Model Predictive Control

Author: Alexander Jöhl

1 Linear Systems

A Model of a system is usually continuous and and nonlinear. But in MPC discrete and linear systems are needed. Nonlinear System:

$$\dot{x} = f(x, u)$$
$$y = g(x, u)$$

Linearize it at: $\dot{x}_s = f(x_s, u_x) = 0, y_s = g(x_s, u_s)$

$$\dot{x} - \dot{x}_s = \left.\frac{\partial f}{\partial x}\right|_{\substack{x=x_s \\ u=u_s}} (x - x_s) + \left.\frac{\partial f}{\partial u}\right|_{\substack{x=x_s \\ u=u_s}} (u - u_s)$$
$$y - y_s = \left.\frac{\partial g}{\partial x}\right|_{\substack{x=x_s \\ u=u_s}} (x - x_s) + \left.\frac{\partial g}{\partial u}\right|_{\substack{x=x_s \\ u=u_s}} (u - u_s)$$

Which gives us:

$$\Delta\dot{x} = A^c \Delta x + B^c \Delta u$$
$$\Delta y = C\Delta x + D\Delta u$$

The Δ's are usually omitted. Then we discretize the system:

$$x(t_{k+1}) = e^{A^c T_s} x(t_k) + \int_{t_k}^{t_{k+1}} e^{A^c (t_{k+1} - \tau)} B^c \, \mathrm{d}\tau \; u(t_k)$$
$$x(t_{k+1}) = Ax(t_k) + Bu(t_k)$$

Integrating from t_k to t_{k+1} is the same as from 0 to T_s. Other matrices stay the same.

1.1 Analysis of LTI discrete Systems

1.1.1 Stability

System is asymptotically stable if $\lim_{k\to\infty} x(k) = 0 \; \forall x(0)$. Necessary and sufficient condition: All eigenvalues of A are $|\lambda_i| < 1 \; \forall i$. Only for LTI discrete systems.
For general systems: Global Lyapunov Stability.
Consider Equilibrium point $x = 0$ of system $x(k+1) = f(x(k))$. If a function $V : \mathbb{R}^n \to \mathbb{R}$ exists, such that:

$$\|x\| \to \infty \Rightarrow V(x) \to \infty$$
$$V(0) = 0 \quad \text{and} \quad V(x) > 0 \; \forall x \neq 0$$
$$V(x(k+1)) - V(x(k)) < 0 \; \forall x \neq 0$$

then $x = 0$ is globally asymptotically stable. For linear systems we can take $V(x) = x^T P x$ and get the discrete time Lyapunov equation (solvable only if eigenvalues of A inside unit circle):

$$A^T P A - P = -Q, \; Q > 0$$

Can calculate infinite horizon cost to go with P:

$$\Psi(x(0)) = \sum_{k=0}^{\infty} x(k)^T Q x(k) = x(0)^T P x(0)$$

1.1.2 Controllability

System is controllable if it can be controlled from any state to any state in finite time. Controllable if C is full rank:

$$\mathrm{rank}(\mathcal{C}) = \mathrm{rank}\left(B \; AB \ldots \; A^{n-1}B\right) = n$$

1.1.3 Observability

For any state we can distinguish the initial state by the measurements.

$$\mathrm{rank}(\mathcal{O}) = \mathrm{rank}\left(C^T \; (CA)^T \ldots \; (CA^{n-1})^T\right)^T = n$$

2 LQR Control

General finite horizon optimal control problem:

$$J_0^*(x(0)) = \min_{U_0} p(x_N) + \sum_{k=0}^{N-1} q(x_k, u_k)$$
$$\begin{aligned} \text{subject to} \quad & x_{k+1} = g(x_k, u_k), \; k = 0, \ldots, N-1 \\ & h(x_k, u_k) \leq 0, \; k = 0, \ldots, N-1 \\ & x_n \in \mathcal{X}_f \\ & x_0 = x(0) \end{aligned}$$

