5ο Εργαστήριο

Για την αριθμητική επίλυση ενός προβλήματος αρχικών τιμών (Π.Α.Τ.)

$$y'(t) = f(t, y(t)), \quad t \in [a,b], \quad y(0) = y_0$$

θεωρήσαμε μονοβηματικές μεθόδους όπως του Euler, αλλά και άμεσες 2-βηματικές όπως η Adams Bashforth (2) και η μέθοδος του μέσου. Έστω ένας ομοιόμορφος διαμερισμός του $[a,b]$, στα σημεία $t_n = a + nh$, $n = 0,\ldots,N$, με βήμα $h = \dfrac{b-a}{N}$, υπολογίζουμε τις τιμές y_n που αποτελούν προσεγγίσεις στις τιμές $y(t_n)$, $n = 0,\ldots,N$

Άμεση Euler

$$y_{n+1} = y_n + hf(t_n, y_n), \quad n = 0,\ldots,N-1.$$

Μέθοδος του μέσου Για δοσμένα y_0, y_1,

$$y_{n+2} = y_n + 2hf(t_{n+1}, y_{n+1}), \quad n = 0,\ldots,N-2.$$

Μέθοδος Adams Bashforth(2) Για δοσμένα y_0, y_1,

$$y_{n+2} = y_{n+1} + h(\frac{3}{2}f(t_{n+1}, y_{n+1}) - \frac{1}{2}f(t_n, y_n)), \quad n = 0,\ldots,N-2.$$

Άσκηση 1: Έστω $y(t) = e^{0.25-(t-0.5)^2}$, στο $[0,4]$ η οποία είναι λύση στο

$$y'(t) = (1-2t)y(t), \quad t \in [0,4], \quad y(0) = 1.$$

Θεωρείστε τη μέθοδο του μέσου και τη μέθοδο Adams-Bashforth (2). Θεωρήστε ένα διαμερισμό του $[0,4]$ σε $N+1$ σημεία, και για να υλοποιήσετε αυτές τις δύο μεθόδους χρησιμοποιήστε τις αρχικές τιμές $y_0 = 1$ και $y_1 = y(h) = e^{0.25-(h-0.5)^2}$. Για $N = 50$, κατασκευάστε τις προσεγγίσεις που δίνουν οι δύο μέθοδοι, δημιουργείστε τις γραφικές παράστασεις της προσεγγιστικής λύσης και της ακριβούς στο διάστημα [0,4]. Στη συνέχεια βρείτε το σφάλμα $\max_{0\le n\le N} |y_n - y(t_n)|$. Για $N = 100, 200, 300, 400, 500$ βρείτε τη πειραματική τάξη σύγκλισης για κάθε μια από τις δύο μεθόδους

In [ ]:
```python
import numpy as np
import matplotlib.pyplot as plt
```

In [ ]:
```python
def f(t,y):
    s=(1-2*t)*y
    return s
def y_exact(t):
    w=0.25-(t-0.5)**2
    s=np.exp(w)
    return s
```

In [ ]:
```python
N=50
t=np.linspace(0,4,N+1)
h=t[1]-t[0]


y_mid=np.zeros(N+1)
y_ab=np.zeros(N+1)


y_mid[0]=1
y_ab[0]=1

### Εισαγουμε την 1η προσεγγιση για να ξεκινησει η διβηματικη μεθοδος
y_mid[1]=y_exact(t[1])
y_ab[1]=y_exact(t[1])

# Προσοχή στον αριθμο των επαναλήψεων
#Στη διβηματικη μεθοδο κανουμε 1 λιγοτερη επαναληψη
for i in range(N-1):
    #midpoint
    y_mid[i+2]=y_mid[i]+2*h*f(t[i+1],y_mid[i+1])
    #AB(2)
    y_ab[i+2]=y_ab[i+1]+h*((3./2)*f(t[i+1],y_ab[i+1])-(1./2)*f(t[i],y_ab[i]))
```

Άσκηση 2: Επαναλάβετε την Άσκηση 1, αλλά τώρα η τιμή y_1 θα δίνεται χρησιμοποιώντας την άμεση μέθοδο του Euler, δηλαδή $y_1 = y_0 + hf(t_0, y_0)$. Αλλάζει η τάξη σύγκλισης;

In [ ]:
```python
# Το μόνο που αλλάζει είναι οτι y1 προκυπτει με την αμεση Euler
### Εισαγουμε την 1η προσεγγιση για να ξεκινησει η διβηματικη μεθοδος
y_mid[1]=y_mid[0]+h*f(t[0],y_mid[0])
y_ab[1]=y_ab[0]+h*f(t[0],y_ab[0])
```

Άσκηση 3: Για το ΠΑΤ που έχετε θεωρήσει μέχρι τώρα φτιάξτε ένα πίνακα των σφαλμάτων της παρακάτω μορφής χρησιμοποιήστε ακρίβεια 4 δεκαδικών ψηφίων, η y_1 θα δίνεται χρησιμοποιώντας την άμεση μέθοδο του Euler,

N	$ΆμεσηEuler$	p	$ΜέθοδοςΜέσου$	p	$AB(2)$	p
100						
200						
300						
400						
500						

```
Ποιά μέθοδο θα προτιμήσετε;
```

