1 Motivation

Ziel: Ableitung der Nebenbedingung $x_i - s_i = 0$ welche aus dem multiple shooting Verfahren kommt.
Das Problem lässt sich reformulieren zu: Berechnung der Matrizen $Dx_{x_{i-1}}$ und $Dx_{u_{i-1}}$ ohne x zu kennen.

2 Gegebenheiten

Gegeben ist die Differentialgleichung Randwertproblem?

$$\dot{x} = f(x, u) \qquad x(0) = x_0, \quad x(t_N) = x_N$$

welche die Dynamik des Quadkopters beschreibt. Sie wird numerisch mit dem multiple shooting Verfahren gelöst.
Das multiple shooting Verfahren löst N Anfangswertprobleme auf den N Intervallen $[t_{i-1}, t_i], \quad i \in 1, ..., N$. Dabei wird der Anfangswert eines jeden Intervalls jedoch nicht fest vorgegeben, sondern als eine Variable behandelt. Der (unbekannte) Anfangswert für die Differentialgleichung des Intervalls $[t_{i-1}, t_i]$ ist $x(t_{i-1}) = x_{i-1}$. Da die Lösung x der Differentialgleichung auch von der Steuerung u in dem jeweiligen Intervall abhängig ist, kann man die Lösung jedes Anfangswertproblems als eine Funktion $x = x(t, x_{i-1}, u_{i-1})$ angeben. Hierbei ist also x_{i-1} der Anfangswert als Parameter, außerdem gehen wir davon aus, dass die Steuerung im Intervall $[t_{i-1}, t_i]$ konstant gleich $u(t_{i-1}) = u_{i-1}$ ist.
Damit x eine stetige Funktion ist, soll der Anfangswert eines jeden Intervalls mit dem dem Endwert des vorigen Intervalls übereinstimmen. Dies wird als Nebenbedingung folgendermaßen in das bereits bestehenden Optimierungsproblem eingefügt:

$$x(t_{x_i}, x_{i-1}, u_{i-1}) - x_i = 0 \tag{1}$$

Man kann dabei die senkrechte „Bewegung“ des aus dem multiple shooting stammenden Endpunktes als Funktion in Abhängigkeit von x_{i-1} und u_{i-1} ansehen.

Da wir mit dem SQP-Verfahren arbeiten, brauchen wir als Nebenbedingung nicht nur die oben angegebene Nebenbedingung (1), sondern auch deren Ableitung. (Für die Zielfunktion des SQP-Verfahrens ist dann auch noch die Hessematrix von (1).)

3

4 Vorgehen

Bedingungen prüfen, erfüllen unsere Gleichungen das? Erfüllen sie es auch mit numerischen Lösungen?

Gegeben sei die Differentialgleichung:

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

Idee: Anstatt zuerst die Differentialgleichung zu lösen, und dann die Lösung nach x bzw. u abzuleiten, leite zuerst die ODE nach x bzw. u ab und löse erst dann die Differentialgleichung: (Damit das nachfolgende Vorgehen überhaupt erlaubt, bzw. alle Ausdrücke wohldefiniert sind, muss $x(t)$ in allen Variablen stetig differenzierbar sein, damit man den Satz von Schwarz anwenden kann.)
Am Computer muss dieses Verfahren iterativ für jeden Zeitschritt angewendet werden. D.h. in jedem neuen Zeitschritt berechnet man M und N neu (also sozusagen M_{i-1}, N_{i-1})
Warum $i-1$? Ist das jetzt zum Zeitpunkt i oder zu einem anderen?

