# Model Predictive Control

February 28, 2017

John Leth

AALBORG UNIVERSITY
DENMARK

# Agenda



LECTURE I

Formulation of MPC

Solving MPC (unconstrained)

LECTURE II

Solving MPC (constrained)

Terminal Constraints and Stability

Infinite Horizon and Stability (unconstrained)

LECTURE III

Infinite Horizon and Stability (constrained)

The Fake Algebraic Riccati Equation

# Idea of Model Predictive Control



To control a plant in an “optimal” manner, that is with respect to a given cost function, by

- first, optimizing over a certain prediction horizon $H_p$ (that is, $H_p$ samples into the future) to obtain a sequence of predicted optimal control inputs
- secondly, apply the first sample of the determined predicted optimal input to the plant
- thirdly, move one sample and repeat the above procedure (receeding horizon)

# State Space Model

We consider the discrete time linear system

$$
\begin{aligned}
x(k+1) &= Ax(k) + Bu(k) \\
y(k) &= C_y x(k) \\
z(k) &= C_z x(k)
\end{aligned}
$$

with

$$
\begin{aligned}
&x \in \mathbb{R}^n \quad \text{state} \\
&u \in \mathbb{R}^l \quad \text{input} \\
&y \in \mathbb{R}^{m_y} \quad \text{measured output} \\
&z \in \mathbb{R}^{m_z} \quad \text{output to be controlled}
\end{aligned}
$$

# State Space Model



We consider the discrete time linear system

$$
\begin{aligned}
x(k+1) =& Ax(k) + Bu(k) \\
y(k) =& C_y x(k) \\
z(k) =& C_z x(k)
\end{aligned}
$$

Remark

- $y$ and $z$ are usually overlapping
- if $y = z$ set $m = m_y = m_z$ and $C = C_m$

# State Space Model



We consider the discrete time linear system

$$
\begin{aligned}
x(k+1) =& Ax(k) + Bu(k) \\
y(k) =& C_y x(k) \\
z(k) =& C_z x(k)
\end{aligned}
$$

Action steps at time $k$

- obtain measurement $y(k)$
- compute input $u(k)$
- apply input $u(k)$

# MPC Predictions



We use the following notation

- $\hat{u}(k+i|k)$ : future value (at time $k+i$) of $u$, assumed at time $k$

and for $\phi$ denoting either $x$, $y$ or $z$

- $\hat{\phi}(k+i|k)$ : predicted value (at time $k+i$) of $\phi$, made at time $k$ and based on the sequence $\hat{u}(k|k), \hat{u}(k+1|k), \ldots, \hat{u}(k+i-1|k)$ and on a known (or estimated) $\hat{x}(k|k)$

# Cost Function



We optimize with respect to the quadratic cost function

$$V(k) = \sum_{i=H_w}^{H_p} \|\hat{z}(k+i|k) - r(k+i|k)\|^2_{Q(i)} + \sum_{i=0}^{H_u-1} \|\Delta\hat{u}(k+i|k)\|^2_{R(i)}$$

where

- $\|w\|_M = \sqrt{w^T M w}$
- weights: $Q(i), R(i) \geq 0$ (positive semi-definite)
- reference signal $r(k+i|k)$ (may depend on measurements up to time $k$)
- prediction horizon $H_p \geq$ control horizon $H_u$
- window parameter $H_w \geq 1$
- change in input (or control move):

$$\begin{aligned} \Delta\hat{u}(k+i|k) &= \hat{u}(k+i|k) - \hat{u}(k+i-1|k), && i < H_u \\ \Delta\hat{u}(k+i|k) &= 0, && i \geq H_u \end{aligned}$$

# Cost Function



We optimize with respect to the quadratic cost function

$$V(k) = \sum_{i=H_w}^{H_p} \|\hat{z}(k+i|k) - r(k+i|k)\|^2_{Q(i)} + \sum_{i=0}^{H_u-1} \|\Delta\hat{u}(k+i|k)\|^2_{R(i)}$$

- The quantities

  $$Q(i), R(i), H_p, H_u, H_w,$$

  are usually considered as tuning parameters
- The use of $\Delta\hat{u}$ (instead of just $\hat{u}$) is to introduce integral action (to avoid steady state error)

# Cost Function



We optimize with respect to the quadratic cost function

$$V(k) = \sum_{i=H_w}^{H_p} \|\hat{z}(k+i|k) - r(k+i|k)\|^2_{Q(i)} + \sum_{i=0}^{H_u-1} \|\Delta\hat{u}(k+i|k)\|^2_{R(i)}$$

## Example

$$Q(1) = 0, \quad Q(2) = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}, \quad Q(3) = \begin{bmatrix} 0 & 0 \\ 0 & 2 \end{bmatrix}$$

# Constraints



To an optimization problem one often have to consider various constraints:

- actuator slew rate:

$$E \begin{bmatrix} \Delta\mathcal{U}(k) \\ 1 \end{bmatrix} \leq 0, \quad \Delta\mathcal{U}(k) = \left[\Delta\hat{u}(k|k)^T \cdots \Delta\hat{u}(k+H_u-1|k)^T\right]^T$$

- actuator range:

$$F \begin{bmatrix} \mathcal{U}(k) \\ 1 \end{bmatrix} \leq 0, \quad \mathcal{U}(k) = \left[\hat{u}(k|k)^T \cdots \hat{u}(k+H_u-1|k)^T\right]^T$$

- constraints on the controlled variable:

$$G \begin{bmatrix} \mathcal{Z}(k) \\ 1 \end{bmatrix} \leq 0, \quad \mathcal{Z}(k) = \left[\hat{z}(k+H_w|k)^T \cdots \hat{z}(k+H_p|k)^T\right]^T$$

with $E, F$ and $G$ matrices of appropriate dimensions

**Example**

2 input, 2 controlled variables, $H_u, H_w = 1$ and $H_p = 2$

$$-2 \leq \Delta u_1 \leq 2 \Leftrightarrow \left\{ \begin{array}{cccc} -\frac{1}{2}\Delta u_1 & -1 & \leq & 0 \\ \frac{1}{2}\Delta u_1 & -1 & \leq & 0 \end{array} \right\}$$

