

EÖTVÖS LORÁND TUDOMÁNYEGYETEM

TERMÉSZETTUDOMÁNYI KAR

**KÖZÖNSÉGES DIFFERENCIÁLEGYENLETEK
NUMERIKUS MEGOLDÁSI MÓDSZEREI**

Májusi András

BSc szakdolgozat

Témavezető:

Havasi Ágnes

Adjunktus

Alkalmazott Analízis és Számításmatematikai Tanszék



Budapest, 2017

Köszönetnyilvánítás

Ezúton szeretnék köszönetet mondani témavezetőmnek, Havasi Ágnesnek, hogy elvállalta a konzulensi teendőket, bármikor fordulhattam hozzá, ötleteivel és lelkesedésével segítette szakdolgozatom létrejöttét.

Továbbá köszönetet szeretnék mondani családomnak és barátaimnak, akik tanulmányaim alatt végig biztattak és támogattak.

Tartalomjegyzék

1. Bevezetés	4
2. Közönséges differenciálegyenlet	6
2.1. Példa	7
3. Véges különbséges módszerek	8
3.1. Az Explicit Euler-módszer vizsgálata	11
3.2. Az implicit Euler-módszer vizsgálata	12
3.3. A trapézmódszer vizsgálata	12
3.4. A θ -módszer vizsgálata	13
3.5. Numerikus módszerek stabilitása	14
4. Runge–Kutta típusú módszerek	16
4.1. A kétlépcsős explicit Runge–Kutta-módszer vizsgálata	18
4.2. Butcher-táblázat	19
5. Populációs modellek vizsgálata	21
5.1. Egydimenziós populációnövekedési modellek	21
5.2. Lotka–Volterra-modell	26
6. A numerikus módszerek gépidőigényének vizsgálata	27
7. Összefoglalás	29
8. Függelék	30
8.1. A dolgozat során felhasznált Matlab programkódok	30

1. Bevezetés

A differenciálegyenletek segítségével különböző természettudományos, műszaki illetve közgazdasági folyamatokat tudunk modellezni. Két fő típusukat különböztetjük meg, a közönséges differenciálegyenleteket és a parciális differenciálegyenleteket. A köztük lévő különbség abban rejlik, hogy míg a közönséges differenciálegyenletben az ismeretlen függvény egyváltozós, addig a parciális differenciálegyenletben többváltozós függvényt keresünk. Az egyszerűbb differenciálegyenleteket analitikusan is meg tudjuk oldani, azonban a gyakorlatban felmerülő problémák gyakran nemlineáris egyenletekre vagy egyenletrendszerekre vezetnek, amelyek megoldásait csak közelítőleg tudjuk meghatározni különböző numerikus módszerek segítségével.

Dolgozatomban a közönséges differenciálegyenletekkel foglalkozom, azon belül is ezeknek a numerikus megoldásaival. Először ismertetem a különböző módszereket illetve azok elméleti hátterét, majd modelleken keresztül bemutatom a köztük lévő főbb különbségeket a Matlab segítségével.

Az első fejezetben a közönséges differenciálegyenlet pontos megoldásához kapcsolódó alapvető definíciókat és tételeket tekintjük át. A második fejezetben a közönséges differenciálegyenletek numerikus megoldására szolgáló legelterjedtebb módszer családot, a véges különbséges módszerek alapjaival foglalkozunk. Bevezetjük a legalapvetőbb véges különbséges módszereket, majd a közönséges differenciálegyenletek megoldására szolgáló legegyszerűbb numerikus módszereket, és kiszámoljuk ezek konzisztenciarendjét. Ebben a fejezetben megismerkedünk továbbá a numerikus módszerek stabilitásával is. A harmadik fejezetben a Runge–Kutta típusú módszereket vezetjük be, meghatározzuk néhány módszer konzisztenciarendjét, majd bemutatjuk ezen módszerek táblázatos megadását. A negyedik fejezetben először egydimenziós populációdinamikai modelleken keresztül teszteljük a bemutatott módszereket. Demonstráljuk, hogy a numerikus megoldás minden esetben a kiszámolt elméleti rendnek megfelelően konvergál. Ezek után a két skaláris differenciálegyenletet tartalmazó Lotka–Volterra-modellt oldjuk meg az explicit Euler-módszer segítségével, illusztrálva ezzel, hogy a módszereket nem csak skaláris differenciálegyenlet, hanem rendszer megoldására is alkalmazhatjuk. Az ötödik fejezetben egy egyszerű közönséges differenciálegyenleten és egy nagyméretű rendszeren a különböző Runge–Kutta-

módszerek gépidőigényét tanulmányozzuk. Megmérjük hogy egyre csökkenő lépésközöket választva mennyi időt vesz igénybe a különböző módszerek alkalmazása, és összehasonlítjuk a hatékonyságukat.

2. Közönséges differenciálegyenlet

A közönséges differenciálegyenlet olyan egyenlet, amelyben információnk van egy egyváltozós függvény és annak deriváltja közötti kapcsolatáról. Ebben a fejezetben összegyűjtjük a közönséges differenciálegyenletek pontos megoldásával kapcsolatos legfontosabb fogalmakat és tételeket.

1. Definíció. (Közönséges differenciálegyenlet) Legyen $F : \mathbb{R}^{n+2} \rightarrow \mathbb{R}$ képező folytonos függvény. Ekkor n -edrendű közönséges differenciálegyenletnek nevezzük az $F(t, x(t), x'(t), \dots, x^{(n)}(t)) = 0$ alakú egyenletet.

2. Definíció. (Differenciálegyenlet rendje) A differenciálegyenlet rendjén a benne előforduló legmagasabb derivált rendjét értjük.

3. Definíció. (Közönséges differenciálegyenlet-rendszer) Legyen $f : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ képező folytonos függvény. Elsőrendű közönséges differenciálegyenlet-rendszernek nevezzük az $x'(t) = f(t, x(t))$ alakú differenciálegyenlet-rendszert.

4. Definíció. (Közönséges differenciálegyenlet kezdetiérték-feladata) Legyen $G \subset \mathbb{R} \times \mathbb{R}^d$ egy tartomány (azaz összefüggő, nyílt halmaz), $(t_0, u_0) \in G$ egy adott pont ($t_0 \in \mathbb{R}, u_0 \in \mathbb{R}^n$), $f : G \rightarrow \mathbb{R}^n$ egy folytonos leképezés. Az

$$\begin{cases} u'(t) = f(t, u(t)) & (1) \\ u(t_0) = u_0 & (2) \end{cases}$$

feladatot kezdetiérték-feladatnak, vagy más szóval Cauchy-feladatnak nevezzük.

5. Definíció. Az (1),(2) Cauchy-feladat megoldásán olyan $u : I \rightarrow \mathbb{R}^d$ (I egy nyílt intervallum) folytonosan differenciálható függvényt értünk, amelyre:

- $\{(t, u(t)) : t \in I\} \subset G$ és $t_0 \in G$,
- $u'(t) = f(t, u(t)) \quad \forall t \in I$,
- $u(t_0) = u_0$.

Bizonyos feltételek teljesülése esetén garantálható, hogy a Cauchy-feladat egyértelműen megoldható. Egy ilyen általános feltételt ad a következő tétel.

1. Tétel. *Ha f a második változójában Lipschitz-tulajdonságú T -n, azaz $L \geq 0$ állandó mellett $|f(x, u_1) - f(x, u_2)| \leq L|u_1 - u_2| \forall (x, u_1), (x, u_2) \in T$, akkor a Cauchy-feladatnak létezik egyértelmű megoldása.*

2.1. Példa

Legyen

$$f(t, x(t)) = x(t)$$

azaz tekintsük az

$$x'(t) = x(t), \quad t > 0$$

differenciálegyenletet az

$$x(0) = x_0$$

kezdeti feltétel mellett. Ennek megoldása az

$$x(t) = x_0 e^t$$

függvény, ugyanis

$$x'(t) = x_0(t)e^t = x(t), \text{ és } x(0) = x_0.$$

Ez az egyetlen megoldás, mivel f Lipschitz-tulajdonságú $L = 1$ mellett.

Ezt a fejezetet az [1], [2] jegyzetek segítségével dolgoztuk fel.

3. Véges különbséges módszerek

Egy közönséges differenciálegyenlet kezdetiérték-feladatának megoldását általános esetben nem tudjuk előállítani, ugyanis az csak nagyon speciális f függvény esetén adható meg képlet segítségével. Ezért a gyakorlatban legtöbbször valamilyen numerikus megoldási módszert kell alkalmaznunk.

