Algoritam za sumiranje alternirajućih redova*

Saša Singer† Mladen Rogina‡

15. 5. 1986.

Sažetak

Formulira se algoritam za sumiranje vrlo sporo konvergentnih alternirajućih redova. Algoritam je baziran na višestrukoj primjeni postupka usrednjavanja niza parcijalnih suma. Računanje se vrši u formi trokutaste tablice. Na bazi veze između usrednjavanja i Eulerove transformacije polaznog reda pokazuje se konvergencija i daje ocjena pogreške za određenu klasu redova. Analiziraju se numerička svojstva algoritma i pokazuje njegova stabilnost. Za danu klasu redova moguće je naći njihove sume na točnost računala. Slijede rezultati o vremenskoj i prostornoj složenosti algoritma. Daje se detaljan opis efikasne programske realizacije algoritma. Na bazi rezultata na primjeru reda za $\ln 2$ formira se ubrzani algoritam praćenjem pogreške tokom računanja. Dana je usporedba točnosti i složenosti oba algoritma na nekoliko poznatih redova. Na kraju članka opisana je primjena algoritma na računanje Madelungove konstante kubične kristalne rešetke natrijevog klorida.

Abstract

"An algorithm for summation of alternating series": Summation algorithm for slowly convergent alternating series is formulated. The algorithm is based on iterative averageing of partial sum series in form of a triangular table. Connection to Euler transformation is established and convergence of the algorithm is shown. Sharp error estimates are given for a certain class of alternating series. Numerical properties of the algorithm are discussed and unconditional stability is shown. Therefore, machine precision calculation of alternating series sum is possible. Results concerning time and space complexity of the algorithm are given and an efficient program realisation is described. Numerical results for $\ln 2$ series lead to the accelerated algorithm which is based on monitoring the accuracy throughout the computation. Accuracy and complexity comparation for both algorithms is given. An application of the algorithm is presented in which Madelung constant of NaCl-type cubic crystal lattice was calculated to machine precision.

*VIII Međunarodni simpozij "Kompjuter na sveučilištu", Cavtat, 12.–15. 5. 1986.

†PMF–Matematički odjel, Zagreb.

‡PMF–Matematički odjel, Zagreb.

1. Uvod

Problem sumacije redova često se javlja pri računanju vrijednosti funkcija u nekoj točki. Pretpostavimo da se članovi a_n nekog reda mogu lako računati za zadani n. Problem je numerički naći sumu

$$S = \sum_{n=1}^{\infty} a_n$$

tog reda. Ako red brzo konvergira, dovoljno je sumirati prvih nekoliko članova reda i pripadna parcijalna suma

$$S_n = \sum_{k=1}^{n} a_k$$

je aproksimacija sume reda. U praksi, sumacija se vrši član po član, sve dok susjedne parcijalne sume ne postanu dovoljno bliske — na traženu točnost. Ovaj algoritam je prihvatljiv ako broj članova reda koje treba sumirati nije prevelik. Međutim, za sporo konvergentne redove ovaj postupak je nemoguć. U slučaju reda

$$\ln 2 = \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n}$$

trebalo bi sumirati 10^{18} članova za točnost $\ln 2$ na 18 decimala. Tada se, ovisno o svojstvima reda, koriste razne metode za nalaženje njegove sume. Ako red ima sve članove istog znaka najčešće se koriste ekstrapolacije prvih nekoliko članova niza parcijalnih suma (na pr., Aitken, Richardson) ili asimptotske formule (na pr., Euler–Maclaurin) [2]. Za redove čiji članovi alterniraju po znaku, obično se koriste metode usrednjavanja niza parcijalnih suma [1].

2. Metoda usrednjavanja

Najjednostavniji oblik metode usrednjavanja je formiranje aritmetičkih sredina dviju susjednih parcijalnih suma. Označimo $S(n,0) = S_n$ i formiramo niz $S(n,1)$ sa

$$S(n,1) = \frac{S(n,0) + S(n+1,0)}{2}, \quad n \geq 1. \tag{2.1}$$

Ako polazni alternirajući red zadovoljava Leibnitzov kriterij konvergencije (v. [3]), onda je $S(n,1)$ **bolja** aproksimacija sume S od $S(n,0)$ i $S(n+1,0)$, i to tim **bolja**, što je konvergencija članova a_n prema 0 **sporija**. To sugerira ponavljanje usrednjavanja. Iz niza $S(n,1)$ dobivamo niz $S(n,2)$ i tako redom. Definiramo

$$S(n,k) = \frac{S(n,k-1) + S(n+1,k-1)}{2}, \quad n,k \geq 1. \tag{2.2}$$

Niz $S(n,k)$ konvergira po n prema S, za svaki k (v. [1]).

