1 System Theory

1.1 Nonlinear Systems

$$\dot{x}(t) = A^c x(t) + B^c u(t)$$
$$x(t) = e^{A^c(t-t_0)}x_0 + \int_{t_0}^{t} e^{A^c(t-\tau)}Bu(\tau)d\tau$$
$$e^{A^c t} = \sum_{n=0}^{\infty} \frac{(A^c t)^n}{n!}$$

1.2 Linear Systems

$$x_{k+1} = Ax_k + Bu_k$$
$$y_k = Cx_k + Du_k$$
$$x_{k+N} = A^N x_k + \sum_{i=0}^{N-1} A^i B u_{k+N-1-i}$$

1.3 Lyapunov Stability

We define a system to be stable in the sense of Lyapunov, if it stays in any arbitrarily small neighborhood of the origin when it is disturbed slightly

Lyapunov stable if for every $\epsilon > 0$ there exists a $\delta(\epsilon)$ such that

$$\text{norm}\, x(0) < \delta(\epsilon) \rightarrow \text{norm}\, x(k) < \epsilon, \forall k \geq 0$$

asymptotically stable in $\Omega \subseteq \mathbb{R}^n$ if it is Lyapunov stable and attractive $\lim_{k\rightarrow\infty} x(k) = 0, \forall x(0) \in \Omega$.

Lyapunov Function $V : \mathbb{R}^n \rightarrow \mathbb{R}$ must be continous at the origin, finite $\forall x \in \Omega$ and:

$$V(0) = 0 \text{ and } V(x) > 0, \forall x \in \Omega \backslash \{0\}$$
$$V(g(x)) - V(x) \leq -\alpha(x), \forall x \in \Omega \backslash \{0\}$$

where $\alpha : \mathbb{R}^n \rightarrow \mathbb{R}$ is continuous positive definite, equilibrium at $x = 0$ and $\Omega \subset \mathbb{R}^n$ closed and bounded set containing the origin.

Lyapunov Theorem If a system admits Lyapunov function $V(x)$, then $x = 0$ is asymptotically stable in Ω (sufficient but not necessary)
If additionally $\|x\| \rightarrow \infty \Rightarrow V(x) \rightarrow \infty$, then $x = 0$ is globally asymptotically stable.
To check if $V(x) = x^T \boldsymbol{P} x$ is valid Lyapunov function of system $x_{k+1} = \boldsymbol{A}x_k$ check if $(\boldsymbol{APA} - \boldsymbol{P})$ has neg. eigen values.
In other words: Iff eigenvalues of A inside unit circle (i.e. stable) then $\exists$*unique* $\boldsymbol{P} > 0$ that solves $\boldsymbol{A}_{cl}^T \boldsymbol{P} \boldsymbol{A}_{cl} - \boldsymbol{P} = -\boldsymbol{Q}$, $\boldsymbol{Q} > 0$ and $V(x) = x^T \boldsymbol{P} x$ is a lyapunov function.

1.4 Discretization

Euler: $A = I + T_s A^c,\ B = T_s B^c,\ C = C^c,\ D = D^c$

$$x_{k+1} = x_k + T_s g^c(x_k, u_k) = g(x_k, u_k)$$
$$y_k = h^c(x_k, u_k) = h(x_k, u_k)$$

Exact: (assume constant $u(t)$ during T_s)

$$A = e^{A^c T_s},\ B = \int_0^{T_s} e^{A^c(T_s - \tau')} B^c d\tau$$
$$B = (A^c)^{-1}(A - I)B^c, \text{ if } A^c \text{ invertible}$$

1.5 Controllability (reachability) and observability

$$C = [B\ AB\ ...\ A^{n-1}B]$$
$$O = [C^T\ (CA)^T\ ...\ (CA^{n-1})^T]$$

2 Unconstrained Control

2.1 Block Approach (used also for $\bar{w}$ substition)

$$\begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_N \end{bmatrix} = \begin{bmatrix} \boldsymbol{I} \\ \boldsymbol{A} \\ \vdots \\ \boldsymbol{A}^N \end{bmatrix} x(0) + \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} & \dots & \boldsymbol{0} \\ \boldsymbol{B} & \boldsymbol{0} & \dots & \boldsymbol{0} \\ \boldsymbol{AB} & \boldsymbol{B} & \dots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \boldsymbol{0} \\ \boldsymbol{A}^{N-1}\boldsymbol{B} & \dots & \boldsymbol{AB} & \boldsymbol{B} \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N_1} \end{bmatrix}$$

$$\begin{aligned}
x = \boldsymbol{S}^x \cdot x(0) + \boldsymbol{S}^u \cdot u \quad & \text{size}(\boldsymbol{S}^x) = [n_{\text{states}} \cdot (N+1), N] \\
& \text{size}(\boldsymbol{S}^u) = [n_{\text{states}} \cdot (N+1), n_{\text{states}}] \\
\bar{\boldsymbol{Q}} = \text{diag}(\boldsymbol{Q}, \dots, \boldsymbol{Q}, \boldsymbol{P}) \quad & \text{size}(\bar{\boldsymbol{Q}}) = [n_{\text{states}} \cdot (N+1), n_{\text{states}} \cdot (N+1)] \\
\bar{\boldsymbol{R}} = \text{diag}(\boldsymbol{R}, \dots, \boldsymbol{R}) \quad & \text{size}(\bar{\boldsymbol{R}}) = [n_{\text{input}} \cdot N, n_{\text{input}} \cdot N] \\
\boldsymbol{H} = \boldsymbol{S}^{uT} \bar{\boldsymbol{Q}} \boldsymbol{S}^u + \boldsymbol{R} \quad & \boldsymbol{F} = \boldsymbol{S}^{xT} \bar{\boldsymbol{Q}} \boldsymbol{S}^u \\
\boldsymbol{Y} = \boldsymbol{S}^{xT} \bar{\boldsymbol{Q}} \boldsymbol{S}^x &
\end{aligned}$$

