Model Predictive Control

7. Reference Tracking and Disturbance Rejection

Jun.-Prof. Dr.-Ing. Daniel Görges
Juniorprofessur für Elektromobilität
Technische Universität Kaiserslautern

Reference Tracking

Reference Tracking based on Target Calculation

- **Discrete-Time Linear Time-Invariant (LTI) System**

$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k)$	state equation	(7.1)
$\boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k)$	measured output equation	(7.2)
$\boldsymbol{y}_r(k) = \boldsymbol{C}_r\boldsymbol{x}(k)$	controlled output equation	(7.3)

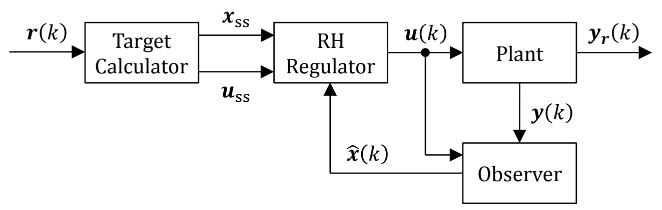
- **Symbols**

$\boldsymbol{x}(k) \in \mathbb{R}^n$ state vector	$\boldsymbol{u}(k) \in \mathbb{U} \subseteq \mathbb{R}^m$ input vector
$\boldsymbol{y}(k) \in \mathbb{Y} \subseteq \mathbb{R}^p$ measured output vector	$\boldsymbol{y}_r(k) \in \mathbb{R}^{p_r}$ controlled output vector
$\boldsymbol{A} \in \mathbb{R}^{n \times n}$ system matrix	$\boldsymbol{B} \in \mathbb{R}^{n \times m}$ input matrix
$\boldsymbol{C} \in \mathbb{R}^{p \times n}$ measured output matrix	$\boldsymbol{C}_r \in \mathbb{R}^{p_r \times n}$ controlled output matrix

- **Remarks**
  - The measured output $\boldsymbol{y}(k)$ is used for the observer
  - The controlled output $\boldsymbol{y}_r(k)$ is considered for reference tracking

Reference Tracking based on Target Calculation

- **Structure**



- **Objective**
  - Control the discrete-time LTI system such that $\boldsymbol{y}_r(k) \to \boldsymbol{r}(k)$ if $\boldsymbol{r}(k) \to$ const. as $k \to \infty$
- **Approach**
  - The target calculator computes the target state $\boldsymbol{x}_{ss}$ and the target input $\boldsymbol{u}_{ss}$
  - The RH regulator controls the discrete-time LTI system to the target pair $(\boldsymbol{x}_{ss}, \boldsymbol{u}_{ss})$

Reference Tracking based on Target Calculation

- **Approach**
  - Consider that the discrete-time LTI system $(7.1)/(7.3)$ is in the steady target state $\boldsymbol{x}_{ss}$, i.e.

$$\begin{aligned} \boldsymbol{x}_{ss} &= \boldsymbol{A}\boldsymbol{x}_{ss} + \boldsymbol{B}\boldsymbol{u}_{ss} \quad &\Leftrightarrow \quad (\boldsymbol{I}_{n\times n} - \boldsymbol{A})\boldsymbol{x}_{ss} - \boldsymbol{B}\boldsymbol{u}_{ss} &= \boldsymbol{0}_{n\times 1} \\ \boldsymbol{y}_{rss} &= \boldsymbol{C}_r\boldsymbol{x}_{ss} = \boldsymbol{r} \quad &\Leftrightarrow \quad \boldsymbol{C}_r\boldsymbol{x}_{ss} &= \boldsymbol{r} \end{aligned}$$

  - Rewrite the equations in matrix form, i.e.

$$\begin{pmatrix} \boldsymbol{I}_{n\times n} - \boldsymbol{A} & -\boldsymbol{B} \\ \boldsymbol{C}_r & \boldsymbol{0}_{p_r\times m} \end{pmatrix} \begin{pmatrix} \boldsymbol{x}_{ss} \\ \boldsymbol{u}_{ss} \end{pmatrix} = \begin{pmatrix} \boldsymbol{0}_{n\times 1} \\ \boldsymbol{r} \end{pmatrix} \tag{7.4}$$

  - The steady target pair $(\boldsymbol{x}_{ss}, \boldsymbol{u}_{ss})$ can be then calculated from (7.4) provided that a solution exists
  - Introduce the state deviation and input deviation

$$\tilde{\boldsymbol{x}}(k) = \hat{\boldsymbol{x}}(k) - \boldsymbol{x}_{ss}, \quad \tilde{\boldsymbol{u}}(k) = \boldsymbol{u}(k) - \boldsymbol{u}_{ss}$$

  - This leads to the discrete-time LTI state equation

$$\tilde{\boldsymbol{x}}(k+1) = \hat{\boldsymbol{x}}(k+1) - \boldsymbol{x}_{ss} = \boldsymbol{A}\hat{\boldsymbol{x}}(k) + \boldsymbol{B}\boldsymbol{u}(k) - (\boldsymbol{A}\boldsymbol{x}_{ss} + \boldsymbol{B}\boldsymbol{u}_{ss}) = \boldsymbol{A}\tilde{\boldsymbol{x}}(k) + \boldsymbol{B}\tilde{\boldsymbol{u}}(k) \tag{7.5}$$

Reference Tracking based on Target Calculation

- **Approach**

  the discrete-time quadratic cost function

$$\tilde{V}_N\left(\tilde{\boldsymbol{x}}(k), \tilde{\boldsymbol{U}}(k)\right) = \tilde{\boldsymbol{x}}^T(k+N)\boldsymbol{P}\tilde{\boldsymbol{x}}(k+N) + \sum_{i=0}^{N-1} \tilde{\boldsymbol{x}}^T(k+i)\boldsymbol{Q}\tilde{\boldsymbol{x}}(k+i) + \tilde{\boldsymbol{u}}^T(k+i)\boldsymbol{R}\tilde{\boldsymbol{u}}(k+i) \quad (7.6)$$

  the state and input constraints

$$\begin{aligned} &\boldsymbol{C}(\tilde{\boldsymbol{x}}(k+i) + \boldsymbol{x}_{ss}) \in \mathbb{Y}, i = 1,2,\dots,N \\ &\tilde{\boldsymbol{u}}(k+i) + \boldsymbol{u}_{ss} \in \mathbb{U}, \quad i = 0,1,\dots,N-1 \end{aligned} \quad (7.7)$$

  – Rewrite Problem 4.1/5.1 w.r.t. the state equation (7.5), cost function (7.6) and constraints (7.7), i.e.

$$\min_{\tilde{\boldsymbol{U}}(k)} \tilde{V}_N(\tilde{\boldsymbol{x}}(k), \tilde{\boldsymbol{U}}(k))$$

$$\text{subject to} \begin{cases} \tilde{\boldsymbol{x}}(k+i+1) = \boldsymbol{A}\tilde{\boldsymbol{x}}(k+i) + \boldsymbol{B}\tilde{\boldsymbol{u}}(k+i), i = 0,1,\dots,N-1 \\ \boldsymbol{C}(\tilde{\boldsymbol{x}}(k+i) + \boldsymbol{x}_{ss}) \in \mathbb{Y}, \quad i = 1,2,\dots,N \\ \tilde{\boldsymbol{u}}(k+i) + \boldsymbol{u}_{ss} \in \mathbb{U}, \quad i = 0,1,\dots,N-1 \end{cases} \quad (7.8)$$

Reference Tracking based on Target Calculation

- **Remark on the Optimization Problem**
  – Problem (7.8) can be formulated as a quadratic program using the methods from Chapter 4 and 5
  – Problem (7.8) relates to a regulation problem, i.e. $\tilde{\boldsymbol{x}}(k) \to \boldsymbol{0}, \tilde{\boldsymbol{u}}(k) \to \boldsymbol{0}$ as $k \to \infty$
  – The reference tracking problem is, however, simultaneously addressed since

$$\tilde{\boldsymbol{x}}(k) \to \boldsymbol{0}, \tilde{\boldsymbol{u}}(k) \to \boldsymbol{0} \Rightarrow \tilde{\boldsymbol{x}}(k) \to \boldsymbol{x}_{ss}, \boldsymbol{u}(k) \to \boldsymbol{u}_{ss} \Rightarrow \boldsymbol{y}_r(k) \to \boldsymbol{r}$$

