

# Numerikus módszerek és optimalizálás

## 2. előadás

## A Fibonacci keresés

Az aranymetszéses eljárásnál láttuk, hogy a bizonytalansági intervallumok hossza minden lépésben azonos arányban csökken, azaz  $L_{k+1} = \alpha L_k$ . A Fibonacci módszernél a redukciós arány lépésről lépésre változik, de megmarad az aranymetszéses módszernek az a jó tulajdonsága, hogy az első intervallumban kettő, a többi intervallumban pedig csak egy függvény-kiértékelésre van szükség. Az eljárás alapját a Fibonacci számok képezik, amelyeket az alábbi rekurzív formula definiál:

$$\begin{aligned} F_0 &= F_1 = 1, \\ F_k &= F_{k-1} + F_{k-2} \quad k = 2, 3, \dots \end{aligned}$$

A sorozat első néhány eleme:

1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, ...



A módszer elején kiválasztunk egy  $n$  természetes számot, amelyet a lépésszám meghatározására fogunk használni a későbbiekben. A Fibonacci módszernél az  $n$  és a Fibonacci számok segítségével az alábbi formulákkal számítjuk ki az  $[a_k, b_k]$  intervallum belsejében a  $c_k, d_k$  pontokat:

$$c_k = a_k + \frac{F_{n-k-1}}{F_{n-k+1}} L_k,$$
$$d_k = a_k + \frac{F_{n-k}}{F_{n-k+1}} L_k.$$

Az első intervallumbeli közbülső pontok számításánál a nevezőben mindig  $F_n$  áll, a számlálóban pedig attól kettővel illetve eggyel kisebb indexű Fibonacci szám áll. A további lépésekben a Fibonacci számok indexe mindig eggyel csökken.

## Megjegyzés

Akár a baloldali, akár a jobboldali intervallum lesz az új bizonytalansági intervallum, mindkét esetben

$$L_{k+1} = \frac{F_{n-k}}{F_{n-k+1}} L_k,$$

és az intervallumok hosszára igaz a Fibonacci számok képzéséhez formailag hasonló összefüggés

$$L_k = L_{k+1} + L_{k+2}, \quad k = 1, 2, \dots, n-3.$$

Akár a baloldali, akár a jobboldali intervallum lesz az új bizonytalansági intervallum, mindkét esetben az új bizonytalansági intervallumbeli egyik pont megegyezik az előző bizonytalansági intervallum egyik belső pontjával, azaz ha az új bizonytalansági intervallum

$$[a_{k+1}, b_{k+1}] = [c_k, b_k], \text{ akkor } d_{k+1} = c_k,$$

ha pedig az új bizonytalansági intervallum

$$[a_{k+1}, b_{k+1}] = [a_k, d_k], \text{ akkor } c_{k+1} = d_k.$$

Abban az esetben amikor  $k = n - 1$ ,

$$c_{n-1} = d_{n-1}.$$



Ez az állítás azt fejezi ki, hogy egy bizonyos lépés után az algoritmus megáll, mivel a két belső pont megegyezik. Összefoglalva, a Fibonacci módszernél az új bizonytalansági intervallum és az abban lévő két pont számítása a következő: Ha  $f(c_k) > f(d_k)$ , akkor az új bizonytalansági intervallum

$$\begin{aligned} [a_{k+1}, b_{k+1}] &= [c_k, b_k] \\ c_{k+1} &= d_k \\ d_{k+1} &= a_{k+1} + \frac{F_{n-k}}{F_{n-k+1}} L_{k+1}, \end{aligned}$$

Ha  $f(c_k) \leq f(d_k)$ , akkor az új bizonytalansági intervallum

$$\begin{aligned} [a_{k+1}, b_{k+1}] &= [a_k, d_k] \\ c_{k+1} &= a_{k+1} + \frac{F_{n-k-1}}{F_{n-k+1}} L_{k+1} \\ d_{k+1} &= c_k, \end{aligned}$$



