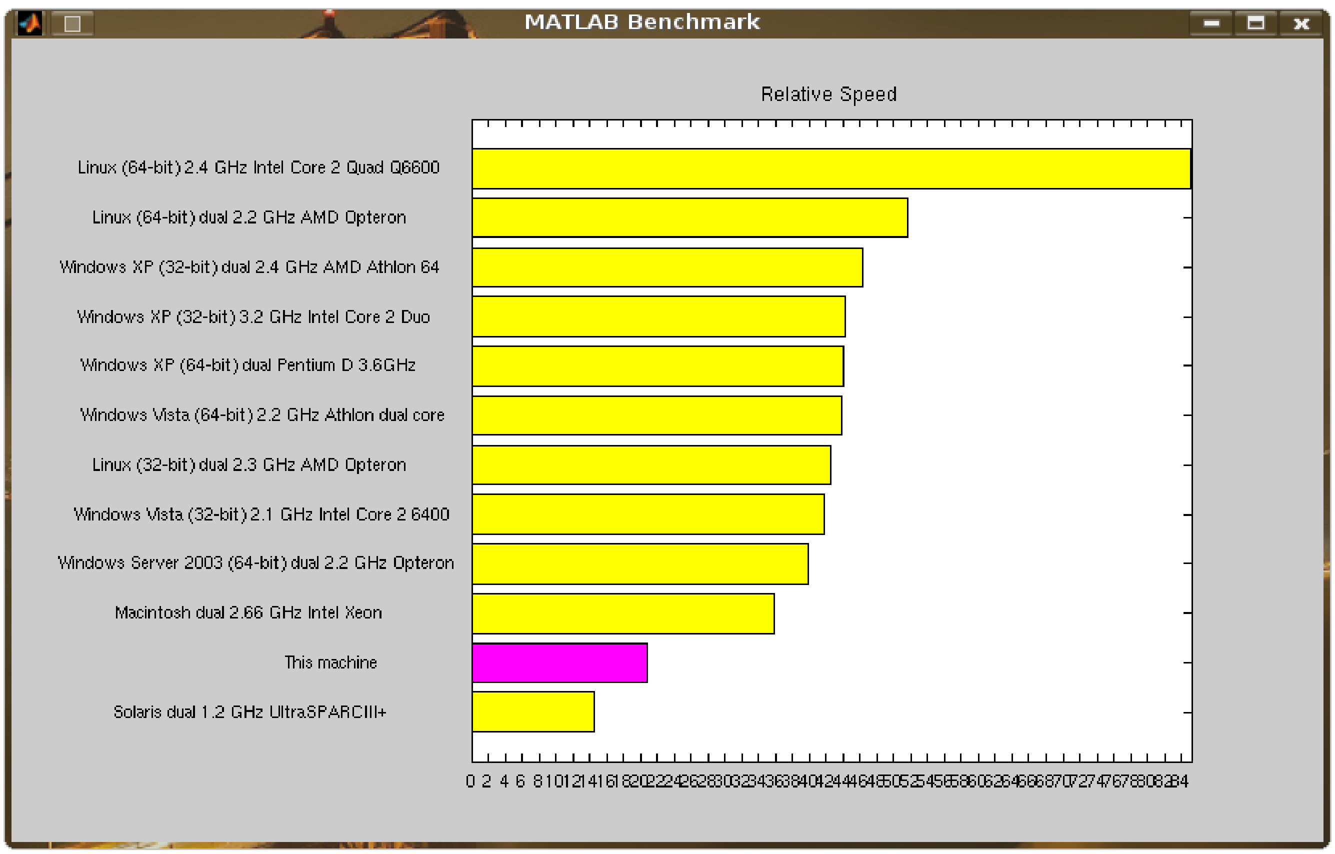
ΕΠΙΣΤΗΜΟΝΙΚΟΣ ΥΠΟΛΟΓΙΣΜΟΣ Ι
Ονοματεπώνυμο: Αραβανής Κων/νος Α .Μ.:3628
1η Εργαστηριακή άσκηση