Optimal cost and control

$$J^*(x_0) = -x_0^T \boldsymbol{FHF}^T x_0 + x_0^T \boldsymbol{Y} x_0$$
$$u^*(x_0) = -\boldsymbol{H}^{-1}\boldsymbol{F}^T x_0 = -\left(\boldsymbol{S}^{uT} \bar{\boldsymbol{Q}} \boldsymbol{S}^u + \boldsymbol{R}\right)^{-1} \boldsymbol{S}^{uT} \bar{\boldsymbol{Q}} \boldsymbol{S}^x x_0$$

2.2 Recursive Approach

$$J_k^*(x_k) = \min_{u_k} I(x_k, u_k) + J_{k+1}(x_{k+1})$$

Is a feedback controller as opposed to the Batch Approach. For LQR solve via Riccati Difference Equation (RDE).

$$\boldsymbol{F}_k = -(\boldsymbol{B}^T \boldsymbol{P}_{k+1} \boldsymbol{B} + \boldsymbol{R})^{-1} \boldsymbol{B}^T \boldsymbol{P}_{k+1} \boldsymbol{A}$$
$$\boldsymbol{P}_k = \boldsymbol{A}^T \boldsymbol{P}_{k+1} \boldsymbol{A} + \boldsymbol{Q} - \boldsymbol{A}^T \boldsymbol{P}_{k+1} \boldsymbol{B}(\boldsymbol{B}^T \boldsymbol{P}_{k+1} \boldsymbol{B} + \boldsymbol{R})^{-1} \boldsymbol{B}^T \boldsymbol{P}_{k+1} \boldsymbol{A}$$

$$u_k^* = \boldsymbol{F}_k\ x_k \qquad J_k^*(x_k) = x_k^T \boldsymbol{P}_k\ x_k \qquad \boldsymbol{P}_N = \boldsymbol{P}$$

For unconstrained Infinite Horizon Problem, substituting $\boldsymbol{P}_\infty = \boldsymbol{P}_k = \boldsymbol{P}_{k+1}$ into RDE gives DARE. Uniquely solvable, iff (A, B) stabilizable and (A, G) detectable, where $\boldsymbol{GG}^T = \boldsymbol{Q}$. Follows from closed-loop system $x_{k+1} = (\boldsymbol{A} + \boldsymbol{BF}_k)x_k$

3 (Convex) Optimization

General Problem $\min_{x \in \text{dom}(f)} f(x)$ s. t. $g_i(x) \leq 0$ and $h_j(x) = 0$.

Norm $f(x) : \mathbb{R}^n \rightarrow \mathbb{R}$

$$\begin{aligned}
f(x) = 0 \Rightarrow x = 0, \quad & f(x) \geq 0 \\
f(\alpha \cdot x) = |\alpha| \cdot f(x) \quad & \text{for scalar } \alpha \\
f(x + y) \leq f(x) + f(y) \quad & \forall x, y \in R^n
\end{aligned}$$

3.1 Convexity

Convex set $\mathcal{X}$ iff $\forall \lambda \in [0, 1] \forall x, y \in \mathcal{X}\ \lambda x + (1 - \lambda)y \in \mathcal{X}$.
Intersection preserves convexity, union does not.

Affine set $\mathcal{X}$ $= \{x \in \mathbb{R}^n | \boldsymbol{A}x = b\}$ for some $\boldsymbol{A}, b$

Subspace is affine set through origin, i.e. $b = \boldsymbol{0}$, aka Nullspace of $\boldsymbol{A}$.

Hyperplane $\mathcal{X}$ $= \{x \in \mathbb{R}^n | a^T x = b\}$ for some a, b.

Halfspace $\mathcal{X}$ $= \{x \in \mathbb{R}^n | a^T x \leq b\}$ for some a, b.

Polyhedron $\mathcal{P}$ $= \{x | a_i^T x \leq b_i, i = 1, \dots, n\} = \{x | \boldsymbol{A}x \leq b\}$

Cone $\mathcal{X}$ if for all $x \in \mathcal{X}$, and for all $\theta > 0, \theta x \in \mathcal{X}$.

Ellipsoid $\mathcal{E}$ $= \{x | (x - x_c)^T \boldsymbol{A}^{-1}(x - x_c) \leq 1\}$, x_c center point.

Convex function $f : \text{dom}(f) \rightarrow \mathbb{R}$ is convex iff $\text{dom}(f)$ is convex and $f(\lambda x + (1 - \lambda)y) \geq \lambda f(x) + (1 - \lambda)f(y), \forall \lambda \in (0, 1), \forall x, y \in \text{dom}(f)$.

Norm ball is convex (for all norms).

Epigraph set $f : \text{dom}(f) \rightarrow \mathbb{R}$ is the set

$$\text{epi}(f) := \left\{ \begin{bmatrix} x \\ t \end{bmatrix} | x \in \text{dom}(f), f(x) \leq t \right\} \subseteq \text{dom}(f) \times \mathbb{R}$$

Level set L_a of a function f for value a is the set of all $x \in \text{dom}(f)$ for which $f(x) = a$: $L_a = \{x | x \in \text{dom}(f), f(x) = a\}$.

Sublevel set C_a is defined by $C_a = \{x | x \in \text{dom}(f), f(x) \leq a\}$.

3.2 Linear Programming (LP)

Problem statement $\min c^T x$ such that $\boldsymbol{G}x \leq h$ and $\boldsymbol{A}x = b$.

