

Řešení soustavy nelineárních rovnic algoritmus "postup08"

Petr Byczanski

Ústav geoniky AV ČR
Studentská 1768 Ostrava – Poruba

Algoritmus "postup08" je určen pro řešení soustavy nelineárních rovnic

$$F(x) = Y \quad (1)$$

splňující

- (A) Složky funkce $F : D \subset E_n \rightarrow E_n$ mají spojité první parciální derivace.
- (B) Je zadán výchozí bod X_0 , o němž je předpokládáno, že z něj lze spojitou změnou přejít do hledaného řešení.

Pro enumeraci spojité změny si zavedme proměnnou t nabývající nezáporných hodnot. Hodnotě 0 nechť odpovídá počáteční bod.

Řešený problém lze přeformulovat. Hledáme funkci $X(t)$ vyhovující

$$\begin{aligned} X(0) &= X_0 \\ \partial_x F(X(t)) \times \partial_t X(t) &= Y - F(X(t)) \end{aligned} \quad (2)$$

Vzhledem k neúplnosti specifikace spojité změny v (B) je na pravé straně (2) použito nejjednodušší možná forma zajišťující postup k cíli. Fakticky nás bude z hledané funkce zajímat pouze její hodnota v okamžiku, kdy dojde ke splnění rovnic soustavy (1).

Přeformulovaný problém lze myšleně vyřešit jednoduchým postupem. Vyjdeme z počátečních hodnot $t = 0$, $X = X_0$. S infinitezimálním zvětšením t provedeme infinitezimální změnu X určenou podle rovnice

$$\partial_x F(X) \times dX = (Y - F(X)) \cdot dt \quad (3)$$

Pokud nevyjedeme z definičního oboru a pokud nenastane $\det(\partial_x F) = 0$, tak buď někdy nastane $F(X) = Y$, tehdy skončíme, anebo budeme pokračovat donekonečna.

Předkládaný algoritmus vychází z uvedeného teoretického myšlenkového postupu. Infinitezimální změny jsou nahrazeny konečnými změnami.

Na začátku vyjdeme z bodu $X = X_0$. Uvažujme obecně, že jsme v nějakém postupovém bodě X .

Levou stranu (1) lze napsat ve tvaru

$$F(X) + \partial_x F(X) \times \Delta X + .?.$$

Kdyby platil lineární rozvoj funkce F , tj. zanedbali bychom $.?.$, a chtěli bychom splnit (1), tak by jsme změnu X určili podle rovnice

$$\partial_x F(X) \times \Delta_{max} X = Y - F(X) \quad (4)$$

Indexem "max" je vyznačen fakt, že jde o maximální možnou konečnou změnu. Skutečnou konečnou změnu musíme totiž zvolit tak, abychom zůstali v oblasti alespoň přibližné platnosti lineárního rozvoje F

$$\Delta X = \eta \cdot \Delta_{max} X \quad \eta \in (0, 1 > \quad (5)$$

Násobek η určíme tak, aby chyba $.\text{?}$. nepřesáhla únosnou mez. Po přičtení konečné změny se dostaneme do dalšího postupového bodu.

Zpřesňování – přechody z předchozího do následujícího postupového bodu – ukončíme, když se dostatečně blízko přiblížíme k požadovanému Y anebo když odhalíme tendenci ke směřování donekonečna.

Přírozenou mírou únosnosti chyby $.\text{?}$. je

$$\|.\text{?}\| \leq \frac{\|\partial_x F \times \Delta X\|}{\text{štíhlost}} \quad (6)$$

Hranice přípustné odchylky mezi linearizovanou a skutečnou funkční hodnotou je úměrná délce pohybu. Je-li koeficient *štíhlost* větší než 1, převyšuje lineární člen rozvoje F součet všech vyšších členů. Tím je opodstatněno jejich zanedbání.

Hodnota η v (5) vyhovující (6) nemusí být určena zcela přesně. Menší hodnota je dokonce příznivější, protože odpovídá menší chybě. Proto lze η stanovit binárním půlením.

