

Ένα παράδειγμα για την κατανόηση της
απαίτησης για απόλυτη ακρίβεια από τους
υπολογιστές

Δημήτριος Θ. Χριστόπουλος^{1,2}

¹Εθνικό Καποδιστριακό Πανεπιστήμιο Αθηνών, Τμήμα Οικονομικών Επιστημών
²dchristop@econ.uoa.gr

10 Απριλίου 2011

Περιεχόμενα

1	Λύση συστήματος διαφορικών εξισώσεων	3
1.1	Διατύπωση του προβλήματος	3
1.2	Θεωρητική λύση του προβλήματος	4
1.3	Ακριβής λύση με το wxMaxima	5
1.4	Προσεγγιστική λύση με το MATLAB	8
1.4.1	Λύση με την χρήση της μεθόδου ode45	8
1.4.2	Λύση με την χρήση του εκθετικού πίνακα	10
1.5	Προσεγγιστική λύση με το Octave	13
1.5.1	Λύση με την χρήση της μεθόδου ode45	13
1.5.2	Λύση με την χρήση του εκθετικού πίνακα	15
2	Προβληματισμοί και Συμπεράσματα	19

1 Λύση συστήματος διαφορικών εξισώσεων

1.1 Διατύπωση του προβλήματος

Ένας οικονομολόγος, με ειδίκευση στα μακρο-οικονομικά, στην προσπάθειά του να βρει ένα υπόδειγμα το οποίο θα διέπει την χρονική εξέλιξη πέντε βασικών μεγεθών μιας οικονομίας, ας τις ονομάσουμε x_1, x_2, x_3, x_4, x_5 , αφού μελέτησε τα δεδομένα του παρελθόντος και σκέφτηκε κατέληξε στο ακόλουθο σύστημα γραμμικών διαφορικών εξισώσεων:

$$\begin{cases} x_1'(t) = & & & & +4x_5(t) \\ x_2'(t) = & x_1(t) & & & -16x_5(t) \\ x_3'(t) = & & x_2(t) & & +25x_5(t) \\ x_4'(t) = & & & x_3(t) & -19x_5(t) \\ x_5'(t) = & & & & x_4(t) +7x_5(t) \end{cases} \quad (1)$$

Επίσης έστω ότι στον χρόνο μηδέν, δηλ. όταν διατύπωσε το υπόδειγμά του, οι τιμές των μακρο-οικονομικών μεγεθών είναι οι ακόλουθες:

$$x_1(0) = \frac{1}{10}, x_2(0) = -\frac{1}{15}, x_3(0) = \frac{3}{7}, x_4(0) = \frac{4}{9}, x_5(0) = \frac{3}{4} \quad (2)$$

Οι 1 μαζί με τις 2 αποτελούν ένα Πρόβλημα Αρχικών Τιμών (Π.Α.Τ.) το οποίο και θέλει να λύσει ο οικονομολόγος του παραδείγματός μας και μάλιστα ενδιαφέρεται να προβλέψει την τιμή των μεγεθών μετά από $t = 10$ έτη. Για να τον βοηθήσουμε τού προτείνουμε να γράψει το πρόβλημά του υπό μορφή εξίσωσης πινάκων ως εξής:

$$\begin{aligned} x'(t) &= A \cdot x(t) \\ A &= \begin{pmatrix} 0 & 0 & 0 & 0 & 4 \\ 1 & 0 & 0 & 0 & -16 \\ 0 & 1 & 0 & 0 & 25 \\ 0 & 0 & 1 & 0 & -19 \\ 0 & 0 & 0 & 1 & 7 \end{pmatrix} \\ x_0 &= \begin{pmatrix} \frac{1}{10} \\ -\frac{1}{15} \\ \frac{3}{7} \\ \frac{4}{9} \\ \frac{3}{4} \end{pmatrix} \end{aligned} \quad (3)$$

1.2 Θεωρητική λύση του προβλήματος

Έχοντας διατυπώσει το πρόβλημα του οικονομολόγου σε μορφή πινάκων γνωρίζουμε¹ ότι η λύση του συστήματος 3 θα δίνεται από τον τύπο:

$$x(t) = e^{A \cdot t} \cdot x_0 \quad (4)$$

Το πρόβλημα λοιπόν είναι ο υπολογισμός του εκθετικού πίνακα $e^{A \cdot t}$. Εάν προσπαθήσουμε να κάνουμε τον υπολογισμό 'με το χέρι' θα δούμε ότι ο πίνακας A είναι μη απλής δομής² και γι' αυτόν τον λόγο δεν διαγωνοποιείται. Πρέπει λοιπόν να υπολογίσουμε την κανονική μορφή Jordan αυτού και κατόπιν να βρούμε τον εκθετικό πίνακα που ζητάμε. Ο υπολογισμός της κανονικής μορφής Jordan είναι από μόνος του ένα δύσκολο πρόβλημα στην συγκεκριμένη περίπτωση, αλλά μετά από αρκετή διαδικασία καταλήγουμε στην εξής ανάλυση:

$$A = P \cdot J \cdot P^{-1}$$
$$P = \begin{pmatrix} -14 & 1 & \frac{11}{4} & -\frac{1}{4} & 1 \\ 49 & 0 & -\frac{33}{4} & -2 & 0 \\ -63 & -6 & \frac{143}{16} & \frac{75}{16} & 0 \\ 35 & 8 & -\frac{33}{8} & -\frac{49}{16} & -\frac{1}{2} \\ -7 & -3 & \frac{11}{16} & \frac{5}{8} & \frac{3}{16} \end{pmatrix} \quad (5)$$
$$J = \begin{pmatrix} 2 & 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Τώρα γνωρίζουμε, λ.χ. κοιτάζτε στην σελ. 557 του G. Golub & C. Van Loan (1996), ότι ο εκθετικός πίνακας θα είναι:

$$e^{A \cdot t} = P \cdot e^{J \cdot t} P^{-1} \quad (6)$$

¹Βλέπετε λ.χ. στην σελ. 456 του βιβλίου Boyce & DiPrima (1997)