Norm l_∞ $\min_x \|x\|_\infty = \min_{x\in\mathbb{R}^n}[\max\{x,\dots,x_n,-x_1,\dots,-x_n\}]$:

$$\begin{aligned} &\min_{x,t} t && \text{subject to} \quad x_i \le t, -x_i \le t, && \boldsymbol{F}x \le g \\ \iff &\min_{x,t} t && \text{subject to} \quad -\mathbf{1}t \le x \le \mathbf{1}t, && \boldsymbol{F}_x \le g. \end{aligned}$$

Norm l_1 $\min_x \|x\|_1 = \min_x \left[\sum_{i=1}^m \max\{x_i, -x_i\}\right]$:

$$\begin{aligned} &\min_t t_1 + \dots + t_m && \text{subject to} \quad x_i \le t_i, -x_i \le t_i, && \boldsymbol{F}x \le g \\ \iff &\min_t \mathbf{1}^T t && \text{subject to} \quad -t \le x \le t, && \boldsymbol{F}_x \le g. \end{aligned}$$

Note that for $\dim x = 1$, l_1 and l_∞ are the same. Note also that t is scalar for norm l_∞ and a vector in norm l_1.

Piecewise Affine

$$\min_x \left[\max_{i=1,\dots,m}\{c_i^T x + d_i\}\right] \quad \text{s.t. } \boldsymbol{G}x \le h$$
$$\iff \min_{x,t} t \quad \text{s.t. } c_i^T x + d_i \le t, \boldsymbol{G}x \le h$$

3.3 Duality

Lagrangian Dual Function

$$L(x,\lambda,\nu) = f(x) + \sum_{i=1}^m \lambda_i g_i(x) + \sum_{i=1}^p \nu_i h_i(x)$$
$$d(\lambda,\nu) = \inf_{x\in\mathcal{X}} L(x,\lambda,\nu) \quad \text{i.e. } \nabla_x L(x,\lambda,\nu) = 0$$

Dual Problem (always convex) $\max_{\lambda,\nu} d(\lambda,\nu)$ s. t. $\lambda \ge 0$.
Optimal value is lower bound for primal: $d^* \le p^*$.
If primal convex, Slater condition (strict feasibility) implies *strong duality*: $\{x \mid Ax = b, f_i(x) < 0,\} \ne \emptyset \Rightarrow d^* = p^*$

Karush-Kuhn-Tucker (KKT) Conditions are necessary for optimality (and sufficient if primal convex).

Primal Feasability	$f_i(x^*) \le 0$	$i = 1,\dots,m$
	$h_i(x^*) = 0$	$i = 1,\dots,p$
Dual Feasability	$\lambda^* \ge 0$	
Complementary slackness	$\lambda_i^* \cdot f_i(x^*) = 0$	$i = 1,\dots,m$
Stationarity	$\nabla_x L(x^*,\lambda^*,\nu^*) = 0$	

Dual of LP

$$\min_x c^T x \quad \text{s.t. } \boldsymbol{A}x = b, \boldsymbol{C}x \le e$$
$$\iff \max_{\lambda,\nu} -b^T\nu - e^T\lambda \quad \text{s.t. } A^T\nu + C^T\lambda + c = 0, \lambda \ge 0$$

Dual of QP

$$\min_x \frac{1}{2}x^T\boldsymbol{Q}x + c^T x \quad \text{s.t. } \boldsymbol{C}x \le e$$
$$\iff \min_{\lambda,\nu} \frac{1}{2}\lambda^T\boldsymbol{C}\boldsymbol{Q}^{-1}\boldsymbol{C}^T\lambda + (\boldsymbol{C}\boldsymbol{Q}^{-1}c + e)^T\lambda + \frac{1}{2}c^T\boldsymbol{Q}^{-1}c$$
$$\text{s.t. } \boldsymbol{Q}x + \nu + c^T\lambda = 0, \lambda \ge 0$$

4 Constrained Finite Time Optimal Control (CFTOC)

4.1 MPC with linear cost

$$J(x_0,u) = \|\boldsymbol{P}x_N\|_p + \sum_{i=0}^{N-1}\|\boldsymbol{Q}x_i\|_p + \|\boldsymbol{R}u_i\|_p.$$

The CFTOC problem can be formulated as an ∞-norm LP problem as shown below.

$$\begin{aligned} &\min_z \epsilon_0^x + \dots + \epsilon_N^x + \epsilon_0^u + \dots + \epsilon_{N-1}^u \\ &\text{s.t.} \quad -\mathbf{1}_n\epsilon_i^x \le \pm\boldsymbol{Q}\left[\boldsymbol{A}^i x_0 + \sum_{j=0}^{i-1}\boldsymbol{A}^j\boldsymbol{B}u_{i-1-j}\right] \\ &\qquad -\mathbf{1}_r\epsilon_N^x \le \pm\boldsymbol{P}\left[\boldsymbol{A}^N x_0 + \sum_{j=0}^{N-1}\boldsymbol{A}^j\boldsymbol{B}u_{N-1-j}\right] \\ &\qquad -\mathbf{1}_m\epsilon_N^u \le \pm\boldsymbol{R}u_i \\ &\qquad x_i = \boldsymbol{A}^i x_0 + \sum_{j=0}^{i-1}\boldsymbol{A}^j\boldsymbol{B}u_{i-1-j} \in \mathcal{X} \\ &\qquad x_N = \boldsymbol{A}^N x_0 + \sum_{j=1}^{N-1}\boldsymbol{A}^j\boldsymbol{B}u_{N-1-j} \in \mathcal{X} \\ &\qquad u_i \in \mathcal{U} \end{aligned}$$

Converting to LP form:

$$\begin{aligned} &\min_z c^T z \\ &\text{s.t. } \bar{\boldsymbol{G}}z \le \bar{w} + \bar{s}x_k \\ &z = \begin{bmatrix}\epsilon_0^x & \dots & \epsilon_N^x & \epsilon_0^u & \dots & \epsilon_{N-1}^u & u_0^T & \dots & u_{N-1}^T\end{bmatrix} \\ &c = \begin{bmatrix}1 & \dots & 1 & 1 & \dots & 1 & 0 & \dots & 0\end{bmatrix} \\ &\bar{\boldsymbol{G}} = \begin{bmatrix}\boldsymbol{G}_\epsilon & \boldsymbol{G}_u \\ 0 & \boldsymbol{G}\end{bmatrix}, \quad \bar{w} = \begin{bmatrix}w_\epsilon \\ w\end{bmatrix} \\ &\bar{s} = \begin{bmatrix}s_\epsilon \\ s\end{bmatrix} \end{aligned}$$

Where $\boldsymbol{G}$ is the normal problem constraints and $[\boldsymbol{G}_\epsilon\boldsymbol{G}_u]$ form the constraints of the newly introduced variable ϵ as given in the first 3 constraints in the section above.