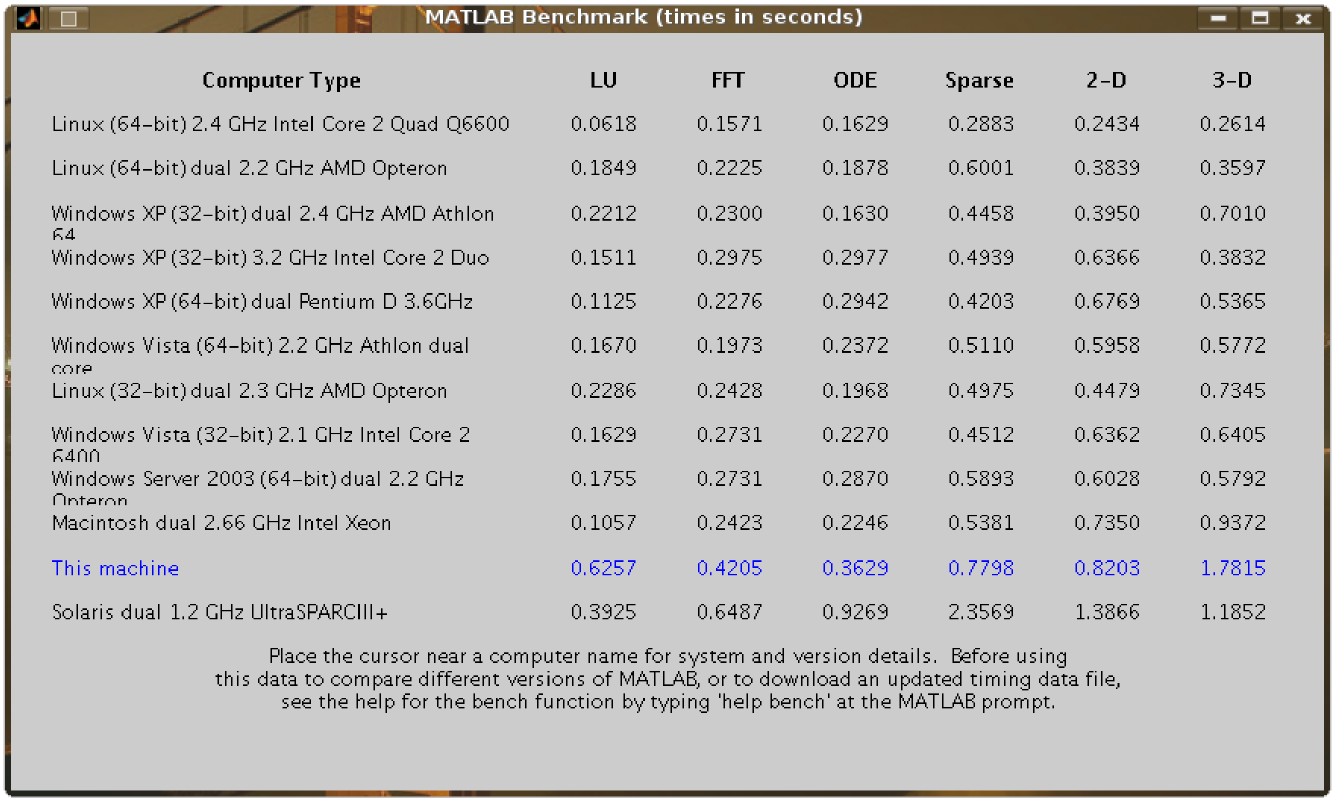
Εισαγωγικό σημείωμα: Αυτή η αναφορά συνοδεύεται από ένα zip-αρισμένο αρχείο στο οποίο μπορείτε να βρείτε τα σχήματα, τους κώδικες αλλα και τα στοιχεια των πινάκων που παρουσιάζονται παρακάτω. Συγκεκριμένα το κάθε σχήμα, ο κάθε κώδικας και ο κάθε πίνακας μετά την παραθεσή του στην αναφορά ακολουθείτε από μια λεζάντα που λέει το όνομα που έχει στο αρχείο που παραδόθηκε.

I)Αρχικά θα περιγράψουμε το σύστημα στο οποίο τρέξαμε τα πειράματα μας:
Το σύστημα μας είχε το λειτουργικό Linux/Debian Lenny και ο επεξεργαστής του ήταν ένας Intel Core Duo T2300 στα 1,66GHz. Είχαμε μία ιεραρχεία μνήμης δύο επιπέδων με το πρώτο επίπεδο να αποτελείτε από δύο κρυφές μνήμες, η μία για data (L1 D-Cache) και η άλλη για inst. (L1 D-Cache) και οι δύο των 32KBytes και με τα ακόλουθα χαρακτηριστικά: 8 way set-associative, 64-byte line size. Η κύρια μνήμη (L2 Cache) μας είχε μέγεθος 2048KBytes και ακριβώς τα ίδια χαρακτηριστικά με τις κρυφές. Όσον αφορά το λογισμικό στο οποίο εκτελέστηκαν τα προγράμματα μας ήταν η Matlab 7.6.0(R2008a).