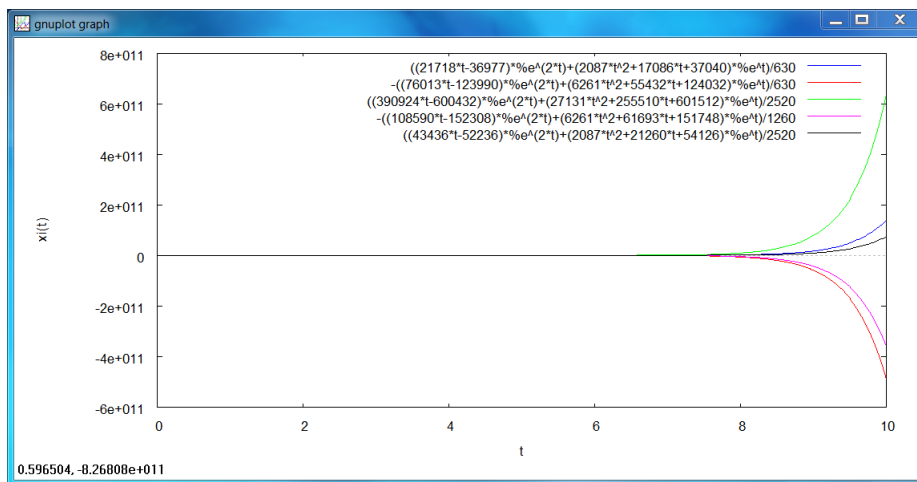
²Ένας πίνακας λέγεται μη απλής δομής όταν ο αριθμός των ιδιοδιανυσμάτων κάθε ιδιοτιμής δεν είναι ίσος με την πολλαπλότητα της ιδιοτιμής.

1.3 Ακριβής λύση με το wxMaxima

Υπολογίζουμε τον εκθετικό πίνακα με το wxMaxima χρησιμοποιώντας τις εντολές:

```
load("eigen")$
load("diag")$
A:matrix([0, 0, 0, 0, 4], [1, 0, 0, 0, -16],
[0, 1, 0, 0, 25], [0, 0, 1, 0, -19],
[0, 0, 0, 1, 7])$
X0:matrix([1/10], [-1/15], [3/7], [4/9], [3/4])$
et:ratsimp(mat_function(exp,t*A))$
xt:ratsimp(et.X0)$
x(t)=xt;
plot2d(makelist(xt[m][1],m,1,length(xt)), [t,0,10],
[xlabel, "t"], [ylabel, "xi(t)"])$
xmax:float(subst(10,t,xt))$
x(10)=xmax;
```

Έχουμε την γραφική παράσταση των πέντε συναρτήσεων $x_i(t)$, $i = 1, \dots, 5$:



Σχήμα 1: Γράφημα της λύσης του συστήματος 3 με το wxMaxima

Φυσικά μας ενδιαφέρει η λύση στο τέλος του χρόνου $t = 10$ οπότε και είναι:

$$x(10) = \begin{pmatrix} 1.3878952370488977 \cdot 10^{11} \\ -4.8993923789736987 \cdot 10^{11} \\ 6.3708244797780591 \cdot 10^{11} \\ -3.5950560884359155 \cdot 10^{11} \\ 7.357291154178479 \cdot 10^{10} \end{pmatrix} \quad (7)$$

Η απάντηση του wxMaxima φαίνεται ακολούθως:

```
(%i1) load("eigen")$
load("diag")$
A:matrix([0, 0, 0, 0, 4], [1, 0, 0, 0, -16],
[0, 1, 0, 0, 25], [0, 0, 1, 0, -19],
[0, 0, 0, 1, 7])$
X0:matrix([1/10], [-1/15], [3/7], [4/9], [3/4])$
et:ratsimp(mat_function(exp,t*A))$
xt:ratsimp(et.X0)$
x(t)=xt;
plot2d(makelist(xt[m][1],m,1,length(xt)), [t,0,10],
[xlabel, "t"], [ylabel, "xi(t)"])$
xmax:float(subst(10,t,xt))$
x(10)=xmax;
```

(%i3)

(%i4)

$$x(t) = \begin{pmatrix} \frac{(21718t - 36977)e^{2t} + (2087t^2 + 17086t + 37040)e^t}{630} \\ \frac{(76013t - 123990)e^{2t} + (6261t^2 + 55432t + 124032)e^t}{630} \\ \frac{(390924t - 600432)e^{2t} + (27131t^2 + 255510t + 601512)e^t}{2520} \\ \frac{(108590t - 152308)e^{2t} + (6261t^2 + 61693t + 151748)e^t}{1260} \\ \frac{(43436t - 52236)e^{2t} + (2087t^2 + 21260t + 54126)e^t}{2520} \end{pmatrix}$$

(%i10)

$$x(10) = \begin{pmatrix} 1.3878952370488977 \cdot 10^{11} \\ -4.8993923789736975 \cdot 10^{11} \\ 6.3708244797780579 \cdot 10^{11} \\ -3.5950560884359149 \cdot 10^{11} \\ 7.357291154178479 \cdot 10^{10} \end{pmatrix}$$