## Example

2 input, 2 controlled variables, $H_u, H_w = 1$ and $H_p = 2$

$$-2 \leq \Delta u_1 \leq 2 \Leftrightarrow \left\{ \begin{array}{cccc} -\frac{1}{2}\Delta u_1 & -1 & \leq & 0 \\ \frac{1}{2}\Delta u_1 & -1 & \leq & 0 \end{array} \right\}$$

$$E \begin{bmatrix} \Delta\mathcal{U}(k) \\ 1 \end{bmatrix} = \begin{bmatrix} -\frac{1}{2} & 0 & -1 \\ \frac{1}{2} & 0 & -1 \end{bmatrix} \begin{bmatrix} \Delta\hat{u}_1(k|k) \\ \Delta\hat{u}_2(k|k) \\ 1 \end{bmatrix} \leq 0$$

### Example

2 input, 2 controlled variables, $H_u, H_w = 1$ and $H_p = 2$

$$\left\{ \begin{array}{ccc} z_1 & \geq & 0 \\ z_2 & \geq & 0 \\ 3z_1 + 5z_2 & \leq & 15 \end{array} \right\} \Leftrightarrow \left\{ \begin{array}{cccccc} -z_1 & & & \leq & 0 \\ & -z_2 & & \leq & 0 \\ \frac{1}{5}z_1 & +\frac{1}{3}z_2 & -1 & \leq & 0 \end{array} \right\}$$

## Example

2 input, 2 controlled variables, $H_u, H_w = 1$ and $H_p = 2$

$$\left\{ \begin{array}{ccc} z_1 & \geq & 0 \\ z_2 & \geq & 0 \\ 3z_1 + 5z_2 & \leq & 15 \end{array} \right\} \Leftrightarrow \left\{ \begin{array}{ccccc} -z_1 & & & \leq & 0 \\ & -z_2 & & \leq & 0 \\ \frac{1}{5}z_1 & +\frac{1}{3}z_2 & -1 & \leq & 0 \end{array} \right\}$$

$$G \begin{bmatrix} \mathcal{Z}(k) \\ 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ \frac{1}{5} & \frac{1}{3} & 0 & 0 & -1 \\ 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & \frac{1}{5} & \frac{1}{3} & -1 \end{bmatrix} \begin{bmatrix} \hat{z}_1(k+1|k) \\ \hat{z}_2(k+1|k) \\ \hat{z}_1(k+2|k) \\ \hat{z}_2(k+2|k) \\ 1 \end{bmatrix} \leq 0$$

# Optimization Problem



We consider the optimization problem

$$\min_{\Delta\mathcal{U}(k)} V(k) = \sum_{i=H_w}^{H_p} \|\hat{z}(k+i|k) - r(k+i|k)\|^2_{Q(i)} + \sum_{i=0}^{H_u-1} \|\Delta\hat{u}(k+i|k)\|^2_{R(i)}$$

subject to the dynamics

$$\begin{aligned} \hat{x}(k+i+1|k) &= A\hat{x}(k+i|k) + B\hat{u}(k+i|k) \\ \hat{y}(k+i|k) &= C_y\hat{x}(k+i|k) \\ \hat{z}(k+i|k) &= C_z\hat{x}(k+i|k) \end{aligned}$$

and the constraints

$$E\begin{bmatrix} \Delta\mathcal{U}(k) \\ 1 \end{bmatrix} \leq 0, \; F\begin{bmatrix} \mathcal{U}(k) \\ 1 \end{bmatrix} \leq 0, \; G\begin{bmatrix} \mathcal{Z}(k) \\ 1 \end{bmatrix} \leq 0$$

# Important observations



- The cost is quadratic
- The dynamics is linear
- The constraints are linear inequalities
- The constraints and the “parameters” $Q(i), R(i), H_p, H_u, H_w$, do not depend on time
- The prediction horizon $H_p$ should be chosen bigger than the slowest dynamics
- The matrices $E, F, G$ can become large

# Reformulation

Since we optimize over $\Delta\mathcal{U}$, it is advantageous to reformulate the dynamics and constraints in therms of $\Delta\mathcal{U}$, that is, reformulate $\mathcal{U}$ and $\mathcal{Z}$ in terms of $\Delta\mathcal{U}$.

# Reformulation



reformulate $\mathcal{U}$ and $\mathcal{Z}$ in terms of $\Delta\mathcal{U}$.

$$
\begin{aligned}
\hat{x}(k+1|k) &= A\hat{x}(k|k) + B\hat{u}(k|k) \\
\hat{x}(k+2|k) &= A\hat{x}(k+1|k) + B\hat{u}(k+1|k) \\
&= A^2\hat{x}(k|k) + AB\hat{u}(k|k) + B\hat{u}(k+1|k) \\
&\vdots \\
\hat{x}(k+i|k) &= A^i\hat{x}(k|k) + A^{i-1}B\hat{u}(k|k) + \cdots \\
&\quad \cdots + A^{i-j}B\hat{u}(k+j-1|k) + \cdots + B\hat{u}(k+i-1|k) \\
&\vdots \\
\hat{x}(k+H_p|k) &= A^{H_p}\hat{x}(k|k) + A^{H_p-1}B\hat{u}(k|k) + \cdots + B\hat{u}(k+H_p-1|k)
\end{aligned}
$$

# Reformulation



reformulate $\mathcal{U}$ and $\mathcal{Z}$ in terms of $\Delta\mathcal{U}$.

$$\begin{aligned}
\hat{u}(k|k) &= \Delta\hat{u}(k|k) + u(k-1) \\
\hat{u}(k+1|k) &= \Delta\hat{u}(k+1|k) + \Delta\hat{u}(k|k) + u(k-1) \\
&\vdots \\
\hat{u}(k+H_u-1|k) &= \Delta\hat{u}(k+H_u-1|k) + \cdots + \Delta\hat{u}(k|k) + u(k-1)
\end{aligned}$$

