states, the model predictive control problem at the sampling instant k is defined as a constrained optimization problem whereby the future manipulated input moves $\mathbf{u}(k|k), \mathbf{u}(k+1|k)......\mathbf{u}(k+q-1|k)$ are determined by minimizing an objective function J defined as follows

$$\begin{aligned} J &= \left[\widehat{\mathbf{z}}(k+p)-\mathbf{z}_s(k)\right]^T\mathbf{w}_\infty\left[\widehat{\mathbf{z}}(k+p)-\mathbf{z}_s(k)\right]\\ &\quad+\sum_{j=1}^{p}\left[\widehat{\mathbf{z}}(k+j)-\mathbf{z}_s(k)\right]^T\mathbf{w}_x\left[\widehat{\mathbf{z}}(k+j)-\mathbf{z}_s(k)\right] && (209)\\ &\quad+\sum_{j=0}^{q-1}\left[\mathbf{u}(k+j|k)-\mathbf{u}_s(k)\right]^T\mathbf{w}_u\left[\mathbf{u}(k+j|k)-\mathbf{u}_s(k)\right] && (210) \end{aligned}$$

Here, $\mathbf{w}_x$ represents positive definite state error weighting matrix and $\mathbf{w}_u$ represents positive definite the input weighting matrix. The terminal state weighting matrix $\mathbf{w}_\infty$ can be found by solving discrete Lyapunov equation. When poles of $\mathbf{\Phi}$ are inside the unit circle, the terminal state weighting matrix $\mathbf{w}_\infty$ can be found by solving discrete Lyapunov equation given as

$$\mathbf{w}_\infty=\mathbf{w}_x+\mathbf{\Phi}^T\mathbf{w}_\infty\mathbf{\Phi} \tag{211}$$

When some poles of $\mathbf{\Phi}$ are outside unit circle, the procedure for computing the terminal weighting matrix is given in Muske and Rawlings (1993). The above minimization problem is subject to following operating constraints

- Model prediction equations (200-201)
- Manipulated input constraints

$$\mathbf{u}(k+q|k) = \mathbf{u}(k+q+1|k) = ... = \mathbf{u}(k+p-1|k) = \mathbf{u}(k+q-1|k) \tag{212}$$

$$\begin{aligned} \mathbf{u}^L &\leq \mathbf{u}(k+j|k) \leq \mathbf{u}^H && (213) \\ \Delta\mathbf{u}^L &\leq \Delta\mathbf{u}(k+j|k) \leq \Delta\mathbf{u}^H && (214) \\ j &= 0, 1, 2, ...q-1 \end{aligned}$$

- If linearized version of a mechanistic model is used for developing the MPC scheme, then it is possible to impose constraints on the predicted states as well

$$\begin{aligned} \mathbf{x}_L &\leq \widehat{\mathbf{z}}(k+j) \leq \mathbf{x}_H && (215) \\ j &= p_1, p_1+1, ...p \end{aligned}$$

  where p_1 denotes the constraint horizon.

4.5 Nominal Stability (Goodwin et al., 2009)

Similar to LQ formulation, MPC formulation is based on minimization of a performance measure. The question that naturally arises is whether the resulting formulation guarantee stable closed loop behavior. It is relatively easy to answer this question under the nominal conditions i.e. in absence of MPM and under the deterministic settings. In this sub-section, we examine nominal stability of a deterministic version of MPC under the following simplifying assumptions

- There is no model plant mismatch or unmeasured disturbances are absent and both internal model (i.e. observer) and plant evolve according to

$$\mathbf{x}(k+1) = \mathbf{\Phi}\mathbf{x}(k) + \mathbf{\Gamma}\mathbf{u}(k) \tag{216}$$

- The true states are perfectly measurable
- It is desired to control the system at the origin.

With the above assumptions, the MPC formulation developed in the previous section can be posed as minimization of a loss function of the form

$$\begin{aligned} J(\mathbf{x}(k), \mathbf{U}_p(k)) &= \mathbf{x}(k+p)^T \mathbf{w}_\infty \mathbf{x}(k+j) + \sum_{j=1}^{p} \mathbf{x}(k+j)^T \mathbf{w}_x \mathbf{x}(k+j) \\ &\quad + \sum_{j=0}^{p-1} \mathbf{u}(k+j|k)^T \mathbf{w}_u \mathbf{u}(k+j|k) && (217) \end{aligned}$$

with respect to $\mathbf{U}_p(k)$. Though we have formulated MPC as minimization of a quadratic loss function for the sake of keeping the development simple and facilitate connection with LQOC, the MPC objective function can be formulated using any other suitable positive function of the vectors involved. Thus, let us formulate MPC in terms of a generalized loss function

$$J\left[\mathbf{x}(k), \mathbf{U}_p(k)\right] = \varphi\left[\mathbf{x}(k+p)\right] + \sum_{j=1}^{p} \pi\left[\mathbf{x}(k+j), \mathbf{u}(k+j|k)\right] \tag{218}$$

