

Model Predictive Control

Silvan Stadelmann - 5. November 2025 - v1.1.0

github.com/silvasta/summary-mpc



Contents

1	Nominal MPC	1
2	Practical MPC	1
2.1	Steady-state Target Problem	1
2.2	Offset-free Reference Tracking	1
2.3	Soft Constraints	1
3	Robust MPC	2
3.1	Robust Constraint-Tightening MPC	2
3.2	Robust Tube MPC	2
4	Stochastic MPC	2
5	Nonlinear MPC	2
6	Invariance	2
6.1	Control Invariance	2
6.2	Robust Invariance	2
6.3	Computing Invariant Sets and Pre-sets	2
7	Introduction to Systems and Controls	3
7.1	Linearization	3
7.2	Discretization	3
7.3	Analysis of Discrete-Time LTI Systems	3
8	Optimal Control	3
8.1	Unconstrained Finite Horizon Control Problem	3
8.2	Infinite Horizon Control Problem	3
8.3	Constrained Finite Time Optimal Control Problem	3
9	Optimization	4
9.1	Convex Sets	4
9.2	Convex Functions	4
9.3	Optimality Conditions	4
9.4	Convex Optimization Problems	4
10	Implementation, needed?	4

Requirements and Steps to MPC

- 1 **Model of the System** dynamics to state space
- 2 **State Estimator** track trajectory and disturbance
- 3 **Optimal Control Problem** define strategy
- 4 **Optimization problem** mathematical formulation
- 5 **Get Optimal Control Sequence** solve optimization
- 6 **Verify Closed-Loop Performance** iterative tests

1 Nominal MPC

MPC Mathematical Formulation

$$\operatorname{argmin}_U \sum_{i=0}^{N-1} l(x_i, u_i) + l_f(x_N) \quad (1)$$

Constraints $x_0 = x(k)$
 $x_{i+1} = Ax_i + Bu_i$
 $x_i \in \mathcal{X}$
 $u_i \in \mathcal{U}$
 $x_N \in \mathcal{X}_f$
 $l_f(\cdot)$ and \mathcal{X}_f are chosen to mimic an infinite horizon.

What can go wrong with *standard* MPC?

- No feasibility guarantee, the problem may not have a solution
- No stability guarantee, trajectories may not converge to origin

Stability of MPC - Main Result

Assumptions

- 1 Stage cost is strictly positive and only zero at the origin
- 2 Terminal set is **invariant** under local control law $\kappa_f(x_i)$:

$$x_{i+1} = Ax_i + B\kappa_f(x_i) \in \mathcal{X}_f \quad \forall x_i \in \mathcal{X}_f$$

All state and input **constraints** are satisfied in \mathcal{X}_f :

$$\mathcal{X}_f \in X, \kappa_f(x_i) \in U \quad \forall x_i \in \mathcal{X}_f$$

- 3 Terminal cost is a continuous **Lyapunov function** s.t.

$$l_f(x_{i+1}) - l_f(x_i) \leq -l(x_i, \kappa_f(x_i)) \quad \forall x_i \in \mathcal{X}_f$$

Theorem 1. Under the previous assumptions, the closed-loop system under the MPC control law $u_0^*(x)$ is asymptotically stable and the set \mathcal{X}_f is positive invariant for

$$x(k+1) = Ax(k) + Bu_0^*(x(k))$$

Finite-horizon MPC may not satisfy constraints for all time!

Finite-horizon MPC may not be stable!

- An infinite-horizon provides stability and invariance.
- Infinite-horizon *faked* by forcing final state into an invariant set for which there exists invariance-inducing controller, whose infinite-horizon cost can be expressed in closed-form.
- Extends to non-linear systems, but compute sets is difficult!

2 Practical MPC

2.1 Steady-state Target Problem

- Reference is achieved by the target state x_s if $z_s = Hx_s = r$
- Target state should be a steady-state, i.e. $x_s = Ax_s + Bu_s$

$$\begin{aligned} x_s &= Ax_s + Bu_s \\ z_s &= Hx_s = r \end{aligned} \iff \begin{bmatrix} \mathbb{I} - A & -B \\ H & 0 \end{bmatrix} \begin{bmatrix} x_s \\ u_s \end{bmatrix} = \begin{bmatrix} 0 \\ r \end{bmatrix}$$

♯ solution $\rightarrow \min (Hx_s - r)^T Q_s (Hx_s - r)$ (closest x to r)

If \exists multiple feasible $u_s \rightarrow$ compute $\min u_s^T R_s u_s$ (cheapest)

$$\min_U |z_N - Hx_s|_{P_z}^2 + \sum_{i=1}^{N-1} |z_i - Hx_s|_{Q_z}^2 + |u_i - u_s|_R^2$$