(%o10)

Σχήμα 2: Λύση του συστήματος 3 με το wxMaxima

Σημειώνουμε εδώ ότι η ακριβής λύση του Π.Α.Τ. 3 είναι:

$$x(t) = \begin{pmatrix} \frac{(21718t-36977)e^{2t} + (2087t^2 + 17086t + 37040)e^t}{630} \\ -\frac{(76013t-123990)e^{2t} + (6261t^2 + 55432t + 124032)e^t}{630} \\ \frac{(390924t-600432)e^{2t} + (27131t^2 + 255510t + 601512)e^t}{2520} \\ -\frac{(108590t-152308)e^{2t} + (6261t^2 + 61693t + 151748)e^t}{1260} \\ \frac{(43436t-52236)e^{2t} + (2087t^2 + 21260t + 54126)e^t}{2520} \end{pmatrix} \quad (8)$$

Μπορούμε να κάνουμε επαλήθευση, είτε με πράξεις στο χέρι είτε με το wxMaxima, για να είμαστε απόλυτα σίγουροι για το ακριβές της λύσης μας:

$$x'(t) - A \cdot x(t) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

ή με εντολές του wxMaxima ως εξής:

```
(%i14) diff(xt,t)-ratsimp(A.xt)$ratsimp(%);
(%o15)  $\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$ 
```

1.4 Προσεγγιστική λύση με το MATLAB

1.4.1 Λύση με την χρήση της μεθόδου ode45

Λύνουμε τώρα το πρόβλημα 1 και 2 χρησιμοποιώντας το MATLAB και την εντολή ode25, δηλ. την μέθοδο Runge-Kutta 45. Για τον σκοπό αυτό δημιουργούμε τα παρακάτω δύο m-files:

m-file:sys5.m

```
function out1 = sys5(t,x)
out1 = [4*x(5);x(1)-16*x(5);25*x(5)+x(2);x(3)-19*x(5);7*x(5)+x(4)];
```

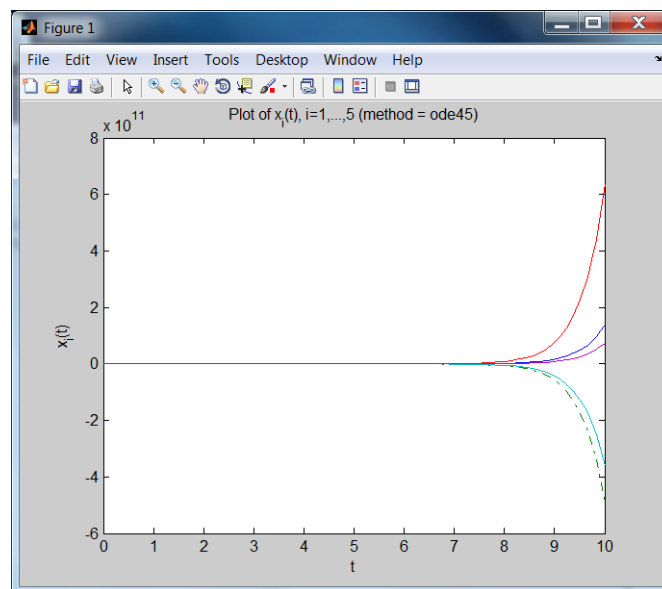
m-file: sysode45.m

```
function [x,xend]=sysode45(t1,t2,x0)
[t,x] = ode45('sys5',[t1 t2],x0);
clf
plot(t,x(:,1),'-',t,x(:,2),'-.',t,x(:,3),'-',t,x(:,4),'-',t,x(:,5),'-')
set(gca,'XTick',t1:t2)
xlabel('t')
ylabel('x_{i}(t)')
title('Plot of x_{i}(t), i=1,...,5 (method = ode45)')
xend=[x(end,1);x(end,2);x(end,3);x(end,4);x(end,5)];
end
```

Τώρα έχουμε στο MATLAB τα αποτελέσματα:

```
>> format compact
>> format long e
>> [xall45,xmat45]=sysode45(0,10,[1/10;-1/15;3/7;4/9;3/4]);
>> xmat45
xmat45 =
    1.388295292439706e+011
   -4.900805164349407e+011
    6.372662504697011e+011
   -3.596094003855919e+011
    7.359417359143037e+010
```


Επίσης το γράφημα:



Σχήμα 3: Γράφημα της λύσης του συστήματος 3 με το MATLAB (ode45)

Βλέπουμε ότι και το MATLAB έδωσε περίπου τα ίδια αποτελέσματα. Εάν συγκρίνουμε τις δύο λύσεις στο τέλος του χρόνου $t = 10$ θα δούμε ότι η επί τοις εκατό διαφορά τους είναι:

```
>> xmax=[1.3878952370488977e+011;  
-4.8993923789736975e+011;  
6.3708244797780579e+011;  
-3.5950560884359149e+011;  
7.357291154178479e+010]  
xmax =  
    1.387895237048898e+011  
   -4.899392378973698e+011  
    6.370824479778058e+011  
   -3.595056088435915e+011  
    7.357291154178479e+010  
>> (xmat45-xmax)./xmax*100  
ans =  
    2.882461010953169e-002  
    2.883593038541592e-002  
    2.885066014276175e-002  
    2.887063218131414e-002  
    2.889929078518063e-002
```

Δηλαδή έχουμε ένα σχετικό σφάλμα στην λύση του MATLAB της τάξης του 0.03%, το οποίο μπορεί να θεωρηθεί μικρό. Εάν όμως δούμε το απόλυτο σφάλμα, τότε αυτό είναι:

```
>> xmat45-xmax
ans =
    4.000553908081055e+007
   -1.412785375709229e+008
    1.838024918952637e+008
   -1.037915420004273e+008
    2.126204964558411e+007
```

Έχουμε ένα απόλυτο σφάλμα στην λύση του MATLAB της τάξης των 100 εκατομμυρίων, πρόκειται για πολύ μεγάλη τιμή σφάλματος.