Ebben a részben olyan típusú eljárások szerepelnek, ahol valamely rögzített időpontbeli közelítés az azt közvetlenül megelőző egy időpontbeli közelítés felhasználásával határozható meg.

Célunk numerikus eljárás konstruálása az előző fejezetben definiált (1), (2) kezdetiérték-feladat megoldására. Ehhez első lépésben definiálunk egy úgynevezett rácshálót a feladat időintervallumán:

$$\omega_\tau = \{t_n = n\tau, n = 0, 1, 2, \dots\}$$

ahol

$$\tau > 0 \text{ adott paraméter.}$$

Tehát ω_τ egy ekvidisztáns rácshálót jelöl, azaz a szomszédos pontok egyenlő távolságra helyezkednek el egymástól. Ezután olyan $y_\tau : \omega_\tau \rightarrow \mathbb{R}$ rácsfüggvényt szeretnénk meghatározni, amelyre

$$y_\tau(t_n) \approx u(t_n)$$

vagyis amely a $t_n \in \omega_\tau$ pontokban "jól közelíti" az $u(t)$ megoldásfüggvényt. (A későbbiekben tisztázzuk, mint értünk jó közelítés alatt.) A kérdés az, hogy az y_τ rácsfüggvényt hogyan határozzuk meg. Ehhez vezessük be a következő jelöléseket:

$$y_\tau(t_n) =: y_n$$

$$u(t_n) =: u_n$$

Mivel u a pontos megoldás, így u speciálisan $t = t_n$ -ben is kielégíti az (1) differenciálegyenletet, vagyis

$$u'(t_n) = f(t_n, u(t_n)) \quad (t_n \in \omega_\tau \text{ tetszőleges})$$

Az elvárásunk az, hogy y_n jól közelítse az $u(t_n)$ pontos megoldást az ω_τ rácspontjaiban. Első lépésben a megoldás t_1 -beli értékét fogjuk közelítőleg meghatározni. Ehhez alkalmazunk közelítő deriválást! A t_0 -beli deriváltat az

$$u'(t_0) \approx \frac{u(t_1) - u(t_0)}{\tau}$$

jobb oldali különbségi hányadossal közelítjük. A bal oldal az (1),(2) feladat szerint $f(t_0, u(t_0))$ -al egyenlő. Itt és a jobb oldalon az $u(t_0) = u_0$ kezdeti feltétel ismert. Az egyenletből az $u(t_1)$ közelítőleg kifejezhető, ez lesz a megoldás t_1 -beli értékének közelítése:

$$u(t_1) \approx f(t_0, y_0)\tau + u(t_0) =: y_1$$

Hasonlóan léphetünk tovább a t_2 pontra:

$$u'(t_1) \approx \frac{u(t_2) - u(t_1)}{\tau} \approx \frac{u(t_2) - y_1}{\tau}.$$

Itt $u'(t_1) = f(t_1, u(t_1)) \approx f(t_1, y_1)$. Ebből

$$u(t_2) \approx f(t_1, y_1)\tau + y_1 =: y_2$$

és így tovább. Így a következő algoritmushoz jutottunk:

$$y_{n+1} = y_n + \tau f(t_n, y_n), \quad n = 0, 1, 2, \dots, N-1.$$

Vagyis y_n -ből y_{n+1} közvetlenül számítható, ezt egy lépéses rekurciónak nevezzük. Azt a módszert, amelynek során a fent leírt módon számítjuk ki a rácsfüggvényt, explicit Euler-módszernek nevezzük.

Azonban a rácsfüggvényt máshogy is meghatározhatjuk. Vegyük észre, hogy az explicit Euler-módszer bal oldalán szereplő hányados egyben a megoldásfüggvény t_{n+1} -beli deriváltjának bal oldali véges különbséges közelítése is. Ezért az explicit Euler-módszer képletében a jobb oldalt cseréljük ki az $f(t_{n+1}, y_{n+1})$ értékre.

$$\begin{aligned} t &= t_{n+1} \\ u'(t_{n+1}) &= f(t_{n+1}, u(t_{n+1})) \approx \\ \frac{y_{n+1} - y_n}{\tau} &= f(t_{n+1}, y_{n+1}) \end{aligned}$$

Az így definiált módszert implicit Euler-módszernek nevezzük.

Fontos még egy módszert megemlíteni az egy lépéses módszer családon belül, ez pedig a trapéz módszer, amelyben a numerikus megoldást meghatározó egyenlőség jobb oldalán $f(t_n, y_n)$ és $f(t_{n+1}, y_{n+1})$ számtani közepe szerepel. Erről később belátjuk, hogy pontosabb, mint az explicit, illetve az implicit Euler-módszer. A trapéz módszer a következő alakban írható fel:

$$\frac{y_{n+1} - y_n}{\tau} = \frac{1}{2}(f(t_n, y_n) + f(t_{n+1}, y_{n+1})) \quad n = 0, 1, \dots \quad (y_0 = u_0)$$

Az eddig ismertetett módszerek a következő, úgynevezett általános θ -módszer speciális esetei:

$$\frac{y_{n+1} - y_n}{\tau} = \theta f(t_n, y_n) + (1 - \theta)f(t_{n+1}, y_{n+1})$$

A $\theta \in [0, 1]$ különböző értékeire különböző módszereket kapunk. Ezekre példák:

- ha $\theta = 1$, az explicit Euler-módszert kapjuk;
- ha $\theta = 0$, az implicit Euler-módszert kapjuk;
- ha $\theta = \frac{1}{2}$, a trapéz módszert kapjuk.

Azonban még mindig kérdés, hogy teljesül-e az $y_n \sim u(t_n)$ közelítés. Tekintsük az ω_τ rácshálók sorozatát $\tau \rightarrow 0$ mellett, valamint vezessük be a következő definíciókat.

6. Definíció. (Konvergencia) Tegyük fel hogy $t^* \in \omega_\tau \forall \tau$ -ra. Azt mondjuk, hogy a numerikus módszer konvergens a $t = t^*$ pontban, ha $\lim_{\tau \rightarrow 0} |y_n - u(t^*)| = 0$, ahol $n\tau = t^*$.

7. Definíció. Azt mondjuk, hogy a numerikus módszer p -ed rendben konvergens, ha $|y_n - u(t^*)| = o(\tau^p)$.

8. Definíció. Azt mondjuk, hogy a numerikus módszer konvergens a $[0, T]$ intervallumon, ha $\forall t^* \in [0, T]$ pontban konvergens.

Minden véges különbséges módszerrel szemben az az egyik alapvető elvárásunk, hogy a numerikus megoldása a fenti értelemben tartson a feladat pontos megoldásához.

3.1. Az Explicit Euler-módszer vizsgálata

9. Definíció. (Hibafüggvény) Az $e_n = y_n - u(t_n)$ rácsfüggvényt hibafüggvénynek vagy globális hibafüggvénynek nevezzük.

Ha behelyettesítjük az explicit Euler-módszer rekurziójába az $y_n = e_n + u_n$ összeget, az

$$\frac{e_{n+1} + u_{n+1} - e_n + u_n}{\tau} = f(t_n, e_n + u_n)$$

összefüggéshez jutunk. Ezt átrendezve

$$\frac{e_{n+1} - e_n}{\tau} = f(t_n, e_n + u_n) - \left[\frac{u_{n+1} - u_n}{\tau} \right]$$

Ha a jobb oldalhoz hozzáadjuk és ki is vonjuk az $f(t_n, u_n)$ értéket, az

$$\frac{e_{n+1} - e_n}{\tau} = \left[f(t_n, u_n) - \frac{u_{n+1} - u_n}{\tau} \right] + [f(t_n, e_n + u_n) - f(t_n, u_n)]$$

egyenlőséghez jutunk. Vezessük be a következő jelöléseket!

$$\begin{aligned} f(t_n, u_n) - \frac{u_{n+1} - u_n}{\tau} &=: \psi_n^{(1)} \\ f(t_n, e_n + u_n) - f(t_n, u_n) &=: \psi_n^{(2)} \end{aligned}$$

Hibaegyenletnek nevezzük az

$$e_{n+1} = e_n + \tau(\psi_n^{(1)} + \psi_n^{(2)})$$

egyenletet, ahol az $\psi_n^{(1)}$ tagot lokális approximációs hibának nevezzük.