2.2 Offset-free Reference Tracking

$$\begin{aligned} \Delta x_{k+1} &= x_{k+1} - x_s \\ &= A\Delta x_k + Bu_k - (Ax_s + Bu_s) \\ &= A\Delta x_k + B\Delta u_k \\ G_x x &\leq h_x \Rightarrow G_x \Delta x \leq h_x - G_x x_s \\ G_u u &\leq h_u \Rightarrow G_u \Delta u \leq h_u - G_u u_s \end{aligned}$$

Convergence

Assume feasible target with $x_s \in \mathcal{X}, u_s \in \mathcal{U}$, choose terminal weight $V_f(x)$ and constraint \mathcal{X}_f as in regulation case satisfying

$$V_f(x(k+1)) - V_f(x(k)) \leq -l(x(k), Kx(k))$$

$$\text{and } (A + BK)x \in \mathcal{X} \quad \forall x \in \mathcal{X}_f \text{ for both}$$

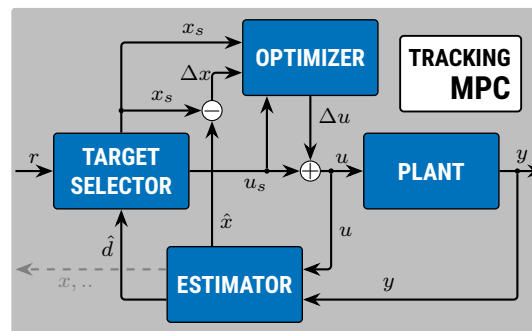
If in addition the target reference x_s, u_s is such that

$$x_s \oplus \mathcal{X}_f \subseteq \mathcal{X}, K\Delta x + u_s \in \mathcal{U}, \quad \forall \Delta x \in \mathcal{X}_f$$

then the closed loop system converges to the target reference.

Proof. Invariance under local control law is inherited from regulation case. Constraint satisfaction is provided by extra conditions and convergence comes from the asymptotic stability of the regulation problem: $\Delta x(k) \rightarrow 0$ for $k \rightarrow \infty$ \square

Terminal set use $\mathcal{X}_f^{\text{scaled}} = \alpha \mathcal{X}_f$ (s.t. constraints satisfied)



Disturbance Cancellation

Approach Model the disturbance, use the measurements and model to estimate the state and disturbance and find control inputs that use the disturbance estimate to remove offset.

Augmented Model

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + B_d d_k \\ y_k &= Cx_k + C_d d_k \end{aligned}$$

Constant disturbance $d_{k+1} = d_k$

Observable iff $\begin{bmatrix} A-I & B_d \\ C & C_d \end{bmatrix}$ has full rank (assuming $n_x = n_d$)

Observer For Augmented Model

$$\begin{bmatrix} \hat{x}_{k+1} \\ \hat{d}_{k+1} \end{bmatrix} = \begin{bmatrix} A & B_d \\ 0 & \mathbb{I} \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \hat{d}_k \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u_k + \begin{bmatrix} L_x \\ L_d \end{bmatrix} (Cx_k + C_d \hat{d}_k - y_k)$$

Error Dynamics \Rightarrow choose L s.t error dynamics converge to 0

$$\begin{bmatrix} x_{k+1} - \hat{x}_{k+1} \\ d_{k+1} - \hat{d}_{k+1} \end{bmatrix} = \left(\begin{bmatrix} A & B_d \\ 0 & \mathbb{I} \end{bmatrix} + \begin{bmatrix} L_x \\ L_d \end{bmatrix} \begin{bmatrix} C & C_d \end{bmatrix} \right) \begin{bmatrix} x_k - \hat{x}_k \\ d_k - \hat{d}_k \end{bmatrix}$$

Lemma 1. Steady-state of an asym. stable observer satisfies:

$$\begin{bmatrix} A - \mathbb{I} & B \\ C & 0 \end{bmatrix} \begin{bmatrix} \hat{x}_\infty \\ u_\infty \end{bmatrix} = \begin{bmatrix} -B_d \hat{d}_\infty \\ y_\infty - C_d \hat{d}_\infty \end{bmatrix} \quad (\text{for } n_y = n_d)$$

\Rightarrow Observer output $C\hat{x}_\infty + C_d \hat{d}_\infty$ tracks y_∞ without offset

Reference Tracking with Disturbance Cancellation

Goal Track constant reference: $Hy(k) = z(k) \rightarrow r, k \rightarrow \infty$

$$x_s = Ax_s + Bu_s + B_d \hat{d}_\infty$$

$$z_s = H(Cx_s + C_d \hat{d}_\infty) = r$$

$$\begin{bmatrix} A - \mathbb{I} & B \\ HC & 0 \end{bmatrix} \begin{bmatrix} x_s \\ u_s \end{bmatrix} = \begin{bmatrix} 0 \\ r - HC_d \hat{d} \end{bmatrix}$$