II)Στο σημείο αυτό εκτελέσαμε την συνάρτηση bench της Matlab και τα αποτελέσματα που λάβαμε ήταν αυτά που παρουσιάζονται στις ακόλουθες εικόνες:



snapshot1.png


MATLAB Benchmark (times in seconds)

Computer Type	LU	FFT	ODE	Sparse	2-D	3-D
Linux (64-bit) 2.4 GHz Intel Core 2 Quad Q6600	0.0618	0.1571	0.1629	0.2883	0.2434	0.2614
Linux (64-bit) dual 2.2 GHz AMD Opteron	0.1849	0.2225	0.1878	0.6001	0.3839	0.3597
Windows XP (32-bit) dual 2.4 GHz AMD Athlon 64	0.2212	0.2300	0.1630	0.4458	0.3950	0.7010
Windows XP (32-bit) 3.2 GHz Intel Core 2 Duo	0.1511	0.2975	0.2977	0.4939	0.6366	0.3832
Windows XP (64-bit) dual Pentium D 3.6GHz	0.1125	0.2276	0.2942	0.4203	0.6769	0.5365
Windows Vista (64-bit) 2.2 GHz Athlon dual core	0.1670	0.1973	0.2372	0.5110	0.5958	0.5772
Linux (32-bit) dual 2.3 GHz AMD Opteron	0.2286	0.2428	0.1968	0.4975	0.4479	0.7345
Windows Vista (32-bit) 2.1 GHz Intel Core 2 6400	0.1629	0.2731	0.2270	0.4512	0.6362	0.6405
Windows Server 2003 (64-bit) dual 2.2 GHz Opteron	0.1755	0.2731	0.2870	0.5893	0.6028	0.5792
Macintosh dual 2.66 GHz Intel Xeon	0.1057	0.2423	0.2246	0.5381	0.7350	0.9372
This machine	0.6257	0.4205	0.3629	0.7798	0.8203	1.7815
Solaris dual 1.2 GHz UltraSPARCIII+	0.3925	0.6487	0.9269	2.3569	1.3866	1.1852

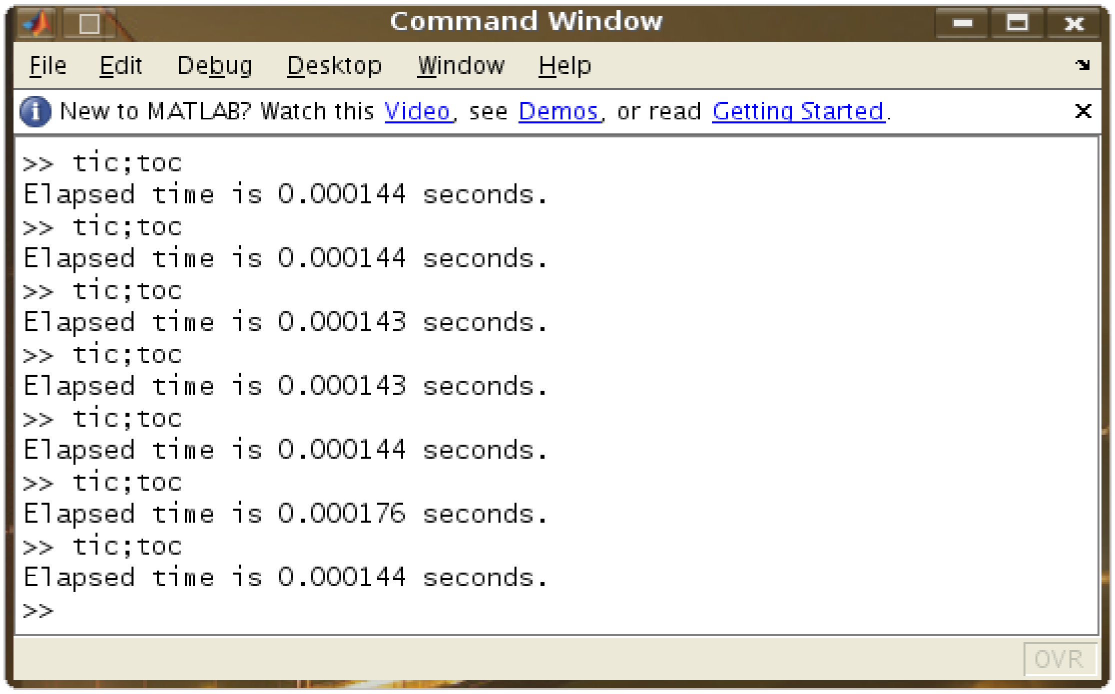
Place the cursor near a computer name for system and version details. Before using this data to compare different versions of MATLAB, or to download an updated timing data file, see the help for the bench function by typing 'help bench' at the MATLAB prompt.

