Multiple Shooting

Mar 8, 2018

- **Example Problem:**

  Given: $\vec{x}_0, \vec{x}_F, T, \dot{\vec{x}} = \vec{f}(\vec{x}, \vec{u})$

  find: $\vec{u}(t)$

  min: $J(\vec{x}(t), \vec{u}(t)) = \int_0^T g(\vec{x}(t), \vec{u}(t))\,dt$

  Subject to: $\vec{x}(0) = \vec{x}_0 \quad \vec{x}(T) = \vec{x}_F$

Transcription via Multiple Shooting: Euler's Method, uniform grid, one step per segment

N = number of steps = number of segments $\rightarrow h = T/N$

Dec. Vars: $(\vec{x}_0), \vec{x}_1, \vec{x}_2, \dots \vec{x}_{N-1}, (\vec{x}_N)$ *

$\vec{u}_0, \vec{u}_1, \vec{u}_2, \dots \vec{u}_{N-1}$

- Objective: $J = h\sum_{k=0}^{N-1} g(\vec{x}_k, \vec{u}_k)$

dynamics = constraints $\quad \vec{C} = [\vec{C}_0; \vec{C}_1, \vec{C}_2, \dots, \vec{C}_{N-1}]^T \leftarrow$ column vector

$$\vec{C} = \begin{bmatrix} \vec{C}_0 = (\vec{x}_0 + h\vec{f}(\vec{x}_0, \vec{u}_0)) - \vec{x}_1 \\ \vec{C}_1 = (\vec{x}_1 + h\vec{f}(\vec{x}_1, \vec{u}_1)) - \vec{x}_2 \\ \vdots \\ C_{N-1} = (\vec{x}_{N-1} + h\vec{f}(\vec{x}_{N-1}, \vec{u}_{N-1})) - \vec{x}_N \end{bmatrix}$$

* Boundary Constraints applied to x_0 and x_N

→ This is a non-linear program (NLP) – Can be solved w/ FMINCON

-

Multiple Shooting Gradients

Mar. 9, 2018

- $\frac{\partial \vec{c}}{\partial \vec{z}}$ = Gradient of constraints wrt decision variables

$\vec{z} = [x_0, x_1, \ldots, x_N, \; u_0, u_1, \ldots, u_{N-1}]^T$ ← column vector

$$\frac{\partial \vec{c}}{\partial \vec{z}} = \begin{bmatrix} \frac{\partial c_0}{\partial x_0} & \cdots & \frac{\partial c_0}{\partial x_N}, & \frac{\partial c_0}{\partial u_0} & \cdots & \frac{\partial c_0}{\partial u_{N-1}} \\ \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial c_{N-1}}{\partial x_0} & \cdots & \frac{\partial c_{N-1}}{\partial x_N}, & \frac{\partial c_{N-1}}{\partial u_0} & \cdots & \frac{\partial c_{N-1}}{\partial u_{N-1}} \end{bmatrix}$$

This will be a sparse matrix: Most of these blocks are zero.

- $$\frac{\partial c_k}{\partial x_p} = \frac{\partial}{\partial x_p}\left(x_k + h f(x_k, u_k) - x_{k+1}\right)$$

$$= \frac{\partial x_k}{\partial x_p} + h f_x(k) \cdot \frac{\partial x_k}{\partial x_p} + h f_u(k) \cdot \cancelto{0}{\frac{\partial u_k}{\partial x_p}} - \frac{\partial x_{k+1}}{\partial x_p}$$

$$= \frac{\partial x_k}{\partial x_p}\left(1 + h f_x(k)\right) - \frac{\partial x_{k+1}}{\partial x_p}$$

$$\frac{\partial c_k}{\partial x_p} = \begin{cases} 1 + h f_x(k) & k = p \\ -1 & k+1 = p \\ 0 & \text{otherwise} \end{cases}$$

*Not recursive!

-

Multiple Shooting Gradients

Mar. 9, 2018

- $$\frac{\partial c_k}{\partial u_p} = \frac{\partial}{\partial u_p}\left(x_k + h f(x_k, u_k) - x_{k+1}\right)$$

$$= \frac{\partial x_k}{\partial u_p} + h f_x(k)\cdot\frac{\partial x_k}{\partial u_p} + h f_u(k)\cdot\frac{\partial u_k}{\partial u_p} - \frac{\partial x_{k+1}}{\partial u_p}$$

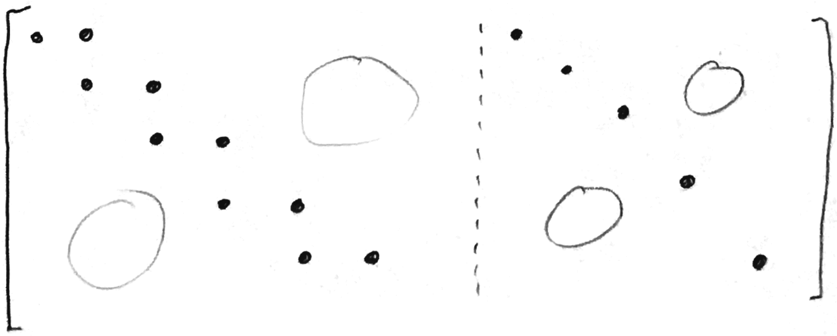
$$= \left(1 + h f_x(k)\right)\cancel{\frac{\partial x_k}{\partial u_p}}^{\,0} + h f_u(k)\cdot\frac{\partial u_k}{\partial u_p} - \cancel{\frac{\partial x_{k+1}}{\partial u_p}}^{\,0}$$

$$\frac{\partial c_k}{\partial u_p} = \begin{cases} h f_u(k) & k = p \\ 0 & \text{otherwise} \end{cases}$$

*Not Recursive!

- $$\frac{\partial \vec{c}}{\partial \vec{z}} = \begin{array}{c} \begin{array}{ccccccc} x_0 & x_1 & x_2 & \cdots & u_0 & u_1 & \cdots \end{array} \\ \left[\begin{array}{ccc|c|ccc} (1+hf_x(0)) & -1 & 0 & & hf_u(0) & 0 & 0 \\ 0 & (1+hf_x(1)) & -1 & & 0 & hf_u(1) & 0 \\ 0 & 0 & (1+hf_x(2)) & & 0 & 0 & hf_u(2) \\ & & & \vdots & & & \end{array}\right] \end{array} \begin{array}{c} \\ c_0 \\ c_1 \\ c_2 \\ \vdots \end{array}$$

$$\frac{\partial \vec{c}}{\partial \vec{z}} =$$



Sparsity Pattern