- **Remark on the Target Calculator**
  – Generally it is not possible to control the state $\boldsymbol{x}(k)$ to an arbitrary target state $\boldsymbol{x}_{ss}$
  – E.g. it is not possible to maintain a constant position and a constant velocity of a car simultaneously
  – A sufficient condition for the existence of a solution of (7.4) for any reference input $\boldsymbol{r}$ is that

$$\begin{pmatrix} \boldsymbol{I}_{n\times n} - \boldsymbol{A} & -\boldsymbol{B} \\ \boldsymbol{C}_r & \boldsymbol{0}_{p_r\times m} \end{pmatrix} \in \mathbb{R}^{(n+p_r)\times(n+m)} \text{ has full rank } n + p_r$$

  – This implies that $\boldsymbol{C}_r$ must have full rank and number of controlled outputs $p_r \leq$ number of inputs m
  – The solution may not be unique

Reference Tracking based on Target Calculation

- **Remark on the Constrained Case**
  - For the constrained case target pair $(\boldsymbol{x}_{\text{ss}}, \boldsymbol{u}_{\text{ss}})$ must fulfill output constraint $\mathbb{Y}$ and input constraint $\mathbb{U}$
  - For this purpose the target calculator based on (7.4) must be modified to

$$\min_{\boldsymbol{x}_{\text{ss}},\boldsymbol{u}_{\text{ss}}} \frac{1}{2}\left((\boldsymbol{C}_r\boldsymbol{x}_{\text{ss}} - \boldsymbol{r})^T\boldsymbol{Q}_{\text{ss}}(\boldsymbol{C}_r\boldsymbol{x}_{\text{ss}} - \boldsymbol{r}) + (\boldsymbol{u}_{\text{ss}} - \boldsymbol{u}_{\text{ss}}^{\text{unc}})^T\boldsymbol{R}_{\text{ss}}(\boldsymbol{u}_{\text{ss}} - \boldsymbol{u}_{\text{ss}}^{\text{unc}})\right)$$

$$\text{subject to} \begin{cases} \begin{pmatrix} \boldsymbol{I}_{n\times n} - \boldsymbol{A} & -\boldsymbol{B} \\ \boldsymbol{C}_r & \boldsymbol{0}_{p_r\times m} \end{pmatrix} \begin{pmatrix} \boldsymbol{x}_{\text{ss}} \\ \boldsymbol{u}_{\text{ss}} \end{pmatrix} = \begin{pmatrix} \boldsymbol{0}_{n\times 1} \\ \boldsymbol{r} \end{pmatrix} \\ \boldsymbol{C}\boldsymbol{x}_{\text{ss}} \in \mathbb{Y} \\ \boldsymbol{u}_{\text{ss}} \in \mathbb{U} \end{cases} \tag{7.9}$$

  where $\boldsymbol{u}_{\text{ss}}^{\text{unc}}$ is the target input resulting from (7.4) for the unconstrained case and $\boldsymbol{Q}_{\text{ss}} = \boldsymbol{Q}_{\text{ss}}^T \succcurlyeq \boldsymbol{0}$ and $\boldsymbol{R}_{\text{ss}} = \boldsymbol{R}_{\text{ss}}^T \succ \boldsymbol{0}$ are weighting matrices
  - Problem (7.9) can be formulated as a quadratic program
  - Feasibility of Problem (7.9) is discussed in [RM09, Section 1.5.1]

Reference Tracking based on Target Calculation

- **Remark on Receding Horizon Control**
  1. Estimate the current state $\hat{\boldsymbol{x}}(k)$
  2. Solve Problem (7.9) for the given reference input $\boldsymbol{r}(k)$ to determine the target pair $(\boldsymbol{x}_{\text{ss}}, \boldsymbol{u}_{\text{ss}})$
  3. Solve Problem (7.8) for $\tilde{\boldsymbol{x}}(k) = \hat{\boldsymbol{x}}(k) - \boldsymbol{x}_{\text{ss}}$ to determine the optimal input sequence $\widetilde{\boldsymbol{U}}^*(k)$
  4. Compute first element of the optimal input sequence $\tilde{\boldsymbol{u}}^*(k) = (\boldsymbol{I}_{m\times m} \quad \boldsymbol{0}_{m\times m} \quad \cdots \quad \boldsymbol{0}_{m\times m})\widetilde{\boldsymbol{U}}^*(k)$
  5. Implement the optimal input $\boldsymbol{u}^*(k) = \tilde{\boldsymbol{u}}^*(k) + \boldsymbol{u}_{\text{ss}}$
  6. Increment the time instant $k \coloneqq k + 1$ and go to 1.
- **Further Remarks**
  - The extension for state constraints and controlled output constraints is straightforward
  - The structure on Slide 7-3 is essentially equivalent to the state-command structure on Slide 2-49ff
  - Disturbances and uncertainties in $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}_r$ lead to a steady-state error or offset
  - More details and references are given in [RM09, Section 1.5.1], [BBM15, Section 13.6], and [MR93]

Reference Tracking based on the Delta Input Formulation

- **Discrete-Time Linear Time-Invariant (LTI) System**

  $\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k)$ state equation (7.10)

  $\boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k)$ output equation (7.11)

- **Symbols**

  $\boldsymbol{x}(k) \in \mathbb{R}^n$ state vector $\quad$ $\boldsymbol{u}(k) \in \mathbb{U} \subseteq \mathbb{R}^m$ input vector

  $\boldsymbol{y}(k) \in \mathbb{Y} \subseteq \mathbb{R}^p$ output vector

  $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ system matrix $\quad$ $\boldsymbol{B} \in \mathbb{R}^{n \times m}$ input matrix

  $\boldsymbol{C} \in \mathbb{R}^{p \times n}$ output matrix

- **Objective**
  - Control the discrete-time LTI system such that $\boldsymbol{y}(k) \to \boldsymbol{r}(k)$

Reference Tracking based on the Delta Input Formulation

- **Approach**
  - Introduce the input increment

    $\Delta\boldsymbol{u}(k) = \boldsymbol{u}(k) - \boldsymbol{u}(k-1)$
  - Introduce the discrete-time quadratic cost function

$$V_N\big(\boldsymbol{y}(k), \Delta\boldsymbol{U}(k), \boldsymbol{r}(k)\big) = \big(\boldsymbol{y}(k+N) - \boldsymbol{r}(k)\big)^T \boldsymbol{P}\big(\boldsymbol{y}(k+N) - \boldsymbol{r}(k)\big) + \sum_{i=0}^{N-1} \big(\boldsymbol{y}(k+i) - \boldsymbol{r}(k)\big)^T \boldsymbol{Q}\big(\boldsymbol{y}(k+i) - \boldsymbol{r}(k)\big) + \Delta\boldsymbol{u}^T(k+i)\boldsymbol{R}\Delta\boldsymbol{u}(k+i) \qquad (7.12)$$

    with the reference input $\boldsymbol{r}(k)$ and the weighting matrices $\boldsymbol{P} = \boldsymbol{P}^T \succ \boldsymbol{0}$, $\boldsymbol{Q} = \boldsymbol{Q}^T \succ \boldsymbol{0}$, $\boldsymbol{R} = \boldsymbol{R}^T \succ \boldsymbol{0}$
  - Note that $\boldsymbol{r}(k)$ is held over the whole prediction horizon
  - Note that $\boldsymbol{y}(k+i) - \boldsymbol{r}(k)$ is the control error

Reference Tracking

Reference Tracking based on the Delta Input Formulation

- **Approach**
  - Reformulate Problem 4.1/5.1 w.r.t. the system (7.10)/(7.11), cost function (7.12) and $\Delta\boldsymbol{u}(k)$, i.e.

$$\min_{\Delta\boldsymbol{U}(k)} V_N(\boldsymbol{y}(k), \Delta\boldsymbol{U}(k), \boldsymbol{r}(k))$$
$$\text{subject to} \begin{cases} \boldsymbol{x}(k+i+1) = \boldsymbol{A}\boldsymbol{x}(k+i) + \boldsymbol{B}\boldsymbol{u}(k+i), & i = 0,1,\dots,N-1 \\ \boldsymbol{y}(k+i) = \boldsymbol{C}\boldsymbol{x}(k+i), & i = 0,1,\dots,N \\ \boldsymbol{y}(k+i) \in \mathbb{Y}, & i = 1,2,\dots,N \\ \boldsymbol{u}(k+i) \in \mathbb{U}, & i = 0,1,\dots,N-1 \\ \boldsymbol{u}(k+i) = \boldsymbol{u}(k+i-1) + \Delta\boldsymbol{u}(k+i), & i = 0,1,\dots,N-1 \end{cases} \tag{7.13}$$

