Direct Approach for Numerical Optimal Control

- Discretization
- Direct Single Shooting
- Direct Multiple Shooting
- Direct Collocation

Problem Formulation

- Optimal control problem in continuous time

$$\min_{x(\cdot),u(\cdot)} \quad \underbrace{\int_{t_0}^{t_f} \ell(x(t),u(t))dt}_{\textsc{Lagrangian term}} + \underbrace{\mathcal{M}(x(t_f))}_{\textsc{Mayer term}}$$

subject to

$$\begin{cases} \dot{x}(t) & = & f(x(t),u(t)), \quad t \in [t_0, t_f] & \textsc{Dynamic equation} \\ 0 & = & x(t_0) - \hat{x} & \textsc{Initial condition} \\ 0 & \geq & h(x(t),u(t)), \quad t \in [t_0, t_f] & \textsc{Path constraints} \\ 0 & \geq & r(x(t_f)) & \textsc{Terminal constraints} \end{cases}$$

- Main idea of direct optimal control

discretize, then optimize

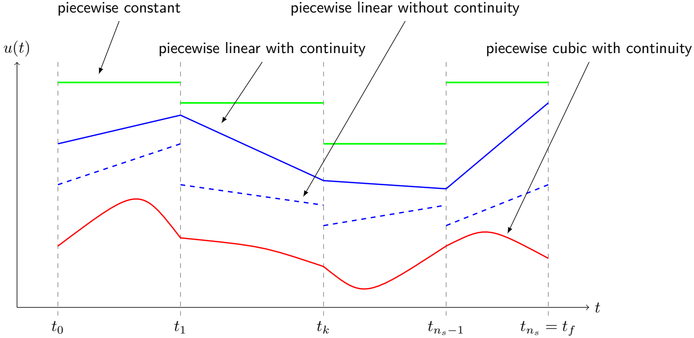
Contents

Input Discretization

- Parametrize $u(\cdot)$ by finitely many parameters u_k, $k = 1, ..., N_{\text{opt}}$,

$$u(t) = \sum_{k=1}^{N_{\text{opt}}} u_k \phi_k(t),\ \phi_k \text{ basis functions.}$$

- Example of basis functions



piece-wise constant input parametrizations commonly used

Dynamics Discretization

Main idea:

$$\begin{cases} \dot{x}(t) & = f(x(t), u(t)),\ t \in [t_k, t_{k+1}] \\ x(t_k) & = x_k \\ u(t) & \text{with piece-wise basis} \end{cases} \implies x_{k+1} = \xi(x_k, u_k)$$

- Taylor model based integrator;
- Explicit Runge-Kutta integrator:
  - Euler's method;
  - Heun's method;
  - RK 4;
  - ...
- Implicit Runge-Kutta integrator.

Objective Discretization

- Consider piece-wise constant input parametrizations.
- **Direct discretization** with time grid $\{t_k\}_0^N$ and constant $\Delta t = t_{k+1} - t_k$

$$\int_{t_0}^{t_f} \ell(x(t), u(t))dt \approx \sum_{k=0}^{N-1} \ell(x_k, u_k) \ \text{ with } \ \begin{cases} t_N = t_f \\ u(t) = u_k, \ t \in [t_k, t_{k+1}]. \end{cases}$$

- **Indirect discretization** defines the augmented state $\tilde{x} = (x, z)$ with

$$\dot{\tilde{x}}(t) = \begin{pmatrix} x(t) \\ z(t) \end{pmatrix} = F(\tilde{x}(t), u(t)) = \begin{bmatrix} f(x(t), u(t)) \\ \ell(x(t), u(t)) \end{bmatrix}$$

such that the Lagrangian term is transferred to a Mayer term

$$\int_{t_0}^{t_f} \ell(x(t), u(t))dt = z(t_f).$$

Then, integrate F.

Contents

Direct Single Shooting

Consider

- uniform grid $\{t_k\}_0^N$ with $t_N = t_f$ and constant $\Delta t = t_{k+1} - t_k$;
- piece-wise constant input parameterization, i.e., $u(t) = u_k$, $t \in [t_k, t_{k+1}]$.

Discretized OCP:

$$\min_{x_0, U_N} \quad \sum_{k=0}^{N-1} \ell(X_k(x_0, U_k), u_k) + \mathcal{M}(X_N(x_0, U_N))$$

subject to

$$\begin{cases} 0 &= x_0 - \hat{x} & \textsc{Initial condition} \\ X_k(x_0, U_k) &= \xi(\xi(...\xi(\xi(x_0, u_0), u_1), ...), u_{k-1}) & \textsc{k-step model} \\ 0 &\geq h(X_k(x_0, U_k), u_k) & \textsc{Path constraints} \\ 0 &\geq r(X_N(x_0, U_N)) & \textsc{Terminal constraints} \end{cases}$$

with $U_k = (u_0, u_1, ..., u_{k-1})$ for all $k \in \{1, ..., N\}$.