# Reformulation



reformulate $\mathcal{U}$ and $\mathcal{Z}$ in terms of $\Delta\mathcal{U}$.

$$\begin{aligned}
\hat{x}(k+1|k) &= A\hat{x}(k|k) + B(\Delta\hat{u}(k|k) + u(k-1)) \\
\hat{x}(k+2|k) &= A^2\hat{x}(k|k) + AB(\Delta\hat{u}(k|k) + u(k-1)) \\
&\quad + B(\Delta\hat{u}(k+1|k) + \Delta\hat{u}(k|k) + u(k-1)) \\
&= A^2\hat{x}(k|k) + (A+I)B\Delta\hat{u}(k|k) + B\Delta\hat{u}(k+1|k) + (A+I)Bu(k-1) \\
&\vdots \\
\hat{x}(k+H_u|k) &= A^{H_u}\hat{x}(k|k) + (A^{H_u-1} + \cdots + A + I)B\Delta\hat{u}(k|k) + \cdots \\
&\quad + B\Delta\hat{u}(k+H_u-1|k) + (A^{H_u-1} + \cdots + A + I)Bu(k-1) \\
\hat{x}(k+H_u+1|k) &= A^{H_u+1}\hat{x}(k|k) + (A^{H_u} + \cdots + A + I)B\Delta\hat{u}(k|k) + \cdots \\
&\quad + (A+I)B\Delta\hat{u}(k+H_u-1|k) + (A^{H_u} + \cdots + A + I)Bu(k-1) \\
&\vdots \\
\hat{x}(k+H_p|k) &= A^{H_p}\hat{x}(k|k) + (A^{H_p-1} + \cdots + A + I)B\Delta\hat{u}(k|k) + \cdots \\
&\quad + (A^{H_p-H_u} + \cdots + A + I)B\Delta\hat{u}(k+H_u-1|k) + (A^{H_p} + \cdots + A + I)Bu(k-1)
\end{aligned}$$

# Reformulation

In matrix notation this yields

$$\mathcal{X}(k) = \mathcal{A}\hat{x}(k|k) + \mathcal{B}_u u(k-1) + \mathcal{B}_{\Delta u}\Delta\mathcal{U}(k)$$

where

$$\mathcal{X}(k) = \begin{bmatrix} \hat{x}(k+1|k) \\ \vdots \\ \hat{x}(k+H_u|k) \\ \hat{x}(k+H_u+1|k) \\ \vdots \\ \hat{x}(k+H_p|k) \end{bmatrix}$$

# Reformulation



In matrix notation this yields

$$\mathcal{X}(k) = \mathcal{A}\hat{x}(k|k) + \mathcal{B}_u u(k-1) + \mathcal{B}_{\Delta u}\Delta\mathcal{U}(k)$$

where

$$\mathcal{A} = \begin{bmatrix} A \\ \vdots \\ A^{H_u} \\ A^{H_u+1} \\ \vdots \\ A^{H_p} \end{bmatrix}$$

# Reformulation



In matrix notation this yields

$$\mathcal{X}(k) = \mathcal{A}\hat{x}(k|k) + \mathcal{B}_u u(k-1) + \mathcal{B}_{\Delta u}\Delta\mathcal{U}(k)$$

where

$$\mathcal{B}_u = \begin{bmatrix} B \\ \vdots \\ \sum_{i=0}^{H_u-1} A^i B \\ \sum_{i=0}^{H_u} A^i B \\ \vdots \\ \sum_{i=0}^{H_p} A^i B \end{bmatrix}$$

In matrix notation this yields

$$\mathcal{X}(k) = \mathcal{A}\hat{x}(k|k) + \mathcal{B}_u u(k-1) + \mathcal{B}_{\Delta u}\Delta\mathcal{U}(k)$$

where

$$\mathcal{B}_{\Delta u} = \begin{bmatrix} B & 0 & 0 & \cdots & 0 \\ AB+B & B & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & & \vdots \\ \sum_{i=0}^{H_u-1} A^i B & & \cdots & & B \\ \sum_{i=0}^{H_u} A^i B & & \cdots & & AB+B \\ \vdots & & \vdots & & \vdots \\ \sum_{i=0}^{H_p} A^i B & & \cdots & & \sum_{i=0}^{H_p-H_u} A^i B \end{bmatrix}$$

# Reformulation



In matrix notation this yields

$$\mathcal{X}(k) = \mathcal{A}\hat{x}(k|k) + \mathcal{B}_u u(k-1) + \mathcal{B}_{\Delta u}\Delta\mathcal{U}(k)$$

and hence

$$\mathcal{Z}(k) = \Psi\hat{x}(k|k) + \Upsilon u(k-1) + \Theta\Delta\mathcal{U}(k)$$

where

$$\mathcal{Z}(k) = \begin{bmatrix} \hat{z}(k+H_w|k) \\ \vdots \\ \hat{z}(k+H_p|k) \end{bmatrix}$$

# Reformulation



In matrix notation this yields

$$\mathcal{X}(k) = \mathcal{A}\hat{x}(k|k) + \mathcal{B}_u u(k-1) + \mathcal{B}_{\Delta u}\Delta\mathcal{U}(k)$$

and hence

$$\mathcal{Z}(k) = \Psi\hat{x}(k|k) + \Upsilon u(k-1) + \Theta\Delta\mathcal{U}(k)$$

where

$$\mathcal{C} = diag(C_z), \quad \Psi = \mathcal{C}\mathcal{A}, \quad \Upsilon = \mathcal{C}\mathcal{B}_u, \quad \Theta = \mathcal{C}\mathcal{B}_{\Delta u}$$

**Warrning: the matrices are often ill-conditioned**

# Reformulation



In matrix notation this yields

$$\mathcal{X}(k) = \mathcal{A}\hat{x}(k|k) + \mathcal{B}_u u(k-1) + \mathcal{B}_{\Delta u}\Delta\mathcal{U}(k)$$

and hence

$$\mathcal{Z}(k) = \Psi\hat{x}(k|k) + \Upsilon u(k-1) + \Theta\Delta\mathcal{U}(k)$$