  - Problem (7.13) can be formulated as a quadratic program using the methods from Chapter 4 and 5
  - Problem (7.13) relates to a regulation problem, i.e. $\boldsymbol{y}(k) - \boldsymbol{r}(k) \to \boldsymbol{0}, \Delta\boldsymbol{u}(k) \to \boldsymbol{0}$ in steady state
  - Note that the reference tracking problem has been transformed into a regulation problem using $\boldsymbol{y}(k+i) - \boldsymbol{r}(k)$ instead of $\boldsymbol{x}(k+i)$ and $\Delta\boldsymbol{u}(k)$ instead of $\boldsymbol{u}(k)$ (generally $\boldsymbol{u}(k) \nrightarrow \boldsymbol{0}$ in steady state)

Reference Tracking

Reference Tracking based on the Delta Input Formulation

- **Approach**
  - The prediction model (4.4), the cost function (4.5), and the constraint model (5.1) must be reformulated to obtain a quadratic program
  - For this purpose the discrete-time LTI system (7.10)/(7.11) is augmented w.r.t. $\Delta\boldsymbol{u}(k)$, i.e.

$$\underbrace{\begin{pmatrix} \boldsymbol{x}(k+1) \\ \boldsymbol{u}(k) \end{pmatrix}}_{\tilde{\boldsymbol{x}}(k+1)} = \underbrace{\begin{pmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{0} & \boldsymbol{I} \end{pmatrix}}_{\tilde{\boldsymbol{A}}} \underbrace{\begin{pmatrix} \boldsymbol{x}(k) \\ \boldsymbol{u}(k-1) \end{pmatrix}}_{\tilde{\boldsymbol{x}}(k)} + \underbrace{\begin{pmatrix} \boldsymbol{B} \\ \boldsymbol{I} \end{pmatrix}}_{\tilde{\boldsymbol{B}}} \Delta\boldsymbol{u}(k) \tag{7.14}$$

$$\underbrace{\boldsymbol{y}(k)}_{\boldsymbol{y}(k)} = \underbrace{\begin{pmatrix} \boldsymbol{C} & \boldsymbol{0} \end{pmatrix}}_{\tilde{\boldsymbol{C}}} \underbrace{\begin{pmatrix} \boldsymbol{x}(k) \\ \boldsymbol{u}(k-1) \end{pmatrix}}_{\tilde{\boldsymbol{x}}(k)} \tag{7.15}$$

  - The prediction model (4.4) can then be reformulated w.r.t. the augmented system (7.14)/(7.15) as

$$\tilde{\boldsymbol{X}}(k) = \tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{x}}(k) + \tilde{\boldsymbol{\Gamma}}\Delta\boldsymbol{U}(k) \tag{7.16}$$

where $\tilde{\boldsymbol{X}}(k)$, $\tilde{\boldsymbol{\Phi}}$, $\tilde{\boldsymbol{\Gamma}}$, $\Delta\boldsymbol{U}(k) = \begin{pmatrix} \Delta\boldsymbol{u}^T(k) & \Delta\boldsymbol{u}^T(k+1) & \cdots & \Delta\boldsymbol{u}^T(k+N-1) \end{pmatrix}^T$ are defined as in (4.4)

Reference Tracking

Reference Tracking based on the Delta Input Formulation

- **Approach**
  - The cost function (4.5) can then be reformulated w.r.t. the augmented system (7.14)/(7.15), the prediction model (7.16), and the cost function (7.12) as

$$
\begin{aligned}
&V_N(\tilde{\boldsymbol{x}}(k),\Delta\boldsymbol{U}(k),\boldsymbol{R}(k)) = (\boldsymbol{y}(k)-\boldsymbol{r}(k))^T\boldsymbol{Q}(\boldsymbol{y}(k)-\boldsymbol{r}(k)) + (\boldsymbol{Y}(k)-\boldsymbol{R}(k))^T\boldsymbol{\Omega}(\boldsymbol{Y}(k)-\boldsymbol{R}(k)) + \Delta\boldsymbol{U}^T(k)\boldsymbol{\Psi}\Delta\boldsymbol{U}(k)\\
&= \left(\tilde{\boldsymbol{C}}\tilde{\boldsymbol{x}}(k)-\boldsymbol{r}(k)\right)^T\boldsymbol{Q}\left(\tilde{\boldsymbol{C}}\tilde{\boldsymbol{x}}(k)-\boldsymbol{r}(k)\right) + \left(\tilde{\boldsymbol{\mathcal{C}}}\tilde{\boldsymbol{X}}(k)-\boldsymbol{R}(k)\right)^T\boldsymbol{\Omega}\left(\tilde{\boldsymbol{\mathcal{C}}}\tilde{\boldsymbol{X}}(k)-\boldsymbol{R}(k)\right) + \Delta\boldsymbol{U}^T(k)\boldsymbol{\Psi}\Delta\boldsymbol{U}(k)\\
&= \left(\tilde{\boldsymbol{C}}\tilde{\boldsymbol{x}}(k)-\boldsymbol{r}(k)\right)^T\boldsymbol{Q}\left(\tilde{\boldsymbol{C}}\tilde{\boldsymbol{x}}(k)-\boldsymbol{r}(k)\right) + \left(\tilde{\boldsymbol{\mathcal{C}}}\left(\tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{x}}(k)+\tilde{\boldsymbol{\Gamma}}\Delta\boldsymbol{U}(k)\right)-\boldsymbol{R}(k)\right)^T\boldsymbol{\Omega}\left(\tilde{\boldsymbol{\mathcal{C}}}\left(\tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{x}}(k)+\tilde{\boldsymbol{\Gamma}}\Delta\boldsymbol{U}(k)\right)-\boldsymbol{R}(k)\right) + \Delta\boldsymbol{U}^T(k)\boldsymbol{\Psi}\Delta\boldsymbol{U}(k)\\
&= \cdots\\
&= \Delta\boldsymbol{U}^T(k)(\boldsymbol{\Psi}+\tilde{\boldsymbol{\Gamma}}^T\tilde{\boldsymbol{\mathcal{C}}}^T\boldsymbol{\Omega}\tilde{\boldsymbol{\mathcal{C}}}\tilde{\boldsymbol{\Gamma}})\Delta\boldsymbol{U}(k) + 2\Delta\boldsymbol{U}^T(k)\tilde{\boldsymbol{\Gamma}}^T\tilde{\boldsymbol{\mathcal{C}}}^T\boldsymbol{\Omega}\tilde{\boldsymbol{\mathcal{C}}}\tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{x}}(k) - 2\Delta\boldsymbol{U}^T(k)\tilde{\boldsymbol{\Gamma}}^T\tilde{\boldsymbol{\mathcal{C}}}^T\boldsymbol{\Omega}\boldsymbol{R}(k) + f(\tilde{\boldsymbol{x}}(k),\boldsymbol{R}(k))\\
&= \frac{1}{2}\Delta\boldsymbol{U}^T(k)\underbrace{2(\boldsymbol{\Psi}+\tilde{\boldsymbol{\Gamma}}^T\tilde{\boldsymbol{\mathcal{C}}}^T\boldsymbol{\Omega}\tilde{\boldsymbol{\mathcal{C}}}\tilde{\boldsymbol{\Gamma}})}_{\tilde{\boldsymbol{H}}}\Delta\boldsymbol{U}(k) + \Delta\boldsymbol{U}^T(k)\left(\underbrace{2\tilde{\boldsymbol{\Gamma}}^T\tilde{\boldsymbol{\mathcal{C}}}^T\boldsymbol{\Omega}\tilde{\boldsymbol{\mathcal{C}}}\tilde{\boldsymbol{\Phi}}}_{\tilde{\boldsymbol{F}}}\tilde{\boldsymbol{x}}(k) \underbrace{-2\tilde{\boldsymbol{\Gamma}}^T\tilde{\boldsymbol{\mathcal{C}}}^T\boldsymbol{\Omega}}_{\tilde{\boldsymbol{F}}_{\boldsymbol{R}}}\boldsymbol{R}(k)\right) + f(\tilde{\boldsymbol{x}}(k),\boldsymbol{R}(k))
\end{aligned}
$$

where $\tilde{\boldsymbol{\mathcal{C}}} = \mathrm{diag}(\tilde{\boldsymbol{C}},\ldots,\tilde{\boldsymbol{C}})$, $\boldsymbol{\Omega}$, $\boldsymbol{\Psi}$, $\boldsymbol{Y}(k) = (\boldsymbol{y}^T(k+1) \quad \boldsymbol{y}^T(k+2) \quad \cdots \quad \boldsymbol{y}^T(k+N))^T$ and $\boldsymbol{R}(k) = (\boldsymbol{r}^T(k) \quad \boldsymbol{r}^T(k) \quad \cdots \quad \boldsymbol{r}^T(k))^T$ are defined as in (4.5)

