Közönséges differenciálegyenletek megoldása

Bevezetés

A differenciálegyenlet *rendje* mondja meg, hogy mi a legmagasabb rendű derivált. Belátható, hogy megfelelő új változók bevezetésével minden magasabb rendű differenciálegyenlet átalakítható egy *csatolt* első rendű differenciálegyenletrendszere. A csatoltság azt jelenti, hogy ami az egyik egyenletben a jobb oldalon szerepel, annak a deriváltja előfordul egy másik egyenlet baloldalán. Pl. a

$$\frac{d^2y}{dx^2} + q(x)\frac{dy}{dx} = r(x)$$

másodrendű egyenlet a $z(x) = dy/dx$ új változóval a következő elsőrendű differenciálegyenletre írható át:

$$\frac{dy}{dx} = z(x)$$

$$\frac{dz}{dx} = r(x) - q(x)z(x)$$

Néha praktikus nem egyszerűen a változó deriváltjait bevezetni új változónak, hanem annak valamilyen függvényét pl. a független változóval kombinálva. Ilyenkor gondolni kell a számábrázolásból adódó esetleges problémákra, különösen túlcsordulás fordulhat elő olyan folyamatokban ahol gyors változások vannak.

Általános alakban tehát így írható fel a megoldandó probléma:

$$\frac{dy_i(x)}{dx} = f_i(x, y_1, ..., y_N), \qquad i = 1, ..., N$$

Az egyenletek jobb oldala adott. Cél az $y_i(x)$-k megadása tetszőleges x esetén, tipikusan egy végső pontban vagyunk rá kíváncsiak, vagy meghatározott diszkrét időpontokban. Ahhoz hogy a problémának egyértelmű megoldását kapjuk, szükség van még a peremfeltételek rögzítésére is. Ez leggyakrabban a kezdeti feltételek rögzítését jelenti $(y_i(x = 0) = y_{i0})$. Előfordul azonban, hogy ezek valamilyen algebrai kifejezése adott csak, sőt az is lehetséges, hogy a különböző értékek nem azonos időpontokban ismertek (pl. tudjuk hogy a kezdeti időpontban melyik helyről indul a részecske és ismerjük a végső időpontban sebességét, és ezek segítségével kell a pályát meghatározni). Itt az egyszerűbb első esetet tárgyaljuk, a második típus megoldására találhatók iterációs módszerek [1]-ben. A továbbiakban a fenti differenciálegyenletrendszerből csak egynek a megoldását írjuk fel mindig, és f-nek csak egy y-tól való függését. Az egyenletek indexe helyett a diszkrét lépéseket írjuk ki. Az egyenletrendszer egyenleteit párhuzamosan kell megoldanunk, minden y_i-vel párhuzamosan ugyanazt kell tenni..

$$y_{i,n+1} - y_{i,n} = hf_i(x_n, y_{1,n}, ...y_{N,n}) \qquad helyett$$

$$y_{n+1} - y_n = hf(x_n, y_n)$$

A differenciálegyenletek numerikus megoldása az úgynevezett *Euler módszer*en alapul. Ennek lényege az, hogy a differenciálegyenlteket átírjuk dx lépések helyett véges $h = \Delta x$ lépésű differenciaegyenletekre. Így léptetjük a differenciaegyenletet, és ha a lépésköz kellően kicsi, a valódi megoldást közelítjük. Ennek a módszernek direkt implementálása általában nem optimális.

Az alább leírt *Runge-Kutta* módszer több Euler lépés kombinációjából állít elő egylépést, úgy, hogy a diszkretizálásból származó hiba h-ban másodrendűről ötöd- vagy magasabbrendűre csökken. A módszer sebessége és pontossága javítható, ha a lépésközt adaptív módon annak megfelelően változtatjuk, mekora változások történnek éppen az adott lépésben. Hitrtelen nagy változásoknál kisebb lépésköz szükséges, míg lassú változások esetén nagy lépésköz is elegendő pontosságot adhat és felgyorsíthatja az algoritmust. Ez a módszer klasszikusnak nevezhető, olyan esetekben azonban amikor nagy pontosságra van szükség célszerű használni a *Richardson extrapolációs* (vagy *Bulirsch-Stoer*) módszert amely extrapolálja a véges lépésközzel kapott eredményeket a végtelen kicsi lépésközre.

