Megbízható számítások

Csendes Tibor

A cím kissé rejtélyes: a számítógépes algoritmusok a köztudat szerint pontosak és megbízhatók. Az alku során az ügynökök végső érve gyakran az, hogy a kalkulátor is a mondott számot mutatja. Ezzel szemben sajnos fontos feladatok megoldása során is azzal szembesülünk, hogy a kapott eredmény csak közelítő érték, és gyakran kritikus esetben a kapott szám nagyságrendje, vagy az előjele sem helyes. Ennek ellenére van olyan számítógépes módszertan, amely biztosítani tudja a numerikus számítások tetszőleges pontosságát és megbízhatóságát. Ez úgy érhető el, hogy a többet kell számolni, és tovább tart a megoldás. Gyakran a gép memóriája is nagy kell hogy legyen.

A pontatlanság egyik oka: Adjunk össze 3 számot, majd igazoljuk, hogy ha ezt lebegőpontos számítógépes rendszerben tesszük, akkor az eredmény lényegében tetszőlegesen távol lehet a helyestől! Matlabban például a következőt tapasztalhatjuk. Az

```
>> A+B+C
```

utasításra a $10^{23}, 2016, -10^{23}$ értékekkel az eredmény 0, a helyes nyilván 2016. Érdekes, hogy mind az adatok, mind az eredmények lebegőpontos környezetben ábrázolható számok.

Természetesen több megoldás is van erre a problémára. Ilyen a részeredményeknek több lebegőpontos számba való gyűjtése, az ún. error free transformation (EFT), vagy az intervallum aritmetikán alapuló befoglaló függvények használata.

Legyen $\mathbb{I}$ a kompakt valós intervallumok tere. Az intervallum-aritmetika műveletei ezen a halmazon vannak értelmezve. A műveleteket úgy kell definiálni, hogy az $A \circ B$ eredménye egy olyan C intervallum legyen, amely pontosan azon c valós számok halmaza, amelyekhez léteznek olyan $a \in A$ és $b \in B$ valósok, hogy $c = a \circ b$. Itt $\circ$ a négy alapművelet valamelyikét jelöli. Az ilyen aritmetika segítségével követni lehet a kerekítési hibákat, és az adatainkat terhelő bizonytalanság is tükröződhet az eredményekben.

Az előző definíció mellett az intervallum-aritmetikát lehet kizárólag a valós aritmetikára támaszkodva is definiálni. Az $[a,\ b]$ és $[c,\ d]$ intervallumokra legyen

$$[a,\ b] + [c,\ d] = [a + c,\ b + d],$$

$$[a,\ b] - [c,\ d] = [a - d,\ b - c],$$

$$[a,\ b][c,\ d] = [\min(ac, ad, bc, bd),\ \max(ac, ad, bc, bd)],$$
$$[a,\ b]/[c,\ d] = [a,\ b][1/d,\ 1/c].$$

Az osztást persze csak akkor értelmezzük, ha $0 \notin [c,\ d]$. Érdemes megjegyezni, hogy ez utóbbi feltétel jól megfogalmazott gyakorlati feladatokban a tapasztalataink szerint szinte kivétel nélkül teljesül. A valós műveleteknek ezt a kiterjesztését intervallumokra természetes vagy naiv intervallum-kiterjesztésnek nevezik. Az utóbbi években vizsgálják az olyan intervallum-aritmetikákat is, amelyek nem csak kompakt intervallumokon definiáltak. Ezeken a nullát tartalmazó intervallummal való osztás is értelmezhető. Például $[1, 2]/[-1, 1] = [1, -1] = \mathbb{R}\backslash(-1, 1)$.

Bár az alapműveletek pontosak a fenti értelemben, mégis, a velük kiszámított bonyolultabb függvények durva becslései is lehetnek a megfelelő értékkészletnek. A gyakran emlegetett példa a következő: az $x - x^2$ értékkészlete a $[0,\ 2]$ intervallumon $[-2,\ 0,25]$. Ezzel szemben az intervallum-kiterjesztéssel adódó intervallum $[-4,\ 2]$.