Reference Tracking

Reference Tracking based on the Delta Input Formulation

- **Approach**
  - The constraints in standard form (cf. Slide 5-10) can be reformulated w.r.t. Problem (7.13) as

$$
\begin{aligned}
\boldsymbol{M}\boldsymbol{y}(k+i) + \boldsymbol{E}\Delta\boldsymbol{u}(k+i) &\leq \boldsymbol{b}, i = 0,1,\ldots,N-1\\
\boldsymbol{M}\boldsymbol{y}(k+N) &\leq \boldsymbol{b}
\end{aligned}
$$

  - The constraint model (5.1) can then be reformulated w.r.t. the augmented system (7.14)/(7.15) and the prediction model (7.16) as

$$
\begin{aligned}
\boldsymbol{\mathcal{D}}\tilde{\boldsymbol{\mathcal{C}}}\tilde{\boldsymbol{x}}(k) + \boldsymbol{\mathcal{M}}\tilde{\boldsymbol{\mathcal{C}}}\left(\tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{x}}(k)+\tilde{\boldsymbol{\Gamma}}\Delta\boldsymbol{U}(k)\right) + \boldsymbol{\mathcal{E}}\Delta\boldsymbol{U}(k) &\leq \boldsymbol{\mathcal{b}} && \Leftrightarrow\\
(\boldsymbol{\mathcal{D}}\tilde{\boldsymbol{\mathcal{C}}} + \boldsymbol{\mathcal{M}}\tilde{\boldsymbol{\mathcal{C}}}\tilde{\boldsymbol{\Phi}})\tilde{\boldsymbol{x}}(k) + (\boldsymbol{\mathcal{M}}\tilde{\boldsymbol{\mathcal{C}}}\tilde{\boldsymbol{\Gamma}} + \boldsymbol{\mathcal{E}})\Delta\boldsymbol{U}(k) &\leq \boldsymbol{\mathcal{b}} && \Leftrightarrow\\
\underbrace{(\boldsymbol{\mathcal{M}}\tilde{\boldsymbol{\mathcal{C}}}\tilde{\boldsymbol{\Gamma}} + \boldsymbol{\mathcal{E}})}_{\tilde{\boldsymbol{\mathcal{A}}}}\Delta\boldsymbol{U}(k) &\leq \boldsymbol{\mathcal{b}} + \underbrace{(-\boldsymbol{\mathcal{D}}\tilde{\boldsymbol{\mathcal{C}}} - \boldsymbol{\mathcal{M}}\tilde{\boldsymbol{\mathcal{C}}}\tilde{\boldsymbol{\Phi}})}_{\tilde{\boldsymbol{\mathcal{W}}}}\tilde{\boldsymbol{x}}(k) && \Leftrightarrow
\end{aligned}
$$

where $\boldsymbol{\mathcal{D}}$, $\boldsymbol{\mathcal{M}}$, $\boldsymbol{\mathcal{E}}$ and $\boldsymbol{\mathcal{b}}$ are defined as in (5.1)

Reference Tracking

Reference Tracking based on the Delta Input Formulation

- **Approach**
  - Problem 4.1 is then solved by the optimal state feedback control law

$$\partial/_{\partial\Delta\boldsymbol{U}(k)} V_N\big(\tilde{\boldsymbol{x}}(k),\Delta\boldsymbol{U}(k),\boldsymbol{R}(k)\big) = \tilde{\boldsymbol{H}}\Delta\boldsymbol{U}(k)+\tilde{\boldsymbol{F}}\tilde{\boldsymbol{x}}(k)+\tilde{\boldsymbol{F}}_{\boldsymbol{R}}\boldsymbol{R}(k)=\boldsymbol{0} \Leftrightarrow \Delta\boldsymbol{U}^*(k) = -\tilde{\boldsymbol{H}}^{-1}\tilde{\boldsymbol{F}}\tilde{\boldsymbol{x}}(k)-\tilde{\boldsymbol{H}}^{-1}\tilde{\boldsymbol{F}}_{\boldsymbol{R}}\boldsymbol{R}(k)$$

  - Problem 5.1 can then be formulated as the quadratic program

$$\min_{\Delta\boldsymbol{U}(k)} \frac{1}{2}\Delta\boldsymbol{U}^T(k)\,\tilde{\boldsymbol{H}}\Delta\boldsymbol{U}(k)+\Delta\boldsymbol{U}^T(k)\left(\tilde{\boldsymbol{F}}\tilde{\boldsymbol{x}}(k)+\tilde{\boldsymbol{F}}_{\boldsymbol{R}}\boldsymbol{R}(k)\right)+\cancel{f\big(\tilde{\boldsymbol{x}}(k),\boldsymbol{R}(k)\big)}$$

$$\text{subject to } \tilde{\mathcal{A}}\Delta\boldsymbol{U}(k)\le \boldsymbol{b}+\tilde{\boldsymbol{W}}\tilde{\boldsymbol{x}}(k)$$

Term is independent of $\Delta\boldsymbol{U}(k)$

Term is therefore not relevant!

The reference input sequence $\boldsymbol{R}(k)$ occurs here!

  - The receding horizon controller in the unconstrained case is given by

$$\begin{aligned}\boldsymbol{u}^*(k) &= (\boldsymbol{I}_{m\times m}\ \ \boldsymbol{0}_{m\times m}\ \cdots\ \ \boldsymbol{0}_{m\times m})\Delta\boldsymbol{U}^*(k)+\boldsymbol{u}^*(k-1)\\ &= \underbrace{-(\boldsymbol{I}_{m\times m}\ \ \boldsymbol{0}_{m\times m}\ \cdots\ \ \boldsymbol{0}_{m\times m})\tilde{\boldsymbol{H}}^{-1}\tilde{\boldsymbol{F}}}_{\tilde{\boldsymbol{K}}_{\mathrm{RHC}}}\tilde{\boldsymbol{x}}(k)+\underbrace{-(\boldsymbol{I}_{m\times m}\ \ \boldsymbol{0}_{m\times m}\ \cdots\ \ \boldsymbol{0}_{m\times m})\tilde{\boldsymbol{H}}^{-1}\tilde{\boldsymbol{F}}_{\boldsymbol{R}}}_{\tilde{\boldsymbol{K}}_{\boldsymbol{R}\mathrm{RHC}}}\boldsymbol{R}(k)+\boldsymbol{u}^*(k-1)\end{aligned}$$

  - The receding horizon controller is an affine TI state feedback controller in the unconstrained case

Technische Universität Kaiserslautern

Reference Tracking

Reference Tracking based on the Delta Input Formulation

- **Approach**
  - The feedback matrices $\tilde{\boldsymbol{K}}_{\mathrm{RHC}}$ and $\tilde{\boldsymbol{K}}_{\boldsymbol{R}\mathrm{RHC}}$ can be calculated offline in the unconstrained case
  - The closed-loop system is globally asymptotically stable iff $\rho\big(\tilde{\boldsymbol{A}}+\tilde{\boldsymbol{B}}\tilde{\boldsymbol{K}}_{\mathrm{RHC}}\big)<1$ (cf. Theorem 2.3)
  - The receding horizon controller in the constrained case results as

$$\boldsymbol{u}^*(k) = (\boldsymbol{I}_{m\times m}\quad \boldsymbol{0}_{m\times m}\quad \cdots\quad \boldsymbol{0}_{m\times m})\Delta\boldsymbol{U}^*(k)+\boldsymbol{u}^*(k-1)$$

  - The receding horizon controller is a nonlinear state feedback controller in the constrained case
- **Remarks**
  - The extension for state constraints is straightforward
  - Constraints on $\boldsymbol{u}(k+i-1)$ and thus on $\boldsymbol{u}(k+i)$ can be written as $(\boldsymbol{0}_{m\times n}\quad \boldsymbol{I}_{m\times m})\tilde{\boldsymbol{x}}(k+i)\in\mathbb{U}$
  - Disturbances and uncertainties in $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ do not lead to a steady-state error or offset
  - More details and references are given in [Mac02] and [BBM15, Section 13.6]

Reference Tracking

Preview Control for Reference Tracking

- **Motivation**
  - Sometimes the reference input $\boldsymbol{r}(k+i)$ is known over the prediction horizon
  - E.g. in motion control problems the reference sequence is usually precomputed
  - The reference input $\boldsymbol{r}(k+i)$ can then be included in Problem (7.13) and previewed in this way
  - The receding horizon controller can then work proactively
- **Approach**
  - Problem (7.13) with reference $\boldsymbol{r}(k+i)$ can be solved in "batch" way using quadratic programming
  - Assume that the reference input $\boldsymbol{r}(k+i)$ is known for $i = 0,1,\ldots,N$
  - The reference input $\boldsymbol{r}(k)$ can then be replaced by the reference input $\boldsymbol{r}(k+i)$ in cost function (7.12)
  - The reference input sequence on Slide 7-13ff is then $\boldsymbol{R}(k) = (\boldsymbol{r}^T(k+1) \;\; \boldsymbol{r}^T(k+2) \;\; \cdots \;\; \boldsymbol{r}^T(k+N))^T$
  - The formulation as a quadratic program then follows analogously to Slide 7-12ff