Linear Quadratic Optimal Control for LTI discrete systems:

$$J_0^*(x(0)) = \min_{U_0} \; x_N^T P x_N + \sum_{k=0}^{N-1} x_k^T Q x_k + u_k^T R u_k$$
$$\begin{aligned} \text{subject to} \quad & x_{k+1} = Ax_k + Bu_k, \; k = 0, \ldots, N-1 \\ & x_0 = x(0) \end{aligned}$$

2.1 Finite Time Horizon

2.1.1 Batch Approach

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

Simplified: $\mathcal{X} = \mathcal{S}^x x(0) + \mathcal{S}^u U_0$. Then define cost function

$$J_0 x(0), U_0 = \mathcal{X}^T \bar{Q} \mathcal{X} + U_0^T \bar{R} U_0$$

With $\bar{Q} = \mathrm{blockdiag}(Q, \ldots, Q, P)$, $\bar{R} + \mathrm{blockdiag}(R, \ldots, R)$ we get the optimal input sequence and optimal cost:

$$\begin{aligned} U_0^*(x(0)) =& -(\mathcal{S}^{uT} \bar{Q} \mathcal{S}^u + \bar{R})^{-1} \mathcal{S}^{uT} \bar{Q} \mathcal{S}^x x(0) \\ J_0^*(x(0)) =& x(0)^T \Big(\mathcal{S}^{xT} \bar{Q} \mathcal{S}^x - \ldots \\ & \ldots \mathcal{S}^{xT} \bar{Q} \mathcal{S}^u (\mathcal{S}^{uT} \bar{Q} \mathcal{S}^u - \bar{R})^{-1} \mathcal{S}^{uT} \bar{Q} \mathcal{S}^x\Big) x(0) \end{aligned}$$

2.1.2 Recursive Approach

$$\begin{aligned} u^*(k) =& -(B^T P_{k+1} B + R)^{-1} B^T P_{k+1} A x(k) \\ J_k^*(x(k)) =& x(k)^T P_k x(k) \\ P_k =& A^T P_{k+1} A + Q - \ldots \\ & \ldots A^T P_{k+1} B \left(B^T P_{k+1} B + R\right)^{-1} B^T P_{k+1} A \end{aligned}$$

Is a feedback controller as opposed to the Batch Approach. Last equation is the Riccati Difference Equation.

2.2 Infinite Time Horizon

Take above equations from the recursive approach and replace N, k and $k+1$ with ∞.

3 Uncertainty Modeling

System with $w(k)$ as noise/disturbance input:

$$x_p(k+1) = A_p x(k) + B_p u(k) + F_p w(k)$$
$$y(k) = C_p x_p(k) + G_p w(k)$$

Stochastic Process (colored noise) with $\varepsilon(k)$ as white noise:

$$x_w(k+1) = A_w x_w(k) + B_w \varepsilon(k)$$
$$w(k) = C_w x_w(k) + \varepsilon(k)$$

If mean of noise shifts, integrate $\varepsilon(k)$ and use $\varepsilon_{\mathrm{int}}(k)$ as input.

4 State Estimation

Want to observe the state of a linear system with disturbance noise ε_1 and measurement noise ε_2 with variances R_1, R_2.
Estimator structure:

$$\hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1} + Bu_{k-1}$$
$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k (y(k) - C\hat{x}_{k|k-1})$$

Error dynamics:

$$x_{k|k}^e = (A - K_k CA)\, x_{k-1|k-1}^e + (I - K_k C)\, \varepsilon_1(k-1) - K_k \varepsilon_2(k)$$

Errors go to zero (stable) if $|\mathrm{eig}(A - KCA)| < 1$

4.1 Kalman Filter

Initialize estimate $\hat{x}_{0|0}$ and $P_{0|0}$. Compute filter gain K_k and error covariance matrix $P_{k|k}$ (online or in advance):

$$\begin{aligned} P_{k|k-1} &= AP_{k-1|k-1}A^T + R_1 \\ K_k &= P_{k|k-1}C^T(CP_{k|k-1}C^T + R_2)^{-1} \\ P_{k|k} &= (I - K_kC)P_{k|k-1} \end{aligned}$$

Compute the a priori estimate:

$$\hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1} + Bu(k-1)$$

Get measurement $y(k)$ and compute new estimate:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(y(k) - C\hat{x}_{k|k-1})$$