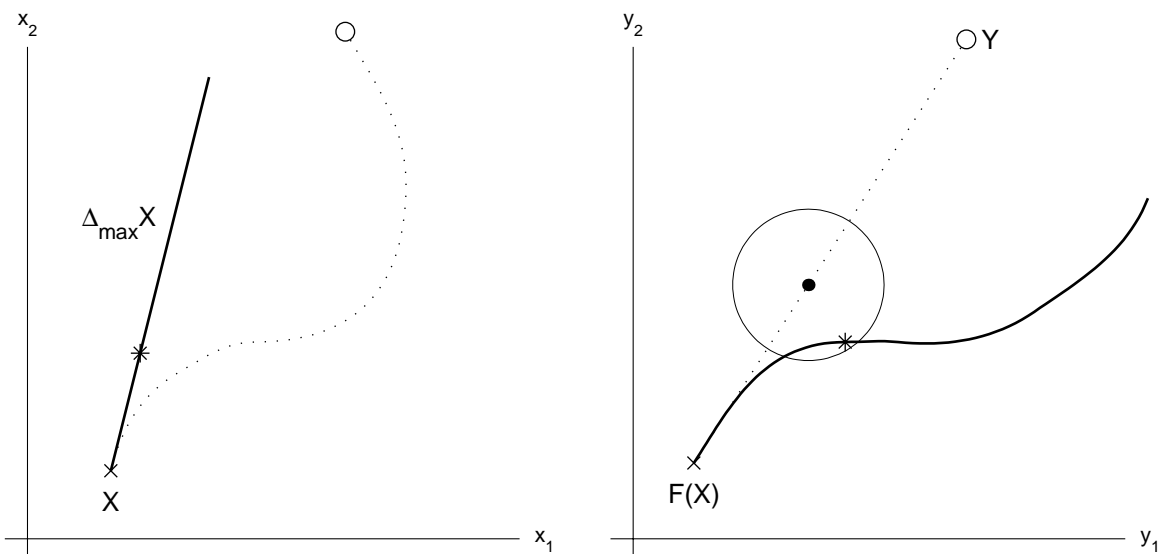
Nejprve zkusíme $\eta = 1$, tj. ověříme platnost

$$\|F(X) + \partial_x F(X) \times \Delta_{max} X - F(X + \Delta_{max} X)\| \leq \frac{\|\partial_x F(X) \times \Delta_{max} X\|}{\text{štíhlost}} \quad (7)$$

Při splnění jsme hotovi. V opačném případě začneme proces půlení. Vyjdeme z intervalu 0/vyhovuje /... 1/nevychovuje/. Pro každou novou prostřední hodnotu η ověříme

$$\|F(X) + \partial_x F(X) \times (\eta \cdot \Delta_{max} X) - F(X + \eta \cdot \Delta_{max} X)\| \leq \frac{\|\partial_x F(X) \times (\eta \cdot \Delta_{max} X)\|}{\text{štíhlost}} \quad (8)$$

Při splnění (8) přejdeme na pravý podinterval, při nesplnění na levý. Okamžikem prvního splnění začneme počítat počet půlení. Skončíme jakmile dosáhneme počet *jemnost*. Mne-motechnické pojmenování je dáno tím, že požadovaným počtem platných dvojkových číslic hodnoty η vlastně určíme, nakolik přesně (jemně) se posuneme dále.



Krok postupu je naznačen na obrázku. Tečkovaně je zakreslen přesný myšlenkový postup podle (3). Skutečný pohyb je vyznačen plnou čarou. Počáteční bod je vyznačen křížkem, následující postupový bod hvězdičkou. Obrázek je nepřesný v tom, že neodpovídá konečné přesnosti η . Naznačené platí pro spojitě η v celém rozsahu $(0, 1)$. Ve skutečnosti se realizují pouze některé z těchto bodů.

Pro jednu hodnotu η odpovídající tečce je naznačeno přípustné okolí skutečných funkčních hodnot, kdy je chyba Δ považována ještě za únosnou. Středů přípustných kruhových okolí se myšleně pohybují na tečkované úsečce, jejich poloměry vzrůstají úměrně vzdálenosti středu od výchozího bodu. Obalová plocha těchto přípustných okolí vytváří kužel; jeho štíhlost dala jméno mnemotechnickému pojmenování koeficientu úměrnosti.