A Runge-Kutta módszer

Az Euler formula

$$y_{n+1} = y_n + hf(x_n, y_n)$$

h-ban elsőrendig pontos, hiszen a Taylor kifejtés értelmében

$$y(x+h) = y(x) + \frac{1}{1!}\frac{dy}{dx}h + \frac{1}{2!}\frac{d^2y}{dx^2}h^2 + ...$$

ez pont a jobb oldal első két tagját adja, vagyis a hiba $O(h^2)$. Az Euler formula fordítva is felírható, vagyis y_n is kifejezhető y_{n+1}-ből. Látjuk azonban az aszimettriát, hogy a derivált (f) mindig az x_n pontban szerepel. Lépjünk e helyett ezzel a deriváltal (k_1) először csak a lépéshossz felével, számoljuk ki ott a derivált értékét, és használjuk ezt a teljes lépésben.

$$k_1 = hf(x_n, y_n)$$

$$k_2 = hf(x_n + 0.5h, y_n + 0.5k_1)$$

$$y_{n+1} = y_n + k_2 + O(h^3)$$

mert

$$y(x+h) = y(x) + hf(x+0.5h, y(x+0.5h)) = y(x) + h\frac{dy}{dx}(x+0.5h) = y(x) + h\frac{dy}{dx}(x) + 0.5h^2\frac{d^2y}{dx^2}(x) + O(h^3)$$

Ezt nevezzük másodrendű Runge-Kutta módszernek.

ábrák.......

Több különböző módon felírhatóak olyan kifejezések, amelyek elsőrendban ugyanazt az eredményt adják, viszont a magasabbrendű korrekciók együtthatói különbözőek. Ezek megfelelő lineárkombinációval kiejthetőek, így csak magasabbrendű korrekció marad. A leggyakrabban használt formula a 4-ed rendű:

$$k_1 = hf(x_n, y_n)$$

$$k_2 = hf(x_n + 0.5h, y_n + 0.5k_1)$$

$$k_3 = hf(x_n + 0.5h, y_n + 0.5k_2)$$

$$k_4 = hf(x_n + h, y_n + k_3)$$

$$y_{n+1} = y_n + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6} + O(h^5)$$

ábra ....

Mivel $\frac{dy}{dx} = f(x,y)$, tekinthetjük ezeket úgy is, hogy az $y(x_n) = y_n$ kezdeti feltételnek eleget tevő differenciálegyenlet integrálját közelítjük valamilyen közelítő integrálformulával (pl. Simpson formula)

$$y(x_{n+1}) = y_n + \int_{x_n}^{x_{n+1}} f(x,y)dx$$

Látható, hogy a fenti 4-ed rendű módszernél négyszer kell kiértékelni a függvényt 1 lépéshez, míg az előtte lévő formulánál csak kétszer kellett. Ez akkor éri meg, ha ugyanakkora pontosság mellettlegalább kétszer akkorát léphetünk. Ez általában teljesül, vagyis érdemes magasabbrendű módszert alkalmazni, de ez általánosan nem bizonyítható.

Adaptív lépéshossz változtatás

Amikor sima, unalmas részszét integráljuka a differenciálegyenletnek, adott pontosság mellett megengedhetünk nagy lépéseket is, míg gyorsan változó szakaszokon kis lépésekben kell haladni. Nyilván nem optimális végig a meredek részek miatt megkövetelt pontossághoz tartozó kis lépésközzel menni. Így nagyságrendi (pl. százszoros) sebességnövekedést is elérhetünk.

Ehhez valahogy folyamatosan figyelni kell a hibát amit egy lépésben elkövetünk. Pl. a 4-ed rendű Runge-Kutta módszernél egy lépést tegyünk meg egyszer teljesen (y_1) egyszer meg két fél lépésben (y_2). Mindegyik lépés 4 függvénykiértékelést igényel (amiből belátható, hogy van kettő amelyik ugyanaz, vagyis összesen 11 kell), viszont így fél lépésnyi pontosságú a felbontásunk. Vagyis 11 függvénykiértékelésünk van a 8 helyett, ugyanazon felbontáshoz. A két eredmény közti különbség

$$\Delta = y_2 - y_1$$

használható a hiba becslésére. Létezik ennél még optimálisabb, kevesebb függvénykiértékeléssel megvalósítható módszer is ami hasonló elvek alapján írható fel (lásd Fehlberg módszer).

A kapott Δ érték segítségével szeretnénk valahogy az optimális lépést megadni. Ha negyed rendű módszert használtunk, akkor Δ skálázása h^5 szerint megy, vagyis ha a próbalépés h_1 eredményeképp Δ_1 hibát kaptunk, akkor azt a h_0 lépésközt, ami az optimális Δ_0 hibát adná így határozhatjuk meg:

$$h_0 = h_1 \left|\frac{\Delta_0}{\Delta_1}\right|^{0.2}$$

Ha több csatolt differenciálegyenletünk van, akkor minden változóra kaphatunk optimális lépést, nyilván a legrosszabbat kell választani. Tehát ha a próbalépés a kívántnál nagyobb hibát adott, akkor az így meghatározott kisebb lépéssel meg kell azt ismételni, ha pedig a hiba kisebb mint a megengedett, akkor a következő lépést már a nagyobb lépésközzel tehetjük meg.