Direct Single Shooting

Remark:

- OCP transformed into NLP on one shooting interval
- $n_u \cdot N + n_x$ decision variables
- Constraints are enforced at discretization points only
- **Unstable systems require good initial guesses**
- Other variants of single shooting rely on variable step-size integrators (direct sequential single shooting); not discussed here
- For linear-quadratic MPC also known as condensing

Contents

Direct Multiple Shooting

Consider

- uniform grid $\{t_k\}_0^N$ with $t_N = t_f$ and constant $\Delta t = t_{k+1} - t_k$;
- piece-wise constant input parameterization, i.e., $u(t) = u_k$, $t \in [t_k, t_{k+1}]$.

Discretized OCP:

$$\min_{X,U} \quad \sum_{k=0}^{N-1} \ell(x_k, u_k) + \mathcal{M}(x_N)$$

subject to

$$\begin{cases} 0 = x_0 - \hat{x} & \textsc{Initial condition} \\ x_{k+1} = \xi(x_k, u_k),\ k \in \{0, 1, ..., N-1\} & \textsc{Discrete-time dynamic} \\ 0 \geq h(x_k, u_k),\ k \in \{0, 1, ..., N-1\} & \textsc{Path constraints} \\ 0 \geq r(x_N) & \textsc{Terminal constraints} \end{cases}$$

with $X = (x_0, x_1, ..., x_N)$ and $U = (u_0, u_1, ..., u_{N-1})$.

Direct Multiple Shooting

Remark:

- OCP transformed into NLP on multiple shooting intervals
- $(n_x + n_u) \cdot N + n_x$ decision variables
- ODE is satisfied upon convergence of NLP solver ($\rightarrow$ simultaneous approach)
- Constraints are enforced at discretization points only
- **Handles unstable systems much better than single shooting**
- Workhorse method for this course (exercises and projects)
- In case of convergence problems, initialize with feasible trajectory
- Other variants of multiple shooting rely on variable step-size integrators; not discussed here

Contents

Parameterization of Controls via Polynominals

- N-stage time splitting: $[t_0, t_f] \to \{[t_0, t_1], ..., [t_{N-1}, t_N]\}$, $t_N = t_f$.
- In each interval $[t_k, t_{k+1}]$ approximate $u(\cdot)$ by polynomial functions

$$[u(t)]_j = [U_k(t, \omega_k)]_j = \sum_{i=0}^{M_u} \omega_k^{j,i} \phi_i^{M_u} \underbrace{\left(\frac{t - t_k}{t_{k+1} - t_k} \right)}_{\tau(t) \in [0,1]}, \ j = 1, .., n_u.$$

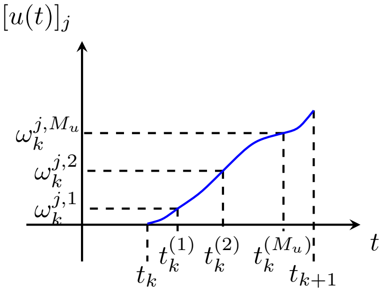
with $[\cdot]_j$ j-th element.

- Decision variables on $[t_k, t_{k+1}] : n_u \cdot (M_u + 1)$
- Collocation points:

$$t_k = t_k^{(0)} \le t_k^{(1)} \le \cdots \le t_k^{(M_u)} \le t_{k+1}$$

- Lagrange polynomials with $\tau_q = \tau(t_k^{(q)})$,

$$\phi_i^M(\tau) = \begin{cases} 1 & \text{if } M = 0 \\ \prod_{q=0, q \neq i}^{M} \frac{\tau - \tau_q}{\tau_i - \tau_q} & \text{if } M \ge 1 \end{cases}$$



Direct Collocation (Direct Transcription Method)

- State collocation: $[x(t)]_j,\ j = 1, .., n_x$ expressed via polynomial functions

$$[x(t)]_j = [X_k(t, \alpha_k)]_j := \sum_{i=0}^{M_x} \alpha_k^{j,i} \phi_i^{M_x} \left(\frac{t - t_k}{t_{k+1} - t_k} \right)$$