4.2 QP with substitution (see also Batch approach)

$$J^*(x_k) = \min_u \begin{bmatrix}u^T & x_k^T\end{bmatrix}\begin{bmatrix}\boldsymbol{H} & \boldsymbol{F}^T \\ \boldsymbol{F} & \boldsymbol{Y}\end{bmatrix}\begin{bmatrix}u \\ x_k\end{bmatrix}$$
$$\text{s. t. } \boldsymbol{G}\ u \le w + \boldsymbol{E}\ x_k$$

Latter gives three sets (same for without substitution)

$$\mathcal{X} = \{x \mid A_x\ x \le b_x\}$$
$$\mathcal{U} = \{u \mid A_u\ u \le b_u\}$$
$$\mathcal{X}_f = \{x \mid A_f\ x \le b_f\}$$

State equations are in cost matrix, usually
$\boldsymbol{A}_x = \begin{bmatrix}1 \\ -1\end{bmatrix}, b_x = \begin{bmatrix}b_{\max} \\ -b_{\min}\end{bmatrix}$

$$\boldsymbol{G} = \begin{bmatrix} \boldsymbol{A}_u & 0 & \dots & 0 \\ 0 & \boldsymbol{A}_u & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \boldsymbol{A}_u \\ 0 & 0 & \dots & 0 \\ \boldsymbol{A}_x\boldsymbol{B} & 0 & \dots & 0 \\ \boldsymbol{A}_x\boldsymbol{A}\boldsymbol{B} & \boldsymbol{A}_x\boldsymbol{B} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{A}_x\boldsymbol{A}^{N-2}\boldsymbol{B} & \boldsymbol{A}_x\boldsymbol{A}^{N-3}\boldsymbol{B} & \dots & 0 \\ \boldsymbol{A}_f\boldsymbol{A}^{N-1}\boldsymbol{B} & \boldsymbol{A}_f\boldsymbol{A}^{N-2}\boldsymbol{B} & \dots & \boldsymbol{A}_f\boldsymbol{B} \end{bmatrix} \quad \boldsymbol{E} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ -\boldsymbol{A}_x \\ -\boldsymbol{A}_x\boldsymbol{A} \\ -\boldsymbol{A}_x\boldsymbol{A}^2 \\ \vdots \\ -\boldsymbol{A}_x\boldsymbol{A}^{N-1} \\ -\boldsymbol{A}_f\boldsymbol{A}^N \end{bmatrix} \quad \boldsymbol{W} = \begin{bmatrix} b_u \\ b_u \\ \vdots \\ b_u \\ b_x \\ b_x \\ b_x \\ \vdots \\ b_x \\ b_f \end{bmatrix}$$

4.3 QP without substitution

State equations represented in equality constraints (k fixed, usually $k = 0$).

$$J^*(x_k) = \min_z \begin{bmatrix}z^T & x_k^T\end{bmatrix}\begin{bmatrix}\bar{\boldsymbol{H}} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{Q}\end{bmatrix}\begin{bmatrix}z \\ x_k\end{bmatrix}$$
$$\text{s.t. } \boldsymbol{G}\ z \le w + \boldsymbol{E}\ x_k$$
$$\boldsymbol{G}_{\text{eq}}\ z = \boldsymbol{E}_{\text{eq}}\ x_k, \quad \text{system dynamics}$$

$\bar{\boldsymbol{H}} = \text{diag}(\boldsymbol{Q},\dots,\boldsymbol{Q},\boldsymbol{P},\boldsymbol{R},\dots,\boldsymbol{R})$

$$z = \begin{bmatrix} x_1 \\ \vdots \\ x_N \\ u_0 \\ \vdots \\ u_{N-1} \end{bmatrix} \quad \boldsymbol{G}_{\text{eq}} = \left[\begin{array}{cccc|cccc} \boldsymbol{I} & & & & -\boldsymbol{B} & & & \\ -\boldsymbol{A} & \boldsymbol{I} & & & & -\boldsymbol{B} & & \\ & \ddots & \ddots & & & & \ddots & \\ & & -\boldsymbol{A} & \boldsymbol{I} & & & & -\boldsymbol{B} \end{array}\right] \quad \boldsymbol{E}_{\text{eq}} = \begin{bmatrix} \boldsymbol{A} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$w = \begin{bmatrix} b_x \\ \vdots \\ b_f \\ b_u \\ \vdots \\ b_u \end{bmatrix} \quad \boldsymbol{G} = \left[\begin{array}{ccc|ccc} 0 & & & & & \\ & \boldsymbol{A}_x & & & & \\ & & \ddots & & & \\ & & \boldsymbol{A}_x & & & \\ \hline & & & \boldsymbol{A}_d & & \\ & & & & \ddots & \\ & & & & & \boldsymbol{A}_d \end{array}\right] \quad \boldsymbol{E} = \begin{bmatrix} -\boldsymbol{A}_x^T \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

4.4 Invariance

Def.: $x(k) \in O \Rightarrow x(k+1) \in O \forall k$.