10. Definíció. (Konzisztencia) Azt mondjuk, hogy a numerikus módszer konzisztens, ha $\lim_{n \rightarrow \infty} \psi_n^{(1)} = 0$. Ha $\psi_n^{(1)} = o(\tau^p)$, akkor a módszert p -ed rendben konzisztensnek nevezzük.

A konzisztenciarendek vizsgálata során tegyük fel, hogy $u(t)$ kétszer folytonosan differenciálható.

Az explicit Euler-módszer konzisztenciája:

$$\psi_n^{(1)} = -\frac{u_{n+1} - u_n}{\tau} + f(t_n, u_n) = \frac{u(t_{n+1}) - u(t_n)}{\tau} + f(t_n, u(t_n))$$

Fejtsük Taylor-sorba az $u(t_{n+1})$ -et a t_n pont körül:

$$\begin{aligned} u(t_{n+1}) &= u(t_n + \tau) = u(t_n) + u'(t_n)\tau + u''(t_n)\frac{\tau^2}{2} + o(\tau^3) \\ \frac{u(t_{n+1}) - u(t_n)}{\tau} &= u'(t_n) + \frac{\tau}{2}u''(t_n) + o(\tau^2). \end{aligned}$$

A közönséges differenciálegyenletek kezdetiérték-feladatának definíciója alapján:

$$u'(t) = f(t, u(t)) \text{ és } u'(t_n) = f(t_n, u(t_n))$$

Ebből következik, hogy

$$\psi_n^{(1)} = -\frac{\tau}{2}u''(t_n) + o(\tau^2) \equiv o(\tau),$$

vagyis az explicit Euler-módszer elsőrendben konzisztens.

3.2. Az implicit Euler-módszer vizsgálata

Az implicit Euler-módszer esetén a lokális approximációs hiba a következőképpen írható fel:

$$\psi_n^{(1)} = -\frac{u_{n+1} - u_n}{\tau} + f(t_{n+1}, u(t_{n+1}))$$

Továbbá:

$$\frac{u_{n+1} - u_n}{\tau} = \frac{u(t_{n+1}) - u(t_{n+1} - \tau)}{\tau}$$

Fejtsük Taylor-sorba az $u(t_{n+1} - \tau)$ -t a t_{n+1} pont körül:

$$\begin{aligned} u(t_{n+1} - \tau) &= u(t_{n+1}) - \tau u'(t_{n+1}) + \frac{\tau^2}{2}u''(t_{n+1}) + o(\tau^3) \\ \frac{u_{n+1} - u_n}{\tau} &= u'(t_{n+1}) - \frac{\tau}{2}u''(t_{n+1}) + o(\tau^2) \end{aligned}$$

A lokális approximációs hiba:

$$\psi_n^{(1)} = -u'(t_{n+1}) + \frac{\tau}{2}u''(t_{n+1}) + u'(t_{n+1}) + o(\tau^2) \equiv o(\tau).$$

Vagyis az implicit Euler-módszer szintén elsőrendben konzisztens.

3.3. A trapézmódszer vizsgálata

A lokális approximációs hiba a következőképpen írható fel:

$$\psi_n^{(1)} = -\frac{u_{n+1} - u_n}{\tau} + \frac{1}{2}[f(t_n, u(t_n)) + f(t_{n+1}, u(t_{n+1}))]$$

Az explicit Euler-módszernél már láttuk, hogy:

$$\frac{u_{n+1} - u_n}{\tau} = u'(t_n) + \frac{\tau}{2}u''(t_n) + o(\tau^2)$$

Tekintsük a lokális approximációs hiba második tagját:

$$\frac{1}{2}[u'(t_n) + u'(t_{n+1})]$$

Ezt a t_n pont körül sorba fejtve kapjuk, hogy:

$$\begin{aligned} \frac{1}{2}[u'(t_n) + u'(t_n) + \tau u''(t_n) + o(\tau^2)] = \\ u'(t_n) + \frac{\tau}{2}u''(t_n) + o(\tau^2) \equiv o(\tau^2). \end{aligned}$$

Vagyis a trapézmódszer másodrendben konzisztens.

3.4. A θ -módszer vizsgálata

A θ -módszer lokális approximációs hibája a következő:

$$\psi_n^{(1)} = -\frac{u(t_{n+1}) - u(t_n)}{\tau} + \theta f(t_n, u(t_n)) + (1 - \theta)f(t_{n+1}, u(t_{n+1}))$$

Fejtsük Taylor-sorba az $u(t_{n+1})$ -et a t_n pont körül, illetve alkalmazzuk az $f(t_n, u(t_n)) = u'(t_n)$ helyettesítést, ezzel:

$$\psi_n^{(1)} = -u'(t_n) - \frac{\tau}{2}u''(t_n) - \frac{\tau^2}{6}u'''(t_n) + \theta u'(t_n) + (1 - \theta)[u'(t_n) + \frac{\tau}{2}u''(t_n) + \frac{\tau^2}{2}u'''(t_n)] + o(\tau^3)$$

Összevonva a kifejezést, illetve kiemelve a $\left(-\frac{1}{2} - \theta\right)$ és $\left(-\frac{1}{3} - \theta\right)$ különbségeket, a következőhöz jutunk:

$$\begin{aligned} \left(-\frac{1}{2} - \theta\right) (\tau u''(t_n)) + \left(-\frac{1}{3} - \theta\right) \left(\frac{\tau^2}{2}u'''(t_n) + \tau u''(t_n) + \frac{\tau^2}{2}u'''(t_n)\right) + o(\tau^3) = \\ \left(\frac{1}{2} - \theta\right) (\tau u''(t_n)) + \left(\frac{1}{3} - \theta\right) \left(\frac{\tau^2}{2}u'''(t_n)\right) + o(\tau^3). \end{aligned}$$

Tehát, ha $\theta = \frac{1}{2}$, akkor a módszer másodrendű, a θ többi értékére a módszer viszont elsőrendű marad.

3.5. Numerikus módszerek stabilitása

A numerikus megoldás konvergenciájához nem elegendő, ha a módszer konzisztens. Ha azonban emellett teljesül egy további tulajdonság, az ún. 0-stabilitás, akkor a kettő együtt már biztosítja a konvergenciát. A 0-stabilitás azt jelenti, hogy ha a kezdeti feltételt csak kicsit változtatjuk meg, akkor a numerikus megoldások is az egész időintervallumon csak kicsit térnek el egymástól.

A stabilitás vizsgálata során jelölje e_i a globális hibát, vagyis az összes lépés alatt keletkező hibát, d_i pedig a lokális hibát, vagyis az egy lépés alatt keletkező hibát.

2. Tétel. *Egy numerikus módszer 0-stabil, ha $\exists K > 0$, melyre $|e_i| \leq K(|e_0| + \sum_{j=0}^{i-1} |d_j|\tau)$, $\forall i$ $1 \leq i \leq N$.*

3. Tétel. *Ha egy numerikus módszer p -ed rendben konzisztens és 0-stabil, akkor p -ed rendben konvergens.*

Az alkalmazások során fontos a rögzített rácson való stabilitás vizsgálata is. Tekintsük a következő feladatot:

$$\begin{cases} y'(t) = \lambda y(t) & , \lambda \in \mathbb{C}, \operatorname{Re}(\lambda) < 0 \quad t \in (0, +\infty) \\ y(0) = c & \quad c > 0, c \in \mathbb{R} \end{cases}$$

Ezt Dahlquist-féle tesztfeladatnak nevezzük. Világos, hogy ennek a feladatnak a pontos megoldása korlátos, sőt $t \rightarrow \infty$ esetén nullához tart. A numerikus módszertől szintén elvárjuk, hogy rendelkezzen ezzel a tulajdonsággal.

11. Definíció. *Egy numerikus módszer A-stabil, ha a tesztfeladatra minden rögzített $\tau > 0$ esetén kapott y_0, y_1, \dots közelítő sorozatra $\lim_{n \rightarrow \infty} y_n = 0$.*

Legyen

$$z = \lambda\tau.$$

Ekkor egy egylépéses numerikus módszer n -edik lépése, ha a módszert a Dahlquist-féle tesztfeladatra alkalmazzuk:

$$y_{n+1} = R(z)y_n$$

alakban írható fel, ahol az $R(z)$ függvényt a módszer stabilitási függvényének nevezzük.

12. Definíció. Az $R(z)$ stabilitási függvénnyel rendelkező módszer stabilitási tartományán a $\{z \in \mathbb{C} : |R(z)| < 1\}$ komplex számhalmazt értjük.