Increment k.
P_∞ satisfies ARE:

$$P_\infty = AP_\infty A^T - AP_\infty C^T \left(CP_\infty C^T + R_2\right)^{-1} CP_\infty A^T + R_1$$

5 Convex Optimization

General Optimization Problem:

$$\begin{aligned} \min_{x\in\mathcal{X}} \quad & f_0(x) \\ \text{subject to :} \quad & f_i(x) \le 0 \quad i = 1, \dots, m \\ & h_i(x) = 0 \quad i = 1, \dots, p \end{aligned}$$

5.1 Convexity

Convex set $\mathcal{X} \Leftrightarrow \lambda x + (1-\lambda)y \in \mathcal{X},\ \forall \lambda \in [0,1],\ \forall x, y \in \mathcal{X}$
Convex function $f \Leftrightarrow \mathrm{dom}(f)$ convex and
$f(\lambda x + (1-\lambda)y) \le \lambda f(x) + (1-\lambda) f(y)$

5.2 Convex Optimization Problem

$$\begin{aligned} \min_{x\in\mathcal{X}} \quad & f_0(x) \\ \text{subject to :} \quad & f_i(x) \le 0 \quad i = 1, \dots, m \\ & a_i^T x = b_i \quad i = 1, \dots, m \end{aligned}$$

With $f_i,\ i = 0, \dots, m$ convex.
Linear Program if $f_i,\ i = 0, \dots, m$ affine.
Quadratic Program if f_0 quadratic and $f_i,\ i = 1, \dots, m$ affine.

5.3 Lagrangian Dual Function

$$\begin{aligned} g(\lambda, \nu) &= \inf_{x\in\mathcal{X}} L(x, \lambda, \nu) \\ &= \inf_{x\in\mathcal{X}} \left[f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x) \right] \end{aligned}$$

Dual Problem:

$$\begin{aligned} \max_{\lambda,\nu} \quad & g(\lambda, \nu) \\ \text{subject to :} \quad & \lambda \ge 0 \end{aligned}$$

Is a convex problem, optimal value is $d^* \le p^*$. In a convex problem with Slater condition (strict feasibility) fulfilled:

$$\{x \mid Ax = b, f_i(x) < 0,\ \forall i \in \{1, \dots, m\}\} \neq \emptyset$$

Then $p^* = d^*$

5.4 Karush-Kuhn-Tucker Conditions (KKT)

- Primal Feasibility:

$$\begin{aligned} f_i(x^*) \le 0 \quad i = 1, \dots, m \\ h_i(x^*) = 0 \quad i = 1, \dots, p \end{aligned}$$

- Dual Feasibility: $\lambda^* \ge 0$
- Complementary Slackness:

$$\lambda_i^* \cdot f_i(x^*) = 0 \qquad i = 1, \dots, m$$

- Stationarity:

$$\begin{aligned} &\nabla_x L(x^*, \lambda^*, \nu^*) = 0 = \\ &\nabla f_0(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla f_i(x^*) + \sum_{i=1}^{p} \nu_i^* \nabla h_i(x^*) \end{aligned}$$

5.5 Sensitivity

Perturbed optimization problem (u_i, v_i instead of 0):

$$\begin{aligned} \min_{x\in\mathcal{X}} \quad & f_0(x) \\ \text{subject to :} \quad & f_i(x) \le u_i \quad i = 1, \dots, m \\ & h_i(x) = v_i \quad i = 1, \dots, p \end{aligned}$$

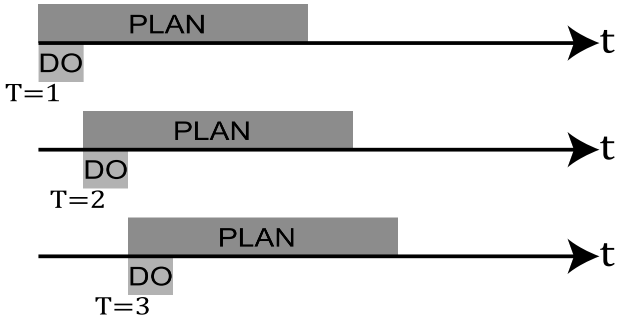
Weak duality for perturbed problem implies (strong duality in unperturbed case assumed):

$$p^*(u, v) \ge p^*(0, 0) - u^T \lambda^* - v^T \nu^*$$

Optimal values can change a lot or little depending on the sign of u_i, v_i.