$$\text{pre}(S) := \{x \mid g(x) \in S\} \qquad = \{x \mid Ax \in S\}$$

Max invariant set calculation: $\Omega_{i+1} \leftarrow \text{pre}(\Omega_i) \cap \Omega_i$, terminating when $\Omega_{i+1} = \Omega_i$.

tim: We need more here, pos. inv. set, max. pos.inv O_∞

4.5 Stability and Feasability

Main Idea: Choose $\mathcal{X}_f$ and $\boldsymbol{P}$ to mimic infinite horizon. LQR control law $\kappa(x) = \boldsymbol{F}_\infty x$ from solving DARE. Set terminal weight $\boldsymbol{P} = \boldsymbol{P}_\infty$, terminal set $\mathcal{X}_f$ as maximal invariant set:

$$x_{k+1} = \boldsymbol{A}x_k + \boldsymbol{B}\boldsymbol{F}_\infty\ x_k \in \mathcal{X}_f \quad \forall x_k \in \mathcal{X}_f \text{ terminal set invariant}$$
$$\mathcal{X}_f \subseteq \mathcal{X}, \quad \boldsymbol{F}_\infty\ x_k \in \mathcal{U} \quad \forall x_k \in \mathcal{X}_f \text{ constrainst satisfied}$$

We get: 1. Positive stage cost function, 2. invariant terminal set by construction, 3. Terminal cost is Lyapunov function with

$$x_{k+1}^T\boldsymbol{P}x_{k+1} - x_k^T\boldsymbol{P}x_k = -x_k^T(\boldsymbol{Q} + \boldsymbol{F}_\infty^T\boldsymbol{R}\boldsymbol{F}_\infty)x_k$$

Extension to non-linear (time-invariant) MPC possible since terminal set and cost do not rely on linearity.

5 Practical Issues

5.1 MPC for tracking

Target steady-state conditions $x_s = \boldsymbol{A}x_s + \boldsymbol{B}u_s$ and $y_s = \boldsymbol{C}x_s = r$ and constraints give:

$$\min_{x_s,u_s} u_s^T \boldsymbol{R} u_s \text{ subj. to } \begin{bmatrix} \boldsymbol{I}-\boldsymbol{A} & -\boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} x_s \\ u_s \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ r \end{bmatrix}, x_s \in \mathcal{X}, u_s \in \mathcal{U}$$

Usually assume x_s, u_s unique and feasible. If no solution exists, compute closest steady-state $\min(\boldsymbol{C}x_s - r)^T \boldsymbol{Q}(\boldsymbol{C}x_s - r)$ s. t. $x_s = \boldsymbol{A}x_s + \boldsymbol{B}u_s$.
MPC problem to drive $y \to r$ is:

$$\min_u \|y_N - \boldsymbol{C}x_s\|^2_{P_y} + \sum_{i=0}^{N-1} \|y_i - \boldsymbol{C}x_s\|^2_{Q_y} + \|u_i - u_s\|^2_R$$

5.2 Delta formulation

Reference r, $\Delta x_k = x_k - x_s, \Delta u_k = u_k - u_s$:

$$\begin{aligned}
\min V_f(\Delta x_N) + \sum_{i=0}^{N-1} \Delta x_i^T \boldsymbol{Q} \Delta x_i + \Delta u_i^T \boldsymbol{R} \Delta u_i \\
\text{s.t. } \Delta x_0 = \Delta x_k \\
\Delta x_{k+1} = \boldsymbol{A}\Delta x_k + \boldsymbol{B}\Delta u_k \\
\boldsymbol{H}_x x \le k_x \Rightarrow \boldsymbol{H}_x \Delta x \le k_x - \boldsymbol{H}_x x_s \\
\boldsymbol{H}_u u \le k_u \Rightarrow \boldsymbol{H}_u \Delta u \le k_u - \boldsymbol{H}_u u_s \\
\Delta x_N \in \mathcal{X}_f \quad \text{adjusted accordingly, shift (and scaled)} \\
x_s \oplus \mathcal{X}_f \subseteq \mathcal{X} \\
\boldsymbol{K}\Delta x + u_s \in \mathcal{U}
\end{aligned}$$

Control given by $u_0^* = \Delta u_0^* + u_s$.

5.3 Offset free tracking

$$\begin{aligned}
x_{k+1} &= \boldsymbol{A}x_k + \boldsymbol{B}u_k + \boldsymbol{B}_d d_k \\
d_{k+1} &= d_k \\
y_k &= \boldsymbol{C}x_k + \boldsymbol{C}_d d_k
\end{aligned}$$

$$\begin{bmatrix} \boldsymbol{I}-\boldsymbol{A} & -\boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} x_s \\ u_s \end{bmatrix} = \begin{bmatrix} \boldsymbol{B}_d \hat{d} \\ r - \boldsymbol{C}_d \hat{d} \end{bmatrix}$$

Choice of $\boldsymbol{B}_d, \boldsymbol{C}_d$ requires that $(\boldsymbol{A}, \boldsymbol{C})$ is observable and $\begin{bmatrix} \boldsymbol{A}-\boldsymbol{I} & \boldsymbol{B}_d \\ \boldsymbol{C} & \boldsymbol{C}_d \end{bmatrix}$ has full $(n_x + n_d)$ column frank (i.e. det $\neq 0$).
Intuition: for fixed y_s at steady-state, d_s is uniquely determined.
If plant has no integrator we can choose $\boldsymbol{B}_d = \boldsymbol{0}$ since $\det(\boldsymbol{A} - \boldsymbol{I}) \neq 0$.

$$\begin{bmatrix} \hat{x}_{k+1} \\ \hat{d}_{k+1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{B}_d \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \hat{d}_k \end{bmatrix} + \begin{bmatrix} \boldsymbol{B} \\ \boldsymbol{0} \end{bmatrix} u_k + \begin{bmatrix} \boldsymbol{L}_x \\ \boldsymbol{L}_d \end{bmatrix} \left(-y_k^m + \boldsymbol{C}\hat{x}_k + \boldsymbol{C}_d \hat{d}_k\right)$$

where y_k^m measured output; choose $\begin{bmatrix} \boldsymbol{L}_x \\ \boldsymbol{L}_d \end{bmatrix}$ s.t. error dynamics stable and converge to zero.

tim: Target condition here

If 1) number of dist. = number of outputs, 2) target steady-state problem feasible and no constraints active at steady-state, 3) closed-loop system converges, then target achieved without offset.