13. Definíció. Egy módszert A-stabilnak nevezünk, ha a bal oldali félsík része a stabilitási tartománynak.

Vizsgáljuk ezt meg az explicit Euler-módszer esetén. Ha a módszert a Dahlquist-féle teszt-feladatra alkalmazzuk, akkor:

$$y_{n+1} = y_n + \tau f(t_n, y_n) = y_n + \tau \lambda y_n = y_n(1 + \tau \lambda) = y_n(1 + z).$$

Látható, hogy az explicit Euler-módszer stabilitás függvénye:

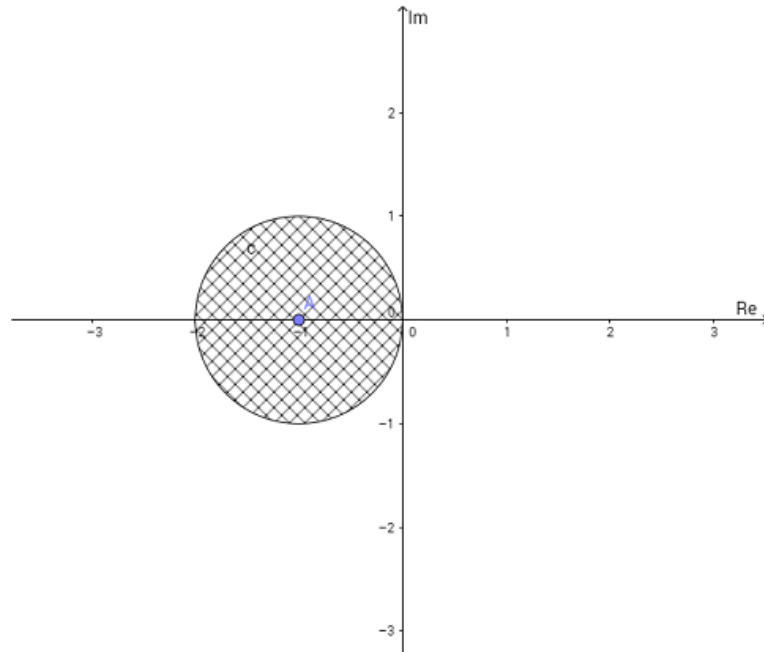
$$R(z) = 1 + z$$

tehát az explicit Euler-módszer stabilitási tartománya azon $z \in \mathbb{C}$ pontok halmaza, amelyekre

$$|1 + z| < 1$$

Ezt az 1. ábra mutatja be. Láthatjuk, hogy a stabilitási tartomány nem foglalja magába a bal oldali félsíkot, tehát az explicit Euler-módszer nem A-stabil. Ez azt jelenti, hogy a Dahlquist-féle feladatra alkalmazva az explicit Euler-módszert, túlságosan hosszú τ lépésközre a megoldás stabilitása elromlik, miközben $t \rightarrow \infty$.

Ezt a fejezetet az [1], [4], [5], [6], [7], [8] jegyzetek segítségével dolgoztuk fel.



1. ábra. Az explicit Euler-módszer stabilitási tartománya

4. Runge–Kutta típusú módszerek

Ebben a fejezetben a Runge-Kutta típusú módszerekkel foglalkozunk. Ezek szintén egy-lépéses módszerek, azonban a numerikus megoldás meghatározásához előbb több úgynevezett lépcsőt kell kiszámolnunk, így magasabb rendű módszereket tudunk definiálni.

Az explicit Euler-módszer rendjén szeretnénk javítani, ezt úgy tudjuk megtenni, hogy először az explicit Euler-módszerrel egy fél lépést megtéve kiszámoljuk az $f(t_{n+\frac{1}{2}}, y_{n+\frac{1}{2}})$ értékét, majd ezzel az új meredekséggel teszünk egy egész lépést az (t_n, y_n) pontból. Az így kapott módszert javított Euler-módszernek nevezzük, melynek képlete:

$$\frac{y_{n+1} - y_n}{\tau} = f(t_{n+\frac{1}{2}}, y_{n+\frac{1}{2}}),$$

ahol az $y_{n+\frac{1}{2}}$ közelítést úgy számoljuk ki, hogy a (t_n, y_n) pontból kiindulva $0,5\tau$ távolságon alkalmazzuk az explicit Euler-módszert, azaz:

$$y_{n+\frac{1}{2}} = y_n + \frac{1}{2}\tau f(t_n, y_n).$$

Így tehát a javított Euler-módszer képlete:

$$y_{n+1} = y_n + \tau f(t_{n+\frac{1}{2}}, y_n + 0, 5\tau f(t_n, y_n))$$

Kérdés az, hogy az így kapott módszer milyen pontosságú. Ehhez határozzuk meg a lokális approximációs hibát, ami a következő:

$$\psi_n^{(1)} = -\frac{u(t_{n+1}) - u(t_n)}{\tau} + f(t_{n+\frac{1}{2}}, u(t_n) + 0, 5\tau f(t_n, u(t_n)))$$

A képlet első feléből a Taylor-sorba fejtés után a

$$-\frac{u(t_{n+1}) - u(t_n)}{\tau} = -u'(t_n) - \frac{\tau}{2}u''(t_n) + o(\tau^2)$$

egyenlőséget kapjuk. Tekintsük a kifejezés második felét:

$$f(t_{n+\frac{1}{2}}, u(t_n) + 0, 5\tau f(t_n, u(t_n)))$$

Ennek átalakításához használjuk fel a következő Taylor-sorfejtést:

$$\begin{aligned} f(x + \Delta x, y + \Delta y) &= f(x, y) + \Delta x \partial_1 f(x, y) + \Delta y \partial_2 f(x, y) + \dots \\ f(t_{n+\frac{1}{2}}, u(t_n) + 0, 5\tau f(t_n, u(t_n))) &= \\ f(t_n, u(t_n)) + 0, 5\tau \partial_1 f(t_n, u(t_n)) + 0, 5\tau f(t_n, u(t_n)) \partial_2 f(t_n, u(t_n)) &+ o(\tau^2) = \\ u'(t_n) + 0, 5\tau [\partial_1 f|_{(t_n, u(t_n))} + f \partial_2 f|_{(t_n, u(t_n))}] &+ o(\tau^2) \end{aligned}$$

Innen

$$u''(t_n) = \partial_1 f|_{(t, u(t))} + f \partial_2 f|_{(t, u(t))}$$

felhasználásával az

$$\begin{aligned} u'(t_n) + 0, 5\tau [\partial_1 f|_{(t_n, u(t_n))} + f \partial_2 f|_{(t_n, u(t_n))}] + o(\tau^2) &= u'(t_n) + 0, 5\tau u''(t_n) + o(\tau^2) \\ \psi_n^{(1)} &\equiv o(\tau^2) \end{aligned}$$

összefüggés adódik. Így a módszer másodrendben konzisztens.

Ezek után próbáljuk meg általánosítani az előző módszert, mégpedig úgy, hogy két rácspont között ne csak egy, hanem s helyen számoljunk ki újabb értékeket. Ehhez vezessük be a következő jelöléseket:

$$\begin{aligned} k_1 &= f(t_n, y_n) \\ k_2 &= f(t_n + a_2\tau, y_n + \tau b_{21}k_1) \\ k_3 &= f(t_n + a_3\tau, y_n + \tau(b_{31}k_1 + b_{32}k_2)) \end{aligned}$$

⋮

$$\begin{aligned}k_s &= f(t_n + a_s\tau, y_n + \tau(b_{s1}k_1 + b_{s2}k_2 + \dots b_{ss-1}k_{s-1})) \\y_{n+1} &= y_n + \tau(\sigma_1k_1 + \sigma_2k_2 + \dots \sigma_s k_s)\end{aligned}$$

ahol $a_1, a_2 \dots a_s, b_{ij}$ és σ_j adott paraméterek.

Az így kapott módszert s -lépcsős Runge-Kutta-módszernek nevezzük. Az

$$a_2 = 0,5$$

$$b_{21} = 0,5$$

$$\sigma_1 = 0$$

$$\sigma_2 = 1$$

$$\begin{aligned}y_{n+1} &= y_n + \tau(\sigma_1k_1 + \sigma_2k_2) \\k_2 &= f(t_n + 0,5\tau, y_n + 0,5\tau k_1)\end{aligned}$$

paraméter-megválasztás mellett a már említett kétlépcsős módszert kapjuk.