Az intervallum-aritmetika műveleteinek tulajdonságaival foglalkozik az intervallum-algebra. Számos, a valós műveletekre érvényes tulajdonság változatlanul teljesül az intervallum-műveletekre is (pl. a kommutativitás, asszociativitás az összeadásra és a szorzásra), de általában nincs inverz, és érvényes a szubdisztribúciós tulajdonság: $A(B + C) \subseteq AB + AC$.

Az alapműveletekhez hasonlóan könnyen lehet definiálni az elemi függvények intervallum-kiterjesztését is, tehát a számítógépen kiszámítható függvényeket szinte kivétel nélkül meg lehet valósítani természetes intervallum-kiterjesztésben is.

Az intervallum-aritmetika alkalmazása szempontjából alapvető fogalom a befoglaló függvény. Az $F(X) : \mathbb{I}^n \to \mathbb{I}$ az $f(x)$ n-változós valós függvény befoglaló függvénye, ha $f(x) \in F(X)$ érvényes minden $x \in X$ pontra és $X \in \mathbb{I}^n$ intervallumra. Az intervallum-matematika fontos eredménye, hogy az $f(x)$ valós függvényből természetes (vagy naiv) intervallum-kiterjesztéssel adódó $F(X)$ függvény befoglaló függvény.

A befoglaló függvényektől természetes azt elvárni, hogy bővebb argumentum-intervallumra ne adjanak szűkebb eredmény-intervallumot. Ezt a feltételt fogalmazza meg az izotonitás: egy $F(X)$ befoglaló függvény akkor izoton, ha $X \subseteq Y$-ból következik $F(X) \subseteq F(Y)$. Az izotonitás szinte minden intervallum-aritmetika implementációra érvényes.

A befoglaló függvények minőségének fontos mutatója a rend: azt mondjuk, hogy az $F(X)$ befoglaló függvény rendje $\alpha > 0$, ha létezik olyan c valós konstans, hogy $w(F(X)) - w(f(X)) \leq cw(X)^\alpha$ teljesül minden $X \in \mathbb{I}^n$-re, ahol $w(X)$ az X intervallum szélessége, és $f(X)$ az $f(x)$ értékkészlete az X intervallumon. A természetes intervallum-kiterjesztéssel adódó befoglaló függvények elsőrendűek, de kidolgozott a magasabbrendű befoglaló függvények elmélete is. Az egynél szélesebb intervallumokra a természetes intervallum-kiterjesztést, a kisebbekre pedig a magasabbrendű befoglaló függvényeket szokták ajánlani.

A számítógépes megvalósítás során minden intervallum-művelet végrehajtása után a kapott intervallumot módosítani szokás. Az intervallum alsó határát lefelé, felső határát felfelé kell kerekíteni a legközelebbi ábrázolható számra. Ezzel az úgynevezett *kifelé kerekítés*i eljárással el lehet érni, hogy a befoglalási tulajdonság a kerekítési hibák ellenére is fennmaradjon. Ezen a módon számítógéppel automatizálható a garantált megbízhatóságú befoglaló függvények előállítása.

Az intervallum-aritmetikához használatos speciális kerekítéseket az IEEE 754-1985 szabvány biztosítja, ezeket napjaink szinte minden számítógépes processzora támogatja. A hetvenes évek közepétől elérhetők olyan programozási nyelvek, amelyek az INTERVAL adattípus használatát támogatják. Ilyen nyelveken (mint például a C-XSC) még az intervallum-aritmetikát megvalósító szubrutinokat sem kell megírni: a megfelelő befoglaló függvény implementálásához elegendő a függvény kiszámításához használt változók típusát megváltoztatni.