5.4 Soft-constraints via slack variables

$$\min_x f(z) + l_\epsilon(\epsilon) \quad \text{s.t. } g(z) \le \epsilon, \epsilon \ge 0$$

Requirement: Softened problem has same minimiser as original problem if feasible.
Quadratic error function $l_\epsilon(\epsilon) = v\epsilon + w\epsilon^2$, $w > 0$ gives smoothness, choose $v > \lambda^* \ge 0$ for exact penalty (above requirement fulfilled).

Move Blocking main idea to set a number of inputs as the same, $u_2 = u_3 = \cdots = u_N$, to reduce computational burden, at the slight cost of sub-optimality.

6 Robust MPC

Enforcing terminal constraints by robust invariance:

$$x \in O^{\mathcal{W}} \Rightarrow g(x, w) \in \Omega^W \; \forall w \in \mathcal{W}$$
$$\text{pre}^{\mathcal{W}}(\Omega) = \{x | g(x, w) \in \Omega \; \forall w \in \mathcal{W}\}$$

Enforcing sequential constraints for uncertain system ϕ:

$$\phi_i(x_0, u, w) = \left\{ x_i + \sum_{j=0}^{i-1} \boldsymbol{A}^j w_j \middle| w \in \mathcal{W}^i \right\} \subseteq \mathcal{X}$$
$$\phi_N(x_0, u, w) \in \mathcal{X}_f \quad \text{as well}$$

The uncertain system evolves with the summation of all the disturbances up to time i, hence we have to restrict the open-loop (determine control before disturbance is measured):

$$\boldsymbol{A}_x x \le b_x \text{ becomes } \boldsymbol{A}_x x_i + \boldsymbol{A}_x \sum_{j=0}^{i-1} \boldsymbol{A}^j w_k \le b_x :$$
$$\begin{aligned}
x_i &\in \mathcal{X} \ominus (\mathcal{W} \oplus \boldsymbol{A}\mathcal{W} \oplus \cdots \oplus \boldsymbol{A}^{i-1}\mathcal{W}) \\
&= \left(\bigoplus_{j=0}^{i-1} \boldsymbol{A}^j \mathcal{W} \right) = \begin{bmatrix} \boldsymbol{A}^0 & \ldots & \boldsymbol{A}^{i-1} \end{bmatrix} \mathcal{W}^i
\end{aligned}$$

For example: Robust invariant set calculation of $x_{k+1} = 0.5x_k + w_k$ under $-10 \le x \le 10$ and $-1 \le w \le 1$.

$$\begin{aligned}
\Omega_0 &= [-10, 10] \\
\text{pre}^{\mathcal{W}}(\Omega_0) &= \{x | -10 \le 0.5x + w \le 10 \text{ for } -1 \le w \le 1\} \\
&= \{x | -20 - 2w \le x \le 20 + 2w \text{ for } -1 \le w \le 1\} \\
&= \{x | -18 \le x \le 18\} \\
\Omega_1 &= [-10, 10] \cap [-18, 18] = [-10, 10] = \mathcal{O}_\infty^{\mathcal{W}}
\end{aligned}$$

For example: Terminal set calculation of $x_{k+1} = w_k$, $-1 \le w \le 1$, N= 2:

$$\mathcal{X}_f^{\mathcal{W}} = \mathcal{X}_f \ominus \left(\bigoplus_{j=0}^{1} \boldsymbol{A}^j \mathcal{W} \right) = \mathcal{X}_f \ominus 2\mathcal{W} = [-10, 10] \ominus [-2, 2] = [-8, 8]$$

Tube-MPC We want nominal system $z_k = \boldsymbol{A}z_k + \boldsymbol{B}v_k$ with "tracking" controller $u_k = \boldsymbol{K}(x_k - z_k) + v_k$ i.e. closed-loop, $\boldsymbol{K}$ found offline.
Step 1: Compute the minimal robust invariant set $\mathcal{E} = \bigoplus_{j=1}^{\infty} \boldsymbol{A}_{cl}^j \mathcal{W}$.
Step 2: Shrink Constraints:

$$\begin{aligned}
\{z_i\} \oplus \mathcal{E} \subseteq \mathcal{X} \quad &\Rightarrow \{z_i\} \in \mathcal{X} \ominus \mathcal{E} \\
u_i \in \boldsymbol{K}\mathcal{E} \oplus \{v_i\} \subset \boldsymbol{U} \quad &\Rightarrow \{v_i\} \in \mathcal{U} \ominus \boldsymbol{K}\mathcal{E} \\
z_n \in \mathcal{X}_f \ominus \mathcal{E} \quad &
\end{aligned}$$

Also check that the set $\mathcal{X}_f$ is invariant for the nominal system with tightened constraints: $(A + BK)\mathcal{X}_f \subseteq \mathcal{X}_f$, and that it satisfies the constraints: $\mathcal{X}_f \subseteq \mathcal{X} \ominus \mathcal{E}$ and $K\mathcal{X}_f \subseteq \mathcal{U} \ominus K\mathcal{E}$.

7 Explicit MPC

$z^*(x_k)$ is continuous and polyhedral piecewise affine over feasible set.