Πεπλεγμένες πολυβηματικές μέθοδοι Μια πεπλεγμένη μονοβηματική μέθοδος είναι η **πεπλεγμένη Euler**

$$y_{n+1} = y_n + hf(t_{n+1}, y_{n+1}), \quad n = 0,\ldots,N-1.$$

Για να υπολογίσουμε την τιμή y_{n+1} χρειάζεται να λύσουμε (στη γενική περίπτωση που η f είναι μη γραμμική ως προς y) μια μη γραμμική εξίσωση. Στην περίπτωση της άσκησης 1, η f είναι γραμμική ως προς y. Οπότε η πεπλεγμένη Euler παίρνει την ακόλουθη μορφή

$$y_{n+1} = y_n + h(1-2t_{n+1})y_{n+1}, \quad n = 0,\ldots,N-1.$$

Οπότε λύνοντας ως προς y_{n+1} παίρνουμε

$$(1 - h(1-2t_{n+1}))y_{n+1} = y_n \quad n = 0,\ldots,N-1$$

ή

$$y_{n+1} = \frac{1}{1 - h(1-2t_{n+1})}y_n, \quad n = 0,\ldots,N-1.$$

Μια πεπλεγμένη διβηματική μέθοδος είναι η μέθοδος του Simpson
Μέθοδος του Simpson Για δοσμένα y_0, y_1,

$$y_{n+2} = y_n + \frac{h}{3}(f(t_{n+2}, y_{n+2}) + 4f(t_{n+1}, y_{n+1}) + f(t_n, y_n)), \quad n = 0,\ldots,N-2.$$

Για να υπολογίσουμε την τιμή y_{n+1} χρειάζεται να λύσουμε (στη γενική περίπτωση που η f είναι μη γραμμική ως προς y) μια μη γραμμική εξίσωση. Στην περίπτωση της άσκησης 1, η f είναι γραμμική ως προς y. Οπότε η πεπλεγμένη Euler παίρνει την ακόλουθη μορφή

$$y_{n+2} = y_n + \frac{h}{3}((1-2t_{n+2})y_{n+2} + 4(1-2t_{n+1})y_{n+1} + (1-2t_n)y_n), \quad n = 0,\ldots,N-2.$$

Οπότε λύνοντας ως προς y_{n+2} παίρνουμε

$$(1 - \frac{h}{3}(1-2t_{n+2}))y_{n+2} = y_n + \frac{h}{3}(4(1-2t_{n+1})y_{n+1} + (1-2t_n)y_n) \quad n = 0,\ldots,N-1$$

ή

$$y_{n+2} = \frac{1}{1 - \frac{h}{3}(1-2t_{n+1})}(y_n + \frac{h}{3}(4(1-2t_{n+1})y_{n+1} + (1-2t_n)y_n)) \quad n = 0,\ldots,N-1$$

Άλλες πεπλεγμένες διβηματικές μέθοδοι είναι η μέθοδος Adams Multon(2) και η BDF(2)
Μέθοδος Adams Multon(2) Για δοσμένα y_0, y_1,

$$y_{n+2} = y_{n+1} + h(\frac{5}{12}f(t_{n+2}, y_{n+2}) + \frac{2}{3}f(t_{n+1}, y_{n+1}) - \frac{1}{12}f(t_n, y_n)), \quad n = 0,\ldots,N-2.$$

Μέθοδος BDF(2) Για δοσμένα y_0, y_1,

$$y_{n+2} - \frac{4}{3}y_{n+1} + \frac{1}{3}y_n = \frac{2}{3}hf(t_{n+2}, y_{n+2}), \quad n = 0,\ldots,N-2.$$

Άσκηση 4: Για το ΠΑΤ της Άσκησης 1 θεωρείστε την πεπλεγμένη μέθοδο του Euler. Βρείτε το σφάλμα $\max_{0\le n\le N} |y_n - y(t_n)|$. Για $N = 100, 200, 300, 400, 500$ βρείτε τη πειραματική τάξη σύγκλισης για την πεπλεγμένη μεθόδο του Euler.

Άσκηση 4: Για το ΠΑΤ που έχετε θεωρήσει μέχρι τώρα φτιάξτε ένα πίνακα των σφαλμάτων της παρακάτω μορφής χρησιμοποιήστε ακρίβεια 4 δεκαδικών ψηφίων, η y_1 θα δίνεται χρησιμοποιώντας τη πεπλεγμένη μέθοδο του Euler,

N	$Πεπλεγμενη\ Euler$	p	$Μέθοδος\ Simpson$	p	$AM(2)$	p	$BDF(2)$	p
100								
200								
300								
400								
500								

Ποιά μέθοδο θα προτιμήσετε;

In [ ]:
```python
def f(t,y):
    s=g(t)*y
    return s
def y_exact(t):
    w=0.25-(t-0.5)**2
    s=np.exp(w)
    return s
def g(t):
    s=(1-2*t)
    return s
```

In [ ]:
```python
#### Μεθοδος Simpson για N=50
#### Οι αλλες υλοποιούνται ανάλογα

t=np.linspace(0,4,N+1)
h=t[1]-t[0]


y_simp=np.zeros(N+1)

y_simp[0]=1


### Εισαγουμε την 1η προσεγγιση για να ξεκινησει η διβηματικη μεθοδος
### Παράδειγματος χάριν y1 προκηπτει με την πεπλεγμενη Euler
y_simp[1]=y_simp[0]/(1-h*g(t[1]))


# Προσοχή στον αριθμο των επαναλήψεων
#Στη διβηματικη μεθοδο κανουμε 1 λιγοτερη επαναληψη
for i in range(N-1):

    #simpson
    y_simp[i+2]=(y_simp[i]+(h/3)*(4*f(t[i+1],y_simp[i+1])+f(t[i],y_simp[i])))/(1-(h/3)*g(t[i+2]))
```