

Parameterschätzung an einem Anfangswertproblem aus der Physiologie

Erkki Silde und Horatio Cuesdeanu

13. Juli 2010

Gegeben sei das Zielfunktional

$$J(y, u) = \frac{1}{2} \left(\sum_{i=1}^m (y_1(t_i) - \hat{y}_i)^2 + \sum_{i=1}^4 \beta_i (u_i - \hat{u}_i)^2 \right) \quad (1)$$

unter der Nebenbedingung

$$\begin{aligned} \dot{y}(t) &= A(u)y(t) + g(t, u) \quad t \in [0, t_f] \\ y(0) &= y^0 = 0. \end{aligned} \quad (2)$$

Dabei wird für Gleichung (1) mit $\hat{u} = (0.8, 5, -0.8, 5)^T$ der Bereich für physiologisch sinnvolle Werte festgelegt und \hat{y}_1 sind Messwerte für $y_1(t)$.

Für Gleichung (2) benutzen wir die folgenden Definitionen:

$$A(u) = \begin{pmatrix} -\frac{u_4}{u_1} & \frac{u_4}{u_1} \\ \frac{u_4}{u_2} & \frac{-u_4 - u_3}{u_2} \end{pmatrix} \quad (3)$$

$$g(t, u) = \frac{1}{u_1} \begin{pmatrix} q_{inf}(t) \\ 0 \end{pmatrix}. \quad (4)$$

Also stellt die AWA $\dot{y}(t) = A(u)y(t) + g(t, u)$ eine lineare inhomogene DGL dar, die eine eindeutige Lösung besitzt.

Wir definieren:

$$\hat{J}(y(u), u) := J(y, u)$$

und lösen das Problem mit Hilfe eines 'Lagrange-Ansatzes'. Dazu formulieren wir die Nebenbedingung wie folgt um:

$$E(y, u) = \begin{pmatrix} \dot{y} - f(\cdot, y(\cdot), u) \\ y(0) - y^0 \end{pmatrix} = \begin{pmatrix} 0 \in L^2 \\ 0 \in \mathbb{R}^2 \end{pmatrix}.$$

$$\implies \mathcal{L}(y, u, p, p^0) = J(y, u) + \int_0^{t_f} (\dot{y}(t) - f(t, y(t), u(t)))^T p(t) dt + y(0)p^0 \quad (5)$$

Aus $0 = D_y \mathcal{L}(y^*, u^*, p, p^0) \cdot h$ gewinnen wir wie in (2.26a) aus der Vorlesung die Differentialgleichung für die adjungierte Variable:

$$-\dot{p}(t) - A(u)^T p(t) = - \sum_{i=1}^{m-1} (y_1(t_i) - \hat{y}_i) \delta_{t_i} \quad t \in [0, t_f]. \quad (12)$$

$$h(0) = 0 \Rightarrow p(t_f) = - \begin{pmatrix} y_1(t_f) - \hat{y}_m \\ 0 \end{pmatrix} \quad (13)$$

$$h(t_f) = 0 \Rightarrow p(0) = p^0 \quad (14)$$

Aus $0 = D_u \mathcal{L}(y^*, u^*, p, p^0) \cdot h$ gewinnen wir wie in (2.26b) aus der Vorlesung die Darstellung für den Gradienten des reduzierten Zielfunktional:

$$\left(\nabla \hat{J}_p(p) \right)_i = \beta_i (u_i^* - \hat{u}_i) - \int_0^{t_f} (y^T(t) A_{u_i}^T(u) + g_{u_i}^T(t, u)) p(t) dt = 0 \quad (17)$$

Die oben hergeleiteten Bedingungen können wir ausnutzen, um das Problem numerisch zu lösen. Dazu verwenden wir den folgenden Algorithmus. Dazu sei ein Startwert u^0 gegeben ($n=0$).

Die oben hergeleiteten Bedingungen können wir ausnutzen, um das Problem numerisch zu lösen. Dazu verwenden wir den folgenden Algorithmus. Dazu sei ein Startwert u^0 gegeben ($n=0$).

while $n \leq n_{max}$ & $\|\hat{J}_u(u^n)\| > \varepsilon_1$ & $\|\hat{J}(u^n) - \hat{J}(u^{n-1})\| > \varepsilon_2$ **do**

Die oben hergeleiteten Bedingungen können wir ausnutzen, um das Problem numerisch zu lösen. Dazu verwenden wir den folgenden Algorithmus. Dazu sei ein Startwert u^0 gegeben ($n=0$).

while $n \leq n_{max}$ & $\|\hat{J}_u(u^n)\| > \varepsilon_1$ & $\|\hat{J}(u^n) - \hat{J}(u^{n-1})\| > \varepsilon_2$ **do**
1. Berechnen $y(\cdot, u^n)$ mit Hilfe des klassischen Runge-Kutta Verfahrens ($O(h^4)$) gemäß (2).