1.4.2 Λύση με την χρήση του εκθετικού πίνακα

Θα δούμε ότι η κατάσταση βελτιώνεται όταν χρησιμοποιήσουμε την συνάρτηση `funm()` για να παίρνουμε την λύση με την βοήθεια του εκθετικού πίνακα. Αυτό γίνεται με το να ορίσουμε την ανώνυμη συνάρτηση:

```
>> xt=@(A,t,x0)funm(A*t,@exp)*x0
xt =
    @(A,t,x0)funm(A*t,@exp)*x0
```

Κατόπιν δημιουργούμε το επόμενο αρχείο - συνάρτηση με το οποίο βρίσκουμε την λύση οποιουδήποτε Π.Α.Τ. της μορφής 3 για το χρονικό διάστημα $[0, t_1]$, λαμβάνουμε την γραφική παράσταση της λύσης στο ίδιο διάστημα, ενώ μας επιστρέφεται τόσο η συνολική λύση όσο και η λύση στο τέλος του χρόνου t_1 :

```
function [xsol,solfin]=syssol(A,t1,x0,xt)
t=linspace(0,t1);
nn=size(t);
n=nn(2);
xsol=zeros(n,5);
for i=1:n
xsol(i,:)=xt(A,t(i),x0)';
end
clf
plot(t,xsol)
set(gca,'XTick',0:t1)
```

```

xlabel('t')
ylabel('x_{i}(t)')
title('Plot of x_{i}(t), i=1,...,5 (method = matrix exponential)')
solfin=[xsol(end,1);xsol(end,2);xsol(end,3);xsol(end,4);xsol(end,5)];
end

```

Τώρα ορίζουμε τον πίνακα A και το διάνυσμα x0 με τις αρχικές τιμές της αρεσκείας μας και καλούμε την συνάρτηση syssol που ήδη έχουμε ορίσει:

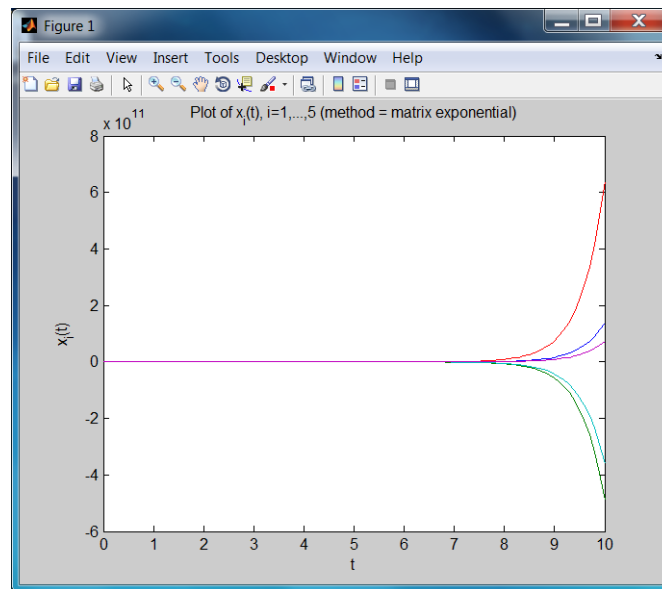
```

>> A=[0, 0, 0, 0, 4;
1, 0, 0, 0, -16;
0, 1, 0, 0, 25;
0, 0, 1, 0, -19;
0, 0, 0, 1, 7]
A =
     0     0     0     0     4
     1     0     0     0    -16
     0     1     0     0    25
     0     0     1     0   -19
     0     0     0     1     7
>> x0=[1/10;-1/15;3/7;4/9;3/4]
x0 =
 1.0000000000000000e-001
-6.666666666666667e-002
 4.285714285714286e-001
 4.444444444444444e-001
 7.500000000000000e-001
>> [xall,xmatexp]=syssol(A,10,x0,xt);
>> xmatexp
xmatexp =
 1.387895237048510e+011
-4.899392378972300e+011
 6.370824479776191e+011
-3.595056088434828e+011
 7.357291154176115e+010
>> (xmatexp-xmax)./xmax*100
ans =
-2.796922875644423e-011
-2.852813104178819e-011
-2.929691571396144e-011
-3.023697283358469e-011
-3.212576988255903e-011
>> xmatexp-xmax

```

```
ans =  
-3.881835937500000e-002  
 1.397705078125000e-001  
-1.866455078125000e-001  
 1.087036132812500e-001  
-2.363586425781250e-002
```

Τώρα το μέγιστο απόλυτο σφάλμα είναι της τάξης του 0.2 , ενώ το σχετικό σφάλμα είναι αμελητέο. Επομένως μόνον με την χρήση συναρτήσεων πινάκων καταφέραμε να λύσουμε το πρόβλημα ικανοποιητικά χρησιμοποιώντας το MATLAB. Επίσης το γράφημα δεν αλλάζει αισθητά:



Σχήμα 4: Γράφημα της λύσης του συστήματος 3 με το MATLAB (e^{At})

Πιθανόν η ακρίβεια της λύσης να μπορεί να βελτιωθεί περαιτέρω. Εάν έχετε κάποια πρόταση γι αυτόν τον σκοπό μην διστάσετε να την αποστείλετε στον γράφοντα³, ώστε να ενσωματωθεί σε κάποια επόμενη έκδοση του παρόντος παραδείγματος.