Furthermore define the vector of future predicted errors with $\Delta\mathcal{U}(k) = 0$

$$\mathcal{E}(k) = \mathcal{T}(k) - \Psi\hat{x}(k|k) - \Upsilon u(k-1)$$

where

$$\mathcal{T}(k) = \begin{bmatrix} r(k+H_w|k) \\ \vdots \\ r(k+H_p|k) \end{bmatrix}$$

# Reformulating the Cost



With

$$
\begin{aligned}
\mathcal{X}(k) &= \mathcal{A}\hat{x}(k|k) + \mathcal{B}_u u(k-1) + \mathcal{B}_{\Delta u}\Delta\mathcal{U}(k) \\
\mathcal{Z}(k) &= \Psi\hat{x}(k|k) + \Upsilon u(k-1) + \Theta\Delta\mathcal{U}(k) \\
\mathcal{E}(k) &= \mathcal{T}(k) - \Psi\hat{x}(k|k) - \Upsilon u(k-1) \\
\mathcal{Q} &= diag(Q(H_w), \dots, Q(H_p)) \\
\mathcal{R} &= diag(R(0), \dots, R(H_u - 1))
\end{aligned}
$$

we reformulate the cost

$$
\begin{aligned}
V(k) &= \sum_{i=H_w}^{H_p} \|\hat{z}(k+i|k) - r(k+i|k)\|^2_{Q(i)} + \sum_{i=0}^{H_u-1} \|\Delta\hat{u}(k+i|k)\|^2_{R(i)} \\
&= \|\mathcal{Z}(k) - \mathcal{T}(k)\|^2_{\mathcal{Q}} + \|\Delta\mathcal{U}(k)\|^2_{\mathcal{R}} \\
&= \|\Theta\Delta\mathcal{U}(k) - \mathcal{E}(k)\|^2_{\mathcal{Q}} + \|\Delta\mathcal{U}(k)\|^2_{\mathcal{R}}
\end{aligned}
$$

in terms of signals which are 'known' (if the state vector is known) at time k and $\Delta\mathcal{U}(k)$

# Reformulating the Cost (again)



For computational reasons we (once again) rewrite the cost

$$\begin{aligned} V(k) &= \|\Theta\Delta\mathcal{U}(k) - \mathcal{E}(k)\|^2_{\mathcal{Q}} + \|\Delta\mathcal{U}(k)\|^2_{\mathcal{R}} \\ &= \mathcal{E}(k)^T\mathcal{Q}\mathcal{E}(k) - 2\Delta\mathcal{U}(k)^T\Theta^T\mathcal{Q}\mathcal{E}(k) + \Delta\mathcal{U}(k)^T[\Theta^T\mathcal{Q}\Theta + \mathcal{R}]\Delta\mathcal{U}(k) \\ &= const - \Delta\mathcal{U}(k)^T\mathcal{G} + \Delta\mathcal{U}(k)^T\mathcal{H}\Delta\mathcal{U}(k) \end{aligned}$$

with

$$\mathcal{G} = 2\Theta^T\mathcal{Q}\mathcal{E}(k), \qquad \mathcal{H} = \Theta^T\mathcal{Q}\Theta + \mathcal{R}$$

We first solve the unconstrained optimization problem

$$\min_{\Delta\mathcal{U}(k)} V(k) = \sum_{i=H_w}^{H_p} \|\hat{z}(k+i|k) - r(k+i|k)\|^2_{Q(i)} + \sum_{i=0}^{H_u-1} \|\Delta\hat{u}(k+i|k)\|^2_{R(i)}$$

subject to the dynamics

$$\begin{aligned} \hat{x}(k+i+1|k) &= A\hat{x}(k+i|k) + B\hat{u}(k+i|k) \\ \hat{y}(k+i|k) &= C_y\hat{x}(k+i|k) \\ \hat{z}(k+i|k) &= C_z\hat{x}(k+i|k) \end{aligned}$$

which is equivalent to solving

$$\min_{\Delta\mathcal{U}(k)} V(k) = const - \Delta\mathcal{U}(k)^T\mathcal{G} + \Delta\mathcal{U}(k)^T\mathcal{H}\Delta\mathcal{U}(k)$$

with

$$\mathcal{G} = 2\Theta^T\mathcal{Q}\mathcal{E}(k), \qquad \mathcal{H} = \Theta^T\mathcal{Q}\Theta + \mathcal{R}$$

# Unconstrained Control Law

The unconstrained control law can hence be found by setting the gradient of $V$ equal to zero

$$0 = \nabla_{\Delta\mathcal{U}(k)} V(k) = -\mathcal{G} + 2\mathcal{H}\Delta\mathcal{U}(k)$$

and then solve with respect to $\Delta\mathcal{U}(k)$

$$\Delta\mathcal{U}(k)_{opt} = \frac{1}{2}\mathcal{H}^{-1}\mathcal{G}$$

# Unconstrained Control Law



The unconstrained control law can hence be found by setting the gradient of $V$ equal to zero

$$0 = \nabla_{\Delta\mathcal{U}(k)} V(k) = -\mathcal{G} + 2\mathcal{H}\Delta\mathcal{U}(k)$$

and then solve with respect to $\Delta\mathcal{U}(k)$

$$\Delta\mathcal{U}(k)_{opt} = \frac{1}{2}\mathcal{H}^{-1}\mathcal{G}$$

To guarantee a minimum we need

$$\frac{\partial^2 V(k)}{\partial \Delta\mathcal{U}(k)^2} = 2\mathcal{H} = 2(\Theta^T \mathcal{Q}\Theta + \mathcal{R}) > 0$$

$$\mathcal{Q} = diag(Q(H_w), \ldots, Q(H_p)), \quad \mathcal{R} = diag(R(0), \ldots, R(H_u - 1))$$

