National Conference with International Participation
ENGINEERING MECHANICS 2003
Svratka, Czech Republic, May 12 – 15, 2003
paper no. 328

KINEMATIC ACCURACY OF A SIMPLE OPEN CHAIN BY USING AUTOMATIC DIFFERENTIATION

M. Němeček*

Summary : *The mail idea of this paper is to compute errors in the position and orientation of an end-effector by the efficient way. We have a simple open chain with revolute joints only. The position and orientation is described by mens of transformation matrix between the basic and the end-efector spaces. We will use the linear model of errors. According tolinear theory we must differentiate this transformation matrix with respect to angle and length chain parameters. This derivation we will to generate efficiently by means of Automatic Differentiation.*

1. Úvod

Tento článek se zabývá efektivním způsobem výpočtu a popisu chyb v poloze a orientaci koncového členu robotické struktury, která tvoří jednoduchý otevřený kinematický řetězec s rotačními vazbami. Matematicky je poloha a orientace koncového členu popsána transformační maticí mezi systémem základního rámu 0 a systémem koncového členu n Obecně lze odvodit lineární a nelineární model chyb v závislosti na jejich předpokládané velikosti. My budeme očekávat malé chyby tj. lineární model. Podle něj je nutné derivovat transformační matici podle délkových nebo úhlových parametrů vyskytujících se v řetězci. Tuto derivaci lze efektivně provést pomocí metody Automatického Derivování.

2. Automatické derivování

Automatické derivování (AD) je metoda, která umožňuje efektivní cestou vytvářet derivace skalárních a vektorových funkcí. AD bývá často zaměňováno se symbolickým derivováním (SD). SD je pro složitější mimoškolní vědecké výpočty velmi nevhodné, protože vytváří dlouhé řetězce výrazů, které se nedokáží ani zobrazit a tak dojde k jejich odtržení. AD derivátor, jehož princip budeme matematicky simulovat, dlouhé řetězce postupně substituuje a výsledkem derivování není dlouhý řetězec, ale procedura obsahující dílčí příkazy a v konečném důsledku počítá hodnotu funkce a její derivace ve zvoleném bodě. Příkazy uvnitř jsou optimalizovány tak, aby se neopakovaly často vyskytující se výrazy.

V této kapitole se tedy budeme snažit vytvořit vhodný matematický aparát AD, který nám umožní jeho efektivní použití k derivaci transformační matice. Budeme tak matematicky simulovat AD derivátor a vytvářet programy-kódy, které zapíšeme v matematické formě.

* Ing. Martin Němeček
Ústav Mechaniky FSI ČVUT Praha, Karlovo náměstí 13,
e-mail: nemecekm@fsik.cvut.cz

Cílem derivování obecně je získat Jacobiovu matici **J** z **m** rozměrné vektorové funkce **F** vzhledem k **n** rozměrnému vektoru nezávislých souřadnic **x** . Nechť **F** je možno rozložit na soustavu y_1 až y_p elementárních funkcí,

$$\begin{aligned} y_1 &= f_1(x_1,\ldots,x_n) \\ y_2 &= f_2(x_1,\ldots,x_n,y_1) \\ &\vdots \\ y_p &= f_p(x_1,\ldots,x_n,y_1,\ldots,y_{p-1}) \\ \mathbf{F}(x_1,\ldots,x_n) &\equiv \mathbf{F}^{p+1}(x_1,\ldots,x_n,y_1,\ldots,y_p) \end{aligned} \tag{1}$$

které představují elementární matematické operace (součet, násobení, sin, …) a poslední elementární funkce v soupisu (1) může obecně obsahovat **n** nezávislých proměnných a **p** závislých proměnných, pak můžeme soustavu zapsat stručně v anulovaném tvaru

$$\begin{aligned} \mathbf{F}^E(\mathbf{x},\mathbf{y}) &= \mathbf{0} \\ \mathbf{F}(\mathbf{x}) &\equiv \mathbf{F}^{p+1}(\mathbf{x},\mathbf{y}) \end{aligned} \tag{2}$$