where $\varphi\left[\mathbf{x}(k+p)\right]$ and $\pi\left[\mathbf{x}(k+j), \mathbf{u}(k+j|k)\right]$ are positive functions of their respective arguments such that $J\left[\mathbf{x}(k), \mathbf{U}_p(k)\right] \to \infty$ as $\|\mathbf{x}(k)\| \to \infty$. Thus, $J\left[\mathbf{x}(k), \mathbf{U}_p(k)\right]$ is a candidate Lyapunov function. It is desired to minimize $J\left[\mathbf{x}(k), \mathbf{U}_p(k)\right]$ subject to

$$\mathbf{U}_p(k) \in \Xi_U$$

where Ξ_U is a closed and bounded set. Let us assume that an additional constraint of the form

$$\mathbf{x}(k+p) = \overline{\mathbf{0}}$$

is placed on the moving horizon optimization problem each time. Let us denote the optimal solution of the resulting constrained optimization problem at instant k as

$$\mathbf{U}_p^*(k) = \begin{bmatrix} \mathbf{u}^*(k|k)^T & \mathbf{u}^*(k+1|k)^T & .... & \mathbf{u}^*(k+p-1|k)^T \end{bmatrix}^T \tag{219}$$

In MPC formulation, we set $\mathbf{u}(k) = \mathbf{u}^*(k|k)$ and the optimization problem is solved again over window $[k+1, k+p+1]$. It may be noted that $\mathbf{u}^*(k|k)$ is some function of $\mathbf{x}(k)$, say $\mathbf{u}^*(k|k) = h\left[\mathbf{x}(k)\right]$. Let

$$\mathbf{x}(k+1) = \mathbf{\Phi}\mathbf{x}(k) + \mathbf{\Gamma}\mathbf{u}^*(k|k) = \mathbf{\Phi}\mathbf{x}(k) + \mathbf{\Gamma}h\left[\mathbf{x}(k)\right] \tag{220}$$

and $\mathbf{x}(k+p) = \overline{\mathbf{0}}$ be terminal state resulting from application of $\mathbf{U}_p^*(k)$ on the system (216).

Now, let

$$\mathbf{U}_p^*(k+1) = \begin{bmatrix} \mathbf{u}^*(k+1|k+1)^T & .... & \mathbf{u}^*(k+p-1|k+1)^T & \mathbf{u}^*(k+p|k+1)^T \end{bmatrix}^T \tag{221}$$

represent the optimum solution of the MPC problem over the window $[k+1, k+p+1]$ and we want to examine

$$\Delta J\left[\mathbf{x}, \mathbf{U}_f\right] = J\left[\mathbf{x}(k+1), \mathbf{U}_p^*(k+1)\right] - J\left[\mathbf{x}(k), \mathbf{U}_p^*(k)\right] \tag{222}$$

A *non-optimal but feasible* solution for the optimization problem over window $[k+1, k+p+1]$ is

$$\mathbf{U}_p(k+1) = \begin{bmatrix} \mathbf{u}^*(k+1|k)^T & \mathbf{u}^*(k+2|k)^T & .... & \mathbf{u}^*(k+p-1|k)^T & \overline{\mathbf{0}} \end{bmatrix}^T \tag{223}$$

for this feasible solution, the following inequality holds

$$J\left[\mathbf{x}(k+1), \mathbf{U}_p^*(k+1)\right] \leq J\left[\mathbf{x}(k+1), \mathbf{U}_p(k+1)\right]$$

Thus, it follows that

$$J\left[\mathbf{x}(k+1), \mathbf{U}_p^*(k+1)\right] - J\left[\mathbf{x}(k), \mathbf{U}_p^*(k)\right] \leq J\left[\mathbf{x}(k+1), \mathbf{U}_p(k+1)\right] - J\left[\mathbf{x}(k), \mathbf{U}_p^*(k)\right] \quad (224)$$

Now, since $\mathbf{x}(k+p) = \overline{\mathbf{0}}$ (by assumption), with application of the last element of $\mathbf{U}_p(k+1)$ (i.e. $\overline{\mathbf{0}}$) on (216) yields

$$\mathbf{x}(k+p+1) = \overline{\mathbf{0}}$$

and consequently

$$\pi\left[\mathbf{x}(k+p+1), \overline{\mathbf{0}}\right] = 0$$

This, together with the fact that $\mathbf{U}_p^*(k)$ and $\mathbf{U}_p(k+1)$ share (p-1) sub-vectors, it follows that

$$\begin{aligned}
& J\left[\mathbf{x}(k+1), \mathbf{U}_f(k+1)\right] - J\left[\mathbf{x}(k), \mathbf{U}_f^*(k)\right] \\
= & \sum_{j=2}^{p} \pi\left[\mathbf{x}(k+j), \mathbf{u}^*(k+j|k)\right] - \sum_{j=1}^{p} \pi\left[\mathbf{x}(k+j), \mathbf{u}^*(k+j|k)\right] \\
= & -\pi\left[\mathbf{x}(k), h\left[\mathbf{x}(k)\right]\right] < 0 \text{ if } \mathbf{x}(k) \neq \overline{\mathbf{0}} \qquad (225)
\end{aligned}$$

Thus, it follows that

$$\Delta J\left[\mathbf{x}, \mathbf{U}_p\right] = J\left[\mathbf{x}(k+1), \mathbf{U}_p^*(k+1)\right] - J\left[\mathbf{x}(k), \mathbf{U}_p^*(k)\right] \leq -\pi\left[\mathbf{x}(k), h\left[\mathbf{x}(k)\right]\right] < 0 \text{ if } \mathbf{x}(k) \neq \overline{\mathbf{0}}$$

and the nominal closed loop system is globally asymptotically stable.