³Δημήτριος Θ. Χριστόπουλος, dchristop@econ.uoa.gr

1.5 Προσεγγιστική λύση με το Octave

1.5.1 Λύση με την χρήση της μεθόδου ode45

Θα χρησιμοποιήσουμε την συνάρτηση των παραγώγων και τις επιλογές που δίνονται ακολουθώς:

```
octave-3.2.4.exe:1> format compact
octave-3.2.4.exe:2> format long e
octave-3.2.4.exe:3> octsys=@(t,x) [4*x(5),x(1)-16*x(5),25*x(5)+x(2),\
x(3)-19*x(5),7*x(5)+x(4)]
octsys =

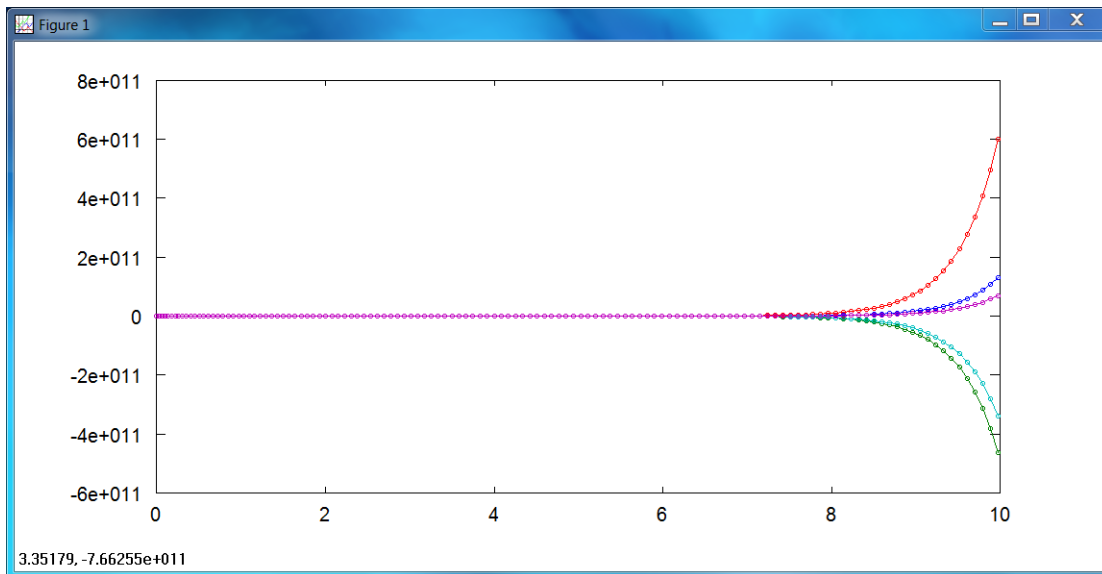
@(t, x) [4 * x(5), x(1) - 16 * x(5), 25 * x(5) + x(2), x(3) - 19 * x(5),\
7 * x(5) + x(4)]

octave-3.2.4.exe:4> vopt = odeset ("OutputFcn", @odeplot);
```

Τώρα απλά καλούμε την ode45 και παρακολουθούμε σε γράφημα - animation την κατασκευή της λύσης:

```
octave-3.2.4.exe:4>sol=ode45(octsys, [0 10],[1/10 -1/15 3/7 4/9 3/4],vopt);
warning: Option "RelTol" not set, new value 0.000001 is used
warning: Option "AbsTol" not set, new value 0.000001 is used
warning: Option "InitialStep" not set, new value 1.000000 is used
warning: Option "MaxStep" not set, new value 1.000000 is used
warning: isstr is obsolete and will be removed from a future
version of Octave,
please use ischar instead
```

Στο τέλος έχουμε το γράφημα (με αρκετή χρονική καθυστέρηση βέβαια):



Σχήμα 5: Γράφημα της λύσης του συστήματος 3 με το Octave (ode45)

Η λύση στο τέλος του χρόνου $t = 10$ είναι:

```
octave-3.2.4.exe:5> xall=sol.y;
octave-3.2.4.exe:6> xoct45=[xall(end,1);xall(end,2);xall(end,3);\
xall(end,4);xall(end,5)]
xoct45 =
```

```
1.38789118116933e+011
-4.89937809310527e+011
6.37080595744849e+011
-3.59504567786361e+011
7.35726997186237e+010
```

Τόσο το σχετικό όσο και το απόλυτο σφάλμα είναι μικρότερο από αυτό του MATLAB. Το απόλυτο σφάλμα είναι περίπου ένα εκατομμύριο. Πάντα αναφερόμαστε στις έτοιμες (default) ρυθμίσεις που έχει το κάθε πρόγραμμα για την ode45.

```
octave-3.2.4.exe:7> xmax=[1.3878952370488977e+011;\
-4.8993923789736975e+011;6.3708244797780579e+011;\
-3.5950560884359149e+011;7.357291154178479e+010]
xmax =
```

```
1.38789523704890e+011
-4.89939237897370e+011
```

```
6.37082447977806e+011
-3.59505608843591e+011
7.35729115417848e+010
```

```
octave-3.2.4.exe:8> (xoct45-xmax)./xmax*100
ans =
```

```
-2.92232400163888e-004
-2.91584493013144e-004
-2.90736774028634e-004
-2.89580247043527e-004
-2.87909172946772e-004
```

```
octave-3.2.4.exe:9> xoct45-xmax
ans =
```

```
-4.05587956298828e+005
1.42858684289551e+006
-1.85223295715332e+006
1.04105723022461e+006
-2.11823161132812e+005
```