Disturbance Rejection

Reference Tracking based on Target Calculation with Disturbance Estimation

- **Discrete-Time Linear Time-Invariant (LTI) System**

$$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k) + \boldsymbol{B}_w\boldsymbol{w}(k) \quad \text{state equation} \tag{7.17}$$

$$\boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k) + \boldsymbol{C}_w\boldsymbol{w}(k) \quad \text{measured output equation} \tag{7.18}$$

$$\boldsymbol{y}_r(k) = \boldsymbol{C}_r\boldsymbol{x}(k) + \boldsymbol{C}_{rw}\boldsymbol{w}(k) \quad \text{controlled output equation} \tag{7.19}$$

- **Symbols**

  $\boldsymbol{x}(k) \in \mathbb{R}^n$ state vector — $\boldsymbol{u}(k) \in \mathbb{U} \subseteq \mathbb{R}^m$ input vector

  $\boldsymbol{w}(k) \in \mathbb{R}^{m_w}$ disturbance vector

  $\boldsymbol{y}(k) \in \mathbb{Y} \subseteq \mathbb{R}^p$ measured output vector — $\boldsymbol{y}_r(k) \in \mathbb{R}^{p_r}$ controlled output vector

  $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ system matrix — $\boldsymbol{B} \in \mathbb{R}^{n\times m}$ input matrix

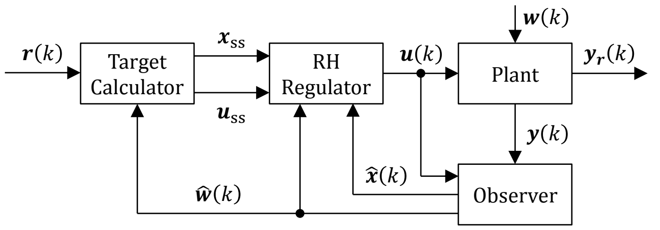
  $\boldsymbol{B}_w \in \mathbb{R}^{n\times m_w}$ disturbance input matrix

  $\boldsymbol{C} \in \mathbb{R}^{p\times n}$ measured output matrix — $\boldsymbol{C}_r \in \mathbb{R}^{p_r\times n}$ controlled output matrix

  $\boldsymbol{C}_w \in \mathbb{R}^{p\times m_w}$ — $\boldsymbol{C}_{rw} \in \mathbb{R}^{p_r\times m_w}$

Reference Tracking based on Target Calculation with Disturbance Estimation

- **Structure**



- **Objective**
  - Control the discrete-time LTI system such that $y_r(k) \to r(k)$ if $r(k), w(k) \to$ const. as $k \to \infty$
- **Approach**
  - The observer estimates the current state $\hat{x}(k)$ and the disturbance $\hat{w}(k)$
  - The target calculator and RH regulator are essentially utilized as outlined on Slide 7-3

Reference Tracking based on Target Calculation with Disturbance Estimation

- **Approach**
  - Introduce the disturbance model

    $$w(k+1) = w(k)$$
  - Augment the discrete-time LTI system (7.17)/(7.18) by the disturbance model, i.e.

    $$\begin{pmatrix} x(k+1) \\ w(k+1) \end{pmatrix} = \begin{pmatrix} A & B_w \\ 0 & I \end{pmatrix} \begin{pmatrix} x(k) \\ w(k) \end{pmatrix} + \begin{pmatrix} B \\ 0 \end{pmatrix} u(k) \quad (7.20)$$

    $$y(k) = \begin{pmatrix} C & C_w \end{pmatrix} \begin{pmatrix} x(k) \\ w(k) \end{pmatrix} \quad (7.21)$$
  - Design an observer for the augmented discrete-time LTI system (7.20)/(7.21) based on the methods developed in Chapter 2

Disturbance Rejection

Reference Tracking based on Target Calculation with Disturbance Estimation

- **Approach**
  - Consider that the discrete-time LTI system (7.17)/(7.19) is in the steady target state $\boldsymbol{x}_{ss}$, i.e.

$$\boldsymbol{x}_{ss} = \boldsymbol{A}\boldsymbol{x}_{ss} + \boldsymbol{B}\boldsymbol{u}_{ss} + \boldsymbol{B}_w\widehat{\boldsymbol{w}} \iff (\boldsymbol{I}_{n\times n} - \boldsymbol{A})\boldsymbol{x}_{ss} - \boldsymbol{B}\boldsymbol{u}_{ss} = \boldsymbol{B}_w\widehat{\boldsymbol{w}}$$
$$\boldsymbol{y}_{rss} = \boldsymbol{C}_r\boldsymbol{x}_{ss} + \boldsymbol{C}_{rw}\widehat{\boldsymbol{w}} = \boldsymbol{r} \iff \boldsymbol{C}_r\boldsymbol{x}_{ss} = \boldsymbol{r} - \boldsymbol{C}_{rw}\widehat{\boldsymbol{w}}$$

  - Note that the estimated disturbance $\widehat{\boldsymbol{w}}$ is used since the steady-state disturbance $\boldsymbol{w}_{ss}$ is unknown
  - Rewrite the equations in matrix form, i.e.

$$\begin{pmatrix} \boldsymbol{I}_{n\times n} - \boldsymbol{A} & -\boldsymbol{B} \\ \boldsymbol{C}_r & \boldsymbol{0}_{p_r\times m} \end{pmatrix} \begin{pmatrix} \boldsymbol{x}_{ss} \\ \boldsymbol{u}_{ss} \end{pmatrix} = \begin{pmatrix} \boldsymbol{B}_w\widehat{\boldsymbol{w}} \\ \boldsymbol{r} - \boldsymbol{C}_{rw}\widehat{\boldsymbol{w}} \end{pmatrix} \quad (7.22)$$

  - The steady target pair $(\boldsymbol{x}_{ss}, \boldsymbol{u}_{ss})$ can be then calculated from (7.22) provided that a solution exists
  - Note that the estimated disturbance $\widehat{\boldsymbol{w}}$ is regarded in the target pair $(\boldsymbol{x}_{ss}, \boldsymbol{u}_{ss})$
  - Introduce the state deviation and input deviation

$$\widetilde{\boldsymbol{x}}(k) = \widehat{\boldsymbol{x}}(k) - \boldsymbol{x}_{ss}, \quad \widetilde{\boldsymbol{u}}(k) = \boldsymbol{u}(k) - \boldsymbol{u}_{ss}$$

TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

Disturbance Rejection

Reference Tracking based on Target Calculation with Disturbance Estimation

- **Approach**
  - This leads to the discrete-time LTI state equation

$$\widetilde{\boldsymbol{x}}(k+1) = \widehat{\boldsymbol{x}}(k+1) - \boldsymbol{x}_{ss} = \boldsymbol{A}\widehat{\boldsymbol{x}}(k) + \boldsymbol{B}\boldsymbol{u}(k) + \boldsymbol{B}_w\widehat{\boldsymbol{w}}(k) - (\boldsymbol{A}\boldsymbol{x}_{ss} + \boldsymbol{B}\boldsymbol{u}_{ss} + \boldsymbol{B}_w\boldsymbol{w}_{ss}) = \boldsymbol{A}\widetilde{\boldsymbol{x}}(k) + \boldsymbol{B}\widetilde{\boldsymbol{u}}(k) \quad (7.23)$$

    the discrete-time quadratic cost function

$$\widetilde{V}_N\left(\widetilde{\boldsymbol{x}}(k), \widetilde{\boldsymbol{U}}(k)\right) = \widetilde{\boldsymbol{x}}^T(k+N)\boldsymbol{P}\widetilde{\boldsymbol{x}}(k+N) + \sum_{i=0}^{N-1} \widetilde{\boldsymbol{x}}^T(k+i)\boldsymbol{Q}\widetilde{\boldsymbol{x}}(k+i) + \widetilde{\boldsymbol{u}}^T(k+i)\boldsymbol{R}\widetilde{\boldsymbol{u}}(k+i) \quad (7.24)$$

    the state and input constraints

$$\begin{aligned} &\boldsymbol{C}(\widetilde{\boldsymbol{x}}(k+i) + \boldsymbol{x}_{ss}) + \boldsymbol{C}_w\widehat{\boldsymbol{w}}(k) \in \mathbb{Y}, i = 1,2,\ldots,N \\ &\widetilde{\boldsymbol{u}}(k+i) + \boldsymbol{u}_{ss} \in \mathbb{U}, \qquad i = 0,1,\ldots,N-1 \end{aligned} \quad (7.25)$$