It is remarkable that we are able to construct a Lyapunov function using the MPC loss function. Thus, under the nominal conditions, MPC controller guarantees global asymptotic stability and optimal performance.

5 Summary

These lecture notes introduce various facets of model predictive control, which is formulation derived from a discrete linear state space model. To begin with, linear quadratic optimal regulator is developed and the resulting control law is implemented using states estimated from a state estimator. Stability of the closed loop system, consisting of the observer and the plant, is established under the nominal conditions by establishing the separation principle. The LQ formulation is further modified by introducing time varying target states and target inputs to deal with the model plant mismatch and drifting unmeasured disturbances and to track changing setpoints. We later move to develop the conventional MPC formulation, which is aimed at dealing with the operating constraints. Unconstrained MPC formulation is shown to be a form of state feedback controller. Constrained MPC is later formulated as a quadratic programming problem to enhance efficiency of the on-line computations. We then develop a state space model based formulation of MPC, which is similar to the LQ formulation with constraints. The closed loop stability is established under the nominal conditions for the deterministic linear model based MPC formulation.

These notes only touch upon tip of the iceberg called MPC and are only meant to be an introduction to a novice. A lot remains to be discussed. For example, important issue of robust stability and robust design under MPM is not touched. Extensions, such as explicit MPC or adaptive MPC, are not discussed. MPC formulations based on nonlinear dynamic models is another well developed area. Meadows and Rawlings (1997) provide a tutorial introduction to nonlinear model based predictive control. Henson (1998) and Qin and Badgwell (2003) have reviewed nonlinear model based MPC formulations. Interested reader is referred to the literature cited if he / she wants to know more about this exciting area. Hope this gentle introduction is sufficient to tread that path.

6 References

Astrom, K. J. and B. Wittenmark, Computer Controlled Systems, Prentice Hall, 1990.

Camacho, E. C. and C. Bourdons, 1999, "Model Predictive Control", Springer Verlag, London.

Franklin, G. F. and J. D. Powell, Digital Control of Dynamic Systems, Addison-Wesley, 1989.

Garcia, C. E., Prett, D. M. , Morari, M. Model predictive control: Theory and practice - A survey. Automatica, 25 (1989), 335-348.

Glad, T., Ljung, L. Control Theory: Multivariable and Nonlinear Methods, Taylor and Francis, 2000.

Goodwin, G., Graebe, S. F., Salgado, M. E., Control System Design, Phi Learning, 2009.

Henson, M.A. (1998). Nonlinear Model Predictive Control : Current status and future directions. *Computers and Chemical Engg,* **23** , 187- 202.

Lee, J.H. (1998). Modeling and Identification for Nonlinear Model Predictive control:Requirements present status and future needs. International Symposium on Nonlinear Model Predictive control,Ascona, Switzerland.

Kothare, M. V., B. Mettler, M. Morari and P. Bendotti (2000) Level Control in the Steam Generator of a Nuclear Power Plant, IEEE Transactions On Control Systems Technology, Vol. 8, No. 1, 55-69.

Meadows, E.S. , Rawlings, J. B. (1997) *Nonlinear Process Control*, (M.A. Henson and D.E. Seborg (eds.), New Jersey: Prentice Hall, Chapter 5.

Morari, M. , Lee, J.H., (1999), Model Predictive Control: Past, Present and Future, *Comp. Chem. Engg.*, 23, 667-682.

Muske, K. R. , Rawlings, J. B., (1993). Model Predictive control with linear models, *AIChE J.*, 39, 262-287.

Muske, K. R. ;Badgwell, T. A. Disturbance modeling for offset-free linear model predictive control. *Journal of Process Control*, 12 (2002), 617-632.

Patwardhan S.C. and S.L. Shah (2005) From data to diagnosis and control using generalized orthonormal basis filters. Part I: Development of state observers, *Journal of Process Control*,15,7, 819-835.

Qin, S.J., Badgwell, T.A. A survey of industrial model predictive control technology, Control Engineering Practice 11 (2003) 733-764.

Rawlings, J. B. and D. Q. Mayne, (2009), Model Predictive Control Theory and Design, North Hill Publication.

Ricker, N. L., 1990, Model Predictive Control with State Estimation, *Ind. Eng. Chem. Res.,* 29, 374-382.

Sodderstrom, T. Discrete Time Stochstic Systems. Springer, 2003.

Yu, Z. H. , Li , W., Lee, J.H. , Morari, M. (1994), State Estimation Based Model Predictive Control applied to Shell Control Problem: A Case Study, *Chem. Eng. Sci.*, 14-22.