Offset-free Tracking - Main Result

Theorem 2. Assuming RHC recursively feasible, $n_d = n_y$, unconstrained for $k \geq j$, and the closed loop system

$$x(k+1) = Ax(k) + B\kappa(\cdot) + B_d d \quad \text{with } (\cdot) = (\hat{x}, \hat{d}, r)$$

$$\hat{x}(k+1) = (A + L_x C)\hat{x}(k) + (B_d + L_x C_d)\hat{d}(k) + B\kappa(\cdot) - L_x y(k)$$

$$\hat{d}(k+1) = L_d C\hat{x}(k) + (\mathbb{I} + L_d C_d)\hat{d}(k) - L_d y(k)$$

converges, then $z(k) = Hy(k) \rightarrow r$ as $k \rightarrow \infty$

2.3 Soft Constraints

Input constraints are dictated by physical constraints on the actuators and are **usually hard**

State/output constraints arise from practical restrictions on the allowed operating range and are **rarely hard**

Hard state/output constraints always lead to **complications in the controller implementation**

Soft Constrained MPC

$$\min_u \sum_{i=0}^{N-1} x_i^\top Q x_i + u_i^\top R u_i + l_\epsilon(\epsilon_i) + x_N^\top P x_N + l_\epsilon(\epsilon_N) \quad (2)$$

Quadratic penalty $l_\epsilon(\epsilon_i) = \epsilon_i^\top S \epsilon_i$ (e.g. $S = Q$)
+ linear norm penalty $l_\epsilon(\epsilon_i) = v|\epsilon_i|_1 / \infty$

Constraints

$$\begin{aligned} x_{i+1} &= A x_i + B u_i \\ H x_i &\leq k_x + \epsilon_i \\ H u_i &\leq k_u \\ \epsilon_i &\geq 0 \text{ slack variable} \end{aligned}$$

Original	$\min_z f(z)$	Softened	$\min_{z, \epsilon} f(z) + l_\epsilon(\epsilon)$
	s.t. $g(z) \leq 0$		s.t. $g(z) \leq \epsilon$ $\epsilon \geq 0$

Requirement on $l_\epsilon(\epsilon)$ If the original problem has a feasible solution z^* , then the softened problem should have the same solution z^* , and $\epsilon = 0$.

Theorem 3 (Exact Penalty Function). $l_\epsilon(\epsilon) = v \cdot \epsilon$ satisfies requirement for any $v > \lambda^*$ where λ^* is optimal Lagrange multiplier for original problem

3 Robust MPC

Uncertain System $x(k+1) = g(x(k), u(k), w(k)); \theta$

Robust Constraint Satisfaction

Idea Compute a set of tighter constraints such that if the nominal system meets these constraints, then the uncertain system will too. We then do MPC on the nominal system.

Goal Ensure constraints satisfied for the MPC sequence.

Disturbance reachable set $F_i = \bigoplus_{j=0}^{i-1} A^j \mathcal{W}$

Robust Open-Loop MPC

$$\min_U \sum_{i=0}^{N-1} l(x_i, u_i) + l_f(x_N) \quad \begin{aligned} x_0 &= x(k) \\ x_i &\in \mathcal{X} \ominus F_i \\ u_i &\in \mathcal{U} \\ x_N &\in \mathcal{X}_f \ominus F_N \end{aligned}$$

s.t. $x_{i+1} = A x_i + B u_i$

Closed Loop Robust MPC

Idea Separate the available control authority into two parts:

$$z(k+1) = A z(k) + B v(k)$$

steers noise-free *nominal* system to origin

$$u_i = K(x_i - z_i) + v_i$$

compensates for deviations, i.e. a *tracking* controller, to keep the real trajectory close to the nominal system.

\Rightarrow We fix the linear feedback controller K offline, and optimize over the nominal inputs $\{v_0, \dots, v_{N-1}\}$ and nominal trajectory $\{z_0, \dots, z_N\}$, which results in a convex problem.

$$e_{i+1} = x_{i+1} - z_{i+1} = (A + BK)e_i + w_i$$

3.1 Robust Constraint-Tightening MPC

Robust Constraint-Tightening MPC

$$\min_{Z, V} \sum_{i=0}^{N-1} l(z_i, v_i) + l_f(z_N) \quad \begin{aligned} z_0 &= x(k) \\ z_i &\in \mathcal{X} \ominus F_i \\ u_i &\in \mathcal{U} \ominus K(F_i) \\ z_N &\in \mathcal{X}_f^{ct} \ominus F_N \end{aligned}$$

subj. to $z_{i+1} = A z_i + B v_i$
 $F_0 := 0 \quad F_i := \mathcal{W} \oplus A_K \mathcal{W} \oplus \dots \oplus A_K^{i-1} \mathcal{W}$

Control Law $u(k) = v_0^* + K(x(k) - z_0) = v_0^*$

3.2 Robust Tube MPC

Idea Ignore noise and plan the nominal trajectory, bound maximum error at any time with RPI set $\mathcal{E} : \epsilon_i \in \mathcal{E} \epsilon_{i+1} \in \mathcal{E}$

Ideally \mathcal{E} is selected as the minimum RPI set F_∞

We know that the real trajectory stays 'nearby' the nominal one $x_i \in z_i \oplus \mathcal{E}$ because we plan to apply the controller $u_i = K(x_i - z_i) + v_i$ in the future.