Die oben hergeleiteten Bedingungen können wir ausnutzen, um das Problem numerisch zu lösen. Dazu verwenden wir den folgenden Algorithmus. Dazu sei ein Startwert u^0 gegeben ($n=0$).

while $n \leq n_{max}$ & $\|\hat{J}_u(u^n)\| > \varepsilon_1$ & $\|\hat{J}(u^n) - \hat{J}(u^{n-1})\| > \varepsilon_2$ **do**

1. Berechnen $y(\cdot, u^n)$ mit Hilfe des klassischen Runge-Kutta Verfahrens ($O(h^4)$) gemäß (2).
2. Berechnen $p(\cdot)$ mit Hilfe des klassischen Runge-Kutta Rückwärtsverfahrens ($O(h^4)$) gemäß (12).

Die oben hergeleiteten Bedingungen können wir ausnutzen, um das Problem numerisch zu lösen. Dazu verwenden wir den folgenden Algorithmus. Dazu sei ein Startwert u^0 gegeben ($n=0$).

- while** $n \leq n_{max}$ & $\|\hat{J}_u(u^n)\| > \varepsilon_1$ & $\|\hat{J}(u^n) - \hat{J}(u^{n-1})\| > \varepsilon_2$ **do**
1. Berechnen $y(\cdot, u^n)$ mit Hilfe des klassischen Runge-Kutta Verfahrens ($O(h^4)$) gemäß (2).
 2. Berechnen $p(\cdot)$ mit Hilfe des klassischen Runge-Kutta Rückwärtsverfahrens ($O(h^4)$) gemäß (12).
 3. Berechnen $\hat{J}_u(u^n)$ mit Hilfe der Simpson-Regel ($O(h^4)$) gemäß (17).

Die oben hergeleiteten Bedingungen können wir ausnutzen, um das Problem numerisch zu lösen. Dazu verwenden wir den folgenden Algorithmus. Dazu sei ein Startwert u^0 gegeben ($n=0$).

- while** $n \leq n_{max}$ & $\|\hat{J}_u(u^n)\| > \varepsilon_1$ & $\|\hat{J}(u^n) - \hat{J}(u^{n-1})\| > \varepsilon_2$ **do**
1. Berechnen $y(\cdot, u^n)$ mit Hilfe des klassischen Runge-Kutta Verfahrens ($O(h^4)$) gemäß (2).
 2. Berechnen $p(\cdot)$ mit Hilfe des klassischen Runge-Kutta Rückwärtsverfahrens ($O(h^4)$) gemäß (12).
 3. Berechnen $\hat{J}_u(u^n)$ mit Hilfe der Simpson-Regel ($O(h^4)$) gemäß (17).
 4. Ermitteln die Schrittweite t^n mit Hilfe der Armijo-Regel.

Die oben hergeleiteten Bedingungen können wir ausnutzen, um das Problem numerisch zu lösen. Dazu verwenden wir den folgenden Algorithmus. Dazu sei ein Startwert u^0 gegeben ($n=0$).

while $n \leq n_{max}$ & $\|\hat{J}_u(u^n)\| > \varepsilon_1$ & $\|\hat{J}(u^n) - \hat{J}(u^{n-1})\| > \varepsilon_2$ **do**

1. Berechnen $y(\cdot, u^n)$ mit Hilfe des klassischen Runge-Kutta Verfahrens ($O(h^4)$) gemäß (2).
2. Berechnen $p(\cdot)$ mit Hilfe des klassischen Runge-Kutta Rückwärtsverfahrens ($O(h^4)$) gemäß (12).
3. Berechnen $\hat{J}_u(u^n)$ mit Hilfe der Simpson-Regel ($O(h^4)$) gemäß (17).
4. Ermitteln die Schrittweite t^n mit Hilfe der Armijo-Regel.
5. Setzen $u^{n+1} = u^n + t^n \frac{\hat{J}_u(u^n)}{\|\hat{J}_u(u^n)\|}$.

Setzen $n := n + 1$.

end while