6 Model Predictive Control

Receding Horizon Control and Feedback: Plan next N control actions, but only use first action, repeat every step.



General Finite Horizon Optimal Control Problem:

$$\begin{aligned} V_N(x) = \min_{\{u_i\}_{i=0}^{N-1}} & \sum_{i=0}^{N-1} l(x_i, u_i) \\ \text{subject to :} \quad & x_i \in \mathbb{X},\ u_i \in \mathbb{U} \quad \forall i \in \{0, ..., N-1\} \\ & x_{i+1} = f(x_i, u_i), \quad x_0 = x \end{aligned}$$

Optimization has to be done online, recursive feasibility not guaranteed, stability not guaranteed.

6.1 Standard Linear MPC Problem

$$\begin{aligned} V(x) = \min_{u} & \sum_{i=0}^{N-1} x_i^T Q x_i + u_i^T R u_i \\ \text{subject to :} \quad & x_{i+1} = Ax_i + Bu_i,\ x_0 = x \\ & x_i \in \mathbb{X}, u_i \in \mathbb{U} \quad \forall i \in \{0, ..., N-1\} \end{aligned}$$

Stack states and inputs into big vectors:

$$\mathbf{x} = (x_0; x_1; \dots; x_{N-1}), \qquad \mathbf{u} = (u_0; u_1; \dots; u_{N-1})$$

$$\mathcal{A} = \begin{bmatrix} I \\ A \\ A^2 \\ \vdots \\ A^{N-1} \end{bmatrix}, \qquad \mathcal{B} = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 \\ B & 0 & 0 & \dots & 0 \\ AB & B & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ A^{N-2}B & \dots & AB & B & 0 \end{bmatrix}$$

$$\mathbf{x} = \mathcal{A}x_0 + \mathcal{B}\mathbf{u}$$

Correspondingly the cost:

$$\begin{aligned} &\mathcal{Q} = \mathrm{diag}(Q, \dots, Q), \quad \mathcal{R} = \mathrm{diag}(R, \dots, R) \\ &V(x) = \mathbf{x}^T \mathcal{Q} \mathbf{x} + \mathbf{u}^T \mathcal{R} \mathbf{u} \end{aligned}$$

The constraints:

$$\mathcal{E}_u \mathbf{u} \le \mathcal{F}_u, \quad \mathcal{E}_x \mathbf{x} \le \mathcal{F}_x$$

Complete problem (QP) with $\mathbf{x}$ eliminated:

$$\begin{aligned} \min_{\mathbf{u},\mathbf{x}} \quad & \mathbf{u}^T \left[\mathcal{R} + \mathcal{B}^T \mathcal{Q} \mathcal{B}\right] \mathbf{u} + 2\mathbf{u}^T \mathcal{B}^T \mathcal{Q} \mathcal{A} x_0 \\ \text{subject to :} \quad & \begin{pmatrix} \mathcal{E}_u \\ \mathcal{E}_x \mathcal{B} \end{pmatrix} \mathbf{u} \leq \begin{pmatrix} \mathcal{F}_u \\ \mathcal{F}_x - \mathcal{E} \mathcal{A} x_0 \end{pmatrix} \end{aligned}$$

Can also have linear cost (LP) (minimize max value, minimize sum). Advantages, Disadvantages of QP vs LP:

- LP: easy to compute, QP: a little bit harder
- LP: non-unique solutions, QP: always unique solution
- LP: far from origin, conservative, QP: far from origin, large input
- LP: close to origin, discontinuity, dead-beat behaviour, QP: smooth
- LP: hard to tune
- QP: somewhat like minimizing energy/power