## A Fibonacci keresés algoritmus

```
1  INPUT  $[a_1, b_1]$  és  $n$ 
2   $L_1 = b_1 - a_1$ 
3   $c_1 = a_1 + \frac{F_{n-2}}{F_n} L_1$ 
4   $d_1 = a_1 + \frac{F_{n-1}}{F_n} L_1$ 
5  FOR  $k = 1$  TO  $n - 1$  DO
6      IF  $f(c_k) > f(d_k)$ 
7          THEN  $a_{k+1} = c_k, b_{k+1} = b_k, L_{k+1} = b_{k+1} - a_{k+1}$ 
8               $c_{k+1} = d_k, d_{k+1} = a_{k+1} + \frac{F_{n-k}}{F_{n-k+1}} L_{k+1}$ 
9          ELSE  $a_{k+1} = a_k, b_{k+1} = d_k, L_{k+1} = b_{k+1} - a_{k+1}$ 
10              $c_{k+1} = a_{k+1} + \frac{F_{n-k-1}}{F_{n-k+1}} L_{k+1}, d_{k+1} = c_k$ 
```

Az eljárást  $k = n - 1$  lépésig folytatjuk és a  $c_{n-1} = d_{n-1}$  közös értékre választjuk a minimumpontot. Ekkor a hiba, amit elkövethetünk legfeljebb  $L_{n-1}/2$ . Könnyen ellenőrizhető, hogy a közös érték nem más, mint a  $k = n - 1$ -edik intervallum középpontja.

Az  $n$  értékét a következőképpen választjuk: Az első intervallum esetén kettő, a további intervallumokban csak egy függvény-kiértékelés szükséges. Ha az utolsó, azaz a  $k = n - 1$ -edik intervallumban is kiszámítjuk a függvény minimumát, akkor a függvény-kiértékelések száma  $n$ , tehát az előre nem definiált  $n$  a függvény-kiértékelések számát jelenti. Ha előírjuk, hogy a választott minimumpontnak legfeljebb  $\varepsilon > 0$  legyen a hibája, azaz

$$L_{n-1} < 2\varepsilon,$$

akkor az  $n$  értéke ennek a formulának a segítségével



Ezt felhasználva,  $k = n - 2$  helyettesítéssel

$$L_{n-1} = \frac{F_2}{F_n} L_1 = \frac{2}{F_n} L_1,$$

A hibára vonatkozó képlet használható formában az alábbiak szerint írható

$$\frac{L_1}{F_n} < \varepsilon,$$

amelyből az adott kezdeti intervallumhossz ( $L_1$ ) és egy előre megadott pontossági tűrés ( $\varepsilon$ ) ismertében  $n$  értékét úgy kell megválasztani, hogy

$$F_n > \frac{L_1}{\varepsilon}.$$

## Példa

Oldjuk meg az  $f(x) = x^2 - 7x + 12 \rightarrow \min!$  egyváltozós optimalizálási feladatot Fibonacci módszerrel, ha a bizonytalansági intervallum  $[a, b] = [2, 4]$ .

## Példa

Oldjuk meg az  $f(x) = x^2 - 7x + 12 \rightarrow \min!$  egyváltozós optimalizálási feladatot Fibonacci módszerrel, ha a bizonytalansági intervallum  $[a, b] = [2, 4]$ .

## Megoldás

Let  $n = 4$ . Válasszuk meg az  $n$  értékét  $n = 4$ -re. Az algoritmushoz szükséges Fibonacci számok:

$$F_0 = F_1 = 1, F_2 = 2, F_3 = 3, F_4 = 5.$$

1. lépés: Az első közelítésbeli bizonytalansági intervallum  $[a_1, b_1] = [2, 4]$ , melynek hossza  $L_1 = 2$ . A két közbülső pont és a hozzájuk tartozó függvényértékek:

$$c_1 = a_1 + \frac{F_2}{F_4} L_1 = 2 + \frac{2}{5} \cdot 2 = 2.8,$$

$$f(c_1) = 0.24,$$

$$d_1 = a_1 + \frac{F_3}{F_4} L_1 = 2 + \frac{3}{5} \cdot 2 = 3.2,$$

$$f(d_1) = -0.16.$$



## Megoldás

2. lépés: Mivel  $f(c_1) > f(d_1)$ , így az új bizonytalansági intervallum  $[a_2, b_2] = [2.8, 4]$ , melynek hossza  $L_2 = 1.2$ . A két közbülső pont és a hozzájuk tartozó függvényértékek az alábbiak. A  $c_2$  pont az előző intervallumbeli  $d_1$  értékkel azonos.