# Unconstrained Control Law



The unconstrained control law can hence be found by setting the gradient of $V$ equal to zero

$$0 = \nabla_{\Delta\mathcal{U}(k)} V(k) = -\mathcal{G} + 2\mathcal{H}\Delta\mathcal{U}(k)$$

and then solve with respect to $\Delta\mathcal{U}(k)$

$$\Delta\mathcal{U}(k)_{opt} = \frac{1}{2}\mathcal{H}^{-1}\mathcal{G}$$

Now we pick out the control move at time k as the first $l$ rows of $\Delta\mathcal{U}(k)_{opt}$

$$\Delta\hat{u}(k|k)_{opt} = \begin{bmatrix} I_l & 0_l & \cdots & 0_l \end{bmatrix} \Delta\mathcal{U}(k)_{opt} \qquad (= \Delta u(k)_{opt}\ )$$

# Unconstrained Control Law



Recalling that

$$\mathcal{G} = 2\Theta^T \mathcal{Q}\mathcal{E}(k) = 2\Theta^T \mathcal{Q}(\mathcal{T}(k) - \Psi\hat{x}(k|k) - \Upsilon u(k-1))$$

$$\mathcal{H} = \Theta^T \mathcal{Q}\Theta + \mathcal{R}$$

we obtain

$$\Delta\hat{u}(k|k)_{opt} = \begin{bmatrix} I_l & 0_l & \cdots & 0_l \end{bmatrix} (\Theta^T \mathcal{Q}\Theta + \mathcal{R})^{-1}\Theta^T \mathcal{Q}(\mathcal{T}(k) - \Psi\hat{x}(k|k) - \Upsilon u(k-1))$$

- This control law is equivalent to a finite horizon LQ controller with feedback from the current state vector (of which $u(k-1)$ should be an element in this case).
- The control law also has feed forward from the reference trajectory $\mathcal{T}(k)$ which is equal to the term found using LQ

- We always need to guarantee

$$\mathcal{H} = (\Theta^T \mathcal{Q} \Theta + \mathcal{R}) > 0$$

with

$$\mathcal{Q} = diag(Q(H_w), \ldots, Q(H_p)), \quad \mathcal{R} = diag(R(0), \ldots, R(H_u - 1))$$

- Never compute the inverse of $\mathcal{H}$ in

$$\Delta\mathcal{U}(k)_{opt} = \frac{1}{2}\mathcal{H}^{-1}\mathcal{G}$$

Instead use Cholesky or SVD factorization of $\mathcal{Q}$ and $\mathcal{R}$ to obtain a least squares problem and then solve by QR factorization.

# Reformulation of Constrains



In the case where constraints are present we want to express these in terms of $\Delta\mathcal{U}(k)$ and variables known at time $k$.

For

$$F\begin{bmatrix}\mathcal{U}(k)\\ 1\end{bmatrix} \leq 0$$

we write $F = \begin{bmatrix} F_1 & \cdots & F_{H_u} & f \end{bmatrix}$ such that

$$F\begin{bmatrix}\mathcal{U}(k)\\ 1\end{bmatrix} = \sum_{j=1}^{H_u} F_j \hat{u}(k+j-1|k) + f \leq 0$$

which can be written in therms of $\Delta\hat{u}$ as

$$\sum_{i=1}^{H_u}\sum_{j=i}^{H_u} F_j \Delta\hat{u}(k+i-1|k) + \sum_{j=l}^{H_u} F_j \hat{u}(k-1) + f \leq 0$$

and then collected to the matrix inequality in $\Delta\mathcal{U}$

$$\mathbf{F}\Delta\mathcal{U}(k) \leq -\mathbf{F}_1 u(k-1) - f$$

with $\mathbf{F} = \begin{bmatrix} \mathbf{F}_1 & \cdots & \mathbf{F}_{H_u} \end{bmatrix}$ and $\mathbf{F}_i = \sum_{j=i}^{H_u} F_j$

For

$$G\begin{bmatrix}\mathcal{Z}(k)\\1\end{bmatrix} = \begin{bmatrix}\Psi\hat{x}(k|k) + \Upsilon u(k-1) + \Theta\Delta\mathcal{U}(k)\\1\end{bmatrix} \leq 0$$

we write $G = \begin{bmatrix}\Gamma & g\end{bmatrix}$ to obtain the matrix inequality in $\Delta\mathcal{U}$

$$\Gamma(\Psi\hat{x}(k|k) + \Upsilon u(k-1) + \Theta\Delta\mathcal{U}(k)) + g \leq 0$$
$$\Updownarrow$$
$$\Gamma\Theta\Delta\mathcal{U}(k) \leq -\Gamma(\Psi\hat{x}(k|k) + \Upsilon u(k-1)) - g$$

For

$$E\begin{bmatrix}\Delta\mathcal{U}(k)\\1\end{bmatrix} \leq 0$$

we write $E = \begin{bmatrix}W & -w\end{bmatrix}$ to obtain the matrix inequality in $\Delta\mathcal{U}$

$$W\Delta\mathcal{U}(k)) \leq w$$

# Constrained MPC



The constrained MPC problem may now be formulated like

$$\min_{\Delta\mathcal{U}(k)} V(k) = -\Delta\mathcal{U}(k)^T \mathcal{G} + \Delta\mathcal{U}(k)^T \mathcal{H} \Delta\mathcal{U}(k)$$

subject to

$$\begin{bmatrix} \mathbf{F} \\ \Gamma\Theta\Delta\mathcal{U}(k) \\ W \end{bmatrix} \Delta\mathcal{U}(k) \leq \begin{bmatrix} -\mathbf{F}_1 u(k-1) - f \\ -\Gamma(\Psi\hat{x}(k|k) + \Upsilon u(k-1)) - g \\ w \end{bmatrix}$$

This is a quadratic minimization problem with linear inequality constraints, which is convex and for which standard software exists (e.g. the Matlab plugin 'CVX').