6.2 Reference Tracking

Regulation: reject disturbances around given fix point, *Tracking*: make $y(k)$ follow a reference $r(k)$. If reference unknown for future times, it is usually assumed to be constant over prediction horizon, if known preview control can be used.
Need steady state state and input:

$$\begin{bmatrix} (I-A) & -B \\ C & 0 \end{bmatrix} \begin{pmatrix} x_{ss} \\ u_{ss} \end{pmatrix} = \begin{pmatrix} 0 \\ r \end{pmatrix}$$

Plug those into cost (rest stays the same):

$$\begin{aligned} V(x) = \min_{\mathbf{u}} \sum_{i=0}^{N-1} & (x_i - x_{ss})^T Q (x_i - x_{ss}) + \ldots \\ & \ldots (u_i - u_{ss})^T R (u_i - u_{ss}) \end{aligned}$$

6.3 Stability and Feasibility

MPC is not guaranteed to be stabilizing, the optimization may not remain feasible. Remedies:

- derive lower bound on N, such that stability is guaranteed for all $Q \succeq 0,\ R \succ 0$
- additional constraint to ensure feasibility and stability
- check stability and feasibility of designed system a-posteriori

6.3.1 Ensure stability

- infinite prediction horizon: $N \to \infty$
- terminal state constraint: $x_N = 0$ or $x_N \in \mathbb{X}_N$
- contraction constraint: $\|x_1\| \leq \alpha \|x_0\|, \quad \alpha < 1$
- use cost function as a Lyapunov function

6.3.2 General stability method for MPC

$$\begin{aligned} \min_{\mathbf{u}} \quad & \sum_{i=0}^{N-1} l(x_i, u_i) + \mathbf{\Psi}(x_N) \\ \text{subject to :} \quad & x_i \in \mathbb{X}, u_i \in \mathbb{U}\ \forall i \in \{0, \ldots, N-1\} \\ & x_{i+1} = f(x_i, u_i), \quad x_0 = x \\ & x_N \in \mathbb{X}_N \end{aligned}$$

And some additional controller $u = K(x)$. Terminal constraint have to satisfy: $\mathbb{X}_N \subset \mathbb{X},\ x \in \mathbb{X}_N \Rightarrow K(x) \in \mathbb{U}$ and $x \in \mathbb{X}_N \Rightarrow f(x, K(x)) \in \mathbb{X}_N$.
Terminal Lyapunov condition:
$\Psi(f(x, K(x))) - \Psi(x) \leq -l(x, K(x))\ \forall x \in \mathbb{X}_N$

6.3.3 Linear MPC Stability

Assume a linear, constrained, possibly unstable system with quadratic cost. Terminal controller $K(x) = Kx$, Terminal state weight $\Psi(x) = x^T P x$, found by Riccati Equation:

$$(A + BK)^T P (A + BK) - P = -Q - K^T R K$$

Terminal set: an invariant set $\mathbb{X}_N$ for $x_{k+1} = (A + BK)x_k$.

7 MPC: Explicit Solution

Optimization required at each timestep, lots of computational effort. Move it offline.

7.1 Parametric Programming

$$\begin{aligned} & f^*(\theta) = \inf_z f(z, \theta) \\ \text{subject to :} \quad & g(z, \theta) \leq 0 \end{aligned}$$

$z \in \mathbb{R}^n$ is the optimization variable and $\theta \in \mathbb{R}^n$ is the parameter. Can do sensitivity analysis or find solutions depending on the parameter. Then we get critical regions in the θ feasible space ($\mathcal{X}$), where the KKT conditions do not change.