$$\begin{aligned}c_2 &= d_1 = 3.2 & f(c_2) &= -0.16, \\d_2 &= a_2 + \frac{F_2}{F_3} L_2 = 2.8 + \frac{2}{3} \cdot 1.2 = 3.6, & f(d_2) &= -0.24.\end{aligned}$$

3. lépés: Mivel  $f(c_2) > f(d_2)$ , így az új intervallum  $[a_3, b_3] = [3.2, 4]$ , melynek hossza  $L_3 = 0.8$ . A két közbülső pont és a hozzájuk tartozó függvényértékek az alábbiak. Most a  $c_3$  pont lesz az előző intervallumbeli  $d_2$  értékkel azonos.

$$\begin{aligned}c_3 &= d_2 = 3.6 & f(c_3) &= -0.24, \\d_3 &= a_3 + \frac{F_1}{F_2} L_3 = 3.2 + \frac{1}{2} \cdot 0.8 = 3.6, & f(d_3) &= -0.24.\end{aligned}$$

## Megoldás

4. lépés: Mivel  $c_3 = d_3$ , így az algoritmust be kell fejezni. A minimumpont közelítése  $x_{\min} \approx 3.6$ , vagyis a közbülső pontok értéke.

## Megjegyzés

Megfigyelhető, hogy a bizonytalansági intervallumok hosszának csökkenése minden lépésben a Fibonacci számok hányadosával arányos. Például az  $L_3 = \frac{F_2}{F_3} L_2$ .

Ha az  $n$  értékét önkényesen választjuk meg, akkor - mint ahogy tapasztaltuk - nem tudjuk befolyásolni a pontosságot.

Amennyiben azt szeretnénk, hogy a pontossági tűrés például  $\varepsilon = 0.05$  legyen, akkor az  $n$  értékét olyanra kell választani, hogy fennálljon az

$$F_n > \frac{L_1}{\varepsilon}.$$

összefüggés, ez pedig azt jelenti példánkban, hogy

$F_n > \frac{2}{0.05} = 40$ . A Fibonacci számokból adódik, hogy  $n = 9$ , mivel  $F_9 = 55$ .

Ahhoz, hogy az  $f(x)$  függvénynek minimuma legyen az  $x^*$  pontban szükséges feltétel a következő:  $f'(x^*) = 0$ .

A direkt gyökkereső eljárások lényege az  $f'(x) = 0$  egyenlet gyökeinek a megtalálása, közelítése.

Három gyökkereső algoritmust használhatunk:

- A Newton módszer,
- A kvázi-Newton módszer,
- A szelő módszer.

## Newton módszer

Tekintsük az  $f(x)$  függvény kvadratikus közelítését (a Taylor sort alkalmazva) az  $x = x_k$  pontban:

$$f(x) = f(x_k) + f'(x_k)(x - x_k) + \frac{1}{2}f''(x_k)(x - x_k)^2$$

Ezt az egyenletet deriváljuk. Az így kapott egyenletnek egyenlőnek kell lennie 0-val (ez a minimumpontra vonatkozó feltétel):

$$f'(x) = f'(x_k) + f''(x_k)(x - x_k) = 0$$

Ha  $x_k$  jelöli az  $f$  függvény minimumpontjának közelítését, akkor a következő approximációt kapjuk:

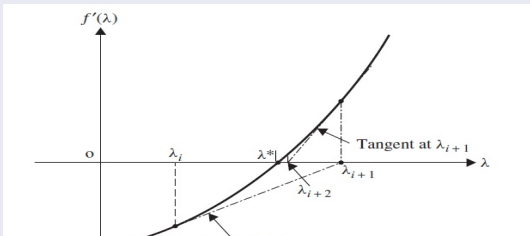
$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)}$$