  - Note that $\widehat{\boldsymbol{w}}(k) = \boldsymbol{w}_{ss}$ is assumed in (7.23) as detailed in [PR03] and [RM09, Section 1.5.1]
  - Note that the deviations are resubstituted in (7.25) since the constraint sets $\mathbb{Y}$ and $\mathbb{U}$ are considered

Disturbance Rejection

Reference Tracking based on Target Calculation with Disturbance Estimation

- **Approach**
  - Rewrite Problem 4.1/5.1 w.r.t. the state equation (7.23), cost function (7.24) and constr. (7.25), i.e.

$$\min_{\tilde{\boldsymbol{U}}(k)} \tilde{V}_N(\tilde{\boldsymbol{x}}(k), \tilde{\boldsymbol{U}}(k))$$
$$\text{subject to} \begin{cases} \tilde{\boldsymbol{x}}(k+i+1) = \boldsymbol{A}\tilde{\boldsymbol{x}}(k+i) + \boldsymbol{B}\tilde{\boldsymbol{u}}(k+i), i = 0,1,\dots,N-1 \\ \boldsymbol{C}(\tilde{\boldsymbol{x}}(k+i) + \boldsymbol{x}_{\text{ss}}) + \boldsymbol{C}_w\hat{\boldsymbol{w}}(k) \in \mathbb{Y}, \quad i = 1,2,\dots,N \\ \tilde{\boldsymbol{u}}(k+i) + \boldsymbol{u}_{\text{ss}} \in \mathbb{U}, \quad i = 0,1,\dots,N-1 \end{cases} \tag{7.26}$$

- **Remark on the Detectability**
  - The augmented discrete-time LTI system (7.20)/(7.21) is detectable iff

    (1) $(\boldsymbol{C}, \boldsymbol{A})$ is detectable

    (2) $\begin{pmatrix} \boldsymbol{I}_{n\times n} - \boldsymbol{A} & -\boldsymbol{B}_w \\ \boldsymbol{C} & \boldsymbol{C}_w \end{pmatrix} \in \mathbb{R}^{(n+p)\times(n+m_w)}$ has full rank $n + m_w$
  - This implies that number of disturbances $m_w \leq$ number of measured outputs p is required for detectability of the disturbance. Note that $\boldsymbol{B}_w$ and $\boldsymbol{C}_w$ can always be chosen such that (2) is fulfilled.

Disturbance Rejection

Reference Tracking based on Target Calculation with Disturbance Estimation

- **Remark on the Constrained Case**
  - For the constrained case target pair $(\boldsymbol{x}_{ss}, \boldsymbol{u}_{ss})$ must fulfill output constraint $\mathbb{Y}$ and input constraint $\mathbb{U}$
  - For this purpose the target calculator based on (7.22) must be modified to

$$\min_{\boldsymbol{x}_{\text{ss}}, \boldsymbol{u}_{\text{ss}}} \frac{1}{2}\left((\boldsymbol{C}_r\boldsymbol{x}_{\text{ss}} + \boldsymbol{C}_{rw}\hat{\boldsymbol{w}} - \boldsymbol{r})^T\boldsymbol{Q}_{\text{ss}}(\boldsymbol{C}_r\boldsymbol{x}_{\text{ss}} + \boldsymbol{C}_{rw}\hat{\boldsymbol{w}} - \boldsymbol{r}) + (\boldsymbol{u}_{\text{ss}} - \boldsymbol{u}_{\text{ss}}^{\text{unc}})^T\boldsymbol{R}_{\text{ss}}(\boldsymbol{u}_{\text{ss}} - \boldsymbol{u}_{\text{ss}}^{\text{unc}})\right)$$
$$\text{subject to} \begin{cases} \begin{pmatrix} \boldsymbol{I}_{n\times n} - \boldsymbol{A} & -\boldsymbol{B} \\ \boldsymbol{C}_r & \boldsymbol{0}_{p_r\times m} \end{pmatrix}\begin{pmatrix} \boldsymbol{x}_{\text{ss}} \\ \boldsymbol{u}_{\text{ss}} \end{pmatrix} = \begin{pmatrix} \boldsymbol{B}_w\hat{\boldsymbol{w}} \\ \boldsymbol{r} - \boldsymbol{C}_{rw}\hat{\boldsymbol{w}} \end{pmatrix} \\ \boldsymbol{C}\boldsymbol{x}_{\text{ss}} + \boldsymbol{C}_w\hat{\boldsymbol{w}} \in \mathbb{Y} \\ \boldsymbol{u}_{\text{ss}} \in \mathbb{U} \end{cases} \tag{7.27}$$

    where $\boldsymbol{u}_{\text{ss}}^{\text{unc}}$ is the target input resulting from (7.22) for the unconstrained case and $\boldsymbol{Q}_{\text{ss}} = \boldsymbol{Q}_{\text{ss}}^T \succcurlyeq \boldsymbol{0}$ and $\boldsymbol{R}_{\text{ss}} = \boldsymbol{R}_{\text{ss}}^T \succ \boldsymbol{0}$ are weighting matrices
  - Problem (7.27) can be formulated as a quadratic program

Disturbance Rejection

Reference Tracking based on Target Calculation with Disturbance Estimation

- **Remark on Receding Horizon Control**
  1. Estimate the current state $\hat{\boldsymbol{x}}(k)$ and the disturbance $\hat{\boldsymbol{w}}(k)$
  2. Solve Problem (7.27) for the given $\boldsymbol{r}(k)$ and estimated $\hat{\boldsymbol{w}}(k)$ to determine the target pair $(\boldsymbol{x}_{ss}, \boldsymbol{u}_{ss})$
  3. Solve Problem (7.26) for $\tilde{\boldsymbol{x}}(k) = \hat{\boldsymbol{x}}(k) - \boldsymbol{x}_{ss}$ to determine the optimal input sequence $\tilde{\boldsymbol{U}}^*(k)$
  4. Compute first element of the optimal input sequence $\tilde{\boldsymbol{u}}^*(k) = (\boldsymbol{I}_{m\times m} \quad \boldsymbol{0}_{m\times m} \quad \cdots \quad \boldsymbol{0}_{m\times m})\tilde{\boldsymbol{U}}^*(k)$
  5. Implement the optimal input $\boldsymbol{u}^*(k) = \tilde{\boldsymbol{u}}^*(k) + \boldsymbol{u}_{ss}$
  6. Increment the time instant $k \coloneqq k+1$ and go to 1.
- **Further Remarks**
  - The remarks on the optimization problem and on the target calculator given on Slide 7-6 analogously apply to reference tracking with disturbance estimation
  - Note that reference tracking with disturbance estimation is also denoted as offset-free control
  - More details and references are given in [RM09, Section 1.5.1], [BBM15, Section 13.6], and [PR03]

Disturbance Rejection

Preview Control for Disturbance Rejection

- **Motivation**
  - Consider the discrete-time LTI state equation (7.17) including the disturbance $\boldsymbol{w}(k)$
  - Sometimes the disturbance $\boldsymbol{w}(k+i)$ can be predicted or measured over the prediction horizon
  - E.g. the renewable generation in a power system can be predicted with good accuracy
  - E.g. the road displacement before a car can be measured with a camera sensor
  - The disturbance $\boldsymbol{w}(k+i)$ can then be included in Problem 4.1/5.1 and previewed in this way
  - The receding horizon controller can then work proactively
- **Approach**
  - Problem 4.1/5.1 with disturbance $\boldsymbol{w}(k+i)$ can be solved in "batch" way using quadratic programming
  - The prediction model (4.4), the cost function (4.5), and the constraint model (5.1) must be reformulated w.r.t. the disturbance $\boldsymbol{w}(k+i)$ to this end
  - Assume that the disturbance $\boldsymbol{w}(k+i)$ can be predicted or measured for $i = 0,1,\dots,N-1$