1.5.2 Λύση με την χρήση του εκθετικού πίνακα

Ορίζουμε αρχικά την λύση του προβλήματος 3 με την ανώνυμη συνάρτηση:

```
octave-3.2.4.exe:1> format compact
octave-3.2.4.exe:2> format long e
octave-3.2.4.exe:3> xt=@(A,t,x0)expm(A*t)*x0
xt =
```

```
@(A, t, x0) expm (A * t) * x0
```

Δημιουργούμε την συνάρτηση octsyssol.m:

```
function [xsol,solfin]=octsyssol(A,t1,x0,xt)
t=linspace(0,t1);
nn=size(t);
n=nn(2);
xsol=zeros(n,5);
for i=1:n
xsol(i,:)=xt(A,t(i),x0)';
```

```

end
clf
plot(t,xsol)
set(gca,'XTick',0:t1)
xlabel('t')
ylabel('x_{i}(t)')
title('Plot of x_{i}(t), i=1,...,5 (method = matrix exponential)')
solfin=[xsol(end,1);xsol(end,2);xsol(end,3);xsol(end,4);xsol(end,5)];
end

```

Ορίζουμε τον πίνακα A , το διάνυσμα αρχικών τιμών x_0 και καλούμε την συνάρτηση (m - file) που ήδη έχουμε ορίσει πριν:

```

octave-3.2.4.exe:7> [xall,xoctexp]=octsyssol(A,10,x0,xt);
warning: isstr is obsolete and will be removed from a future
version of Octave,
please use ischar instead
octave-3.2.4.exe:8> [xall,xoctexp]=octsyssol(A,10,x0,xt);
octave-3.2.4.exe:9> xoctexp
xoctexp =

```

```

1.38789523704946e+011
-4.89939237897572e+011
6.37082447978074e+011
-3.59505608843745e+011
7.35729115418177e+010

```

```

octave-3.2.4.exe:10> xmax=[1.3878952370488977e+011;\
-4.8993923789736975e+011;> 6.3708244797780579e+011;\
-3.5950560884359149e+011;7.357291154178479e+010]
xmax =

```

```

1.38789523704890e+011
-4.89939237897370e+011
6.37082447977806e+011
-3.59505608843591e+011
7.35729115417848e+010

```

```

octave-3.2.4.exe:11> (xoctexp-xmax)./xmax*100
ans =

```

```

4.03486908554050e-011
4.12972726652960e-011

```



```
4.20580313879303e-011
4.28342967204571e-011
4.47147578223352e-011
```

```
octave-3.2.4.exe:12> xoctexp-xmax
ans =
```

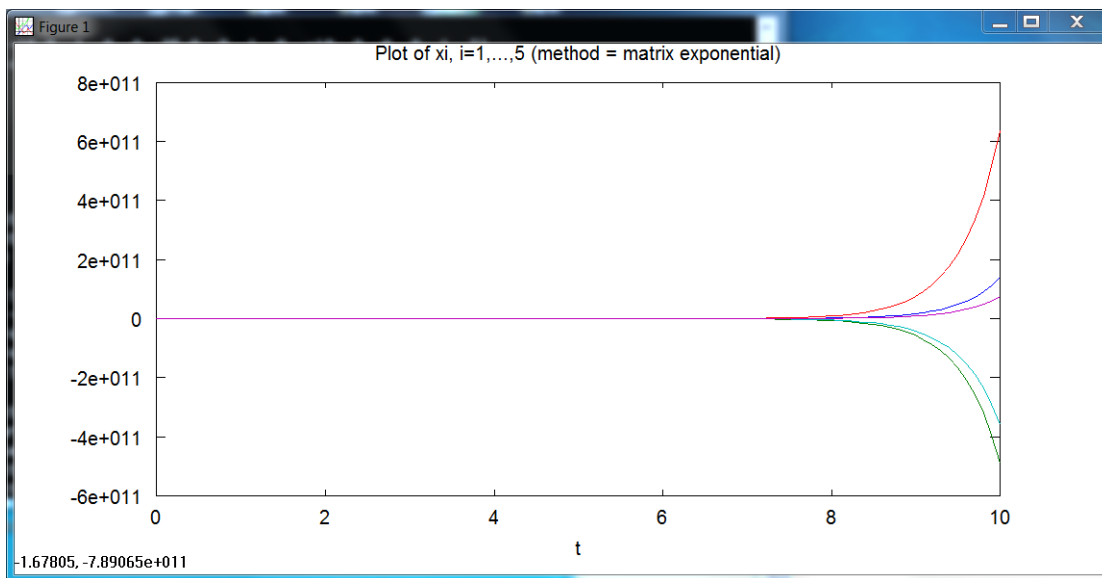
```
5.59997558593750e-002
-2.02331542968750e-001
2.67944335937500e-001
-1.53991699218750e-001
3.28979492187500e-002
```