Man benutzt also:

$$\frac{\partial}{\partial x_{i-1}}\left(\frac{\mathrm{d}}{\mathrm{d}t}x\right) = \frac{\partial}{\partial x_{i-1}}\left(f(x,u)\right)$$

$$\Leftrightarrow \qquad \frac{\mathrm{d}}{\mathrm{d}t}\underbrace{\left(\frac{\partial x}{\partial x_{i-1}}\right)}_{=:M_{i-1}} = \frac{\partial}{\partial x}f(x,u)\cdot\underbrace{\frac{\partial x}{\partial x_{i-1}}}_{=:M_{i-1}}$$

Und:

$$\frac{\partial}{\partial u_{i-1}}\left(\frac{\mathrm{d}}{\mathrm{d}t}x\right) = \frac{\partial}{\partial u_{i-1}}\left(f(x,u)\right)$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\underbrace{\left(\frac{\partial x}{\partial u_{i-1}}\right)}_{=:N_{i-1}} = \frac{\partial}{\partial u}f(x,u)\cdot\underbrace{\frac{\partial x}{\partial u_{i-1}}}_{=:N_{i-1}}$$

Welche Dimension haben M und N??? Wo sind die anderen Partiellen Ableitungen ($\frac{\partial u}{\partial x/u_{i-1}}$)?

Es kann nun also folgendes System gelöst werden, um die Ableitung nach x_{i-1} bzw. u_{i-1} zu berechnen:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} x \\ M_{i-1} \\ N_{i-1}\end{bmatrix} = \begin{bmatrix} f(x,u) & f_x(x,u) & f_u(x,u)\end{bmatrix}\cdot\begin{bmatrix} 1 \\ M_{i-1} \\ N_{i-1}\end{bmatrix}$$

Achtung auf Multiplikation!!! Dimensionen passen? Matrizen? Vektoren?
Diese Differentialgleichung kann nun numerisch gelöst werden.
Setzt man den Zeitpunkt t_{i-1} ein, so erhält man für M_{i-1} und N_{i-1} die Anfangsbedingungen der ODE:

$$M(t_{i-1}) = \frac{\partial x(t_{i-1})}{\partial x_{i-1}} = \mathbb{I}$$

$$N(t_{i-1}) = \frac{\partial x(t_{i-1})}{\partial u_{i-1}} = 0$$

5 Voraussetzungen...

5.1 ...für das SQP-Verfahren

- am besten Hauptsächlich Gleichungsnebenbedingungen
- Zielfunktion und Nebenbedingungen zweimal stetig differenzierbar

6 Einzelne Lösungsschritte

1. SQP-Verfahren
   a) normales Optimierungsverfahren $\rightarrow$ Dynamik muss schon aufgelöst sein
   b) innerhalb noch mal explizites Lösungsverfahren (zB Newton, Newton-Rhapson?)
   c) Approximation der Hessematrix $\nabla^2_{xx} L(x^k, \mu^k)$ möglich (Machen wir nicht?)
   d) hauptsächlich für Gleichungsnebenbedingungen, aber auch für Ungleichungsnebenbedingungen möglich
   e) Globalisierung mit Penalty-Funktion möglich
   f) Schwierigkeiten: mögliche Unzulässigkeit der Teilprobleme (nur bein Ungleichungsnebenbedingungen?)
2. Lösung der Dynamik
   a)

6.1 Genauere Berachtung der Hessematrix der Lagrangefunktion

Lagrangefunktion

$$L : \mathbb{R}^n \times (\mathbb{R}^m) \times \mathbb{R}^p, \quad L(x, \lambda, \mu) = f(x) + (\lambda^T g(x)) + \mu^T h(x)$$

$f(x)$: Zielfunktion
$h(x)$: Gleichungsnebenbedingungen
Achtung: Optimierung bei uns nach zwei Variablen x und u, diese stecken hier beide in x!

Hessematrix für SQP-Verfahren

$$H_k = \nabla^2_{xx} L(x^k, \mu^k) = \nabla^2_{xx} f(x^k) + (\mu^k)^T \nabla^2_{xx} h(x^k)$$

Hier wird die Hessematrix der Nebenbedingungen gebraucht $\rightarrow$ auch die Nebenbedingungen aus Multiple shooting