4.1. A kétlépcsős explicit Runge–Kutta-módszer vizsgálata

Az s lépéses Runge-Kutta-módszer $s = 2$ esetén az alábbi kétlépcsős Runge-Kutta-módszerre vezet:

$$\begin{aligned}k_1 &= f(t_n, y_n) \\k_2 &= f(t_n + a_2\tau, y_n + \tau b_{21}k_1) \\y_{n+1} &= y_n + \tau(\sigma_1k_1 + \sigma_2k_2)\end{aligned}$$

Vagy másképpen:

$$\frac{y_{n+1} - y_n}{\tau} = \sigma_1k_1 + \sigma_2k_2 = \sigma_1f(t_n, y_n) + \sigma_2f(t_n + a_2\tau, y_n + \tau b_{21}f(t_n, y_n))$$

Ennek lokális approximációs hibája:

$$\psi_n^{(1)} = -\frac{u(t_{n+1}) - u(t_n)}{\tau} + \sigma_1f(t_n, u(t_n)) + \sigma_2f(t_n + a_2\tau, u(t_n) + \tau b_{21}f(t_n, u(t_n)))$$

Végezzük el a következő Taylor-sorfejtéseket:

$$\begin{aligned}f(t_n + a_2\tau, u(t_n) + \tau b_{21}f(t_n, u(t_n))) &= \\f(t_n, u(t_n)) + \tau a_2 \partial_1 f(t_n, u(t_n)) + \tau b_{21} f(t_n, u(t_n)) \partial_2 f(t_n, u(t_n)) + o(\tau^2) &= \end{aligned}$$

$$\begin{aligned}
& u'(t_n) + \tau[a_2\partial_1 f(t_n, u(t_n)) + b_{21}f\partial_2 f|_{(t_n, u(t_n))}] + o(\tau^2) \\
& - \frac{u(t_{n+1}) - u(t_n)}{\tau} = -[u'(t_n) + \frac{\tau}{2}u''(t_n)] + o(\tau^2) = \\
& -[u'(t_n) + 0,5\tau(\partial_1 f(t_n, u(t_n)) + f\partial_2 f|_{(t_n, u(t_n))})] + o(\tau^2)
\end{aligned}$$

Alkalmazva az $f(t_n, u(t_n)) = u'(t_n)$ helyettesítést:

$$\begin{aligned}
& \sigma_1 f(t_n, u(t_n)) + \sigma_2 f(t_n + a_2\tau, u(t_n) + \tau b_{21}f(t_n, u(t_n))) = \\
& (\sigma_1 + \sigma_2)u'(t_n) + \tau[\sigma_2 a_2 \partial_1 f(t_n, u(t_n)) + \sigma_2 b_{21} f\partial_2 f|_{(t_n, u(t_n))}] + o(\tau^2)
\end{aligned}$$

Innen már láthatjuk, hogy az explicit kétlépcsős Runge–Kutta-módszer pontosan akkor konzisztens, ha $\sigma_1 + \sigma_2 = 1$, illetve a módszer másodrendben konzisztens, ha $\sigma_2 a_2 = 0,5$ és $\sigma_2 b_{21} = 0,5$.

4.2. Butcher-táblázat

A Runge–Kutta típusú módszerek paramétereit megadhatjuk táblázatos formában is, amelyet Butcher-táblázatnak nevezünk. Ennek általános alakja:

a_1	$b_{11} \ b_{12} \ \dots \ b_{1s}$
a_2	$b_{21} \ b_{22} \ \dots \ b_{2s}$
a_3	$b_{31} \ b_{32} \ \dots \ b_{3s}$
a_s	$b_{s1} \ b_{s2} \ \dots \ b_{ss}$
	$\sigma_1 \ \sigma_2 \ \dots \ \sigma_s$

Az explicit Runge–Kutta-módszereknél a b_{ij} paramétereiből álló mátrix egy szigorúan alsó háromszögmátrix. Ha ennek a mátrixnak a főátlóban vagy afölött is van nemnulla eleme, akkor a megfelelő Runge–Kutta-módszer már implicit lesz.

Nézzük meg az eddig ismertetett módszerek Butcher-táblázatát.

Az explicit Euler-módszer:

0	0
	1

Az implicit Euler-módszer:

1	1
	1

A kétlépcsős explicit Runge–Kutta-módszer:

0	0	0
$\frac{1}{2}$	$\frac{1}{2}$	0
	0	1

Végül nézzünk példát egy néglépcsős explicit Runge–Kutta-módszer Butcher-táblázatára:

0	0	0	0	0
$\frac{1}{2}$	$\frac{1}{2}$	0	0	0
$\frac{1}{2}$	0	$\frac{1}{2}$	0	0
1	0	0	1	0
	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$

Láthatjuk, hogy itt már annyi paraméter szerepel, hogy ha képlettel adnánk meg a módszert, nagyon bonyolult összefüggéshez jutnánk. Ezért hasznos, ha Butcher-táblázatba foglalva írjuk le a paramétereket, mert így sokkal átláthatóbbá válik, hogy pontosan melyik módszert is használjuk.

Ezt a fejezetet az [1] jegyzet segítségével dolgoztuk fel.

5. Populációs modellek vizsgálata

Ebben a fejezetben az előbbieken bevezetett módszerek viselkedését tanulmányozzuk populációdinamikai modellekben. Először két egydimenziós populációnövekedési modellnek határozzuk meg a pontos megoldását. Ezután a Matlab segítségével a már definiált numerikus módszerekkel meghatározzuk a numerikus megoldásokat, és megvizsgáljuk az ezek között lévő különbségeket. Végül Lotka és Volterra zsákmány-ragadozó modelljének numerikus megoldását állítjuk elő szintén a Matlab segítségével.

A modellekben jelölje $N(t)$ a populáció egyedszámát a t időpillanatban. Az egyedszám időbeli változását, vagyis az $N'(t)$ deriváltat a következő folyamatok határozzák meg.

- születés
- elmúlás
- elvándorlás
- betelepülés

5.1. Egydimenziós populációnövekedési modellek

Az egydimenziós korlátlan növekedés modelljében csak a születés határozza meg az egyedszám változást. Ezt a következőképpen írhatjuk fel:

$$\begin{cases} N'(t) = kN(t), & \text{ahol } k \in \mathbb{R}^+ \\ N(0) = N_0 \end{cases}$$

Ennek a pontos megoldása:

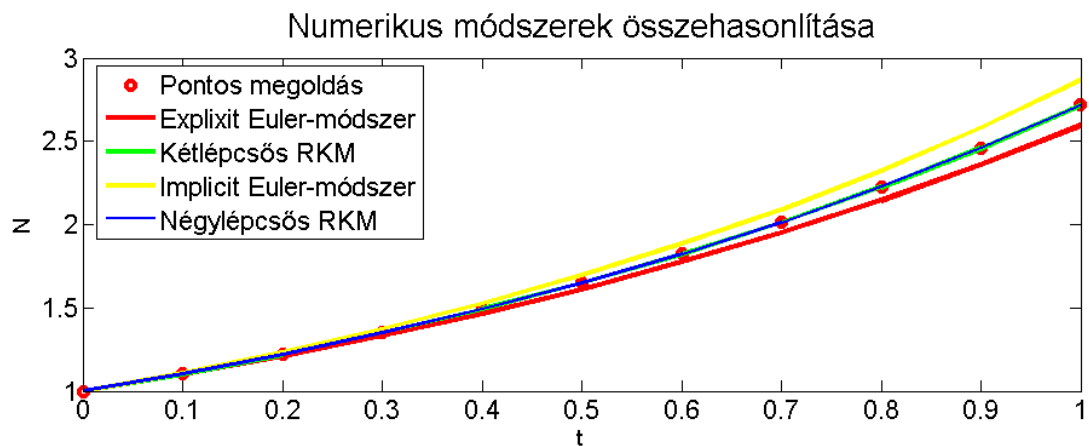
$$N(t) = N_0 e^{kt}, \quad t > 0$$

Vizsgáljuk meg a pontos és a numerikus megoldás közötti különbséget 0,1-es lépéstávolsággal, a $k = 1$ paraméter mellett. Ezt az 1. táblázat és a 2. ábra tartalmazza.