Derivováním systému (2) podle **x** obdržíme vztah pro **J**, kde vystupují Jacobiovy matice funkcí $\mathbf{F}^{p+1}$ a $\mathbf{F}^E$ jenom podle **x** nebo podle **y**

$$\mathbf{J} = \mathbf{F}_\mathbf{x}^{p+1} - \mathbf{F}_\mathbf{y}^{p+1}\left(\mathbf{F}_\mathbf{y}^E\right)^{-1}\mathbf{F}_\mathbf{x}^E \tag{3}$$

Vyhodnocení (3) můžeme provést dvěma způsoby, první označujeme jako **přímý mód AD** a získáme řešením matice **V** (p,n) jako řešení **n** lineárních systémů s dolní trojúhelníkovou maticí $\mathbf{F}_\mathbf{y}^E$, což odpovídá derivaci syst. (2) shora dolů tedy přímo nebo dopředu

$$\begin{aligned} \left(\mathbf{F}_\mathbf{y}^E\right)\mathbf{V} &= \mathbf{F}_\mathbf{x}^E \\ \mathbf{J} &= \mathbf{F}_\mathbf{x}^{p+1} - \mathbf{F}_\mathbf{y}^{p+1}\mathbf{V} \end{aligned} \tag{4}$$

Druhý způsob označovaný jako **zpětný mód AD** obdržíme řešením matice **W** (m,p) jako řešení **m** lineárních systémů s horní trojúhelníkovou transponovanou maticí $\mathbf{F}_\mathbf{y}^E$, což odpovídá derivaci (2) zdola

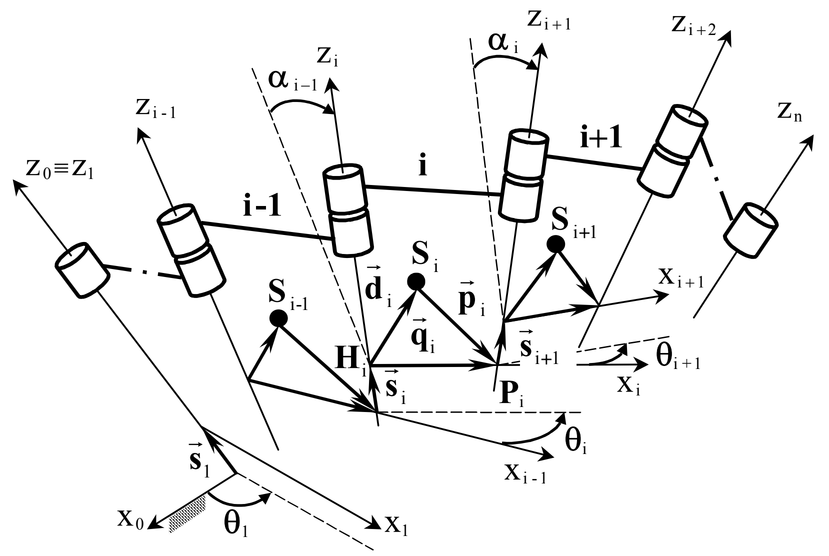
$$\begin{aligned} \left(\mathbf{F}_\mathbf{y}^E\right)^T \mathbf{W}^T &= \left(\mathbf{F}_\mathbf{y}^{p+1}\right)^T \\ \mathbf{J} &= \mathbf{F}_\mathbf{x}^{p+1} - \mathbf{W}\mathbf{F}_\mathbf{x}^E \end{aligned} \tag{5}$$