# Infeasibility



When introducing constraints in a optimizations problem it may become infeasible, e.g. due to

- unexpected large disturbance
- differences between plant and model

This is unacceptable for an on-line controller. There are several approaches to address this problem

- ad hoc strategies
  - applying the same controller $\hat{u}(k|k)$ as last time
  - applying the controller $\hat{u}(k+1|k)$ or $\hat{u}(k+2|k)$ computed last time
- systematic strategies
  - Avoid 'hard' constraints (softening the constraints)
  - Actively manage the constraints and/or the horizons at each times

# Softening the Constraints



To softening the constraints of the MPC problem we introduce slack variables in the form of a vector $\epsilon$ and a scalar $\rho \geq 0$

$$\min_{\Delta\mathcal{U}(k),\epsilon} V(k) = -\Delta\mathcal{U}(k)^T\mathcal{G} + \Delta\mathcal{U}(k)^T\mathcal{H}\Delta\mathcal{U}(k) + \rho\,\|\epsilon\|$$

subject to

$$\begin{bmatrix} \mathbf{F} \\ \Gamma\Theta\Delta\mathcal{U}(k) \\ W \end{bmatrix} \Delta\mathcal{U}(k) \leq \begin{bmatrix} -\mathbf{F}_1 u(k-1) - f \\ -\Gamma(\Psi\hat{x}(k|k) + \Upsilon u(k-1)) - g \\ w \end{bmatrix} + \epsilon$$

$$\epsilon \geq 0$$

To softening the constraints of the MPC problem we introduce slack variables in the form of a vector $\epsilon$ and a scalar $\rho \geq 0$

$$\min_{\Delta\mathcal{U}(k),\epsilon} V(k) = -\Delta\mathcal{U}(k)^T\mathcal{G} + \Delta\mathcal{U}(k)^T\mathcal{H}\Delta\mathcal{U}(k) + \rho\,\|\epsilon\|$$

subject to

$$\begin{bmatrix} \mathbf{F} \\ \Gamma\Theta\Delta\mathcal{U}(k) \\ W \end{bmatrix} \Delta\mathcal{U}(k) \leq \begin{bmatrix} -\mathbf{F}_1 u(k-1) - f \\ -\Gamma(\Psi\hat{x}(k|k) + \Upsilon u(k-1)) - g \\ w \end{bmatrix} + \epsilon$$

$$\epsilon \geq 0$$

Note that

- unconstrained problem is recovered if $\rho = 0$
- original constrained problem is recovered when $\rho \to \infty$

To softening the constraints of the MPC problem we introduce slack variables in the form of a vector $\epsilon$ and a scalar $\rho \geq 0$

$$\min_{\Delta\mathcal{U}(k),\epsilon} V(k) = -\Delta\mathcal{U}(k)^T\mathcal{G} + \Delta\mathcal{U}(k)^T\mathcal{H}\Delta\mathcal{U}(k) + \rho\,\|\epsilon\|$$

subject to

$$\begin{bmatrix} \mathbf{F} \\ \Gamma\Theta\Delta\mathcal{U}(k) \\ W \end{bmatrix} \Delta\mathcal{U}(k) \leq \begin{bmatrix} -\mathbf{F}_1 u(k-1) - f \\ -\Gamma(\Psi\hat{x}(k|k) + \Upsilon u(k-1)) - g \\ w \end{bmatrix} + \epsilon$$

$$\epsilon \geq 0$$

Warning: This formulation can result in violations of the original constraints even when such are not necessary. However

If $\|\| = \|\|_1$ or $\|\| = \|\|_\infty$ and $\rho$ is large enough then constraint violations will only occur if there is no feasible solution to the original problem

For MPC problems it can be difficult to determining how large $\rho$ should be

# Stability



Due to the use of a receding horizon unstable behavior can occur when using MPC. Loosely speaking this is because

each optimization "does not care" about what happens beyond the prediction horizon

## Example

$$\min V(k) = \hat{x}(k+1|k)^T \begin{bmatrix} 1 & 2 \\ 2 & 6 \end{bmatrix} \hat{x}(k+1|k)$$

subject to

$$\hat{x}(k+1|k) = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \hat{x}(k|k) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \hat{u}(k|k)$$

Hence $H_p = 1$ and $R = 0$.

# Guaranteeing Stability



We will look at two ways to guaranty stability

- terminal constraints
- infinite horizon

# Terminal Constraints and Stability

## Theorem

*Assume that a MPC scheme is derived from*

$$\min V(k) = \sum_{i=1}^{H_p} l(\hat{x}(k+i|k), \hat{u}(k+i-1|k))$$

*subject to*

$$\begin{aligned} \hat{x}(k+1|k) &= f(\hat{x}(k|k), \hat{u}(k|k)) \\ \hat{x}(k+i|k) &\in X \\ \hat{u}(k+i|k) &\in U \\ \hat{x}(k+H_p|k) &= 0 \end{aligned}$$

*where $l$ is a decrescent positive definite function and $f(0,0) = 0$. Then the equilibrium point $(x,u) = 0$ is stable, provided that the optimization problem is feasible.*

Infinite Horizon MPC is stable basically due to the ’principle of optimality’

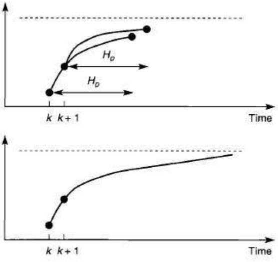


**Figure 6.1** Finite and infinite horizons (no disturbances, perfect model).

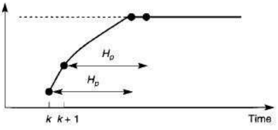


**Figure 6.2** Finite horizon and deadbeat response.

# Modifications in the Infinite Horizon Case



To consider infinite horizon we modify our previous setup by

- setting $Q(i) = Q \geq 0$ and $R(i) = R \geq 0$
- also penalizing control levels $\hat{u}(k+i-1|k)$ with respect to some weight $S > 0$
- wanting to drive the output to zero ($r = 0$)