A táblázatból, illetve a grafikonról is láthatjuk, hogy a legpontosabb a négylépcsős Runge–Kutta-módszer, ezt követi a kétlépcsős Runge–Kutta-módszer, majd az explicit és az implicit Euler-módszer következik. Mindez összhangban van elméleti megfontolásainkkal.

t_i	EEM	IEM	Kétlépcsős RKM	Négylépcsős RKM	Pontos megoldás
0,1	1,1	1,1111	1,105	1,1051708	1,1051709
0,2	1,21	1,2346	1,2210	1,2214025	1,2214027
0,3	1,331	1,3717	1,3492	1,3498584	1,3498588
0,4	1,4641	1,5242	1,4909	1,4918242	1,4918246
0,5	1,605	1,6935	1,6474	1,6487206	1,6487212
0,6	1,7716	1,8817	1,8204	1,8221179	1,8221188
0,7	1,9487	2,0908	2,0116	2,0137516	2,0137527
0,8	2,143	2,3231	2,228	2,2255395	2,2255409
0,9	2,3579	2,5812	2,4562	2,4596014	2,4596031
1	2,5937	2,8680	2,7141	2,7182797	2,7182818

1. táblázat. Az egydimenziós korlátlan növekedési modell numerikus megoldásai, valamint a pontos megoldás 0,1-es lépésköz mellett.



2. ábra. Az egydimenziós korlátlan növekedési modell megvalósítása explicit Euler-módszerrel, implicit Euler-módszerrel, explicit kétlépcsős és négylépcsős Runge–Kutta-módszerrel, illetve a pontos megoldás ábrája 0,1-es lépésköz mellett.

Ezek után írjuk fel az egydimenziós korlátos növekedésű modellt, ahol az élettér $K > 0$

eltartóképességű, azaz maximum $K = N$ egyedeket képes eltartani.

$$\begin{cases} N'(t) = kN(t)\left(1 - \frac{N(t)}{K}\right) \\ N(0) = N_0 \end{cases}$$

A differenciálegyenlet szétválasztható típusú, így pontos megoldását a következőképpen állíthatjuk elő:

$$\int \frac{1}{N(1-\frac{N}{K})} dN = \int k dt$$

Elvégezve a jobb oldali integrálást kapjuk, hogy:

$$\int \frac{1}{N(1-\frac{N}{K})} dN = kt + c_1$$

A bal oldali integrálás eredménye:

$$\begin{aligned} \int \frac{1}{N(K-N)} dN &= \int \frac{1}{N} dN + \int \frac{\frac{1}{K}}{K-N} dN = \\ &= \frac{1}{K} \int \frac{1}{N} dN + \frac{1}{K} \int \frac{1}{K-N} dN = \\ &= \frac{1}{K} \ln |N| - \frac{1}{K} \ln |K - N| + c_2 = \ln \frac{N}{|K-N|} + c_2, \quad c_2 \in \mathbb{R} \end{aligned}$$

A két oldal összevetéséből:

$$\begin{aligned} \ln \frac{N}{|K-N|} &= kt + c = \\ \frac{N}{|K-N|} &= e^{kt+c} = \hat{c}e^{kt}, \quad \hat{c} \in \mathbb{R} \\ \frac{|K-N|}{N} &= \hat{c}e^{-kt} = \\ \frac{K}{N} &= 1 + \hat{c}e^{-kt} \end{aligned}$$

Vagyis a pontos megoldás:

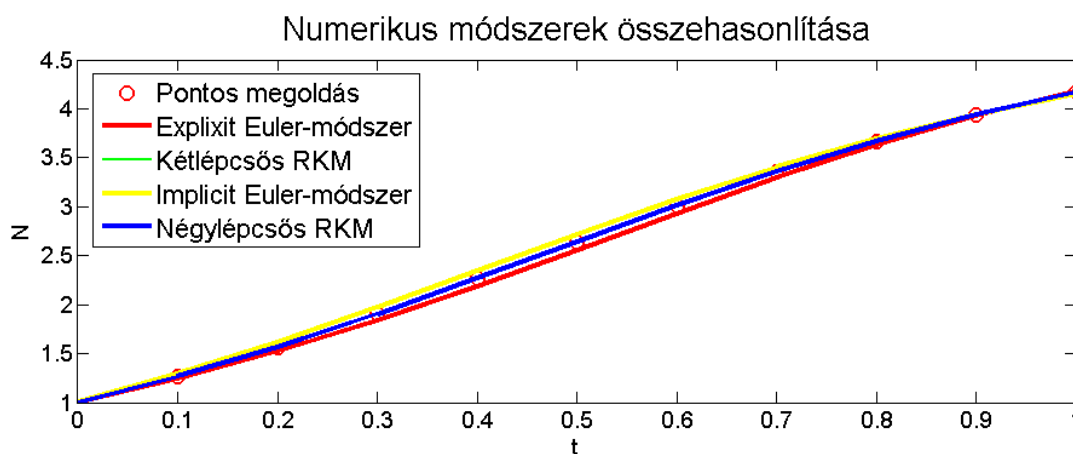
$$\begin{aligned} N(t) &= \frac{K}{1 + \hat{c}e^{-kt}}, \text{ ahol } k, K \text{ adott pozitív számok} \\ N(0) &= \frac{K}{1 + \hat{c}} = N_0, \text{ tehát } \hat{c} = \frac{K}{N_0} - 1. \end{aligned}$$

Vizsgáljuk meg ezen a modellen is a pontos illetve a numerikus megoldás közötti különbségeket $k = 3$, $K = 5$ paraméterek mellett. Ezt a 2. táblázat és a 3. ábra tartalmazza.

A 4. ábrán azt láthatjuk, hogy ha az explicit Euler-módszer használata során túl nagy lépésközt állítunk be, akkor a módszer instabillá válik. Azonban, ha ezzel a lépésközzel

t_i	EEM	IEM	Kétlépcsős RKM	Négylépcsős RKM	Pontos megoldás
0,1	1,24	1,2867	1,2607	1,2616	1,2616
0,2	1,5197	1,6146	1,5634	1,5648	1,5648
0,3	1,8371	1,9730	1,9023	1,9038	1,9038
0,4	2,1857	2,3466	2,2667	2,2678	2,2678
0,5	2,5548	2,7187	2,6416	2,6420	2,6420
0,6	2,9296	3,0739	3,0101	3,0099	3,0099
0,7	3,2935	3,4003	3,3566	3,3561	3,3561
0,8	3,6308	3,6903	3,6689	3,6687	3,6687
0,9	3,9290	3,9407	3,9401	3,9406	3,9407
1	4,1815	4,1520	4,1680	4,1696	4,1696

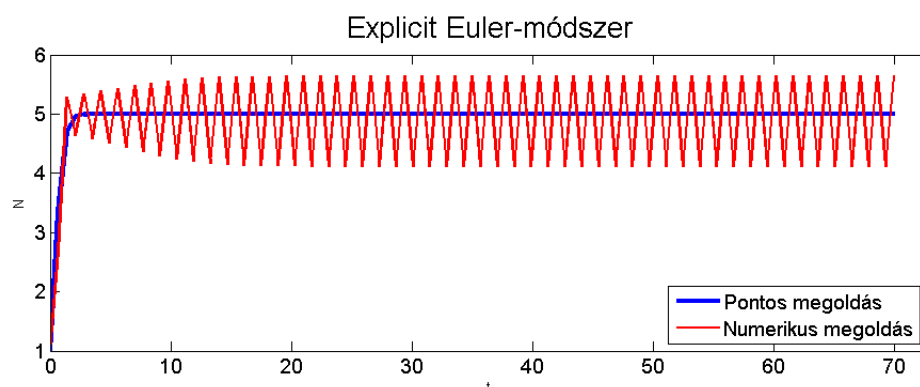
2. táblázat. Az egydimenziós korlátos növekedési modell numerikus megoldásai, valamint a pontos megoldás.



3. ábra. Az egydimenziós korlátos növekedési modell megvalósítása explicit Euler-módszerrel, implicit Euler-módszerrel, explicit kétlépcsős és négylépcsős Runge–Kutta-módszerrel, illetve a pontos megoldás ábrája 0,1-es lépésköz mellett.

dolgozunk, akkor az explicit kétlépcsős Runge–Kutta-módszer és az explicit négylépcsős Runge–Kutta-módszer nem válik instabillá.

A 3. táblázatban a különböző konvergenciarendeket fogjuk vizsgálni. Az első oszlop az



4. ábra. Az explicit Euler-módszer instabil esete.

alkalmazott lépéstávolságokat tartalmazza, a további oszlopok pedig a fejlécben feltüntetett numerikus módszerekkel kapott megoldások hibáját (vagyis a pontos és a numerikus megoldás különbségének az abszolút értékét) a $t = 1$ pontban.