V případě výpočtu gradientu $\mathbf{f}_x$ (n,1) skalární funkce řešíme vektor **w** (p,1) z horního trojúhelníkového systému

$$\begin{aligned} \left(\mathbf{F}_\mathbf{y}^E\right)^T \mathbf{w} &= -\mathbf{f}_\mathbf{y}^{p+1} \\ \mathbf{f}_x(\mathbf{x}) &= \mathbf{f}_\mathbf{x}^{p+1} + \left(\mathbf{F}_\mathbf{x}^E\right)^T \mathbf{w} \end{aligned} \tag{6}$$

3. Ideální model otevřeného kinematického řetězce

Mějme otevřený kinematický řetězec tvořený **n** tělesy spojené rotačními vazbami, který v přesné-nechybné poloze zaujme tvar dle obr.1



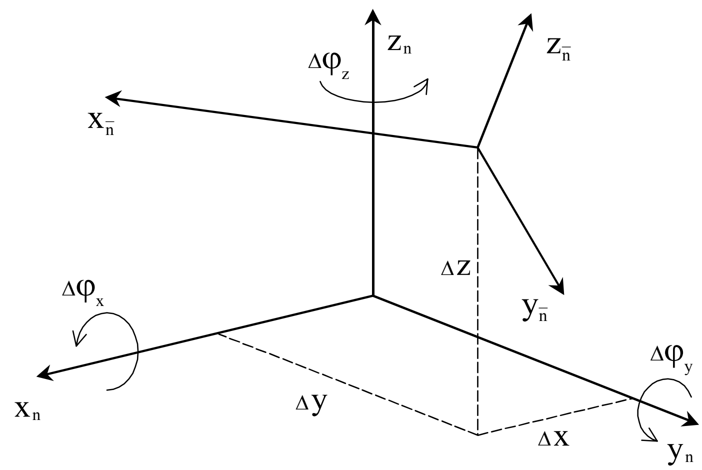
Obrázek1 Ideální model řetězce

4. Matice chyb metodou AD

Jak jsme již uvedli v úvodu polohu a orientaci systému tělesa n vůči rámu popisuje transformační matice, která je závislá na rozměrech nebo úhlech, které popisují řetězec z obr.1. Souhrnně tyto parametry označme vektorem **k** a pro transformační matici platí

$$\mathbf{T}_{0n} = \mathbf{T}_{0n}\left(\mathbf{k}\right) \tag{7}$$

Následkem malé změny Δk_j parametrů k_j koncový člen **n** zaujme polohu $\bar{\mathbf{n}}$ dle obr.2



Obrázek2 Chyba v poloze koncového členu

Této malé změně parametrů **k** odpovídá změna transformační matice zapsané ve tvaru

$$\Delta\mathbf{T}_{0n} = \sum_j \frac{\partial\mathbf{T}_{0n}}{\partial k_j}\Delta k_j = \begin{bmatrix} \frac{\partial\mathbf{T}_{0n}}{\partial k_1} & \dots & \frac{\partial\mathbf{T}_{0n}}{\partial k_N} \end{bmatrix} \begin{bmatrix} \mathbf{E}(\Delta k_1) \\ \vdots \\ \mathbf{E}(\Delta k_N) \end{bmatrix} \tag{8}$$

kde: $\mathbf{F}^C$ je matice citlivostí na parametry **k** a je vyjádřená v prostoru základního rámu. Je to zároveň matice parciálních derivací, kterou budeme efektivně počítat metodou AD.

Chybovou matici s prvky (viz. obr.2) popisujícími chybu v poloze (malé posuvy podél os) a orientaci (malé úhly kolem os) vyjádřenou v systému tělesa **n** lze vyjádřit

$$\mathbf{H}_{n,0n} = \left[\begin{array}{c:c} \Delta\Theta_{n,0n} & {}^{n}\Delta\mathbf{u} \\ \hdashline \mathbf{0} & 0 \end{array}\right] = \begin{bmatrix} 0 & -{}^{n}\Delta\varphi_z & {}^{n}\Delta\varphi_y & {}^{n}\Delta x \\ {}^{n}\Delta\varphi_z & 0 & -{}^{n}\Delta\varphi_x & {}^{n}\Delta y \\ -{}^{n}\Delta\varphi_y & {}^{n}\Delta\varphi_x & 0 & {}^{n}\Delta z \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{9}$$

$$\mathbf{H}_{n,0n} = (\mathbf{T}_{0n})^{-1}\Delta\mathbf{T}_{0n} = (\mathbf{T}_{0n})^{-1}\sum_j \frac{\partial\mathbf{T}_{0n}}{\partial k_j}\Delta k_j \tag{10}$$