7.1 Quadratic Cost

$J^*(x_k)$ is continuous, convex and polyhedral piecewise quadratic.
Using the formulation with substitution,

$$\begin{aligned}
J(x_k) &= \min z^T H z - x_k^T (Y - FH^{-1}F^T) x_k \\
&\text{s.t} \quad Gz \le w + Sx_k \\
z(x_k) &= U + H^{-1}F^T x_k \\
S &= E + GH^{-1}F^T \\
U^* &= z^*(x_k) - H^{-1}F^T x_k
\end{aligned}$$

The first solution gives $u^*(x_k) = \kappa(x_k)$, which is continuous and piecewise affine on polyhedra $\kappa(x) = F_j x + g_j$.

7.2 $1/\infty$-norm

$J^*(x_k)$ is continuous, convex and polyhedral piecewise affine.
Optimal solution: $u_0^* = \begin{bmatrix} 0 & \ldots & 0 & I_m & 0 & \ldots & 0 \end{bmatrix} z^*(x_k)$, and is in the same form as the QP case above.

8 Hybrid MPC

8.1 Piecewise Affine (PWA) Systems

Affine dynamics and output for each region:

$$\begin{Bmatrix} x_{k+1} & = A^i x_k + B^i u_k + f^i \\ y_k & = C^i x_k + D^i u_k + g_i \end{Bmatrix} \text{ if } x_t \in \mathcal{X}_j$$

Polyhedral partition of the (x, u)-space:

$$\{\mathcal{X}_i\}_{j=1}^s = \{x, u | H_j x + J_j u \le K_j\}$$

8.2 Mixed Logical Dynamical Hybrid Model (MLD)

Idea: associate boolean to binary: $p_j \iff \delta_i = 1, \neg p_j \iff \delta_i = 0$.

8.2.1 Translate logic rules to Linear Integer Inequalities

AND	$p_1 \wedge p_2$	$\delta_1 \geq 1, \delta_1 \geq 1$ also $\delta_1 + \delta_2 \geq 2$
OR	$p_1 \vee p_2$	$\delta_1 + \delta_2 \geq 1$
NOT	$\neg p_1$	$1 - \delta_1 \geq 1$ also $\delta_1 = 0$
XOR	$p_1 + p_2$	$\delta_1 + \delta_2 = 1$
IMPLY	$p_1 \rightarrow p_2$	$\delta_1 - \delta_2 \leq 0$
IFF	$p_1 \leftrightarrow p_2$	$\delta_1 - \delta_2 = 0$
ASSIGNMENT	$p_3 \leftrightarrow p_1 \wedge p_2$	$\delta_1 + (1 - \delta_3) \geq 1$, $\delta_2 + (1 - \delta_3) \geq 1$, $(1 - \delta_1) + (1 - \delta_2) + \delta_3 \geq 1$
CNF-Clause	$\neg p_1 \vee \neg p_2 \vee p_3$	$\delta_1 + \delta_2 + \delta_3 \leq 1$

Logic Equality Rules

$$\begin{aligned} \neg(A \wedge B) &= \neg A \vee \neg B \\ A \wedge (B \vee C) &= (A \wedge B) \vee (A \wedge C) \\ A \vee (B \wedge C) &= (A \vee B) \wedge (A \vee C) \end{aligned}$$

8.2.2 Translate continuous and logical components into Linear Mixed-Integer Relations

Event generator: $\delta_e(k) = f_{EG}(x_c(k), u_c(k), t)$.
Consider: $p \Leftrightarrow a^T x \leq b, \mathcal{X} = \{x | a^T x - b \in [m, M]\}$.
Translated to linear inequalities: $m\delta < a^T x - b \leq M(1 - \delta)$, where $[m, M]$ are lower and upper bounds.

Representing Switched Affine Dynamics as "IF-THEN-ELSE" relations

IF p THEN $z_k = a_1^T x_k + b_1$ else $z_k = a_2^T x_k + b_2 \Leftrightarrow$

$$\begin{aligned} (m_2 - M_1)\delta + z_k \leq a_2^T x_k + b_2 \leq -(m_1 - M_2)\delta + z_k \\ (m_1 - M_2)(1 - \delta) + z_k \leq a_1^T x_k + b_1 \leq -(m_1 - M_2)(1 - \delta) + z_k \end{aligned}$$

This results in a linear MLD model

$$\begin{aligned} x_{k+1} &= A x_k + B_1 u_k + B_2 \delta_k + B_3 z_k \\ y_k &= C x_k + D_1 u_k + D_2 \delta_k + D_3 z_k \\ E_2 \delta_k + E_3 z_k &\leq E_4 x_k + E_1 u_k + E_5 \end{aligned}$$

where the last equation describes the relationship between the continuous and integer variables. Physical constraints on cont. variables: $\{\mathcal{C}\} = \left\{ \begin{bmatrix} x_c \\ u_c \end{bmatrix} \in \mathcal{R}^{n_c + m_c} | F x_c + G u_c \leq H \right\}$

8.3 CFTOC for Hybrid Systems

$$\begin{aligned} J^*(x) = \min_U \ & l_N(x_N) + \sum_{k=0}^{N-1} l(x_k, u_k, \delta_k, z_k) \\ \text{s.t} \ & x_{k+1} = A x_k + B_1 u_k + B_2 \delta_k + B_3 z_k \\ & E_2 \delta_k + E_3 z_k \leq E_4 x_k + E_1 u_k + E_5 \\ & x_N \in \mathcal{X}_f, x_0 = x(0) \end{aligned}$$

8.4 MILP/QP

$$\begin{aligned} \min \ & c_c z_c + c_b z_b + d \quad \text{OR} \quad z H z + q z + d \\ \text{s.t} \ & G_c z_c + G_b z_b \leq W \\ & z_c \in R^{s_c}, z_b \in \{0, 1\}^{s_b} \end{aligned}$$

Branch and bound method can be used to efficiently solve the problem. Explict solution is a time varying fb law for both problems: $u_k^*(x_k) = F_k^j x_k + G_k^j$ if $x_k \in \mathcal{R}_k^j$.