A befoglaló függvényekre támaszkodó numerikus algoritmusok érzékenyek a befoglaló függvény minőségére, pontosságára. A vázolt természetes intervallum-kiterjesztés mellett számos más eljárás is ismert a befoglaló függvények előállítására, például a magasabbrendű deriváltakat is használó ún. középponti alakok, az automatikus deriválásra és monotonitás-vizsgálatra épülő stratégiák a befoglaló függvény javítására, illetve az optimális pontosságú befoglaló függvényt generáló eljárás. Ezek a módosítások természetesen növelik az egy befoglaló függvény kiértékeléséhez szükséges számítások mennyiségét.

Az intervallum matematikát részletesen tárgyaló jegyzet vagy magyar nyelvű irodalom sajnos még nincs. Angol és német (esetleg orosz) nyelvű bevezető könyveket tudok ajánlani:

- G. Alefeld, J. Herzberger: Einführung in die Intervallrechnung, Bibliographises Institut AG, Mannheim, 1974.
- G. Alefeld, J. Herzberger: Introduction to Interval Computations, Academic Press, New York, 1983.
- H. Ratschek, J. Rokne: Computer Methods for the Range of Functions, Ellis Horwood Ltd., Chichester, 1984.
- S.A. Kalmikov, Yu.I. Sokin, Z.H. Yuldashev: Az intervallum-analízis módszerei (oroszul), Nauka, 1986.
- H. Ratschek, J. Rokne: New Computer Methods for Global Optimization, Ellis Horwood Ltd., Chichester, 1988.

Egy pozitív példa: A SIAM (Ipari és Alkalmazott Matematikai) Társaság 2002-ben 10 numerikus feladatot tűzött ki[1]. Feladatonként 10 helyes decimális jeggyel 100 dollárt lehetett nyerni. A negyedik megadott feladat a következő függvény minimalizálása volt:

$$\exp(\sin(50x)) + \sin(60e^y) + \sin(70\sin(x)) + \sin(\sin(80y)) -$$

[1] Nick Trefethen: A Hundred-Dollar, Hundred-digit Challenge. SIAM News 35(2002)

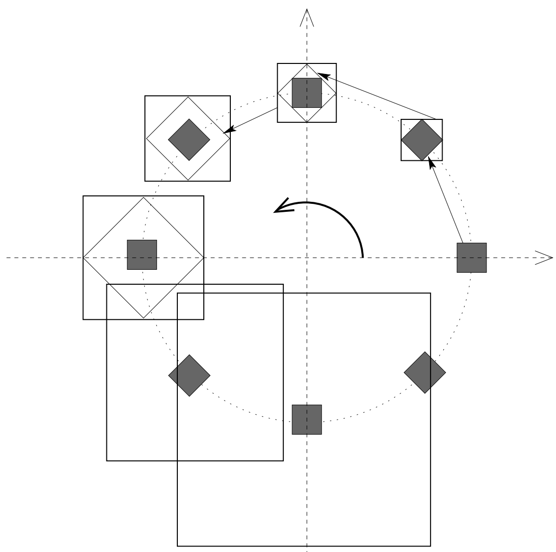
$$-\sin(10(x+y)) + \frac{1}{4}(x^2+y^2).$$

A feladat megoldására egy intervallum aritmetikára alapuló korlátozás és szétválasztás módszert használtunk. A kapott eredmény a $[-10.0, 10.0]$ keresési tartományon a globális minimum értékére a következő alsó- és felső korlátokat adta:

$$[\underline{-3.306868647475}316, \underline{-3.306868647475}196].$$

Az eredményben a kiemelt első 13 jegy matematikai bizonyítóerővel igazoltan helyes. Ehhez mindössze 0.26 másodperc CPU-idő, minimális memóriaigény (75 részintervallum tárolására volt szükség), 1975 célfüggvény-, 1158 gradiens- és 92 Hesse-mátrix kiértékelés kellett.