7.2 mpLP

The primal and the dual problem:

$$\begin{aligned} & J^*(\theta) = \min_z J^*(z, \theta) = c^T z \\ \text{sub. to :} \quad & Gz \leq W + S\theta \end{aligned} \quad \Leftrightarrow \quad \begin{aligned} & \max_\pi (W + S\theta)^T \pi \\ \text{sub. to :} \quad & G^T \pi = c \\ & \pi \leq 0 \end{aligned}$$

Solve the problems for some $\theta = \theta_0$ obtaining $z^*(\theta),\ \pi^*(\theta)$. Further obtain the set of active (inactive) constraints ($\mathcal{I} = \{1, \ldots, q\}$):

$$\begin{aligned} \mathcal{A}(\theta) &= \{i \in \mathcal{I} | \forall z : J(z, \theta) = J^*(\theta) \Rightarrow G_i z - S_i \theta - W_i = 0\} \\ \mathcal{N}(\theta) &= \{i \in \mathcal{I} | \exists z : J(z, \theta) = J^*(\theta) \wedge G_i z - S_i \theta - W_i < 0\} \end{aligned}$$

Then we compute the optimizer $z^*(\theta)$ and the critical region $\mathcal{CR}_0$:

$$\begin{aligned} z^*(\theta) &= G_{\mathcal{A}}^{-1} S_{\mathcal{A}} \theta + G_{\mathcal{A}}^{-1} W_{\mathcal{A}} = F_0 \theta + g_0 \\ \mathcal{CR}_0 &= \{\theta | (G_{\mathcal{N}} F_0 - S_{\mathcal{N}}) \theta < W_{\mathcal{N}} - G_{\mathcal{N}} g_0\} \end{aligned}$$

Replace θ_0 with some $\theta \in \mathcal{X} \setminus \mathcal{CR}_0$. Repeat until all of $\mathcal{X}$ is explored.

8 MPC: Hybrid Systems

System consisting of *continuous dynamics* and *discrete events* (states assume discrete values), often boolean variables p_i which are represented by $\delta_i = \{0, 1\}$.

8.1 Mixed Logical Dynamical Hybrid Model (MLD)

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

8.2 Piecewise Affine Systems (PWA)

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

$$\{\mathcal{D}^i\}_{i=1}^D = \left\{ \begin{bmatrix} x_k \\ u_k \end{bmatrix} | P_x^i x_k + P_u^i u_k \leq P_c^i \right\}$$

Affine dynamics for each region:

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

8.3 MPC for Hybrid Systems

$$J^*(x) = \min_U\ l_N(x_N) + \sum_{k=0}^{N-1} l(x_k, u_k, \delta_k, z_k)$$

$$\text{s.t} \begin{cases} x_{k+1} = A x_k + B_1 u_k + B_2 \delta_k + B_3 z_k \\ E_2 \delta_k + E_3 z_k \leq E_4 x_k + E_1 u_k + E_5 \\ x_N \in \mathcal{X}_f \end{cases}$$

9 Numerical Optimization Methods

$$\begin{aligned} &\text{repeat } x_{i+1} = \Psi(x_i, f, \mathbb{Q}),\ i = 0, 1, \dots, m-1 \\ &\text{until } |f(x_m) - f(x^*)| \leq \epsilon \quad \text{and} \quad \text{dist}(x_m, \mathbb{Q}) \leq \delta \end{aligned}$$

9.1 Unconstrained Optimization

9.1.1 Gradient Methods

L-smoothness:

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\| \ \forall x, y \in \mathbb{R}^n$$

Is the same as an existing quadratic function upper bound for f:

$$f(x) \leq f(y) + \nabla f(y)^T(x - y) + \frac{L}{2}\|x - y\|^2 \ \forall x, y \in \mathbb{R}^n$$

Gradient method:

$$\begin{aligned} \text{Set} \quad & x_0 \\ \text{Repeat} \quad & x_{i+1} = x_i - \frac{1}{L}\nabla f(x_i) \text{ for } i = 0, \dots, m-1 \\ \text{Until} \quad & f(x_m) - f(x^*) \leq \epsilon_1 \text{ or } \|x_m - x_{m-1}\| \leq \epsilon_2 \end{aligned}$$

Fast Gradient Method:

$$\begin{aligned} \text{Set} \quad & x_0, y_0 = x_0 \text{ and } \alpha_0 = (\sqrt{5} - 1)/2 \\ \text{Repeat} \quad & x_{i+1} = y_i - \frac{1}{L}\nabla f(y_i) \\ & \alpha_{i+1} = \alpha_i\left(\sqrt{\alpha_i^2 + 4} - \alpha_i\right)/2 \\ & \beta_i = \frac{\alpha_i(1 - \alpha_i)}{\alpha_i^2 + \alpha_{i+1}} \\ & y_{i+1} = x_{i+1} + \beta_i(x_{i+1} - x_i) \text{ for } i = 0, \dots, m-1 \\ \text{Until} \quad & f(x_m) - f(x^*) \leq \epsilon_1 \text{ or } \|x_m - x_{m-1}\| \leq \epsilon_2 \end{aligned}$$

