Βελτιστοποίηση

Ιωάννης Γ. Τσούλος

Τμήμα Πληροφορικής και τηλεπικοινωνιών
Πανεπιστήμιο Ιωαννίνων

2024

Περίληψη

1. Δεκαδικοί αριθμοί
   - Αναπαράσταση αριθμών
   - Αριθμητικά σφάλματα
2. Εύρεση ριζών
3. Αυτόματη διαφόριση συναρτήσεων

Αριθμοί κινητής υποδιαστολής

1. Ένας αριθμός αναπαρίσταται πάντα στην μορφή $\pm S \times 10^{e}$ $0 \leq S \leq 10$
2. Παράδειγμα $7234.3874 = 7.2343874 \times 10^{3}$
3. Παράδειγμα: $\frac{1}{10} = 0.1 \times 10^{0}$

Μορφή κ-ψηφίων

- Έστω η βάση του συστήματος είναι β
- Κάθε αριθμός δίνεται από την σχέση: $\mathrm{fl}_k(x) = (-1)^s \times (0.d_1 d_2 \ldots d_k)_\beta \times \beta^e$
- k είναι το πλήθος των ψηφίων, s είναι το πρόσημο του αριθμού, e βάση του αριθμού
- Παράδειγμα1: $123.245 = \mathrm{fl}_6(123.245) = (-1)^0 \times (0.12345) \times 10^3$
- Παράδειγμα2: $\mathrm{fl}_6\left(\sqrt{2}\right) = (-1)^0 \times (0.141421) \times 10^1$ (δεν εμφανίζονται όλα τα ψηφία)
- Παράδειγμα3: $\mathrm{fl}_6\left(-9876543\right) = (-1)^1 \times (0.987654) \times 10^7$

IEEE αριθμοί

1. Οι αριθμοί αναπαρίστανται σαν $x = \text{sign} \times \text{fraction} \times \text{base}^{\text{exponet}}$
2. sign είναι το πρόσημο του αριθμού
3. fraction είναι τα ψηφία του αριθμού
4. base είναι η βάση του αριθμητικού συστήματος πχ 2 ή 10.
5. exponent εκθέτης στο αριθμητικό σύστημα

Τύποι δεκαδικών τιμών

1. Υπάρχουν τρεις τύποι δεκαδικών τιμών
2. Στην συνέχεια εμφανίζονται οι διαστάσεις τους σε bits

Τύπος	Συνολικό μήκος	Πρόσημο	Εκθέτης	ψηφία
Single	32	1	8	23
Double	64	1	11	52
Quadruple	128	1	15	112

Ακρίβεια δεκαδικών τιμών

Τύπος	Ακρίβεια (bits)	e_m
Single	24	$2^{-23} \simeq 10^{-7}$
Double	53	$2^{-52} \simeq 2.2 \times 10^{-16}$
Quadruple	113	2^{-112}

isnan και isinf

1. Η συνάρτηση isnan() χρησιμοποιείται στις περισσότερες γλώσσες για να μάθουμε αν ένας αριθμός είναι αόριστος. Πχ, το κλάσμα 0/0 δεν ορίζεται.
2. Η συνάρτηση isinf() χρησιμοποιείται όταν ένας αριθμός αν και υπάρχει δεν μπορεί να αναπαρασταθεί σε υπολογιστή. Πχ. ένας πολύ μεγάλος ή πολύ μικρός αριθμός.
3. Αν και μπορούν να χρησιμοποιηθούν και σε ακέραιες εκφράσεις, συνήθως χρησιμοποιούνται σε δεκαδικές τιμές.

Ορισμοί

- **Σφάλμα αριθμού**: $\text{error}\left(x_{\text{eval}}\right) = x_{\text{true}} - x_{\text{eval}}$
- **Απόλυτο σφάλμα**:$\text{abserror}\left(x_{\text{eval}}\right) = \left\|\text{error}\left(x_{\text{eval}}\right)\right\| = \left\|x_{\text{true}} - x_{\text{eval}}\right\|$
- **Σχετικό Σφάλμα**: $\text{relerror}\left(x_{\text{eval}}\right) = \frac{x_{\text{true}} - x_{\text{eval}}}{x_{\text{true}}}$
- Προσέγγιση αριθμού με k **σημαντικά** ψηφία αν ισχύει: $\text{abserror}\left(x_{\text{eval}}\right) \leq \frac{1}{2} \times 10^{k}$