(we won't actually do this, but it's a valid sub-optimal plan)

We must ensure that all possible state trajectories satisfy the constraints This is now equivalent to ensuring that $x_i \in z_i \oplus \mathcal{E}$ (address input constraints later)

What do we need to make this work?

Compute the set \mathcal{E} that the error will remain inside

Previously we wanted the **maximum robust invariant set**, or the largest set in which our terminal control law works.

We now want the **minimum robust invariant set**, or the smallest set that the state will remain inside despite the noise.

Modify constraints on nominal trajectory $\{z_i\}$

$$x_i \in z_i \oplus \mathcal{E} = \{z_i + e | e \in \mathcal{E}\}$$

Tube MPC

Cost function $J(Z, V) := \sum_{i=0}^{N-1} l(z_i, v_i) + l_f(z_N)$

Feasible set $\mathcal{Z}(x_0) := \left\{ \begin{aligned} z_{i+1} &= A z_i + B v_i \\ z_i &\in \mathcal{X} \ominus \mathcal{E} \\ v_i &\in \mathcal{U} \ominus K \mathcal{E} \\ z_N &\in \mathcal{X}_f \\ x_0 &\in z_0 \oplus \mathcal{E} \end{aligned} \right.$

Optimization $(V^*(x_0), Z^*(x_0)) = \arg\min_{V, Z} \{J(Z, V) | (Z, V) \in \mathcal{Z}(x_0)\}$

Control law $\mu_{\text{tube}}(x) := K(x - z_0^*(x)) + v_0^*(x)$

Theorem 4 (Robust Invariance of Tube MPC). The set $\mathcal{Z} := \{x | \mathcal{Z}(x) \neq \emptyset\}$ is a robust invariant set of the system $x(k+1) = A x(k) + B \mu_{\text{tube}}(x(k)) + w(k)$ subject to the constraints $x, u \in \mathcal{X} \times \mathcal{U}$.

Theorem 5 (Robust Stability of Tube MPC). The state $x(k)$ of the system $x(k+1) = A x(k) + B \mu_{\text{tube}}(x(k)) + w(k)$ converges to the limit of the set \mathcal{E} .

Tube MPC - Quick Summary

To implement tube MPC:

– **Offline** –

1. Stabilizing controller K so that $A + BK$ is (Schur) stable
2. Compute the minimal robust invariant set $E = F_\infty$ for the system $x(k+1) = (A + BK)x(k) + w(k), w \in \mathcal{W}^1$
3. Compute tightened constraints $\bar{\mathcal{X}} := \mathcal{X} \ominus \mathcal{E}, \bar{\mathcal{U}} := \mathcal{U} \ominus K \mathcal{E}$
4. Choose terminal weight function l_f and constraint \mathcal{X}_f satisfying assumptions*

– **Online** –

1. Measure / estimate state x
2. Solve optimization problem for $(V^*(x_0), Z^*(x_0))$
3. Set the input to $u = K(x - z_0^*(x)) + v_0^*(x)$

4 Stochastic MPC

5 Nonlinear MPC

6 Invariance

Definition 1 (Positively Invariant Set \mathcal{O}). For an autonomous or closed-loop system, the set \mathcal{O} is positively invariant if:

$$x(k) \in \mathcal{O} \Rightarrow x(k+1) \in \mathcal{O}, \quad \forall k \in \{0, 1, \dots\}$$

Definition 2 (Maximal Positively Invariant Set \mathcal{O}_∞). A set that contains all \mathcal{O} is the maximal positively invariant set $\mathcal{O}_\infty \subset \mathcal{X}$

Definition 3 (Pre-Sets). The set of states that in the dynamic system $x(k+1) = g(x(k))$ in one time step evolves into the target set S is the **pre-set** of $S \Rightarrow \text{pre}(S) := \{x | g(x) \in S\}$

Lemma 2. **Invariant Sets from Lyapunov Functions**

If $V : \mathbb{R}^n \rightarrow \mathbb{R}$ is a Lyapunov function for $x(k+1) = g(x(k))$, then $Y := \{x | V(x) \leq \alpha\}$ is an invariant set for all $\alpha \geq 0$

Proof. Lyapunov property $V(g(x)) - V(x) < 0$ implies that once $V(x(k)) \leq \alpha, V(x(j)) < \alpha, \forall j \geq k \rightarrow$ Invariance \square

Example System $x(k+1) = A x(k), A^\top P A - P \prec 0 \prec P$ and resulting Lyapunov function $V(x(k)) = x(k)^\top P x(k)$