Disturbance Rejection

Preview Control for Disturbance Rejection

- **Approach**
  - The solution of the discrete-time LTI state equation (7.17) is then given by

$$
\begin{aligned}
\boldsymbol{x}(k+1) &= \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k) + \boldsymbol{B}_w\boldsymbol{w}(k)\\
\boldsymbol{x}(k+2) &= \boldsymbol{A}\boldsymbol{x}(k+1) + \boldsymbol{B}\boldsymbol{u}(k+1) + \boldsymbol{B}_w\boldsymbol{w}(k+1)\\
&= \boldsymbol{A}^2\boldsymbol{x}(k) + \boldsymbol{A}\boldsymbol{B}\boldsymbol{u}(k) + \boldsymbol{A}\boldsymbol{B}_w\boldsymbol{w}(k) + \boldsymbol{B}\boldsymbol{u}(k+1) + \boldsymbol{B}_w\boldsymbol{w}(k+1)\\
&\;\vdots\\
\boldsymbol{x}(k+N) &= \boldsymbol{A}^N\boldsymbol{x}(k) + \boldsymbol{A}^{N-1}\boldsymbol{B}\boldsymbol{u}(k) + \boldsymbol{A}^{N-1}\boldsymbol{B}_w\boldsymbol{w}(k) + \cdots + \boldsymbol{A}\boldsymbol{B}\boldsymbol{u}(k+N-2) + \boldsymbol{A}\boldsymbol{B}_w\boldsymbol{w}(k+N-2)\\
&\quad + \boldsymbol{B}\boldsymbol{u}(k+N-1) + \boldsymbol{B}_w\boldsymbol{w}(k+N-1)
\end{aligned}
$$

  - The prediction model is then given by

$$
\underbrace{\begin{pmatrix}\boldsymbol{x}(k+1)\\ \boldsymbol{x}(k+2)\\ \vdots\\ \boldsymbol{x}(k+N)\end{pmatrix}}_{\boldsymbol{X}(k)} = \underbrace{\begin{pmatrix}\boldsymbol{A}\\ \boldsymbol{A}^2\\ \vdots\\ \boldsymbol{A}^N\end{pmatrix}}_{\boldsymbol{\Phi}}\underbrace{\boldsymbol{x}(k)}_{\boldsymbol{x}(k)} + \underbrace{\begin{pmatrix}\boldsymbol{B} & \boldsymbol{0} & \cdots & \boldsymbol{0}\\ \boldsymbol{A}\boldsymbol{B} & \boldsymbol{B} & \cdots & \boldsymbol{0}\\ \vdots & \vdots & \ddots & \vdots\\ \boldsymbol{A}^{N-1}\boldsymbol{B} & \boldsymbol{A}^{N-2}\boldsymbol{B} & \cdots & \boldsymbol{B}\end{pmatrix}}_{\boldsymbol{\Gamma}}\underbrace{\begin{pmatrix}\boldsymbol{u}(k)\\ \boldsymbol{u}(k+1)\\ \vdots\\ \boldsymbol{u}(k+N-1)\end{pmatrix}}_{\boldsymbol{U}(k)} + \underbrace{\begin{pmatrix}\boldsymbol{B}_w & \boldsymbol{0} & \cdots & \boldsymbol{0}\\ \boldsymbol{A}\boldsymbol{B}_w & \boldsymbol{B}_w & \cdots & \boldsymbol{0}\\ \vdots & \vdots & \ddots & \vdots\\ \boldsymbol{A}^{N-1}\boldsymbol{B}_w & \boldsymbol{A}^{N-2}\boldsymbol{B}_w & \cdots & \boldsymbol{B}_w\end{pmatrix}}_{\boldsymbol{\Gamma}_w}\underbrace{\begin{pmatrix}\boldsymbol{w}(k)\\ \boldsymbol{w}(k+1)\\ \vdots\\ \boldsymbol{w}(k+N-1)\end{pmatrix}}_{\boldsymbol{W}(k)} \quad (7.28)
$$

Disturbance Rejection

Preview Control for Disturbance Rejection

- **Approach**
  - Substituting the prediction model (7.28) into the cost function (4.5) leads to

$$
\begin{aligned}
V_N\big(\boldsymbol{x}(k),\boldsymbol{U}(k),\boldsymbol{W}(k)\big) &= \boldsymbol{x}^T(k)\boldsymbol{Q}\boldsymbol{x}(k) + \boldsymbol{X}^T(k)\,\boldsymbol{\Omega}\boldsymbol{X}(k) + \boldsymbol{U}^T(k)\boldsymbol{\Psi}\boldsymbol{U}(k)\\
&= \boldsymbol{x}^T(k)\boldsymbol{Q}\boldsymbol{x}(k) + \big(\boldsymbol{\Phi}\boldsymbol{x}(k) + \boldsymbol{\Gamma}\boldsymbol{U}(k) + \boldsymbol{\Gamma}_w\boldsymbol{W}(k)\big)^T\boldsymbol{\Omega}\big(\boldsymbol{\Phi}\boldsymbol{x}(k) + \boldsymbol{\Gamma}\boldsymbol{U}(k) + \boldsymbol{\Gamma}_w\boldsymbol{W}(k)\big) + \boldsymbol{U}^T(k)\boldsymbol{\Psi}\boldsymbol{U}(k)\\
&= \boldsymbol{x}^T(k)\boldsymbol{Q}\boldsymbol{x}(k) + \boldsymbol{x}^T(k)\boldsymbol{\Phi}^T\boldsymbol{\Omega}\boldsymbol{\Phi}\boldsymbol{x}(k) + \boldsymbol{x}^T(k)\boldsymbol{\Phi}^T\boldsymbol{\Omega}\boldsymbol{\Gamma}\boldsymbol{U}(k) + \boldsymbol{x}^T(k)\boldsymbol{\Phi}^T\boldsymbol{\Omega}\boldsymbol{\Gamma}_w\boldsymbol{W}(k)\\
&\quad + \boldsymbol{U}^T(k)\boldsymbol{\Gamma}^T\boldsymbol{\Omega}\boldsymbol{\Phi}\boldsymbol{x}(k) + \boldsymbol{U}^T(k)\boldsymbol{\Gamma}^T\boldsymbol{\Omega}\boldsymbol{\Gamma}\boldsymbol{U}(k) + \boldsymbol{U}^T(k)\boldsymbol{\Gamma}^T\boldsymbol{\Omega}\boldsymbol{\Gamma}_w\boldsymbol{W}(k)\\
&\quad + \boldsymbol{W}^T(k)\boldsymbol{\Gamma}_w^T\boldsymbol{\Omega}\boldsymbol{\Phi}\boldsymbol{x}(k) + \boldsymbol{W}^T(k)\boldsymbol{\Gamma}_w^T\boldsymbol{\Omega}\boldsymbol{\Gamma}\boldsymbol{U}(k) + \boldsymbol{W}^T(k)\boldsymbol{\Gamma}_w^T\boldsymbol{\Omega}\boldsymbol{\Gamma}_w\boldsymbol{W}(k) + \boldsymbol{U}^T(k)\boldsymbol{\Psi}\boldsymbol{U}(k)\\
&= \boldsymbol{U}^T(k)(\boldsymbol{\Psi} + \boldsymbol{\Gamma}^T\boldsymbol{\Omega}\boldsymbol{\Gamma})\boldsymbol{U}(k) + 2\boldsymbol{U}^T(k)\boldsymbol{\Gamma}^T\boldsymbol{\Omega}\boldsymbol{\Phi}\boldsymbol{x}(k) + 2\boldsymbol{U}^T(k)\boldsymbol{\Gamma}^T\boldsymbol{\Omega}\boldsymbol{\Gamma}_w\boldsymbol{W}(k)\\
&\quad + \boldsymbol{x}^T(k)(\boldsymbol{Q} + \boldsymbol{\Phi}^T\boldsymbol{\Omega}\boldsymbol{\Phi})\boldsymbol{x}(k) + \boldsymbol{W}^T(k)\boldsymbol{\Gamma}_w^T\boldsymbol{\Omega}\boldsymbol{\Gamma}_w\boldsymbol{W}(k) + 2\boldsymbol{W}^T(k)\boldsymbol{\Gamma}_w^T\boldsymbol{\Omega}\boldsymbol{\Phi}\boldsymbol{x}(k)\\
&= \tfrac{1}{2}\boldsymbol{U}^T(k)\underbrace{2(\boldsymbol{\Psi} + \boldsymbol{\Gamma}^T\boldsymbol{\Omega}\boldsymbol{\Gamma})}_{\boldsymbol{H}}\boldsymbol{U}(k) + \boldsymbol{U}^T(k)(\underbrace{2\boldsymbol{\Gamma}^T\boldsymbol{\Omega}\boldsymbol{\Phi}}_{\boldsymbol{F}}\boldsymbol{x}(k) + \underbrace{2\boldsymbol{\Gamma}^T\boldsymbol{\Omega}\boldsymbol{\Gamma}_w}_{\boldsymbol{F}_w}\boldsymbol{W}(k))\\
&\quad + \boldsymbol{x}^T(k)(\boldsymbol{Q} + \boldsymbol{\Phi}^T\boldsymbol{\Omega}\boldsymbol{\Phi})\boldsymbol{x}(k) + \boldsymbol{W}^T(k)\boldsymbol{\Gamma}_w^T\boldsymbol{\Omega}\boldsymbol{\Gamma}_w\boldsymbol{W}(k) + 2\boldsymbol{W}^T(k)\boldsymbol{\Gamma}_w^T\boldsymbol{\Omega}\boldsymbol{\Phi}\boldsymbol{x}(k)\\
&= \tfrac{1}{2}\boldsymbol{U}^T(k)\,\boldsymbol{H}\,\boldsymbol{U}(k) + \boldsymbol{U}^T(k)\big(\boldsymbol{F}\,\boldsymbol{x}(k) + \boldsymbol{F}_w\,\boldsymbol{W}(k)\big) + f\big(\boldsymbol{x}(k),\boldsymbol{W}(k)\big)
\end{aligned}
$$