Transformační matici (7) a její inverzi vyjádříme pomocí jejich submatic následovně

$$\mathbf{T}_{0n} = \left[\begin{array}{c:c} & {}^{0}x \\ \mathbf{S}_{0n} & {}^{0}y \\ & {}^{0}z \\ \hdashline \mathbf{0} & 0 \end{array}\right] = \left[\begin{array}{c:c} \mathbf{S}_{0n} & {}^{0}\mathbf{u} \\ \hdashline \mathbf{0} & 0 \end{array}\right] \qquad (\mathbf{T}_{0n})^{-1} = \left[\begin{array}{c:c} (\mathbf{S}_{0n})^T & -(\mathbf{S}_{0n})^T\,{}^{0}\mathbf{u} \\ \hdashline \mathbf{0} & 0 \end{array}\right] \tag{11}$$

Ze vztahů (9) až (11) pak můžeme pro jednotlivé submatice chybové matice vyjádřit způsob jejího výpočtu v prostoru základního rámu a koncového členu

$${}^{n}\Delta\mathbf{u} = (\mathbf{S}_{0n})^T\,{}^{0}\Delta\mathbf{u} \tag{12}$$

$$\Delta\Theta_{n,0n} = (\mathbf{S}_{0n})^T\,\Delta\mathbf{S}_{0n} \tag{13}$$

$$\Delta\Theta_{0,0n} = \Delta\mathbf{S}_{0n}\,(\mathbf{S}_{0n})^T \tag{14}$$

kde vektor ${}^{0}\Delta\mathbf{u}$ a submatici $\Delta\mathbf{S}_{0n}$ matice $\Delta\mathbf{T}_{0n}$ napočteme $\mathbf{T}_{0n}$ zpětným módem AD viz. (5).

Transformační matici (7) můžeme pro AD vyjádřit pomocí součinu N základních transformačních matic

$$\mathbf{T}_{0n}(\mathbf{k}) = \mathbf{T}_{Zm1}(k_1)\,\mathbf{T}_{Zm2}(k_2)\dots\mathbf{T}_{Zm(N-1)}(k_{N-1})\,\mathbf{T}_{ZmN}(k_N) \qquad m \in \{1,2,3,4,5,6\} \tag{15}$$

Podle vzoru (5) derivujme matici (15) zpětným módem AD a podle (12) až (14) vygenerujeme pro matici chyb následující kód výpočtu pro případ $N = 4$.