Goal Find the largest α s.t the invariant set $Y_\alpha \in \mathcal{X}$

$$Y_\alpha := \{x | x^\top P x \leq \alpha\} \subset \mathcal{X} := \{x | F x \leq f\}$$

Equivalent to $\max_\alpha \alpha$ s.t. $h_{Y_\alpha}(F_i) \leq f_i \forall i \in \{1, \dots, n\}$

Theorem 6 (Geometric condition for invariance). Set \mathcal{O} is positively invariant set iff $\mathcal{O} \subseteq \text{pre}(\mathcal{O}) \Leftrightarrow \text{pre}(\mathcal{O}) \cap \mathcal{O} = \mathcal{O}$

Proof. **Necessary** if $\mathcal{O} \not\subseteq \text{pre}(\mathcal{O})$, then $\exists \bar{x} \in \mathcal{O}$ s.t $\bar{x} \notin \text{pre}(\mathcal{O}) \rightsquigarrow \bar{x} \in \mathcal{O}, \bar{x} \notin \text{pre}(\mathcal{O})$, thus \mathcal{O} not positively invariant

Sufficient if \mathcal{O} not positive invariant set, then $\exists \bar{x} \in \mathcal{O}$ s.t $g(\bar{x}) \notin \mathcal{O} \rightsquigarrow \bar{x} \in \mathcal{O}, \bar{x} \notin \text{pre}(\mathcal{O})$ thus $\mathcal{O} \not\subseteq \text{pre}(\mathcal{O})$ \square

6.1 Control Invariance

Definition 4 (Control Invariant Set). $\mathcal{C} \subseteq \mathcal{X}$ control invariant if

$$x(k) \in \mathcal{C} \Rightarrow \exists u(k) \in \mathcal{U} \text{ s.t } g(x(k), u(k)) \in \mathcal{C} \forall k$$

Definition 5 (Maximal Control Invariant Set \mathcal{C}_∞). A set that contains all \mathcal{C} is the maximal positively invariant set $\mathcal{C}_\infty \subset \mathcal{X}$

Intuition For all states in \mathcal{C}_∞ exists control law s.t constraints are never violated \rightsquigarrow **The best any controller could ever do**

Pre-set $\text{pre}(S) := \{x | \exists u \in \mathcal{U} \text{ s.t } g(x, u) \in S\}$

Set \mathcal{C} is control invariant iff: $\mathcal{C} \subseteq \text{pre}(\mathcal{C}) \Leftrightarrow \text{pre}(\mathcal{C}) \cap \mathcal{C} = \mathcal{C}$

Control Law from Control Invariant Set

Control law $\kappa(x(k))$ will **guarantee** that the system with control invariant set \mathcal{C} satisfies constraints **for all time** if

$$x(k+1) = g(x(k), u(k)) \rightarrow g(x, \kappa(x)) \in \mathcal{C} \forall x \in \mathcal{C}$$

We can use this fact to **synthesize** control law κ

$$\kappa(x) := \arg\min\{f(x, u) \mid g(x, u) \in \mathcal{C}\}$$

with f as any function (including $f(x, u) = 0$)

Does not ensure that system will converge
 Difficult because calculating control invariant sets is hard
MPC implicitly describes \mathcal{C} s.t easy to represent/compute

6.2 Robust Invariance

Definition 6 (Robust Positive Invariant Set $\mathcal{O}^{\mathcal{W}}$). For the autonomous system $x(k+1) = g(x(k), w(k))$, the set $\mathcal{O}^{\mathcal{W}}$ is robust positive invariant if:

$$x \in \mathcal{O}^{\mathcal{W}} \Rightarrow g(x, w) \in \mathcal{O}^{\mathcal{W}}, \quad \forall w \in \mathcal{W}$$

Given set Ω and dynamic system $x(k+1) = g(x(k), w(k))$,

$$\text{pre}^{\mathcal{W}}(\Omega) := \{x \mid g(x, w) \in \Omega \forall w \in \mathcal{W}\}$$

Definition 7 (Robust Pre-Sets). The set of states that in the dynamic system $x(k+1) = g(x(k), w(k))$ for all disturbance $w \in \mathcal{W}$ in one time step evolves into the target set Ω is the **pre-set** of $\Omega \Rightarrow \text{pre}^{\mathcal{W}}(\Omega) := \{x \mid g(x, w) \in \Omega \forall w \in \mathcal{W}\}$

Theorem 7 (Geometric condition for robust invariance). Set $\mathcal{O}^{\mathcal{W}}$ is robust positive invariant iff $\mathcal{O}^{\mathcal{W}} \subseteq \text{pre}^{\mathcal{W}}(\mathcal{O}^{\mathcal{W}})$

Minimum Robust Invariant Set

$$F_\infty = \bigoplus_{j=0}^{\infty} A_K^j \mathcal{W}, F_0 := \{0\} \Rightarrow F_n = F_{n+1} = F_\infty$$