Παρατηρούμε ότι το Octave έχει ελαφρά μεγαλύτερο απόλυτο σφάλμα από το MATLAB, περίπου μιάμιση φορά μεγαλύτερο για την ακρίβεια:

```
>> dmat=abs(xmatexp-xmax)
dmat =
    3.88183593750000e-002
    1.39770507812500e-001
    1.86645507812500e-001
    1.08703613281250e-001
    2.36358642578125e-002
>> xoctexp=[ 1.38789523704946e+011;
-4.89939237897572e+011;
6.37082447978074e+011;
-3.59505608843745e+011;
7.35729115418177e+010]
xoctexp =
    1.38789523704946e+011
   -4.89939237897572e+011
    6.37082447978074e+011
   -3.59505608843745e+011
    7.35729115418177e+010
>> doct=abs(xoctexp-xmax)
doct =
    5.62438964843750e-002
    2.02270507812500e-001
    2.68188476562500e-001
    1.53503417968750e-001
    3.29132080078125e-002
>> doct./dmat
ans =
```

1.448899371069182e+000
1.447161572052402e+000
1.436886854153041e+000
1.412128017967434e+000
1.392511297611362e+000

Το γράφημα της λύσης με το Octave είναι:



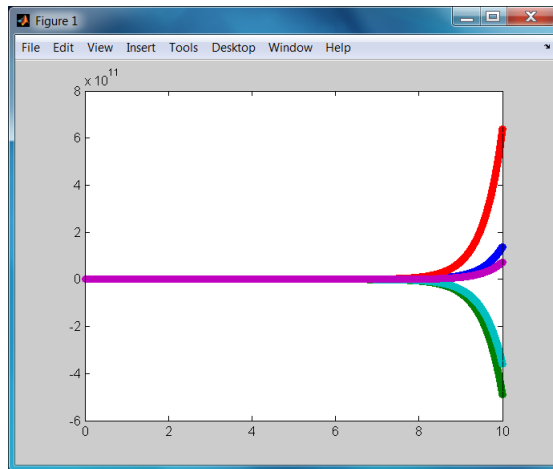
Σχήμα 6: Γράφημα της λύσης του συστήματος 3 με το Octave ($e^{A \cdot t}$)

2 Προβληματισμοί και Συμπεράσματα

1. Μπορούμε να ορίσουμε εμείς τις παραμέτρους της συνάρτησης ode45 στο MATLAB. Δοκιμάζουμε λοιπόν τις εξής παραμέτρους και έχουμε τα ακόλουθα αποτελέσματα:

```
>> format compact
>> ftx=@(t,x) [4*x(5);x(1)-16*x(5);25*x(5)+x(2);x(3)-19*x(5);
7*x(5)+x(4)]
ftx =
    @(t,x) [4*x(5);x(1)-16*x(5);25*x(5)+x(2);x(3)-19*x(5);7*x(5)+x(4)]
>> tic; sol=ode45(ftx, [0 10], [1/10;-1/15;3/7;4/9;3/4], vopt); toc
Elapsed time is 12.246467 seconds.
>> xall=sol.y;
>> xmat45=[xall(1,end);xall(2,end);xall(3,end);
xall(4,end);xall(5,end)]
xmat45 =
    1.387895237050829e+011
   -4.899392378980493e+011
    6.370824479786863e+011
   -3.595056088440857e+011
    7.357291154188542e+010
>> (xmat45-xmax)./xmax*100
ans =
    1.391205269984454e-010
    1.386915645800122e-010
    1.381879372982929e-010
    1.374500460756326e-010
    1.367782132830709e-010
>> xmat45-xmax
ans =
    1.930847167968750e-001
   -6.795043945312500e-001
    8.803710937500000e-001
   -4.941406250000000e-001
    1.006317138671875e-001
```

Το γράφημα είναι τώρα “animation” και η τελική μορφή του είναι η εξής:



Σχήμα 7: Βελτίωση της λύσης του συστήματος 3 με το MATLAB (ode45)

Αν δεν μας ενδιαφέρει να έχουμε animation, τότε ο χρόνος υπολογισμού μικραίνει πολύ:

```
>> vopt=odeset ('RelTol', 1e-12, 'AbsTol', 1e-12, 'NormControl', 'on');
>> tic; sol=ode45(ftx, [0 10], [1/10; -1/15; 3/7; 4/9; 3/4], vopt); toc
Elapsed time is 1.065333 seconds.
```