9 Numerical Optimization – Iterative Methods

9.1 Gradient descent

$x_{i+1} = x_i - h_i \nabla f(x_i)$ with step-size $h_i = \frac{1}{L}$ for L-smooth $f(x)$:

$$\begin{aligned} & \exists L : \|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\| \ \forall x, y, \in \mathbb{R}^n \\ & \iff \nabla f \text{ is Lipschitz continuous} \\ & \iff f \text{ can be upperbounded by a quadratic function:} \\ & f(x) \leq f(y) + \nabla f(y)^T (x - y) + 0.5L \|x - y\|^2 \ \forall x, y \in \mathbb{R}^n \end{aligned}$$

9.2 Newton's Method

$$x_{i+1} = x_i - h_i \Delta x_{nt} := x_i - h_i (\nabla^2 f(x_i))^{-1} \nabla f(x_i)$$

Line search problem: choose $h_i > 0$ s.t. $f(x_i + h_i \Delta x_{nt}) \leq f(x_i)$.
Either compute exact and best h_i using:

$$h_i^* = \text{argmin } x_i - h_i \Delta x_{nt}$$

Or use the backtracking search method:

For $\alpha \in (0, 0.5)$ and $\beta \in (0, 1)$:
initialise $h_i = 1$;
while $f(x_i + h_i \Delta x_{nt}) > f(x_i) + \alpha h_i \nabla f(x_i)^T \Delta x_{nt}$ do $h_i \leftarrow \beta h_i$

For given equality constraint $\boldsymbol{A} x = b$ solve:

$$\begin{bmatrix} \nabla^2 f(x_i) & \boldsymbol{A}^T \\ \boldsymbol{A} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \delta x \\ \lambda \end{bmatrix} = \begin{bmatrix} -\nabla f(x_i) \\ \boldsymbol{0} \end{bmatrix}$$

9.3 Constrained optimization with $g_i(x) \leq 0$

Gradient method $x_{i+1} = \pi_Q(x_i - h_i \nabla f(x_i))$ where π_Q is a projection $\pi_q = \arg\min_x \frac{1}{2}\|x - y\|_2^2$. Projection can be solved directly if simple enough, else solve the dual.

9.4 Interior-Point methods

Assumptions $f(x^*) < \infty$, $\tilde{x} \in \text{dom}(f)$.

Barrier method $\min f(x) + \kappa \phi(x)$. Approximate ϕ using diff'able log barrier(instead of indicator function):

$$\begin{aligned} \phi(x) &= \sum_{i=1}^m I_-(g_i(x)) = -\sum_{i=1}^m \log(-g_i(x)) \\ & \lim_{\kappa \to 0} x^*(\kappa) = x^* \end{aligned}$$

Analytic center: $\arg\min_x \phi(x)$, central path $\{x^*(\kappa) | \kappa > 0\}$.

Path following method

1. Centering $x^*(\kappa) = \arg\min_x f(x) + \kappa \phi(x)$ with newton's method:
1.1. $\Delta x_{\text{nt}} = \left[\nabla^2 f(x) + \kappa \nabla^2 \phi(x)\right]^{-1} (-\nabla f(x) - \kappa \nabla \phi(x))$.
1.2. Line search:

$$\begin{aligned} \text{retain feasability: } & \text{argmax}_{h > 0} \{h | g_i(x + h\Delta x) < 0\} \\ \text{Find } h^* = & \text{argmin}_{h \in [0, h_{\max}]} \{f(x + h\Delta x) + \kappa \phi(x + h\Delta x)\} \end{aligned}$$

2. Update step $x_i = x^*(\kappa_i)$
3. Stop if $m\kappa_i \leq \epsilon$
4. Decrease $\kappa_{i+1} = \kappa_i / \mu$, $\mu > 1$.

Centering step with equality constraints

$$\begin{bmatrix} \nabla^2 f(x) + \kappa \nabla^2 \phi(x) & c^T \\ c & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \Delta x_{\text{nt}} \\ \nu \end{bmatrix} = -\begin{bmatrix} \nabla f(x) + \kappa \nabla \phi(x) \\ 0 \end{bmatrix}$$

Relaxed KKT

$$\begin{aligned} C x^* &= d & g_i(x^*) + s_i^* &= 0 \\ \nabla f(x^*) + \sum_{i=1}^m \lambda_i^* \nabla g_i(x^*) + c^T \nu &= 0 & \lambda_i^* = \kappa \frac{\partial \phi}{\partial g_i} &= -\frac{\kappa}{g_i} \\ \lambda_i^* g_i(x^*) &= -\kappa & \lambda_i^*, s_i^* &\geq 0 \end{aligned}$$

Primal Dual Search Direction Computation

$$\begin{bmatrix} H(x, \lambda) & c^T & G(x)^T & 0 \\ c & 0 & 0 & 0 \\ G(x) & 0 & 0 & \boldsymbol{I} \\ 0 & 0 & \boldsymbol{S} & \boldsymbol{\Lambda} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta u \\ \Delta \lambda \\ \Delta s \end{bmatrix} = -\begin{bmatrix} \nabla f(x) + c^T \nu + G(x)^T \lambda \\ cx - d \\ g(x) + s \\ s\lambda - \nu \end{bmatrix}$$

$\boldsymbol{S} = \text{diag}(s_1, \ldots, s_m), \boldsymbol{\Lambda} = \text{diag}(\lambda_1, \ldots, \lambda_m)$ and ν is a vector for choosing centering parameters.