6.3 Computing Invariant Sets and Pre-sets

System for Pre-Set Computation

$$x(k+1) = A x(k) + B u(k)$$

$$u(k) \in \mathcal{U} := \{u \mid G u \leq g\}$$

$$S := \{x \mid F x \leq f\}$$

Invariant Pre-Set

$$\text{pre}(S) := \{x \mid A x \in S\}$$

$$= \{x \mid F A x \leq f\}$$

Control Invariant Pre-Set

$$\begin{aligned} \text{pre}(S) &:= \{x \mid \exists u \in \mathcal{U}, A x + B u \in S\} \\ &= \{x \mid \exists u \in \mathcal{U}, F A x + F B u \leq f\} \\ &= \left\{x \mid \exists u \in \mathcal{U}, \begin{bmatrix} F A & F B \\ 0 & G \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} \leq \begin{bmatrix} f \\ g \end{bmatrix}\right\} \end{aligned}$$

This is a **projection** operation

System for Robust Pre-Set Computation

$$x(k+1) = A x(k) + w(k)$$

$$\Omega := \{x \mid F x \leq f\}$$

$$J^*(x(k)) = \min_z \begin{bmatrix} z^\top & x(k)^\top \end{bmatrix} \begin{bmatrix} \bar{H} & 0 \\ 0 & Q \end{bmatrix} \begin{bmatrix} z^\top & x(k)^\top \end{bmatrix}^\top$$

$$\text{s.t. } G_{in}z < w_{in} + E_{in}x(k) \quad G_{eq}z = E_{eq}x(k)$$

Construction of QP with substitution

Idea Substitute the state equations.

Step 1 Rewrite cost as

J(x(k)) = U^T H U + 2x(k)^T F U + x(k)^T Y x(k) = [U^T x(k)^T] [H_F^T F_Y^T] [U^T x(k)^T]^T

Step 2 Rewrite constraints compactly as GU ≤ w + E x(k)

G = [A_u 0 ... 0; 0 A_u ... 0; ...; 0 0 ... A_u; A_x B 0 ... 0; A_x A B A_x B ... 0; ...; A_r A^{N-1} B A_r A^{N-2} B ... A_r B], E = [0; 0; ...; -A_x; -A_x A; -A_x A^2; ...; -A_r A^N], w = [b_u; b_u; ...; b_x; b_x; b_x; ...; b_r]

Step 3 Rewrite constrained problem as

J*(x(k)) = min_U [U^T x(k)^T] [H_F^T F_Y^T] [U^T x(k)^T]^T subj. to GU ≤ w + E x(k)

9 Optimization

Mathematical Optimization Problem

Decision variable $x \in \mathbb{R}^n$
Objective function $f : \text{dom}(f) \rightarrow \mathbb{R}$
Inequality constraints g_i ($i \in \# \text{constraints}$)
Equality constraints h_i ($i \in \# \text{constraints}$)
Feasible set $\mathcal{X} := \{x | g(x) \leq 0, h(x) = 0\}$

minimize $f(x)$
subject to:
 $g_i(x) \leq 0$
 $h_i(x) = 0$

Feasible point $x \in \text{dom}(f)$ with $g_i(x) \leq 0, h_i(x) = 0$
Strictly feasible point x with strict inequality $g_i(x) < 0$
Optimal value f^* (or p^*) = $\inf\{f(x) | g_i(x) \leq 0, h_i(x) = 0\}$
 $f^* = +\infty$: OP infeasible, $f^* = -\infty$: OP unbound below
Optimizer set: $\text{argmin}_{x \in \mathcal{X}} f(x) := \{x \in \mathcal{X} | f(x) = f^*\}$

x^* is a **Global Minimum** if $f(x^*) \leq f(x)$
 x^* is a **Local Minimum** if $\exists \epsilon > 0$ s.t. $f(x^*) \leq f(x)$
 $\forall x \in \mathcal{X} \cap B_\epsilon(x^*)$, open ball with center x^* and radius ϵ

9.1 Convex Sets

Definition 11 (Convex Set). Set C is convex if and only if

theta x + (1 - theta) y in C, for all x, y in C, and theta in [0, 1]

Intersection C1, C2 cv => C1 intersect C2 convex (cv)

Image under affine map C subset R^n cv => {Ax + b | x in C} cv

Inverse loaM C subset R^m cv => {x in R^n | Ax + b in C} cv

Definition 12 (Hyperplanes). {x in R^n | a^T x = b}

Definition 13 (Halfspaces). {x in R^n | a^T x <= b}

can be open (strict inequality) or closed (non-strict inequality)

Definition 14 (Polyhedra). Intersection of finite number of closed halfspaces: polyhedra {x in R^n | A q x^n <= b q x^1, }

Definition 15 (Polytope). is a bounded polyhedron.