Řídící parametr *štíhlost* určuje nakolik se může odchýlit skutečný a linearizovaný průběh funkce. Větší hodnotě odpovídá relativně větší přesnost lineární aproximace. Přibližování se k cíli je zajištěno, když hodnota přesahuje 1. Ve většině aplikací bude asi nejvhodnější zvolit hodnotu odpovídající nejrychlejšímu postupu k cíli. Hrubý odhad – maximální přiblížení se k cíli v jednom kroku i v nejméně příznivém případě – vede na hodnotu 2.

Řídící parametr *jemnost* určuje nakolik využijeme možnou přípustnou délku kroku postupu. Jeho volba není kritická. Postačuje jakákoliv intuitivně rozumná hodnota, např. 5.

Před závěrečnou diskusí vlastností algoritmu "postup08" si nejprve všimneme rozdílu oproti jiným běžně užívaným metodám. Zásadní odlišností je korektnost použití Taylorova rozvoje funkce F .

Např. v Newtonově metodě se také používá pouze absolutní a lineární člen. Na jejich základě se určí změna argumentu. Tato se realizuje bez ohledu na platnost použitého rozvoje. Proto se též někdy stává, že tato metoda nekonverguje. Potom se uměle přidává tlumení, realizuje se pouze zlomek změny argumentu. Určení tlumícího faktoru je velmi svízelné, není již součástí Newtonovy metody!

Naproti tomu v předkládaném algoritmu je použití lineárního rozvoje funkce F zcela korektní. Je to zajištěno tím, že skutečně realizovaná změna je dána vhodným násobkem, kdy se pohyb děje ještě v rozsahu postačující platnosti lineárního rozvoje.

Algoritmus "postup08" nahrazuje spojité přibližování se k cíli pohybem po diskrétní posloupnosti bodů. Jsou rozlišovány postupové a průzkumné body. V postupových bodech jsou kromě funkčních hodnot vyčíslovány i jejich derivace. Jsou zjišťovány odchylky od požadované hodnoty a prováděny linearizace funkce. Naproti tomu v průzkumných bodech jsou vyčíslovány pouze funkční hodnoty. Odchylkou mezi těmito skutečnými a linearizovanými hodnotami je ověřována platnost aproximace F .

Prakticky je "postup08" ekvivalentní Newtonově metodě s optimálně určeným tlumením.

Hlavní nevýhodou algoritmu "postup08" je možnost překročení hranice $\det(\partial_x F) = 0$. Toto je způsobeno diskrétností postupu.

Největší výhodou algoritmu "postup08" je univerzálnost použití. Vykazuje totiž kvalitativně shodné chování bez ohledu na míru nelinearity a bez ohledu na vzdálenost počátečního odhadu od neznámého řešení. Délky jednotlivých postupových kroků se automaticky přizpůsobují stupni nelinearity. Je rozlišováno vyčíslování funkčních hodnot bez a včetně jejich derivací. Počet (časově náročných) vyčíslování derivací je automaticky minimalizován.

Při malé hodnotě *štíhlosti* není sledována žádná teoretická linie vedoucí k cíli. Z aktuálního bodu se v y -prostoru začne postupovat podél úsečky mířící k cíli, po odklonu se vytvoří další bod. Situace se bude opakovat, opět zamíříme k cíli, ale již podél jiné úsečky. Naproti tomu při velké hodnotě *štíhlosti* se bude postup k cíli dít v y -prostoru téměř podél úsečky $F(X_0) - Y$ dané výchozím a požadovaným bodem. Tento postup sledující jednu teoretickou linii bude ovšem již podstatně pomalejší oproti předchozímu chaotickému postupu.

Proto byla vytvořena i varianta "postup01", kdy se k cíli postupuje podél úsečky $F(X_0) - Y$. V každém postupovém kroku je nejprve pouze změnou X korigována chyba z předchozího kroku a teprve poté je další změnou X spolu se změnou t postupováno k cíli. Ve variantě "postup01t" je místo úsečky použita obecná křivka $Y(t)$, přičemž F může explicitně záviset i na t ! Pro účely řešení přeurčených soustav (více rovnic než neznámých) byly vytvořeny varianty "postup08mn" a "postup01mn", v nichž jsou rovnice řešeny ve smyslu minimalizace součtu čtverců odchylek.