τ	EEM	IEM	Kétlépcsős RKM	Négylépcsős RKM
0,25	—	0,0541	0,0092	0,001
0,125	0,137	0,0228	0,0024	6,039831e-005
0,0625	0,0082 (0,6)	0,0104 (0,46)	0,0006 (0,25)	3,7292349e-006 (0,062)
0,0313	0,0046 (0,56)	0,005 (0,48)	0,00015 (0,25)	2,3316907e-007 (0,063)
0,0157	0,0025 (0,54)	0,0025 (0,5)	0,00039 (0,26)	1,47259218e-008 (0,063)

3. táblázat. A vizsgált módszerek hibái $t=1$ -ben, és becsült konvergenciarendjük (zárójelben).

Ha visszaemlékszünk a konzisztencia illetve konzisztenciarend definíciójára, akkor láthatjuk, hogy az ott szereplő p határozza meg a módszer konzisztenciarendjét, amely stabilitás esetén egyben a konvergenciarendet is megadja. Ez azt jelenti, hogy elsőrendű módszernél a hibáknak jó közelítéssel feleződniük kell, ha a lépésközöket is mindig felezzük. Egy másodrendű módszer esetében viszont a lépésköz feleződésével a hibáknak már negyedelődniük kell, akkor, ha a lépésközöket feleztük. Ez azért van, mert a konvergenciarend definíciója alapján egy p -ed rendű módszernél a vezető hibtag a lépésköz p -edik hatványával arányos. Ezek alapján egy negyedrendű módszer hibái tizenhatodára

csökkenek, ha a lépésközöket felezzük.

Az elméleti rendeket ellenőrizni tudjuk a 3. táblázat alapján. Az előbb leírtak értelmében az explicit és az implicit Euler-módszer oszlopaiban lévő hibáknak feleződniük, az explicit kétlépcsős Runge–Kutta-módszer oszlopában lévőknek negyedelődniük kell, az explicit néglépcsős Runge–Kutta-módszer esetében pedig tizenhatod részükre kell csökkenniük. A táblázatban ezek a hányadosok zárójelben szerepelnek.

5.2. Lotka–Volterra-modell

Tekintsük Lotka és Volterra zsákmány-ragadozó modelljét, melyben jelölje $r(t)$ a ragadozó állatok egyedszámát a t időpillanatban, $z(t)$ a zsákmányállatok egyedszámát szintén a t időpillanatban, a a zsákmányállatok növekedési rátáját, d pedig a ragadozók halálozási rátáját. Ha egy zsákmányállat találkozik egy ragadozóval, akkor a zsákmányállatok száma csökken, a ragadozóké pedig nő, ezt az összefüggést fejezik ki a b, c együtthatók.

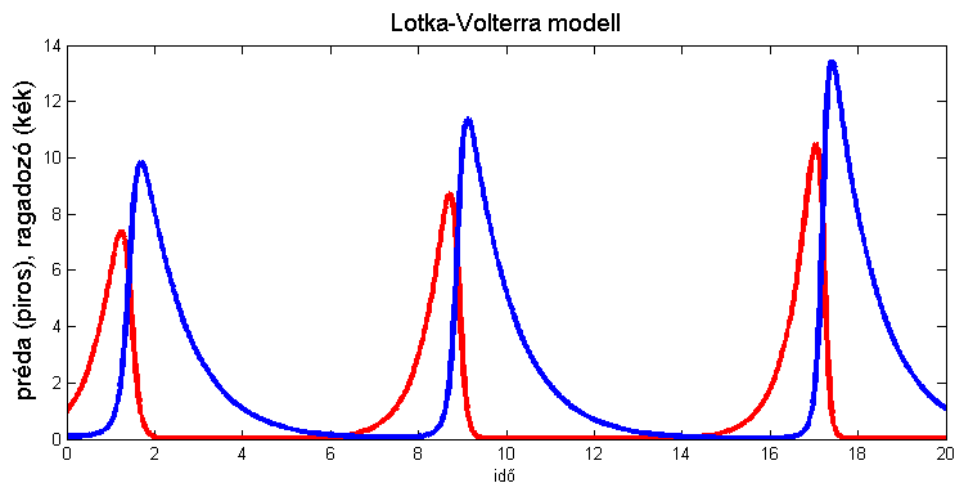
A modellben feltételezzük, hogy a zsákmány soha nem pusztul el természetes halállal, hanem mindig ragadozó végez vele. A modell alapegyenleteit a következőképpen írhatjuk fel:

$$\begin{aligned}z'(t) &= az(t) - bz(t) \\r'(t) &= cr(t)z(t) - dr(t).\end{aligned}$$

Vizsgáljuk meg az egyenletrendszer numerikus megoldását $a = 2$, $b = 1$, $c = -1$, $d = 1$ paraméterek és a $z(0) = 1$, $r(0) = 0,1$ kezdeti feltételek mellett, ezt a 5. ábra szemlélteti.

Az ábráról láthatjuk, hogy először a prédaállatok számának el kell érnie egy megfelelő értéket ahhoz, hogy a ragadozók száma elkezdjen nőni. Ez megfelel az elvárásainknak, hiszen addig nem növekedhet a ragadozók száma, amíg nincs megfelelő számú zsákmányállat a szaporodásukhoz. Szintén észrevehetjük, hogy az állatok száma egyensúlyban van egymással, vagyis ha a prédaállatok száma csökkenni kezd, akkor a ragadozók száma is csökken a táplálék hiánya miatt, azonban ez időben késleltetve játszódik le.

Ezt a fejezetet a [3], [9] jegyzetek segítségével dolgoztuk fel.



5. ábra. A Lotka–Volterra-modell numerikus megoldása explicit Euler-módszerrel.

6. A numerikus módszerek gépidőigényének vizsgálata

Ebben a fejezetben a különböző módszerek gépigényével foglalkozunk. A Matlab segítségével megmérjük az egyes módszerek gépidőigényét egyre csökkenő lépésközökre, majd megvizsgáljuk, hogy adott pontosság eléréséhez mennyi gépidő szükséges.

Tekintsük a következő kezdetiérték-feladatot:

$$y'(t) = -y(t) + t + 1, t \in [0, 100]$$

$$y(0) = 1$$

Vizsgáljuk meg ezen feladat numerikus megoldásainak a gépidőigényét egyre csökkenő lépésközök mellett a már ismert módszerek segítségével. Ezt az 4. táblázat szemlélteti.

A táblázatban szereplő értékekből megállapíthatjuk, hogy a leglassabb módszer az implicit Euler-módszer, annak ellenére, hogy ez csak elsőrendben pontos módszer. Ez azért lehetséges, mert ennek a módszernek a programkódja a legbonyolultabb, ugyanis minden lépésben egy algebrai egyenletet kell megoldani iterációs módszerrel. A többi explicit módszer nagyjából ugyanannyi idő alatt oldja meg az egyenletet, ez azzal magyarázható, hogy egy nem túl bonyolult differenciálegyenletet oldottunk meg.

Most vizsgáljuk meg azt, hogy adott pontosság eléréséhez mekkora gépidő szükséges az egyes módszerek alkalmazása során. Ezt a 5. táblázat szemlélteti.

A táblázatból láthatjuk, hogy ahogy nő az előírt pontosság, úgy nő természetesen a

τ	EEM	IEM	Kétlépcsős RKM	Négylépcsős RKM
0,5	0,011326	0,017557	0,002975	0,000491
0,25	0,003284	0,019960	0,000223	0,003272
0,125	0,000236	0,017964	0,000273	0,000448
0,0625	0,022909	0,008543	0,00346	0,000674
0,0313	0,000341	0,012736	0,000668	0,000684
0,0157	0,000785	0,016086	0,000785	0,000718

4. táblázat. A vizsgált numerikus módszerek gépidőigénye másodpercben.

Hiba nagyságrend	EEM	IEM	Kétlépcsős RKM	Négylépcsős RKM
10^{-1}	0,161797	0,162314	-	-
10^{-2}	0,176139	0,174245	0,191798	-
10^{-3}	0,178247	0,196695	0,209678	0,168725
10^{-4}	0,277752	0,360377	0,209876	0,212467
10^{-5}	0,291514	3,120654	0,227198	0,182852
10^{-6}	0,332100	218,575090	0,254731	0,215566
10^{-7}	3,816281	-	0,257815	0,243178
10^{-8}	33,915342	-	0,261763	0,259091

5. táblázat. Gépidőigény adott pontosság mellett.

gépidőigény is. Ha nagyobb pontossággal akarjuk megoldani a feladatot, akkor érdemes magasabb rendű módszert alkalmazni, mert ugyanazt a pontosságot rövidebb futtatási idővel érhetjük el, mint egy alacsonyabb rendű módszer esetén.