Az úgynevezett "wrapping effect" a differenciálegyenletek megoldását nehezíti, még pontos számítás esetén is a kapott eredmény intervallum mérete nagyon nagy lehet:



Az ábrán azt kell megfigyelni, hogy a jobb szélen lévő négyzet körbeforgatásával a befoglaló intervallumok mérete reménytelenül nő. Ennek az oka a rögzített koordinátarendszerben való intervallumos befoglalás.

Mikor nem kell a megbízhatóság?

- Olyan gyakorlati feladatok esetén, amikor költség vagy haszon jellegű a célfüggvény. Ilyenkor a jó közelítő megoldások elegendők.
- Ilyen H.-P. Schwefel példája: A Siemens megbízásából atomerőművek számára kellett a fűtőelemek helyét meghatároznia úgy, hogy javuljon a

hatékonyság. Az evolúciós módszerrel talált megoldás több mint 1%-al volt jobb, mint a korábban ismert. Bár nem tudja, hogy a talált közelítés akár csak helyileg optimális-e, és azt sem, hogy milyen messze van az optimumtól, a megbízó elégedett volt ($> 10^8$ DEM).

Mikor kell a megbízhatóság?

- Elméleti állítások igazolására mint például a
  - a Kepler-sejtés,
  - a Fekete-feladat,
  - n-dimenziós gömbhöz illeszkedő azonos méretű gömbök száma (kissing number),
  - körpakolási feladatok, vagy:
  - $\min\,(a^n + b^n - c^n)^2 + \sin^2 a\pi + \sin^2 b\pi + \sin^2 c\pi + \sin^2 n\pi$ ahol a, b, c nemnegatív és n nagyobb mint kettő.
- egyes olyan gyakorlati feladatok esetén, ahol a közelítő megoldás ismerete haszontalan (pl. van-e olcsóbb termelés...),
- kritikus alkalmazásokban, mint a műtéti robotok irányítása.

Az érdeklődők számára további anyagok:

A Tudomány Határai, Kossuth Rádió, 2016. VIII. 27., 14:32, interjú Bánhelyi Balázzsal, Csendes Tiborral és Krisztin Tiborral.
`http://www.mediaklikk.hu/radio-lejatszo-kossuth/?date=2016-08-27_14:32:00&ch=mr1`

Csendes Tibor: Egy intervallum-aritmetikán alapuló algoritmus a szinthalmazok korlátainak megkeresésére. Alkalmazott Matematikai Lapok 17(1993) 19-40, `http://real-j.mtak.hu/456/1/ALKMAT_17.pdf`

Csendes Tibor: Közelítő és szimbolikus számítások. Polygon, Szeged, 2007

Csendes Tibor: Számítás garantált pontossággal I., Számábrázolási lehetőségek, Természet Világa 132(2001) 132-134
`http://www.termeszetvilaga.hu/tv2001/tv0103/tartalom.html`

Csendes Tibor: Számítás garantált pontossággal II., Természet Világa 132(2001) 180-183
`http://www.termeszetvilaga.hu/tv2001/tv0104/tartalom.html`

Csendes Tibor: Megbízható Számítógépes Eljárások, Szabadegyetemi előadás,
`https://www.u-szeged.hu/oktatas/megbizhato-szamitogepes/megbizhato-szamitogepes`

Pál László és Csendes Tibor: Egy intervallum alapú globális optimalizálási módszer és alkalmazása szenzor lokalizálási feladatra. Alkalmazott Matematikai Lapok 28(2011) 17-39,
`http://aml.math.bme.hu/wp-content/uploads/2012/06/28-pal_csendes.pdf`

A szerző a Szegedi Tudományegyetem Számítógépes Optimalizálás Tanszéke egyetemi tanára, a fenn megadott cikkek egy része elérhető a

`http://www.inf.u-szeged.hu/~csendes/publh.html`

oldalon is.