Definition 16 (Convex hull). for {v1, ..., vk} in R^d is:

co({v1, ..., vk}) := {x | x = sum_i lambda_i v_i, lambda_i >= 0, sum_i lambda_i = 1}

Definition 17 (Ellipsoid). set: {x | (x - x_c)^T A^-1 (x - x_c) <= 1} where x_c is center of ellipsoid, A > 0 (i.e. positive definite) (Semi-axis lengths are square roots of eigenvalues of A)

Definition 18 (Norm Ball). B_r(x) := {xi in R^n : ||xi - x||_p < r} where p defines the l_p norm, p = {1|2|...|infinity}

Theorem 11. Minkowski-Weyl

The following statements are equivalent for P subset R^d

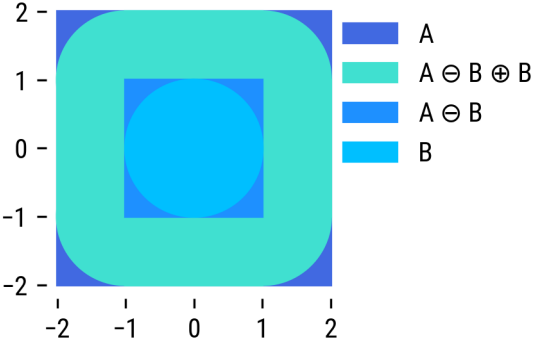
P is a polytope and there exists A, b s.t P = {x | Ax <= b}

P finitely generated, exists finite set {v_i} s.t P = co({v_1, ..., v_s})

Definition 19. Minkowski Sum for A, B subset R^n

A plus B := {x + y | x in A, y in B}

Example [a, b] plus [c, d] = [a + c, b + d]



Definition 20. Pontryagin Difference for A, B subset R^n

A minus B := {x | x + e in A, for all e in B}

Example [a, b] minus [c, d] = [a - c, b - d]

9.2 Convex Functions

Definition 21 (Convex Function). f : C_convex -> R is convex iff

f(theta x + (1 - theta) y) <= theta f(x) + (1 - theta) f(y), for all x, y in C and theta in [0, 1]

f is strictly convex if this inequality is strict.

Definition 22 (Epigraph). f : R^n -> R cv <=> epi(f) is cv set

epi(f) := {(x, t) in R^{n+1} | f(x) <= t}

Check Convexity f is convex if it is composition of simple convex function with convexity preserving operations or if

f : R^n -> R twice differentiable, partial^2 f / partial^2 x >= 0 for all x in R^n

g : R -> R with g(t) = f(x + tv) convex in t for all x, v in R^n

-> f convex (restriction to a line)

- the point wise maximum of convex functions is convex

- the sum of convex functions is convex

- f(Ax + b) is convex if f is convex

9.3 Optimality Conditions

Lagrange Duality

Consider $f^* = \inf_{x \in \mathbb{R}^n} f(x)$ s.t. $g(x) \leq 0, h(x) = 0$ (7)

Lagrangian $\mathcal{L}(x, \lambda, \nu) = f(x) + \lambda^T g(x) + \nu^T h(x)$

Dual Function $d(\lambda, \nu) = \inf_{x \in \mathbb{R}^n} \mathcal{L}(x, \lambda, \nu)$

Proposition 1 (Weak Duality). $d(\lambda, \nu) \leq f^*, \forall \lambda \geq 0, \nu \in \mathbb{R}^h$

Definition 23 (Constraint qualification). Slater's Condition holds if exists at least one strictly feasible point x (h(x) = 0, g(x) < 0)

Proposition 2 (Strong Duality). If Slater's condition holds and OP is convex => exists lambda >= 0, nu in R^{nh} s.t. d(lambda, nu) = f^*

KKT Conditions (Karush-Kuhn-Tucker)

Theorem 12 (KKT Conditions). If Slater's condition holds and (7) is convex -> x* in R^n is a minimizer of the primal (7) and (lambda* >= 0, nu*) in R^{ng} x R^{nh} is a maximizer of the dual <=> is equivalent to the following statements:

KKT-1 (Stationary Lagrangian) $\nabla_x \mathcal{L}(x^*, \lambda^*, \nu^*) = 0$

KKT-2 (primal feasibility) $g(x^*) \leq 0, h(x^*) = 0$

KKT-3 (dual feasibility) $\lambda^*, \nu^* \in \mathbb{R}^{nh} \geq 0$

KKT-4 (complementary slackness) $\lambda^{*T} g(x^*) = 0, \nu^{*T} h(x^*) = 0$

In addition we have: $\sup_{\lambda \geq 0, \nu \in \mathbb{R}^{nh}} q(\lambda, \nu) = \inf_{x \in C} f(x)$

Remark Without Slater, KKT1-4 still implies x* minimizes (7) and lambda, nu maximizes dual, but the converse is no longer true. There can be primal-minimizer/dual-maximizer not satisfy KKT.