Všech pět variant má společnou ústřední myšlenku: Postup k cíli na základě dílčích linearizací v rozsahu jejich přibližných platností.

```

%      postup08
%      *****   varianta ..... kdy v y-prostoru
%                  postupujeme postupne podel mnoha usecek
%                  aktualni_postupovy_bod -->-- pozadovany_bod
%
%      reseni soustavy nelinearnich rovnic F(x)=Y
%      postupem podel dilcich linearizaci
%      v rozsahu jejich pribliznych platnosti
%
%      predpoklady :
%                  spojitost prvnych parcialnich derivaci F
%                  vzajemna "numericka zamennost" x-ovych hodnot
%                  vzajemna "numericka zamennost" y-ovych hodnot
%
%      reprezentace funkce F podprogramem FCE :
%
%                  f      = FCE (x,PAR)
%                  [f,df] = FCE (x,PAR)
%
%      x      n*1 argument
%      PAR   ??? individualni parametry ulohy
%
%      f      n*1 resp. [ ] odpovidajici funkcní hodnota
%      df     n*n resp. nic odpovidajici parcialni derivace
%                  |
%                  df(i,j) == dF(i)/dx(j)
%
%      "x" mimo definicni obor
%
function [ X      ... % n*1 nalezene reseni / posledni postup.bod
,   jake  ... % 1*1 kvalitativni informace o ukonceni
,   pocetv ... % 1*2 pocet vycisleni FCE s 1 a 2 vyst.par.
,   prubeh ... % ? n*? mnozina vseh postupovych bodu
] =
postup08 ( Xp    ... % n*1 realistic. pocatecni bod v x-prostoru
,   pY    ... % n*1 pozadovany koncovy bod v y-prostoru
,   FCE   ... % char nazev podprogramu funkce ulohy
,   PAR   ... % ????? individualni parametry ulohy
,   stop  ... % ? 1*? specifikace podminek ukonceni
,   stihl ... % ? 1*1 stihlost postupu  >=2           2
,   jemn  ... % ? 1*1 jemnost postupu  >=3           5
)
%      |
%      % preskakovane mohou byt []
%
%      % mozne explicitne zadane podminky ukonceni postupu - stop
%
%      (1) relativni presnost reseni v y-prostoru           0
%      (2) absolutni presnost reseni v y-prostoru          0
%      (3) pomerne zpresneni reseni v y-prostoru           0
%      | (rezerva)
%      (8) maximalni pripustny pocet postupovych kroku     100
%
global postup0_ % neprazdne => monitorovani

```

```
%.  
%  
% jake      >=0...uspech  <0...neuspech  
% ''''     pricina ukonceni postupu  
%  
% +1       vel(F(X)-pY)/vel(pY)          < stop(1)  
% +2       vel(F(X)-pY)                  < stop(2)  
% +3       vel(F(X)-pY)/vel(F(Xp)-pY)    < stop(3)  
%  
% 0        vel(F(X)-pY)/vel(F(Xp)-pY) = strojova_nula  
%  
% -8       vycerpan limit postupovych kroku stop(8)  
%          ? postupujeme donekonecna (reseni existuje/neexistuje)  
%          ? silna nelinearita  
%  
% -10      preruseni uzivatelem  
%  
% -11      smerovani mimo definicni obor  
%          nebo      extremni vliv zaokrouhlovacich chyb  
%  
% -12      det(dF) = strojova_nula  
%          nebo      rozsah hodnot mimo strojove moznosti  
%  
% -13      chybny podprogram FCE ! nesouhlas hodnot a derivaci  
%          nebo      "zachyt" u bodu lokalniho extremu  
%  
% -20      strojove nulova vzdalenost pocatecniho a pozadovaneho bodu  
% -21      pocatecni bod mimo definicni obor  
% -28      pocatecni resp. koncovy bod mimo strojove moznosti  
%  
% .....
```