Iz prvih N članova reda možemo izračunati $S(n,k)$ za $k = 0, \ldots, N-1$ i $n = 1, \ldots, N-k$, u formi trokutaste tablice:

$$\begin{array}{cccccc} S(1,0) & S(2,0) & \cdots & S(N-1,0) & S(N,0) \\ S(1,1) & S(2,1) & \cdots & S(N-1,1) & \\ \cdots & \cdots & \cdots & & \\ S(1,N-2) & S(2,N-2) & & & \\ S(1,N-1) & & & & \end{array} \tag{2.3}$$

Problem je koju od vrijednosti $S(n,k)$ treba uzeti kao dobru aproksimaciju sume S polaznog reda. Zbog toga promatramo i Eulerovu transformaciju polaznog reda (v. [1, 4]). Ako članove reda pišemo u obliku

$$a_n = (-1)^{n-1} f_n,$$

tako da su f_n istog znaka, onda vrijedi

$$S(n,k) = S_n + (-1)^n \sum_{i=1}^{k} \frac{(-1)^{i-1}}{2^i} \Delta^{i-1} f_{n+1}. \tag{2.4}$$

To pokazuje da se $S(n,k)$ može dobiti i tako da se prvih n članova reda sumira direktno, a na ostatak s članovima $a_{n+1}, \ldots, a_{n+k}$, tj. sumu

$$(-1)^n \left(f_{n+1} - f_{n+2} + \cdots + (-1)^{k-1} f_{n+k} \right),$$

primijeni se Eulerova transformacija.

3. Konvergencija i ocjena pogreške

Ekvivalencija usrednjavanja i Eulerove transformacije važna je za dokaz konvergencije i ocjenu pogreške. Ako polazni red konvergira, onda konvergira i njegova Eulerova transformacija i to prema istoj sumi [1], pa nizovi $S(n,k)$ konvergiraju prema S po k, za svaki n. To pokazuje da sve vrijednosti $S(N-k,k)$, za $k = 0, \ldots, N-1$, na dijagonali tablice (2.3), također, konvergiraju prema S kad $N \to \infty$.

Za ocjenu pogreške pretpostavljamo da se članovi reda mogu izraziti kao $f_n = f(n)$, za svaki n, gdje je f realna funkcija čije sve derivacije $f^{(k)}(x)$ imaju konstantan znak za $x \geq 1$ i monotono teže prema 0 kad $x \to \infty$. Tada, iz ocjene pogreške Eulerove transformacije [2], slijedi

$$|S - S(n,k)| < \frac{|\Delta^k f_{n+1}|}{2^{k+1}}. \tag{3.1}$$

Ova ocjena nije pogodna za direktnu primjenu u algoritmu usrednjavanja, pa koristimo nešto slabije ocjene

$$|S - S(n,k)| < \frac{|f_1|}{2^{k+1}}. \tag{3.2}$$

Odavde je

$$|S - S(1,N-1)| < \frac{|f_1|}{2^N}, \tag{3.3}$$

pa za aproksimaciju sume S reda uzimamo vrijednost $S(1,N-1)$ **na dnu** tablice (2.3). Vidimo da je s N članova reda moguće naći sumu reda uz relativnu točnost 2^{-N} obzirom na prvi član reda.

4. Numerička stabilnost

Za računanje vrijednosti $S(1,N-1)$ možemo koristiti direktno usrednjavanje parcijalnih suma (2.2) ili Eulerovu transformaciju (2.4). Direktno usrednjavanje je pogodnije jer je numerički idealno stabilno i algoritamski jednostavnije.

Sve vrijednosti $S(n,k)$ u tablici (2.3) su istog znaka i istog reda veličine, osim eventualno u prvim redovima tablice i u slučaju kada je S vrlo blizu 0. Zbog toga, detaljna analiza grešaka zaokruživanja u aritmetici pomičnog zareza pokazuje da je relativna pogreška aritmetičke sredine $S(n,k)$ **istog** reda veličine kao i relativna pogreška polaznih vrijednosti $S(n,k-1)$ i $S(n+1,k-1)$. Dakle, **nema** akumulacije grešaka zaokruživanja, što daje idealnu stabilnost algoritma usrednjavanja.