9.4 Convex Optimization Problems

Theorem 13. For a convex optimization problem, any locally optimal solution is globally optimal (local optima are global optima).

Linear Programming minimize c^T x s.t. Ax - b >= 0, x >= 0

Step 1: L(x, lambda_1, lambda_2) = c^T x - lambda_1^T (Ax - b) - lambda_2^T x, lambda_i >= 0

Step 2: inf_{x in R^n} L = lambda_1^T b, if c - A^T lambda_1 - lambda_2 = 0, else -infinity

Step 3: Dual, maximize b^T lambda s.t. c - A^T lambda >= 0, lambda >= 0 (again LP)

Proposition 3. The optimal solution of a linear program (if it exists) lies always on the boundary of the feasible set and there exists an optimal solution that is a vertex of the feasible set.

Quadratic Programming convex if P = P^T positive semi-definite minimize 1/2 x^T P x + q^T x s.t. G x <= h, Ax = b

Second-Order Cone Program

minimize f^T x s.t. |A_i x + b| <= c_i^T x + d_i, F x = g

Second-order cone C_{n+1} = {(x, t) | x in R^n, t in R, |x| <= t}

|A_i x + b| <= c_i^T x + d_i <=> (A_i x + b, c_i^T x + d_i) in C_{n+1}

Semi-Definite Programming with symmetric F_i, X, A_i

minimize c^T x s.t. sum_{i=1}^n x_i F_i + G <= 0, Ax = b

10 Implementation, needed?

CFTOC problem is multiparametric quadratic program mp-QP

Two options to solve it:

- Iterative optimization methods

- Explicit solution

Let I := 1, ..., m be the set of constraint indices.

Definition 24 (Active Set). A(x) and its complement NA(x)

A(x) := {j in I : G_j z*(x) - S_j x = w_j}
NA(x) := {j in I : G_j z*(x) - S_j x < w_j}

Definition 25 (Critical Region). CR_A is set of parameters x

for which set A subset I of constraints i active at the optimum. For given x in K* let (A, NA) := (A(x), NA(X)). Then

CR_A := {x in K* : A(x) = A} (states share active set)

Online evaluation: Point location

Sequential search

Logarithmic search

Large Scale Convex Optimization

Definition 26 (Lipschitz continuity). q : R^n -> R^m is Lipschitz with constant L if: ||q(x) - q(y)|| <= L ||x - y|| for all x, y in R^n

f is Lipschitz (Lip) with constant L <=> ||nabla f(x)|| <= L

Theorem 14. C subset R^n non-empty closed (cl) convex set, y not in C -> exists a not equal 0, b in R s.t. a^T x + b < a^T y + b, for all x in C

Definition 27 (smoothness). f : R^n -> R is L-smooth (L-sm) if ||nabla f(x) - nabla f(y)|| <= L ||x - y|| for all x, y in R^n

Taylor -> f(y) <= f(x) + nabla f(x)^T (y - x) + L/2 ||x - y||^2

Definition 28 (strong convexity). f : R^n -> R is mu-strongly convex (mu-scv) if f(y) >= f(x) + nabla f(x)^T (y - x) + mu/2 ||x - y||^2

How to find mu/L, Spectra of Hessian nabla^2 f, min/max eigenvalue Gradient Descent

x_{k+1} = x_k - T nabla f(x_k) for k = (k_0, ..., k_N) given x_0, T

Line search optimal step nu_t* = argmin_{nu in R} f(x_t - nu nabla f(x_t))

Definition 29. prox_C(x) = argmin_{y in C} 1/2 ||x - y||^2 C subset R^n

Lemma 4. cl, cv C subset R^n -> ||prox_C(x) - prox_C(y)|| <= ||x - y|| <-> ||prox_C(x) - prox_C(y)||^2 <= (prox_C(x) - prox_C(y))^T (x - y)

Projected Gradient Descent

x_{k+1} = prox_C(x_k - T nabla f(x_k)), for x_0, k_0..N, T in (0, 2/L)

```
def get_next_u(y: Measurement, r: Reference):  
    """  
    System handler for offset-free tracking  
    """  
  
    # approximate state, disturbance  
    x, d = estimator(y)  
    # find steady state und generate delta  
    x_s, u_s = target_selector(x, r, d)  
    x_delta = x - x_s  
    # call solver with new parameter  
    u_delta = mpc_regulator(x_delta, x_s, u_s)  
    u = u_delta + u_s  
  
    return u
```

TRANSFORM P-Nj

l_infinity-Minimization

min_{x in R^m} ||x||_infinity
subj. to F x <= g

mint_{x, t}
subj. to -1_m t <= x <= 1_m t, F x <= g

l_1-Minimization

min_{x in R^m} ||x||_1
subj. to F x <= g

min_{x in R^m, t in R^m} 1_m^T t
subj. to -t <= x <= t, F x <= g