Technische Universität Kaiserslautern

Disturbance Rejection

Preview Control for Disturbance Rejection

- **Approach**
  - Substituting the prediction model (7.28) into the constraint model (5.1) leads to

$$
\begin{aligned}
&\boldsymbol{\mathcal{D}}(k)\boldsymbol{x}(k) + \boldsymbol{\mathcal{M}}(k)\big(\boldsymbol{\Phi}\boldsymbol{x}(k) + \boldsymbol{\Gamma}\boldsymbol{U}(k) + \boldsymbol{\Gamma}_w\boldsymbol{W}(k)\big) + \boldsymbol{\mathcal{E}}(k)\boldsymbol{U}(k) \le \boldsymbol{\mathcal{b}}(k) \quad \Leftrightarrow \\
&(\boldsymbol{\mathcal{D}}(k) + \boldsymbol{\mathcal{M}}(k)\boldsymbol{\Phi})\boldsymbol{x}(k) + \big(\boldsymbol{\mathcal{M}}(k)\boldsymbol{\Gamma} + \boldsymbol{\mathcal{E}}(k)\big)\boldsymbol{U}(k) + \boldsymbol{\mathcal{M}}(k)\boldsymbol{\Gamma}_w\boldsymbol{W}(k) \le \boldsymbol{\mathcal{b}}(k) \quad \Leftrightarrow \\
&\underbrace{\big(\boldsymbol{\mathcal{M}}(k)\boldsymbol{\Gamma} + \boldsymbol{\mathcal{E}}(k)\big)}_{\boldsymbol{\mathcal{A}}(k)}\boldsymbol{U}(k) \le \boldsymbol{\mathcal{b}}(k) + \underbrace{(-\boldsymbol{\mathcal{D}}(k) - \boldsymbol{\mathcal{M}}(k)\boldsymbol{\Phi})}_{\boldsymbol{\mathcal{W}}(k)}\boldsymbol{x}(k) + \underbrace{(-\boldsymbol{\mathcal{M}}(k)\boldsymbol{\Gamma}_w)}_{\boldsymbol{\mathcal{W}}_w(k)}\boldsymbol{W}(k) \quad \Leftrightarrow
\end{aligned}
$$

  - Problem 4.1 is then solved by the optimal state feedback control law

$$
\partial/_{\partial \boldsymbol{U}(k)} V_N\big(\boldsymbol{x}(k), \boldsymbol{U}(k), \boldsymbol{W}(k)\big) = \boldsymbol{H}\boldsymbol{U}(k) + \boldsymbol{F}\boldsymbol{x}(k) + \boldsymbol{F}_w\boldsymbol{W}(k) = \boldsymbol{0} \Leftrightarrow \boldsymbol{U}^*(k) = -\boldsymbol{H}^{-1}\boldsymbol{F}\boldsymbol{x}(k) - \boldsymbol{H}^{-1}\boldsymbol{F}_w\boldsymbol{W}(k)
$$

  - Problem 5.1 can then be formulated as the quadratic program

$$
\min_{\boldsymbol{U}(k)} \frac{1}{2}\boldsymbol{U}^T(k)\boldsymbol{H}\boldsymbol{U}(k) + \boldsymbol{U}^T(k)\big(\boldsymbol{F}\boldsymbol{x}(k) + \boldsymbol{F}_w\boldsymbol{W}(k)\big) + \cancel{f\big(\boldsymbol{x}(k), \boldsymbol{W}(k)\big)}
$$

Term is independent of $\boldsymbol{U}(k)$

Term is therefore not relevant!

$$
\text{subject to } \boldsymbol{\mathcal{A}}(k)\boldsymbol{U}(k) \le \boldsymbol{\mathcal{b}}(k) + \boldsymbol{\mathcal{W}}(k)\boldsymbol{x}(k) + \boldsymbol{\mathcal{W}}_w(k)\boldsymbol{W}(k)
$$

The disturbance sequence $\boldsymbol{W}(k)$ occurs here!

Technische Universität Kaiserslautern

Disturbance Rejection

Preview Control for Disturbance Rejection

- **Approach**
  - The receding horizon controller in the unconstrained case is given by

$$
\begin{aligned}
\boldsymbol{u}^*(k) &= (\boldsymbol{I}_{m\times m} \quad \boldsymbol{0}_{m\times m} \quad \cdots \quad \boldsymbol{0}_{m\times m})\boldsymbol{U}^*(k) \\
&= \underbrace{-(\boldsymbol{I}_{m\times m} \quad \boldsymbol{0}_{m\times m} \quad \cdots \quad \boldsymbol{0}_{m\times m})\boldsymbol{H}^{-1}\boldsymbol{F}}_{\boldsymbol{K}_{\mathrm{RHC}}}\boldsymbol{x}(k) \underbrace{- (\boldsymbol{I}_{m\times m} \quad \boldsymbol{0}_{m\times m} \quad \cdots \quad \boldsymbol{0}_{m\times m})\boldsymbol{H}^{-1}\boldsymbol{F}_w}_{\boldsymbol{K}_{w\mathrm{RHC}}}\boldsymbol{W}(k) \\
&= \boldsymbol{K}_{\mathrm{RHC}}\boldsymbol{x}(k) + \boldsymbol{K}_{w\mathrm{RHC}}\boldsymbol{W}(k)
\end{aligned}
$$

  - The receding horizon controller is an affine TI state feedback controller in the unconstrained case
  - The feedback matrices $\boldsymbol{K}_{\mathrm{RHC}}$ and $\boldsymbol{K}_{w\mathrm{RHC}}$ can be calculated offline in the unconstrained case
  - The closed-loop system is globally asymptotically stable iff $\rho(\boldsymbol{A} + \boldsymbol{B}\boldsymbol{K}_{\mathrm{RHC}}) < 1$ (cf. Theorem 2.3)
  - The receding horizon controller in the constrained case results as

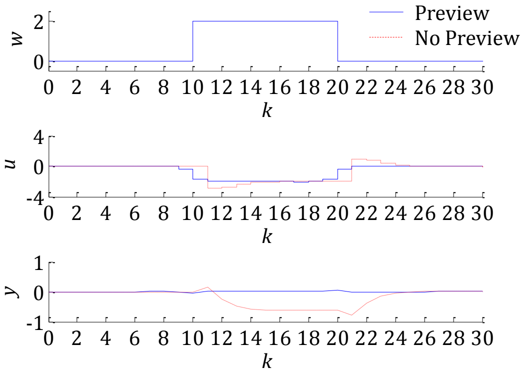
$$
\boldsymbol{u}^*(k) = (\boldsymbol{I}_{m\times m} \quad \boldsymbol{0}_{m\times m} \quad \cdots \quad \boldsymbol{0}_{m\times m})\boldsymbol{U}^*(k)
$$

  - The receding horizon controller is an nonlinear state feedback controller in the constrained case

Disturbance Rejection

Preview Control for Disturbance Rejection

- **Illustrative Example**



Example from Chapter 4

$\boldsymbol{x}(0) = (0 \quad 0)^T$

$y(k) = (-1 \quad 1)\boldsymbol{x}(k)$

No state and input constraints

Disturbance input matrix $\boldsymbol{B}_w = \boldsymbol{B}$

Input weight $R = 0.01$

Terminal weight $\boldsymbol{P} = \boldsymbol{P}_{\mathrm{LQR}}$

RHC (prediction horizon $N = 5$)

Preview RHC works proactively

Very good performance

Literature

Miscellaneous

[MR93] Kenneth R. Muske and James B. Rawlings. Model predictive control with linear models. *AIChE Journal*, 39(2):262–287, 1993.

[PR03] Gabriele Pannocchia and James B. Rawlings. Disturbance models for offset-free model predictive control. *AIChE Journal*, 49(2):426–437, 2003.