Tabulka1 Matice chyb metodou AD

Napočtení nadbytečných souřadnic

$$\mathbf{Y}_1 = \mathbf{T}_{Zm4}\left(k_4\right)$$

$$\mathbf{Y}_2 = \mathbf{T}_{Zm3}\left(k_3\right)\mathbf{Y}_1$$

$$\mathbf{Y}_3 = \mathbf{T}_{Zm2}\left(k_2\right)\mathbf{Y}_2$$

$$\mathbf{Y}_4 = \mathbf{T}_{Zm1}\left(k_1\right)\mathbf{Y}_3$$

Řešení systému s horní trojůhelníkovou maticí

$$\mathbf{W}_4 = \mathbf{E}\left(1\right)$$

$$\mathbf{W}_3 = \mathbf{W}_4\,\mathbf{T}_{Zm1}$$

$$\mathbf{W}_2 = \mathbf{W}_3\,\mathbf{T}_{Zm2}$$

$$\mathbf{W}_1 = \mathbf{W}_2\,\mathbf{T}_{Zm3}$$

Výpočet diference matice $\mathbf{T}_{0n}$

$$\Delta\mathbf{T}_{0n} = \left[\begin{array}{c:c:c:c} \mathbf{W}_4\,\dfrac{\partial\mathbf{T}_{Zm1}}{\partial k_1}\,\mathbf{Y}_3 & \mathbf{W}_3\,\dfrac{\partial\mathbf{T}_{Zm2}}{\partial k_2}\,\mathbf{Y}_2 & \mathbf{W}_2\,\dfrac{\partial\mathbf{T}_{Zm3}}{\partial k_3}\,\mathbf{Y}_1 & \mathbf{W}_1\,\dfrac{\partial\mathbf{T}_{Zm4}}{\partial k_4} \end{array}\right] \begin{bmatrix} \mathbf{E}\left(\Delta k_1\right) \\ \mathbf{E}\left(\Delta k_1\right) \\ \mathbf{E}\left(\Delta k_1\right) \\ \mathbf{E}\left(\Delta k_1\right) \end{bmatrix}$$

Výpočet matic $\left(\mathbf{S}_{0n}\right)^{T}$ *a* $\Delta\mathbf{S}_{0n}$

$$\mathbf{IY} = \left(\mathbf{Y}_4(1:3,1:3)\right)^{T}$$

$$\mathbf{DY} = \Delta\mathbf{T}_{0n}(1:3,1:3)$$

Výpočet vektorů ${}^{0}\Delta\mathbf{u}$, ${}^{n}\Delta\mathbf{u}$ *a matic* $\Delta\Theta_{0,0n}$, $\Delta\Theta_{n,0n}$

$${}^{0}\Delta\mathbf{u} = \Delta\mathbf{T}_{0n}(1:3,4)$$

$${}^{n}\Delta\mathbf{u} = \mathbf{IY}\;{}^{0}\Delta\mathbf{u}$$

$$\Delta\Theta_{0,0n} = \mathbf{DY}\,\mathbf{IY}$$

$$\Delta\Theta_{n,0n} = \mathbf{IY}\,\mathbf{DY}$$

Výpočet matic $\mathbf{H}_{0,0n}$ *a* $\mathbf{H}_{n,0n}$

$$\mathbf{H}_{0,0n} = \left[\begin{array}{c:c} \Delta\Theta_{0,0n} & {}^{0}\Delta\mathbf{u} \\ \hdashline \mathbf{0} & 0 \end{array}\right] \qquad \mathbf{H}_{n,0n} = \left[\begin{array}{c:c} \Delta\Theta_{n,0n} & {}^{n}\Delta\mathbf{u} \\ \hdashline \mathbf{0} & 0 \end{array}\right]$$

5. Závěr

Byla popsána efektivní cesta výpočtu matice chyb polohy a orientace pomocí universálně účinné metody AD. Její rychlý a efektivní výpočet se uplatní při řízení a přesném polohování robotických struktur, kde je nutné výpočet mnohokrát opakovat. Dokážeme tak pružně opravit nepřesný pohyb manipulátoru.

6. Literatura

Coleman T. - Verma A (1998) The efficient computation of sparse Jacobian matrices using automatic differentiation. SIAM *Journal on Scientific Computing*, pp.1210-1233.

Griewank A. (1991) Automatic differentiation of algoritms. *Theory, implementation, and application. SIAM*, Philadelphia, Penn.,

Christianson B. (1992) Automatic Hessians by reverse accumulation. IMA *Journal of Numerical Analysis*, pp.135-150.

Stejskal V. - Valášek M (1996) Kinematics and dynamics of machinery. *Marcel Dekker*, New York.

Valášek M (1990) On the efficient implementation of multibody system formalism. *Universität Stuttgart*, Stuttgart