Δείκτης κατάστασης συνάρτησης

- Δείχνει αν οι υπολογισμοί είναι ευσταθείς για μια συνάρτηση
- Δίνεται από: $E = \left| \frac{f'(x_{true})}{f(x_{true})} x_{true} \right|$
- Παράδειγμα1: $f(x) = x\sqrt{x}$. Δείκτης κατάσταση $3/2$
- Παράδειγμα2: $f(x) = \frac{1}{x^2-4}$. Έχει δείκτη κατάστασης $\frac{2x^2}{x^2-4}$, που σημαίνει ότι κοντά στο ± 2 ο δείκτης κατάστασης παίρνει σχεδόν άπειρες τιμές.

Υπολογισμός ε

Έστ ο αλγόριθμος:

```
eps = 1
one = 1
while(1+eps>one)
{
eps=eps/2;
}
```

Θεώρημα Bolzano

- Αν $y = f(x)$ συνεχής στο $[a, b]$ και ισχύει $f(a)f(b) < 0$,τότε υπάρχει $\xi \in [a, b]$ τέτοιο ώστε $f(\xi) = 0$
- Αν η f(x) διατηρεί την μονοτονία της στο $[a, b]$, τότε αυτή η ρίζα είναι και μοναδική.

Θεώρημα μέσης τιμής

- Αν $y = f(x)$ είναι συνεχής στο $[a, b]$ και διαφορίσιμη, τότε υπάρχει $\xi \in [a, b]$ τέτοιο ώστε

$$\frac{f(b) - f(a)}{b - a} = f'(\xi)$$

Μέθοδος διχοτόμησης

- Εφαρμογή του θεωρήματος bolzano

```
x=(a+b)/2
it=0
while (it<ITMAX) do
if(f(x)=0) then print x; terminate; endif
if(f(a)*f(b)<0) then
b = x
x = (a+b)/2
else
a=x
x=(a+b)/2
end while
```

Μέθοδος εσφαλμένης θέσης

- Η μέθοδος διχοτόμησης δεν λαμβάνει υπόψιν τις τιμές των άκρων f(a), f(b) και επομένως μπορεί να αργήσει

```
x=b-f(b)*(b-a)/(f(b)-f(a))
it=0
while (it<ITMAX) do
if(f(x)=0) then print x; terminate; endif
if(f(a)*f(b)<0) then
b = x
x = b-f(b)*(b-a)/(f(b)-f(a))
else
a=x
x=b-f(b)*(b-a)/(f(b)-f(a))
end while
```

Μέθοδος Newton

1. Η μέθοδος αυτή είναι πιο γρήγορη από τις προηγούμενης γιατί χρησιμοποιεί παράγωγο αλλά και πιο ακριβή σε υπολογισμούς.

```
x=x0
it=0
while(it<ITMAX) do
x=x-f(x)/f'(x)
if(f(x)=0) terminate
it = it+1
end while
```

Παραγώγιση με διαφορές

- Η παράγωγος πολλές φορές δύσκολα υπολογίζεται.
- Μπορεί να υπολολογιστεί έμμεσα με πεπερασμένες διαφορές
- Απαίτηση: η συνάρτηση να είναι συνεχής

Παράγωγος με διαφορές

- $f'(x) \simeq \frac{f(x+h)-f(x)}{h}$
- $f_x(x,y) \simeq \frac{f(x,y+k)-f(x,y-k)}{2k}$
- $f_{xx}(x,y) \simeq \frac{f(x+h,y)-2f(x,y)+f(x-h,y)}{h^2}$
- $f_{yy}(x,y) \simeq \frac{f(x,y+k)-2f(x,y)+f(x,y-k)}{k^2}$
- $f_{xy}(x,y) \simeq \frac{f(x+h,y+k)-f(x+h,y)-f(x,y+k)+2f(x,y)-f(x-h,y)-f(x,y-k)+f(x-h,y-k)}{2hk}$