A fent vázolt lépéshossz szbályzással ellátott Runge-Kutta módszer elég stabil, akkor is ha a diferenciálegyenletek nem folytonos függvényeket is tartalmaznak. Ha viszont sima függvényeink vannak csak, akkor van numerikusan optimálisabb, nagyobb pontosságot lehetővé tevő módszr is:

A Richardson extrapoláció és a Bulirsch-Stoer módszer

A módszer lényege az, hogy megpróbálunk nagyokat lépni (H) úgy hogy a lépés végén lévő értéket kiszámoljuk egyre finomabb $H = nh$ lépésközű felbontásánál, és a kapott eredményekre extrapolálunk a $h = 0$ esetre. Az extrapolációhoz általában polinomot vagy racionális függvényt célszerű választani.

A szükséges függvénykiértékelések számának minimalizására alkalmazható az ún. módosított középponti módszer (modified midpoint method) :

$$z_0 = y(x)$$

$$z_1 = z_0 + hf(x, z_0)$$

$$z_{m+1} = z_{m-1} + 2hf(x + mh, z_m) \qquad m = 1, 2, ..., n-1$$

$$y(x + H) = y_n = 0.5[z_n + z_{n-1} + hf(x + H, z_n)]$$

Ez hasonlít a Runge-Kuttához, viszont a szélső pontok kivételével a következő pontot mindig a kettővel előtte lévőből állítja elő, úgy hogy az előzőben lévő deriváltat használja. Belátható róla, hogy szemben a RungeüKuttával, ahol egy lépéshez 4 függvénykiértékelés kellett, itt átlagosan 1.5 kell ugyanazon 4-ed rendű hibánál.

Implicit módszerek

Ha olyan lineáris differenciálegyenlet rendszrünk van ahol nagyon különböző skálán szerepel a független változó a különböző függő változók megoldásaiban, akkor gyakran előfordul, hogy a stabilitás miatt irreálisan kis lépéseket kell tenni. Vegyük a

$$y' = -cy \qquad c > 0 \;\; konstans$$

egyenletet, melynek megoldása $y(x) = \exp(-cx)$. Ha az egyszerű Euler formulát írjuk fel, akkor

$$y_{n+1} = y_n + hy'_n = (1 - ch)y_n$$

Ezt *explicit*nek nevezhetjük, mert y_{n+1} direkt módon ki van fejezve az előző lépésbeli értékkel. Nyilván, ha $h > 2/c$ akkor $|y_n|$ a végtelenhez tart, ahogy a lépésszámmal haladunk a végtelen felé, ahelyett, hogy a 0-hoz tartana.

Az ilyen problémák sokkal rejtettebben is előjöhetnek pl.:

$$u' = 998u + 1998v$$

$$v' = -999u - 1999v$$

a $u(0) = 1$, $v(0) = 0$ peremfeltételekkel. A

$$u = 2y - z$$

$$v = -y + z$$

transzformációkkal egyszerűen megkaphatjuk a

$$u = 2e^{-x} - e^{-1000x}$$

$$v = -e^{x} + e^{-1000x}$$

megoldást. Amint az előbb láttuk ennek stabilitásához $h < 2/1000$ kell, pedig nyilván a második tagok teljesen elhanyagolhatóak az elsők mellett, így azoknak kellene beállítani a lépés skáláját.

Vegyük az Euler lépést visszafelé:

$$y_{n+1} = y_n + hy'_{n+1}$$

Itt y_{n+1} szerepel a bal oldalon és a jobb oldalon is a deriváltban (*implicit*), de pl. az elején felírt egyszerű esetre kifejezhető:

$$y_{n+1} = \frac{y_n}{1 + ch}$$

Mivel c pozitív, láthatjuk, hogy a módszer tetszőlegesen nagy h értékekre stabil.

Ha lineáris differenciálegyenlet rendszerünk van, akkor, ahhoz, hogy az implicit kifejezésből explicit alakot kapjunk, lineáris egyenletrendszert kell megoldanunk minden lépésben. Ez az ára a stabilitásnak, viszont gyakran jobban megéri, mint kis lépéseket választani.

Ha nemlineáris függvények is szerepelnek, akkor a helyzet még rosszabb, nemlineáris egyenletrendszerekre ugyanis csak iteratív módszerek vannak. Meg lehet próbálkozni (ha h nem túl nagy) a probléma linearizálásával. Ez az ún. *szemi-implicit Euler módszer*.

Természetesen az implicit módszerek is javíthatóak, csökkenthető a hiba rendje, hasonlóan mint a Runge-Kutta (Rosenbrock vagy Kaps-Rentrop módszer) illetve a Bulirsch-Stoer (Bader-Deuflhard módszer) módszer (lásd [1]).