That is, we optimize with respect to the cost

$$V(k) = \sum_{i=1}^{\infty} \|\hat{z}(k+i|k)\|_Q^2 + \|\Delta\hat{u}(k+i-1|k)\|_R^2 + \|\hat{u}(k+i-1|k)\|_S^2$$

Consider

$$V(k) = \sum_{i=1}^{\infty} \|\hat{z}(k+i|k)\|_Q^2 + \|\Delta\hat{u}(k+i-1|k)\|_R^2 + \|\hat{u}(k+i-1|k)\|_S^2$$

and let $V^0(k)$ denote the optimal value of $V(k)$, hence

$$V^0(k) = \sum_{i=1}^{\infty} \left\|\hat{z}^0(k+i|k)\right\|_Q^2 + \sum_{i=1}^{H_u} \left\|\Delta\hat{u}^0(k+i-1|k)\right\|_R^2 + \left\|\hat{u}^0(k+i-1|k)\right\|_S^2$$

Moreover, define $V(k+1)$ by

$$V(k+1) = V^0(k) - \left\|\hat{z}^0(k+1|k)\right\|_Q^2 - \left\|\Delta\hat{u}^0(k|k)\right\|_R^2 - \left\|\hat{u}^0(k|k)\right\|_S^2$$

then

$$\begin{aligned} V^0(k+1) \leq& V(k+1) \\ =& V^0(k) - \left\|\hat{z}^0(k+1|k)\right\|_Q^2 - \left\|\Delta\hat{u}^0(k|k)\right\|_R^2 - \left\|\hat{u}^0(k|k)\right\|_S^2 \\ <& V^0(k) \end{aligned}$$

with $<$ due to $S$

# Infinite Horizon MPC is Stable



Using that

$$V^0(k+1) < V^0(k) \quad \text{and} \quad S > 0$$

we conclude that $\|x(k)\|$ decreases whenever

- the plant is stable, or
- the plant is unstable, $H_u >$ number of unstable modes, and $(A, B)$ is stabilizable

Hence infinite horizon MPC is stable

# Infinite Horizon and Constraints



In the infinite horizon case constraints are only enforced during a finite time period

$$\alpha_i \leq \hat{z}_i(k+i) \leq \beta_i \quad i = C_w, C_w + 1, \ldots, C_p$$

where

- $C_w$ should be chosen large enough such that the problem is feasible at time $k$
- $C_p$ should be chosen large enough such that if the problem is feasible over the finite horizon up time $k + C_p$, then it will remain feasible

Such values always exists (for unstable plants these can depend on $x(k)$)

# Reformulation of the cost (infinite horizon, stable plant)

As in the finite horizon case we wish to write the cost

$$V(k) = \sum_{i=H_u+1}^{\infty} \|\hat{z}(k+i-1|k)\|_Q^2 + \sum_{i=1}^{H_u} \Big( \|\hat{z}(k+i-1|k)\|_Q^2 + \|\Delta\hat{u}(k+i-1|k)\|_R^2 + \|\hat{u}(k+i-1|k)\|_S^2 \Big)$$

as a quadratic function in $\Delta\hat{u}$.

# Reformulation of the cost (infinite horizon, stable plant)



As in the finite horizon case we wish to write the cost

$$V(k) = \sum_{i=H_u+1}^{\infty} \|\hat{z}(k+i-1|k)\|_Q^2 + \sum_{i=1}^{H_u} \Big( \|\hat{z}(k+i-1|k)\|_Q^2 + \|\Delta\hat{u}(k+i-1|k)\|_R^2 + \|\hat{u}(k+i-1|k)\|_S^2 \Big)$$

as a quadratic function in $\Delta\hat{u}$. To do so we need only to focus on the first sum above: First expand

$$\begin{aligned}
\hat{z}(k+H_u|k) &= C_z\hat{x}(k+H_u|k) \\
\hat{z}(k+H_u+1|k) &= C_zA\hat{x}(k+H_u|k) \\
&\vdots \\
\hat{z}(k+H_u+j|k) &= C_zA^j\hat{x}(k+H_u|k)
\end{aligned}$$

then replace

$$\sum_{i=H_u+1}^{\infty} \|\hat{z}(k+i-1|k)\|_Q^2 = \hat{x}(k+H_u|k)^T\bar{Q}\hat{x}(k+H_u|k)$$

where

$$\bar{Q} = \sum_{i=0}^{\infty}(A^T)^iC_z^TQC_zA^i \qquad \text{convergent since } A \text{ stable}$$

# Reformulation of the cost (infinite horizon, stable plant)



As in the finite horizon case we wish to write the cost

$$V(k) = \sum_{i=H_u+1}^{\infty} \|\hat{z}(k+i-1|k)\|_Q^2 + \sum_{i=1}^{H_u} \Big( \|\hat{z}(k+i-1|k)\|_Q^2 + \|\Delta\hat{u}(k+i-1|k)\|_R^2 + \|\hat{u}(k+i-1|k)\|_S^2 \Big)$$

as a quadratic function in $\Delta\hat{u}$. Rewriting we get

$$V(k) = \hat{x}(k+H_u|k)^T \bar{Q} \hat{x}(k+H_u|k) + \sum_{i=1}^{H_u} \Big( \|\hat{z}(k+i-1|k)\|_Q^2 + \|\Delta\hat{u}(k+i-1|k)\|_R^2 + \|\hat{u}(k+i-1|k)\|_S^2 \Big)$$

where

$$\bar{Q} = \sum_{i=0}^{\infty} (A^T)^i C_z^T Q C_z A^i \qquad \text{convergent since } A \text{ stable}$$

can be found as the solution to the discrete Lyapunov (or Stein) equation

$$A^T \bar{Q} A - \bar{Q} + C_z^T Q C_z = 0$$

# Reformulation of the cost (infinite horizon, stable plant)



As in the finite horizon case we wish to write the cost

$$V(k) = \sum_{i=H_u+1}^{\infty} \|\hat{z}(k+i-1|k)\|_Q^2 + \sum_{i=1}^{H_u} \Big( \|\hat{z}(k+i-1|k)\|_Q^2 + \|\Delta\hat{u}(k+i-1|k)\|_R^2 + \|\hat{u}(k+i-1|k)\|_S^2 \Big)$$

as a quadratic function in $\Delta\hat{u}$. Rewriting we get

$$V(k) = \hat{x}(k+H_u|k)^T \bar{Q} \hat{x}(k+H_u|k) + \sum_{i=1}^{H_u} \Big( \|\hat{z}(k+i-1|k)\|_Q^2 + \|\Delta\hat{u}(k+i-1|k)\|_R^2 + \|\hat{u}(k+i-1|k)\|_S^2 \Big)$$

Exactly as in the finite horizon case we may now expand all variables in terms of $\Delta\hat{u}$ to obtain an equivalent quadratic formulation (Exercise).

