

Közönséges differenciálegyenlet-rendszer megoldása explicit Euler-módszerrel

A skaláris eset után tekintsünk most egy példát differenciálegyenlet-rendszer numerikus megoldására, szintén az explicit Euler-módszerrel. Egy egyszerű populáció-dinamikai modell a Lotka–Volterra-modell, amely egy préda- és egy ragadozópopuláció (pl. nyulak és rókák) méretének az időbeli fejlődését írja le. A modell differenciálegyenlet-rendszere

$$y_1'(t) = ay_1(t) - by_1(t)y_2(t), \quad (1)$$

$$y_2'(t) = cy_1(t)y_2(t) - dy_2(t); \quad (2)$$

ahol $y_1(t)$ a nyúlpopuláció mérete a t időpillanatban, $y_2(t)$ a rókapopuláció mérete a t időpillanatban, a a nyúlpopuláció növekedési rátája, d pedig a rókapopuláció halálozási rátája. Amikor egy róka és egy nyúl találkozik, akkor a nyulak számának csökkennie kell, a rókák számának pedig nőnie, ezt fejezik ki a b illetve c együtthatók.

Készítsünk Matlab-programot, amely ezt a feladatot megoldja az explicit Euler-módszerrel. Legyen $a = 2$, $b = 1$, $c = 1$, $d = 1$, továbbá $y_1(0) = 1$, $y_2(0) = 0, 1$. Rajzoltassuk is ki a ragadozók és a prédák számának alakulását az idő függvényében különböző színű görbékkel.

Az explicit Euler-módszer képlete rendszerekre formailag hasonló a skaláris esethez. A megoldandó

$$y' = f(t, y), t \in [0, T]$$

$$y(0) = y_0$$

feladatban most az ismeretlen függvény vektorváltozós, és az explicit Euler-módszer képletében y_i szintén vektort jelöl:

$$t_{i+1} = t_i + h, \quad y_{i+1} = y_i + hf(t_i, y_i).$$

Az y tömbben raktározzuk el a numerikus megoldást úgy, hogy az i -edik oszlopa az i -edik időlépcsőbeli megoldást tartalmazza. A skaláris esetre írt programhoz képest az a fő változás, hogy $y(i)$ helyett $y(:,i)$ írandó.

```
function [t,y] = expeuler_rendszer(diffegy,t0,y0,T,N)
% t0: kezdeti időpillanat
% y0: kezdeti vektor
% N: idolepesek szama a numerikus megoldas soran
t=zeros(N+1,1);
```

```

t(1)=0;
y=zeros(length(y0),N+1)
y(:,1)=y0;
for i=1:N
    t(i+1) = t(i) + h;
    y(:,i+1) = y(:,i) + h * diffegy(t(i),y(:,i));
end
plot(t,y)

```

Az utolsó sor plot utasítása a megoldás összes komponensét kirajzolja az idő függvényében. Ha változtatni szeretnénk a görbék stílusán, külön rajzoltassuk ki az egyes komponenseket, vagyis az y mátrix különböző oszlopait, és adjuk meg a megfelelő ábrázolási stílust, pl. (két ismeretlen függvény esetén)

```

plot(t,y(:,1),'r',t,y(:,2),'g:')

```

Ha ezt a programot a Lotka–Volterra-féle rendszer megoldására szeretnénk használni, szükségünk van még a differenciálegyenlet-rendszert megadó alfüggvényre, amelyet LVde.m néven mentünk el az aktuális könyvtárba:

```

function dydt=LVde(t,y)
dydt=zeros(2,1);
dydt(1)=2*y(1)-y(1)*y(2);
dydt(2)=-y(2)+y(1)*y(2);

```

Itt a második sor biztosítja, hogy a kimenő deriváltvektor oszlopvektor legyen. (Ennek a sornak a hiányában az explicit Euler-módszer ciklusában méretbeli problémák lesznek, erről érdemes is meggyőződnünk a `dydt=zeros(2,1)` sor átmeneti kikommentezésével.) Tegyük fel, hogy a feladatot $N = 1000$ db $h = 0.02$ hosszúságú időlépéssel akarjuk megoldani. Ehhez a parancsablakba írjuk a következőket:

```

» [tee,yee] = expeuler_rendszer(@LVde,0,[1 0.1],0.02,1000)

```

Ugyanennek a feladatnak a megoldására mutatunk egy másik lehetőséget is. Ez a program a populációdinamikai feladat megoldására van szabva, míg az előző esetben a főprogram más rendszer megoldására (és nem csak két-, hanem többismeretlenes rendszer esetére)

is használható, ha más alfüggvénnyel hívjuk meg. A változatosság kedvéért h helyett a T időt használjuk bemenő adatként. Az x és y bemenő adat a két populáció kezdeti mérete. Ezeket minden lépésben felülírjuk úgy, hogy az explicit Euler-módszer képletét alkalmazzuk mindkét skaláris differenciálegyenletre (azaz koordinátáinként alkalmazzuk a módszert). Érdeemes megfigyelni, hogy hogyan töltjük be az **xvektor** és **yvektor** nevű vektorokba minden időlépésben az aktuális x ill. y értéket a tömbök bővítésével. A kezdeti időpontot belül állítjuk be nullára, és nem bemenő adatként definiáljuk. A t vektort pedig nem a ciklusban töltjük fel, hanem vektorutasítással. A programon belül adtuk meg továbbá a differenciálegyenletek jobb oldalát meghatározó függvényeket, és a bemenő adatok között nem szerepel függvény.

```
function LVexpeuler2(x,y,T,N)
% x, y: nyul -ill. rokapopulacio merete kezdetben
% T - ido
% N - idolepesek szama a numerikus megoldas soran
h=T/N;
xvektor=x;
yvektor=y;
for n=1:N
    u=f(x,y);
    v=g(x,y);
    x=x+h*u;
    y=y+h*v;
    xvektor=[xvektor,x];
    yvektor=[yvektor,y];
end
t=0:h:T;
plot(t,xvektor,'r', t, yvektor,'b')
xlabel('idő')
ylabel('préda (piros), ragadozó (kék)')
function U=f(x,y)
U=2*x-x.*y;      % ha a = 2 és b = 1
function V=g(x,y)
V=-1*y+1*x.*y;   % ha c = 1 és d = 1
```

Mentsük el, és az `LVexpeuler2(1,0.1,20,1000)` utasítással számoljuk ki az eredményt.