Strong convexity, lower bound with quadratic function for f:

$$f(x) \geq f(y) + \nabla f(y)^T(x - y) + \frac{\mu}{2}\|x - y\|^2 \ \forall x, y \in \mathbb{R}^n$$

Gives condition number $\kappa = L/\mu$. Lower κ results in faster convergences speed $\rightarrow$ Preconditioning: Do a variable transformation $x = Py$ with P invertible such that the new κ is lower.

9.1.2 Newton's method

Idea: Minimize 2nd order approximation of f. Formula:

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

How to set h_i? Compute best h_i (exact method):

$$h_i^* = \arg\min_{h>0} f(x_i + h_i\Delta x_{nt})$$

Or find a h_i, which decreases f by some percent (inexact, backtracking):

$$\begin{aligned} & \alpha \in (0, 0.5) \text{ and } \beta \in (0, 1) \\ \text{Initialize} \quad & h_i = 1 \\ \text{While} \quad & f(x_i + h_i\Delta x_{nt}) > f(x_i) + \alpha h_i \nabla f(x_i)^T \Delta x_{nt} \\ \text{Do} \quad & h_i \leftarrow \beta h_i \end{aligned}$$

9.2 Constrained Optimization

$$\begin{aligned} & \min f(x) \\ & \text{subject to } x \in \mathbb{Q} \end{aligned}$$

With f and $\mathbb{Q}$ convex and L-smooth.

9.2.1 Gradient Methods

Use unconstraint gradient methods, but project each update x_{i+1} onto $\mathbb{Q}$:

$$x_{i+1} = \pi_{\mathbb{Q}}(x_i - h_i\nabla f(x_i))$$

If projection is easy to compute, fast algorithm. If projection is not easy to compute, solve dual problem instead (maybe slow).

9.2.2 Interior Point Methods

$$\begin{aligned} & \min f(x) \\ & \text{s.t. } g_i(x) \leq 0,\ i = 1, \dots, m \end{aligned}$$

f, g_i convex and twice continuous differentiable, problem strictly feasible. Idea: reformulate problem as unconstrained problem.
Barrier Method:

$$\begin{aligned} & \min f(x) \\ & \text{s.t. } g_i(x) \leq 0,\ i = 1, \dots, m \end{aligned} \Leftrightarrow \begin{aligned} & \min \ f(x) + \kappa\phi(x) \\ & \phi(x) = \sum_{i=1}^m I_-(g_i(x)),\ \kappa = 1 \end{aligned}$$

$\phi(x)$ is the indicator function, with $I_- = 0$ if $u \leq 0$ and ∞ otherwise. This function can be approximated by a logarithmic barrier:

$$\phi(x) = -\sum_{i=1}^m \log(-g_i(x))$$

As $\kappa \rightarrow 0$, approximation improves.
<u>Barrier Interior Point Method</u> (require strictly feasible $x_0, \kappa_0, \mu > 1, \epsilon > 0$):

1. $x^*(\kappa_i) = \min_x f(x) + \kappa_i\phi(x)$ starting from x_{i-1}
2. $x_i := x^*(\kappa_i)$
3. if $m\kappa_i < \epsilon$ STOP
4. $\kappa_{i+1} := \kappa/\mu$

Step 1 is usually solved with Newton's Method:

$$(\nabla^2 f(x) + \kappa\nabla^2\phi(x))\Delta x_{nt} = -\nabla f(x) - \kappa\nabla\phi(x)$$

With additional equality constraints $Cx = d$:

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

<u>Primal-Dual Interior Point Method</u>:

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

Relaxed KKT conditions, solve this system and repeat with decreased κ. This method allows for infeasible start.