Ako nađemo članove a_n reda na točnost računala, onda sve vrijednosti $S(n,k)$, također, imaju točnost računala. Zaključujemo da je za navedenu klasu redova moguće naći njihovu sumu na točnost računala.

Dovoljno je odabrati N takav da je 2^{-N} relativna točnost aritmetike računala. To znači da N treba biti jednak broju bitova mantise u prikazu realnih brojeva (u pomičnom zarezu).

5. Složenost algoritma

Algoritam usrednjavanja treba realizirati tako da se u svakom koraku računa nova dijagonala tablice (2.3). To omogućava da se sve vrijednosti $S(n,k)$ pamte u **jednom** polju S duljine N. U n-tom koraku, polje S sadrži vrijednosti s dijagonale tablice na sljedeći način

$$S[k] = S(k, n-k), \quad k = 1, \ldots, n.$$

Zapis algoritma usrednjavanja u Pascalu ima oblik:

```
S[1] := a(1);
for n := 2 to N do
  begin
  S[n] := S[n - 1] + a(n);  {parcijalna suma S_n}
  for k := n - 1 downto 1 do
    S[k] := ( S[k] + S[k + 1] ) / 2;
  end;
suma := S[1];
```

Prostorna složenost algoritma je N lokacija memorije za polje S. Potreban broj operacija je

$$N \text{ računanja vrijednosti članova } a_n \text{ reda, } n = 1, \ldots, N,$$
$$\frac{(N-1)(N+2)}{2} = \frac{1}{2}(N^2 + N - 2) \text{ realnih zbrajanja,}$$
$$\frac{(N-1)N}{2} = \frac{1}{2}(N^2 - N) \text{ dijeljenja s 2.}$$

Najveći dio vremena otpada na dijeljenja s 2, ako zaista koristimo dijeljenje s 2, ili množenja s 0.5. Međutim, ta operacija se može izuzetno efikasno realizirati tako da u prikazu broja s pomičnim zarezom modificiramo binarni eksponent (karakteristiku) za 1, što se svodi na jedno cjelobrojno zbrajanje, a može se realizirati na pr. i u Pascalu. Ova modifikacija znatno skraćuje ukupno vrijeme algoritma. Faktor skraćenja može biti i 3–4 puta!

6. Analiza rezultata

Ovako modificirani algoritam testiran je na redu za $\ln 2$. Sljedeća tablica daje vrijednosti $S(n,k)$ iz prvog stupca i najdulje dijagonale tablice (2.3) za $N = 10$ i pripadne pogreške obzirom na $\ln 2 \approx 0.69314718$ (na 8 decimala):

k	$S(1,k)$	greška	$S(N-k,k)$	greška
0	1.00000000	−0.30685282	0.64563492	0.04751226
1	0.75000000	−0.05685282	0.69563492	−0.00248774
2	0.70833333	−0.01518615	0.69285714	0.00029004
3	0.69791667	−0.00476949	0.69320437	−0.00005718
4	0.69479167	−0.00164449	0.69312996	0.00001722
5	0.69375000	−0.00060282	0.69315476	−0.00000758
6	0.69337798	−0.00023080	0.69314236	0.00000482
7	0.69323847	−0.00009129	0.69315166	−0.00000448
8	0.69318421	−0.00003703	0.69314081	0.00000637
9	0.69316251	−0.00001533	0.69316251	−0.00001533

Iz tablice se može zaključiti da su vrijednosti na dijagonali točnije pri **sredini** dijagonale nego finalni rezultat. To se pokazalo u većini test–primjera. Pritom

apsolutne vrijednosti razlika susjednih elemenata na dijagonali monotono padaju sve do najtočnije vrijednosti ($S(3,7)$ u gornjem primjeru), a potom rastu do finalnog rezultata $S(1,N-1)$. Za takvo ponašanje postoji opravdanje u preciznoj ocjeni pogreške za $S(n,k)$ iz relacije (3.1).

7. Kontrola točnosti – ubrzani algoritam

Uočeno ponašanje vrijednosti na dijagonalama omogućava kontrolu točnosti za $S(n,k)$ praćenjem razlika susjednih elemenata dijagonale.

U svakom koraku ubrzanog algoritma računamo elemente na novoj dijagonali sve dok razlike susjednih elemenata **padaju** po apsolutnoj vrijednosti i stajemo na elementu koji daje **najmanju** razliku. Ovaj postupak ponavljamo sve dok dobivena najmanja razlika ne padne ispod zadane točnosti 2^{-N}. Nađeni element s najmanjom razlikom je aproksimacija sume reda sa zadanom točnošću.