Ezért a Newton módszer ekvivalens azzal, hogy alkalmazzuk az  $f$  függvény négyzetes közelítését valamint a minimumpontra való szükséges feltételt. Az így kapott iterációs eljárás akkor lesz konvergens, ha a  $f'(x_{k+1})$  derivált közel lesz 0-hoz, azaz

$$|f'(x_{k+1})| \leq \varepsilon$$

ahol  $\varepsilon > 0$  egy adott kicsi hibakorlát. A konvergens eljárás a következő ábrával szemléltethető:

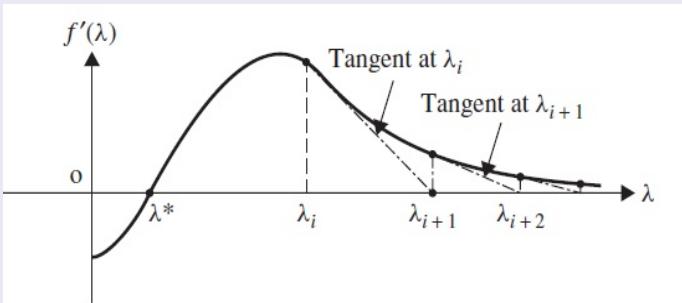


## Megjegyzések

- A Newton módszert a numerikus analízis irodalmában Newton - Raphson módszerként is említik. (Az eljárás alap gondolat Newton-tól származik, de a nemlineeáris egyenlet megoldási módszerét később Raphson újította meg).
- Feltételezzük, hogy az  $f$  függvény kétszer differenciálható.
- Ha  $f''(x_k) = 0$  (a képletben), akkor a Newton módszer a leggyorsabb konvergencia tulajdonsággal rendelkezik.

## Remarks

- Ha a kezdő pontot nem megfelelően válasszuk, akkor a Newton módszer lehet divergens is (és nem közelíti az  $x^*$  megoldást). Például:



### Példa

Keressük a következő függvény minimumát:

$$f(x) = 0.65 - \frac{0.75}{1 + x^2} - 0.65x \operatorname{arctg} \left( \frac{1}{x} \right)$$

Alkalmazzuk a Newton - Raphson módszert az  $x_1 = 0.1$  kezdőpontból kiindulva. Használjuk az  $\varepsilon = 0.01$  értéket a konvergencia ellenőrzéséhez.

## Megoldás

Az első és második deriváltja a függvénynek a következő:

$$f'(x) = \frac{1.5x}{(1+x^2)^2} + \frac{0.65x}{1+x^2} - 0.65 \operatorname{arctg}\left(\frac{1}{x}\right)$$

$$f''(x) = \frac{1.5(1-3x^2)}{(1+x^2)^3} + \frac{0.65(1-x^2)}{(1+x^2)^2} + \frac{0.65}{1+x^2} = \frac{2.8-3.2x^2}{(1+x^2)^3}$$

**1. lépés:**  $x_1 = 0.1$ ,  $f(x_1) = -0.188197$ ,  $f'(x_1) = -0.744832$ ,  $f''(x_1) = 2.68659$ ,

$$x_2 = x_1 - \frac{f'(x_1)}{f''(x_1)} = 0.377241$$

A konvergencia ellenőrzése:  $|f'(x_2)| = |-0.138230| > \varepsilon$ .

## Solution

### 2. lépés:

$$f(x_2) = -0.303279, f'(x_2) = -0.138230, f''(x_2) = 1.57296,$$

$$x_3 = x_2 - \frac{f'(x_2)}{f''(x_2)} = 0.465119.$$

A konvergencia ellenőrzése:  $|f'(x_3)| = |-0.0179078| > \varepsilon$ .

### 3. lépés:

$$f(x_3) = -0.309881, f'(x_3) = -0.0179078, f''(x_3) = 1.17126 \text{ és}$$

$$x_4 = x_3 - \frac{f'(x_3)}{f''(x_3)} = 0.480409.$$

A konvergencia ellenőrzése:  $|f'(x_4)| = |-0.0005033| < \varepsilon$ .

Ezzel az eljárás véget ért, a módszer konvergens és a minimumra kapott közelítés:  $x^* \approx x_4 = 0.480409$