2. Θα κάνουμε ακριβώς το ίδιο και στο Octave για να βελτιώσουμε τις επιδόσεις του. Δοκιμάζουμε λοιπόν τις ίδιες παραμέτρους με το MATLAB και έχουμε τα ακόλουθα αποτελέσματα:

```
octave-3.2.4.exe:1> format compact
octave-3.2.4.exe:2> format long e
octave-3.2.4.exe:3> ftx=@(t,x) [4*x(5), x(1)-16*x(5), \
25*x(5)+x(2), x(3)-19*x(5), 7*x(5)+x(4)]
ftx =

@(t, x) [4 * x(5), x(1) - 16 * x(5), 25 * x(5) + x(2), \
x(3) - 19 * x(5), 7 * x(5) + x(4)]
octave-3.2.4.exe:4> vopt = odeset ("RelTol", 1e-12, \
"AbsTol", 1e-12, > "NormControl", "on", "OutputFcn", @odeplot);
octave-3.2.4.exe:4> [sol]=ode45(ftx, [0 10], [1/10 -1/15 3/7 4/9 3/4], \
vopt); toc
warning: Option "InitialStep" not set, new value 1.000000 is used
warning: Option "MaxStep" not set, new value 1.000000 is used
```

```

warning: isstr is obsolete and will be removed from a future \
version of Octave, please use ischar instead
Elapsed time is 2729.37 seconds.
octave-3.2.4.exe:5>xall=sol.y;
octave-3.2.4.exe:6>xoct45=[xall(end,1);xall(end,2);xall(end,3);\
xall(end,4);xall(end,5)]
xoct45 =

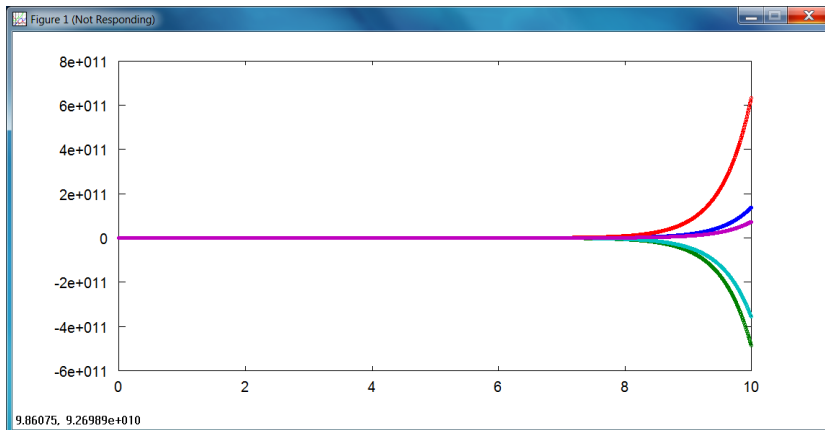
    1.38789523704426e+011
   -4.89939237895737e+011
    6.37082447975687e+011
   -3.59505608842399e+011
    7.35729115415419e+010
octave-3.2.4.exe:7>xmax=[1.3878952370488977*1.0e+011;\
-4.8993923789736987*1.0e+011;6.3708244797780591*1.0e+011;\
-3.5950560884359155*1.0e+011;7.357291154178479*1.0e+010];
octave-3.2.4.exe:8>(xoct45-xmax)./xmax*100
ans =

   -3.33893662473746e-010
   -3.33318198320561e-010
   -3.32555440897001e-010
   -3.31638982218553e-010
   -3.30092804035384e-010
octave-3.2.4.exe:9>xoct45-xmax
ans =

   -4.63409423828125e-001
    1.63305664062500e+000
   -2.11865234375000e+000
    1.19226074218750e+000
   -2.42858886718750e-001

```

Το γράφημα είναι βέβαια “animation”, αλλά γίνεται εξαιρετικά αργά, συγκριτικά με το MATLAB, ενώ η τελική μορφή του είναι η εξής:



Σχήμα 8: Βελτίωση της λύσης του συστήματος 3 με το Octave (ode45)

Εάν δεν μας ενδιαφέρει το ‘ζωντανό’ γράφημα, τότε έχουμε εξαιρετική βελτίωση του χρόνου εκτέλεσης:

```
octave-3.2.4.exe:7> vopt = odeset ("RelTol", 1e-12,\
"AbsTol", 1e-12,"NormControl", "on");
octave-3.2.4.exe:8>tic;[sol]=ode45(ftx, [0 10],\
[1/10 -1/15 3/7 4/9 3/4],vopt);toc
warning: Option "InitialStep" not set, new value 1.000000 is used
warning: Option "MaxStep" not set, new value 1.000000 is used
Elapsed time is 5.19481 seconds.
```

Πρέπει όμως να τονίσουμε ότι το Octave δεν κατάφερε να μειώσει το απόλυτο σφάλμα στα επίπεδα που το έφτασε το MATLAB.

3. Παρατηρούμε λοιπόν ότι για να μπορέσουμε να επιτύχουμε ικανοποιητική ακρίβεια με κάποιο από τα προγράμματα MATLAB ή Octave θα πρέπει να ορίσουμε τόσο μικρές τις παραμέτρους επίλυσης της Runge-Kutta 45, ώστε ο χρόνος υπολογισμού να πρέπει να αυξηθεί αρκετά. Επίσης ακόμα και τότε δεν είμαστε σίγουροι ότι θα επιτύχουμε την πολυπόθητη ακρίβεια και πάντα πρέπει να είμαστε επιφυλακτικοί για τα αποτελέσματά μας.

Αντίθετα, αν χρησιμοποιήσουμε το wxMaxima μπορούμε να υπολογίσουμε με απόλυτη ακρίβεια την λύση για οιοδήποτε χρόνο t και μάλιστα σε ελάχιστο υπολογιστικό χρόνο, δεδομένου ότι κάνουμε μία απλή αντικατάσταση κάθε φορά.

Καταλαβαίνουμε λοιπόν γιατί πρέπει να αφιερώσουμε χρόνο ώστε να αποκτήσουμε την γνώση Συστημάτων Υπολογιστικής Άλγεβρας (CAS), όπως το wxMaxima: για να λύνουμε τουλάχιστον τα γραμμικά προβλήματά μας με απόλυτη ακρίβεια και σε σύντομο υπολογιστικό χρόνο.

Αναφορές

- Boyce, W. & DiPrima, R. (1997), *Στοιχειώδεις Διαφορικές Εξισώσεις και Προβλήματα Αρχικών Τιμών*, 6η έκδοση, Πανεπιστημιακές Εκδόσεις Ε.Μ.Π.
- G. Golub & C. Van Loan (1996), *Matrix Computations*, 3rd edition, The Johns Hopkins University Press, Baltimore, Maryland.