8. Usporedba rezultata

Testiranje oba algoritma obavljeno je na računalu UNIVAC 1100/42 (SRCE, Zagreb), s $N = 60$ članova reda, koliko je potrebno za točnost sume na preciznost računala. Tražena točnost u oba algoritma je $T = 2^{-60} \approx 8.67 \cdot 10^{-19}$, što je i preciznost računala. Sljedeća tablica daje pregled rezultata za nekoliko redova po oba algoritma. Prikazana je točnost, broj izračunatih članova tablice (2.3) (bez prvog reda) i član koji daje sumu.

(a) $f_n = \dfrac{1}{n}$, točna suma = 0.69314 71805 59945 308

algoritam	suma	točnost	članova
osnovni	$S(\ 1,59) = 0.69314\,71805\,59945\,300$	$10\,T$	1770
ubrzani	$S(19,20) = 0.69314\,71805\,59945\,307$	$3\,T$	553

(b) $f_n = \dfrac{1}{2n-1}$, točna suma = 0.78539 81633 97448 307

algoritam	suma	točnost	članova
osnovni	$S(\ 1,59) = 0.78539\,81633\,97448\,301$	$8\,T$	1770
ubrzani	$S(19,20) = 0.78539\,81633\,97448\,305$	$2\,T$	539

(c) $f_n = \dfrac{1}{n^2}$, točna suma = 0.82246 70334 24113 211

algoritam	suma	točnost	članova
osnovni	$S(\ 1,59) = 0.82246\,70334\,24113\,207$	$6\,T$	1770
ubrzani	$S(21,19) = 0.82246\,70334\,24113\,211$	$0\,T$	555

Ovi rezultati potvrđuju izuzetnu efikasnost ubrzanog algoritma.

9. Madelungova konstanta

Madelungova konstanta kubične kristalne rešetke natrijevog klorida je definirana s

$$M = \sum_{z=-\infty}^{+\infty} \sum_{y=-\infty}^{+\infty} \sum_{x=-\infty}^{+\infty} f(x,y,z),$$

gdje je

$$f(x,y,z) = \frac{(-1)^{x+y+z}}{\sqrt{x^2+y^2+z^2}}.$$

Pritom su x, y, z cjelobrojne prostorne koordinate, a točka $x = y = z = 0$ se ne uzima u obzir. Korištenjem simetrije možemo pisati

$$M = 8S_z + 12S_y + 6S_x,$$

gdje je:

$$S_z = \sum_{z=1}^{\infty} Sz(z),$$

$$Sz(z) = \sum_{y=1}^{\infty} S(y,z), \quad S_y = Sz(0),$$

$$S(y,z) = \sum_{x=1}^{\infty} f(x,y,z), \quad S_x = S(0,0).$$

Primijetimo da je $S_x = -\ln 2$, pa je M trodimenzionalna generalizacija reda za $\ln 2$. Algoritmom usrednjavanja redom računamo $S(y,z)$ za $0 \le y,z \le N$, $S(z)$ za $0 \le z \le N$, i na kraju S_z, uz $N = 60$ na točnost računala. Osnovni algoritam bi zahtijevao oko $N^4/2 = 6\,480\,000$ aritmetičkih sredina, dok ubrzanim algoritmom uz $1\,200\,823$ aritmetičkih sredina, za 151 sekundu strojnog vremena dobivamo rezultat

$$M = -1.74756\,45946\,33182\,11.$$

Točna vrijednost je [5]

$$M = -1.74756\,45946\,33182\,16.$$

Time je direktnom sumacijom dobivena vrlo točna vrijednost, što se dosad smatralo praktički nemogućim.

Zahvaljujemo M. Mirniku i D. Babiću koji su nas upoznali s problemom računanja Madelungove konstante, što je iniciralo ovaj rad.

Literatura

[1] G. H. Hardy, *Divergent Series*, Oxford University Press, Oxford, 1949.

[2] T. B. Hildebrand, *Introduction to Numerical Analysis*, McGraw–Hill, New York, 1956.

[3] S. Kurepa, *Matematička analiza 2 (3. izdanje)*, Tehnička knjiga, Zagreb, 1984.

[4] M. Rogina, *O jednoj metodi brze sumacije redova*, Matematika, 4 (1982).

[5] Y. Sakamoto, *Madelung Constants of Simple Crystals Expressed in Terms of Born's Basic Potentials of 15 Figures*, J. Chem. Phys., 28 (1958), str. 164–165.