Η μέθοδος Secant

- Μοιάζει με την μέθοδο Newton αλλά χρησιμοποιεί προσέγγιση για την παράγωγο
- Θέλει δύο αρχικές τιμές x0,x1

```
it=0
while(it<ITMAX)
if(f(x(k))=0) terminate
x(k+1)=x(k)-f(x(k))*(x(k)-x(k-1))/(f(x(k))-f(x(k-1)))
it=it+1
end while
```

Λογική

- Πολλές φορές η παράγωγος υπολογίζεται δύσκολα
- Σε κάποιες περιπτώσεις είναι τεχνικά αδύνατον να υπολογιστεί (πχ αν μεσολαβεί κάποιο άλλο βήμα με άνοιγμα κάποιου αρχείου)
- Υπάρχουν βιβλιοθήκες προγραμματισμού που κάνουν πιο εύκολη την παραπάνω εργασία.
- Συνήθως απαιτούν διατύπωση της συνάρτησης με διαφορετικό τρόπο ή χρησιμοποιούν διαφορετικούς τύπους δεδομένων.

Η βιβλιοθήκη AutoDiff

- Είναι γραμμένη σε C++
- Διαθέσιμη από τον ιστοχώρο `https://github.com/autodiff/autodiff`
- Ο προγραμματιστής πρέπει να χρησιμοποιήσει διαφορετικό τρόπο για να γράψει τις συναρτήσεις
- Απαιτεί την χρήση διαφορετικών τύπων δεδομένων.

Παράδειγμα γραφής συνάρτησης στο AutoDiff

```
dual f(const dual &x, const dual &y, const dual &z)
{
 return (x+y+z)*exp(x*y+z);
}
dual x = 1.0; dual y = 2.0;
dual z = 3.0; dual u = f(x, y, z);
double dudx = derivative(f, wrt(x), at(x, y, z));
double dudy = derivative(f, wrt(y), at(x, y, z));
double dudz = derivative(f, wrt(z), at(x, y, z))
```

Παρατηρήσεις

- Οι συναρτήσεις μπορούν να έχουν οποιοδήποτε αριθμό ορισμάτων
- Ο τύπος double πρέπει να αντικατασταθεί από το **dual**.

Συμβολικός υπολογισμός παραγώγων

- Η αρχική συνάρτηση παρέχεται σε συμβολική μορφή.
- Μεσολαβεί ένας διερμηνευτής που βρίσκει τα σύμβολα της συνάρτησης
- Το τελικό αποτέλεσμα είναι σε συμβολική μορφή.
- Παράδειγμα: η βιβλιοθήκη Ev3 διαθέσιμη από `https://github.com/openturns/ev3`

Παράδειγμα χρήσης της Ev3

```
Ev3::ExpressionParser parser;
parser.SetVariableID("x1", 0);
parser.SetVariableID("x2", 1);
int nerr = 0;
Ev3::Expression expr =
   parser.Parse("x1*sin(x2)", nerr);
Ev3::Expression derivative = Ev3::Diff(expr, 1);
std::cout << derivative->ToString() << std::endl;
```

Σύνοψη

- Παρουσιάστηκαν βασικές έννοιες μαθηματικών
- Παρουσιάστηκαν έννοιες μετατροπής αριθμών.

Βιβλιογραφία I

- Βελτιστοποίηση τεχνικών συστημάτων, Άγγελος Πρωτοπαπάς, 2015, Κάλλιπος, ISBN: 978-960-603-493-0
- Καθολική βελτιστοποίηση: μέθοδοι λογισμικό και εφαρμογές, Ιωάννης Γ. Τσούλος, διδακτορική διατριβή, εθνικό αρχείο διδακτορικών διατριβών.
- Αριθμητικές μέθοδοι βελτιστοποίησης, Ισαάκ Λαγαρής, ιστοσελίδα σημειώσεων διαθέσιμη από http://www.cs.uoi.gr/~lagaris/OPT_UNDER/
- Τεχνικές βελτιστοποίησης, Γεώργιος Α. Ροβιθάκης, 2007, εκδόσεις Τζιόλα, ISBN-13: 978-960-418-141-4 .

Βιβλιογραφία II

Μαθηματική θεωρία βελτιστοποίησης , DingZhu Du, Panos M. Pardalos, Weili Wu, 2005, Εκδόσεις Νέων Τεχνολογιών, ISBN 960-8105-79-X, ISBN-13 978-960-8105-79-9