# Reformulation of the cost (infinite horizon, unstable plant)



For the ’unstable plant’ case we need to make sure that

- the unstable modes are zero at the end of the control horizon
- the infinite sum converges

# Reformulation of the cost (infinite horizon, unstable plant)

For an unstable plant we can still write the cost as

$$
\begin{aligned}
V(k) = &\hat{x}(k+H_u|k)^T \bar{Q} \hat{x}(k+H_u|k) + \sum_{i=1}^{H_u} \Big( \|\hat{z}(k+i-1|k)\|_Q^2 + \|\Delta\hat{u}(k+i-1|k)\|_R^2 \\
&\qquad + \|\hat{u}(k+i-1|k)\|_S^2 \Big)
\end{aligned}
$$

but now $\bar{Q} = \tilde{W}_S^T \Pi \tilde{W}_S$ where $\Pi$ can be found as the solution to the discrete Lyapunov (or Stein) equation

$$
W_S^T C_z^T Q C_z W_S - \Pi + J_s^T \Pi J_s = 0
$$

and $\tilde{W}_\bullet$, $W_\bullet$ and $J_\bullet$ are determined by a Jordan decomposition of $A$ into a stable and unstable part

$$
A = WJW^{-1} = \begin{bmatrix} W_u & W_s \end{bmatrix} \begin{bmatrix} J_u & 0 \\ 0 & J_s \end{bmatrix} \begin{bmatrix} \tilde{W}_u \\ \tilde{W}_s \end{bmatrix}
$$

Moreover, we also need to impose a terminal constraint on the unstable modes:

$$
\tilde{W}_u x(k+H_p) = 0
$$

# The Fake Algebraic Riccati Equation



In the last part we (briefly) mention another method to obtain stability

- the fake algebraic Riccati equation, used in the unconstrained finite horizon case

# The Fake Algebraic Riccati Equation

Based on the (algebraic) Riccati equation for the infinite horizon LQ problem a stabilizing constant state-feedback control law can be designed for a unconstrained finite horizon MPC problem. To do so we introduce the augmented quantities

$$\zeta(k) = \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix}, \quad \tilde{A} = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} B \\ I \end{bmatrix}$$

# The Fake Algebraic Riccati Equation

Based on the (algebraic) Riccati equation for the infinite horizon LQ problem a stabilizing constant state-feedback control law can be designed for a unconstrained finite horizon MPC problem. To do so we introduce the augmented quantities

$$\zeta(k) = \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix}, \quad \tilde{A} = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} B \\ I \end{bmatrix}$$

Note that

$$x(k+1) = Ax(k) + Bu(k) \quad \text{is equivalent to} \quad \zeta(k+1) = \tilde{A}\zeta(k) + \tilde{B}\Delta u(k)$$

# The Fake Algebraic Riccati Equation

Based on the (algebraic) Riccati equation for the infinite horizon LQ problem a stabilizing constant state-feedback control law can be designed for a unconstrained finite horizon MPC problem. To do so we introduce the augmented quantities

$$\zeta(k) = \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix}, \quad \tilde{A} = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} B \\ I \end{bmatrix}$$

and set

$$\|z(k)\|_Q = \|\zeta(k)\|_{\tilde{Q}} \qquad \tilde{Q} = \begin{bmatrix} C_z^T Q C_z & 0 \\ 0 & 0 \end{bmatrix}$$

to obtain the cost

$$V(k) = \left\|\hat{\zeta}(k+H|k)\right\|_P^2 + \sum_{i=1}^{H-1} \left\|\hat{\zeta}(k+i|k)\right\|_{\tilde{Q}}^2 + \|\Delta\hat{u}(k+i|k)\|_R^2$$

where we also have added a final terminal with $P \geq 0$

# The Fake Algebraic Riccati Equation



A stabilizing constant state-feedback control gain for the unconstrained finite horizon MPC problem

$$\min_{\Delta\mathcal{U}(k)} V(k) = \left\|\hat{\zeta}(k+H|k)\right\|_P^2 + \sum_{i=1}^{H-1} \left\|\hat{\zeta}(k+i|k)\right\|_{\tilde{Q}}^2 + \|\Delta\hat{u}(k+i|k)\|_R^2$$

subject to the dynamics

$$\hat{\zeta}(k+i+1|k) \quad = \quad \tilde{A}\hat{\zeta}(k+i|k) + \tilde{B}\Delta\hat{u}(k+i|k)$$

where

$$\zeta(k) = \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix}, \quad \tilde{A} = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} B \\ I \end{bmatrix} \quad \tilde{Q} = \begin{bmatrix} C_z^T Q C_z & 0 \\ 0 & 0 \end{bmatrix}$$

is

$$\tilde{K} = (\tilde{B}^T P \tilde{B} + R)^{-1} \tilde{B}^T P \tilde{A}$$

whenever $(\tilde{A}, \tilde{B})$ is stabilizable, $(\tilde{A}, \tilde{Q}^{1/2})$ is detectable, and

$$P = \tilde{A}^T P \tilde{A} - \tilde{A}^T P \tilde{B}(\tilde{B}^T P \tilde{B} + R)^{-1} \tilde{B}^T P \tilde{A} + \tilde{Q}, \quad P \geq 0, \tilde{Q} \geq 0$$