- Time derivative of parameterized states trajectory:

$$[\dot{x}(t)]_j = \frac{\partial}{\partial t}[X_k(t, \alpha_k)]_j = \frac{1}{t_k - t_{k-1}} \sum_{i=0}^{M_x} \alpha_k^{j,i} \frac{\partial \phi_i^{M_x}}{\partial \tau} \left(\frac{t - t_k}{t_{k+1} - t_k} \right)$$

- The collocation conditions

$$\begin{aligned} x_k &= X_k(t_k, \alpha_k) \\ f(X_k(t_k^{(i)}, \alpha_k), U_k(t_k^{(i)}, \omega_k)) &= \frac{\partial}{\partial t} X_k(t_k^{(i)}, \alpha_k),\ i = 1, ..., M_x \end{aligned}$$

summarized as $c_k(x_k, \alpha_k, \omega_k) = 0$.

Direct Collocation (Direct Transcription Method)

Discretized OCP:

$$\min_{X,\alpha,\omega} \quad \mathcal{M}(x_N) + \sum_{k=0}^{N-1} \ell_k(x_k, \alpha_k, \omega_k)$$

subject to

$$\begin{cases} 0 &= x_0 - \hat{x} & & \textsc{Initial condition} \\ 0 &= c_k(x_k, \alpha_k, \omega_k) & k \in \{0, 1, ..., N-1\} & \textsc{collocation conditions} \\ x_{k+1} &= X_k(t_{k+1}, \alpha_k) & k \in \{0, 1, ..., N-1\} & \textsc{continuity conditions} \\ 0 &\geq h(x_k, \omega_k) & k \in \{0, 1, ..., N-1\} & \textsc{Path constraints} \\ 0 &\geq r(x_N) & & \textsc{Terminal constraints} \end{cases}$$

with $X = (x_0, x_1, ..., x_N)$, $\alpha = (\alpha_0, \alpha_1, ..., \alpha_{N-1})$ and $\omega = (\omega_0, \omega_1, ..., \omega_{N-1})$

Direct Collocation (Direct Transcription Method)

Remark:

- The resulting NLPs are large scale.
- Number of stages and collocation points has to be chosen as a prior.
- Infeasible path method: ODEs satisfied at convergence only $\rightarrow$ computationally efficient, unstable systems doable!
- Stage times can be optimized too.
- Path constraints via inequality constraints at collocation points.
- Pseudospectral Methods
- Variant of orthogonal collocation with 1 stage and high-order polynomials.

Direct Collocation (Direct Transcription Method)

Example: Lotka Volterra fishing problem

$$\min_{u(\cdot)} \int_0^{12} (x_1(t)-1)^2 + (x_2(t)-1)^2 dt$$

$$\text{subject to} \quad \begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{bmatrix} x_1(t) - x_1(t)\cdot x_2(t) - 0.2x_1(t)\cdot u(t) \\ -x_2(t) + x_1(t)\cdot x_2(t) + 0.4x_2(t)\cdot u(t) \end{bmatrix},$$

$$\begin{pmatrix} x_1(0) & x_2(0) \end{pmatrix}^\top = \begin{pmatrix} 0.5 & 0.7 \end{pmatrix}^\top$$

$$u(t) \in [0,1],\ t \in [0,12]$$

with x_1 and x_2 the scaled population densities of a prey and a predator species.

Essential Tricks

- Input Rate Constraints: $\dot{u}(t) \in [\dot{u}_{\min}, \dot{u}_{\max}]$
  - introduce $\tilde{x} = [x^\top, u^\top]^\top$ with $\dot{u} = v$;
  - augmented dynamic $\dot{\tilde{x}} = \begin{bmatrix} f(x,u) \\ v \end{bmatrix}$ with $v \in [\dot{u}_{\min}, \dot{u}_{\max}]$.
- Free End-Time Problems: t_f decision variable
  - time transformation: $t = t_0 + \tau(t_f - t_0)$ with $\tau = \frac{t-t_0}{t_f-t_0} \in [0,1]$;
  - dynamics: $\dot{x}(\tau) = \frac{dx(t)}{dt}\frac{dt}{d\tau} = (t_f - t_0) f(x(\tau), u(\tau))$.
- Scaling of Input and State Variables: adjust the range
  - invertible scaling matrices Σ_x and Σ_u;
  - scaled state and control $\tilde{x} = \Sigma_x x$ and $\tilde{u} = \Sigma_u u$;
  - dynamic:

$$\dot{\tilde{x}} = \Sigma_x f(\Sigma_x^{-1}\tilde{x}, \Sigma_u^{-1}\tilde{u}).$$