Ezek után vizsgáljuk meg a gépidőigényt egy kétszáz egyenletből álló állandó együtthatós homogén lineáris differenciálegyenlet-rendszer esetén szintén egyre csökkenő lépésközökre. Ehhez három explicit módszert használunk fel. A 6. táblázat adatai alapján azt figyelhetjük meg, hogy a kétlépcsős módszer alkalmazása durván kétszer, a négylépcsősé négyszer annyi időt igényel, mint az explicit Euler-módszer használata, mert ekkor az egy időlépcsőben elvégzett függvénykiértékelések száma határozza meg döntően az időigényt.

τ	EEM	Kétlépcsős RKM	Néglépcsős RKM
0,5	1,3129	2,7231	5,0743
0,25	1,5231	3,4062	6,1023
0,125	1,5110	3,7398	6,3326
0,0625	1,5256	2,7538	5,3426
0,0313	1,3655	2,8009	6,2196
0,0157	1,5446	3,1842	6,1076

6. táblázat. A vizsgált numerikus módszerek gépidőigénye másodpercben egy nagyméretű állandó együtthatós homogén lineáris differenciálegyenlet-rendszer esetén.

7. Összefoglalás

A dolgozat első felében áttekintettük a differenciálegyenletekhez kapcsolódó alapvető definíciókat és tételeket. Megismerkedtünk a véges különbséges módszerek alapjaival és főbb fajtaival. Meghatároztuk ezek konzisztenciarendjét, majd foglalkoztunk stabilitásukkal.

Ezek után bevezettük a Runge–Kutta típusú módszereket. Itt szintén meghatároztuk a konzisztenciarendet egy explicit kétlépcsős Runge–Kutta-módszer esetében. Megmutattuk, hogy ezeket az explicit módszereket hogyan érdemes táblázatba foglalni a könnyebb átláthatóság végett.

Az elméleti összefoglaló után a már ismert módszereket a gyakorlatban is alkalmaztuk különböző populációdinamikai feladatokon. Először két egydimenziós populációdinamikai feladaton megmutattuk azt, hogy valóban az explicit néglépcsős Runge–Kutta-módszer a legpontosabb, ahogy ezt az elméleti megfontolásaink alapján vártuk. Meggyőződünk róla, hogy az elméletileg levezetett rendek a gyakorlat során is érvényesülnek. Példával illusztráltuk, hogy ha rosszul választjuk meg a lépésközt, akkor a módszer instabillá válhat. Megoldottuk a Lotka–Volterra-féle zsákmány-ragadozó modell differenciálegyenlet-rendszerét az explicit Euler-módszer segítségével, majd a numerikus megoldásokat ábra segítségével elemeztük.

A dolgozat utolsó fejezetében a gépidőigény mérésével foglalkoztunk. Egy egyszerű közönséges differenciálegyenleten megmértük az egyes módszerek gépidejét csökkenő lé-

pésköz esetén. Itt észrevehettük azt, hogy implicit módszerrel nem érdemes ezt a fajta egyenletet megoldani, valamint hogy ilyen egyszerű esetben lényegében mindegy, hogy melyik explicit módszert használjuk. Nagyobb méretű differenciálegyenlet-rendszer esetén azonban a kétlépcsős módszer alkalmazása mintegy kétszer, a négylépcsősé négyszer annyi időt vesz igénybe, mint az explicit Euler-módszer alkalmazása. Ezek után az előre meghatározott hibahatár eléréséhez szükséges gépidőket mértük meg. Itt azt kaptuk, hogy ha nagy pontosságot szeretnénk elérni, akkor magasabb rendű explicit módszert érdemes használnunk, mert ezek rövidebb futási idővel képesek ugyanazt a pontosságot biztosítani, mint az alacsonyabb rendű módszerek.

8. Függelék

8.1. A dolgozat során felhasznált Matlab programkódok

Explicit Euler-módszer

```
function[t y]= szakexpeuler(szakdiffegy, t0, y0, h, N)
t=zeros(N+1,1);
y=zeros(N+1,1);
t(1)=t0;
y(1)= y0;
for i=1:N
    t(i+1)=t(i)+h;
    y(i+1)= y(i)+h*szakdiffegy(t(i),y(i));
end
```

Lotka-Volterra-modell megoldása explicit Euler-módszerrel

```
function LVexpeuler2(x,y,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)')
title('Lotka-Volterra modell')
function U=f(x,y)
U=2*x-x.*y;
function V=g(x,y)
V=-1*y+1*x.*y;

```

Implicit Euler-módszer

```

function [t,y] = szakimpeuler(diffegy, t0, y0, h, N)
t=zeros(N+1,1);
y = zeros(N+1,1);
t(1) = t0;
y(1) = y0;
for i=1:N
t(i+1) = t(i) + h;
g = @(x)(x-y(i)-h*diffegy(t(i+1),x));
g_der = @(x)(1-h);
hiba = 1e-6;
maxit = 100;
x(1) = y(i)+h*diffegy(t(i),y(i));
y_newton = g(x(1));
it = 0;

```

```

while ((abs(y_newton) > hiba) && (it < maxit))
    it = it + 1;
    x(it+1) = x(it) - g(x(it))/g_der(x(it));
    y_newton = g(x(it+1));
end
y(i+1)=x(it+1);
end

```

Explicit kétlépcsős Runge–Kutta-módszer

```

function [t,y] = midpoint(szakdiffegy, t0, y0, h, N)
t=zeros(N+1,1);
y = zeros(N+1,1);
t(1) = t0;
y(1) = y0;
for i=1:N
    s1=szakdiffegy(t(i),y(i));
    s2=szakdiffegy(t(i)+h/2,y(i)+h/2*s1);
    y(i+1) = y(i) + h * s2;
    t(i+1) = t(i) + h;
end

```

Explicit néglépcsős Runge–Kutta-módszer

```

function [t y]=szakrungekutta(szakdiffegy,t0,y0,h,N)
t=zeros(N+1,1);
y=zeros(N+1,1);
t(1)=t0;
y(1)=y0;
for i=1:N
    k1=szakdiffegy(t(i),y(i));
    k2=szakdiffegy(t(i)+0.5*h,y(i)+h*0.5*k1);
    k3=szakdiffegy(t(i)+0.5*h,y(i)+h*k2*0.5);

```



```

k4=szakdiffegy(t(i)+h,y(i)+h*k3);
t(i+1)=t(i)+h;
y(i+1)=y(i)+h*((1/6)*k1+(1/3)*k2+(1/3)*k3+(1/6)*k4);
end

```

Felhasznált modellek, kezdeti-érték feladatok kódjai

```

function dydt= szakdiffegy(t,y)
dydt(1)=y(1);

function dydt= szakdiffegy2(t,y)
dydt(1)=3*(y(1))*(1-(y(1)/5));

function dydt= szakdiffegy3(t,y)
dydt=-y+t+1;

```

Hivatkozások

- [1] Faragó István *Alkalmazott Analízis 2*, előadás jegyzet, 2015
- [2] Simon Péter *Differenciálegyenletek*, előadás jegyzet, 2014
- [3] Csomós Petra *Folytonos modellezés*, előadás jegyzet, 2016
- [4] Faragó István-Horváth Róbert (2013) *Numerikus Módszerek* Typotex, Url: <http://tankonyvtar.ttk.bme.hu/pdf/30.pdf>
- [5] Havasi Ágnes, *Segédanyag*, URL: <http://nimbus.elte.hu/hagi/segedanyag/fejezet1-7.1.pdf>
- [6] Krebsz Anna, *Közönséges Differenciálegyenletek Numerikus Módszerei Jegyzet*
- [7] Faragó István, *Numerikus Modellezés és Közönséges Differenciálegyenletek Numerikus Megoldási Módszerei*
- [8] Svantnerné Sebestyén Gabriella, *Alkalmazott Analízis 2*, gyakorlat jegyzet
- [9] Havasi Ágnes, *Matlab Segédanyag*, URL: <http://nimbus.elte.hu/hagi/segedanyag/Matlabora/felev2/orakanyaga/gyak07kde2.pdf>