snapshot2.png

Τα αποτελέσματα μόνο για το δικόμας υπολογιστή μαζεμένα φαίνονται στο παρακάτω screenshot του Command Window.

snapshot3.png

Στο ερώτημα αυτό μας ζητήθηκε επίσης να μελετήσουμε τη διακριτότητα του tic;toc. Έπαναλαμβανοντας στη Matlab πολλές φορές το tic;toc διαπυστώσαμε ότι ο χρόνος που παίρναμε πάντα ήταν λίγο μεγαλύτερος από 0.0001 (όπως φαίνεται και στην παρακάτω εικόνα). Με άλλα λόγια η καθυστέρυση που επιβάλλει σε μία χρονομέτρηση όπως αυτές που κάνουμε παρακάτω είναι αξιόλογη. Για το λόγω αυτό κάνουμε πολλές φορές τις πράξεις και βρίσκουμε έναν μέσο όρο για τον χρόνο τους ώστε να “κατεβάσουμε” το σφάλμα σε χαμηλά επίπεδα.



snapshot4.png

III)1)Στο σημείο αυτό υλοποιήσαμε μία συνάρτηση ώστε να χρονομετρήσουμε τις επτά ακόλουθες πράξεις: ανανέωση 1ης τάξης (A+b*b'), πολλαπλασιασμός μητρώων (D=A+A*A), παραγοντοποίηση LU ([L,U]=lu(A)), πίσω/εμπρός αντικατάσταση (x=U\(L\b)), διάσπαση ιδιάζουσων τιμών ([U,S,V]=svd(A)), ταχύς μετασχηματισμός Fourier (x=fft(b)), γινόμενο Hadamard (D=A.*A) για διάφορες διαστάσεις πινάκων, ακολουθώντας πάντα τις οδηγίες της εκφώνησης.

Ο κώδικας που χρησιμοποιήσαμε είναι ο ακόλουθος:

```
%ASKISI 1 ERWTIMA 3o-XRONOMETRISI 7 PRAKSEWN POU AFOROUN
%MITRWA

%gia megaliteri akrivia ta apotelesmata ta tupono kai ta apothikeuo me
%akriveia long
format long

%dimiourgia tuxaiou mitroou B 1024x1024
B=randn(1024,1024);

%dimiourgia pinakon pou tha filane ta apotelesmata tis kathe prakseis gia
%tous diafora mitroa
%px o pinakas first periexei tous xronous tis protis prakseis gia ta mitoa
%me n=64,128,256,512,1024
first=zeros(5,1);
second=zeros(5,1);
third=zeros(5,1);
fourth=zeros(5,1);
fifth=zeros(5,1);
sixth=zeros(5,1);
seventh=zeros(5,1);

%xronomerisi ton zitoumenon prakseon gia ta 5 zitoumena mitroa
for i=4:-1:0
```

```
%dimiourgia tuxaiou dianismatos b 1024x1
    b=randn(2^(10-i),1);
%kataskeui katallilou mitroou analoga me to loop gia tin xronometrisi
%ton zitoumenon prakseon
    C=B(1:2^i:1024,1:2^i:1024);
%opos kai edo etsi kai stis parakato prakseis ektelo mia fora prota tin
%praksi kathos panta i proti ektelesei epistrefei paraplanitika
%apotelesmata kai meta ksekino tin xronometrisi tis polles fores gia na
%paro ton meso oro

%xronometrisi tis D=A+b*b'
    A=C;
    D=A+b*b';
    tic;
for j=1:1:150
    D=A+b*b';
end
first(5-i,1)=toc/150;

%xronometrisi tis D=A+A*A
D=A+A*A;
tic;
for j=1:1:150
    D=A+A*A;
end
second(5-i,1)=toc/150;

%xronometrisi tis [L,U]=lu(A)
[L,U]=lu(A);
tic;
for j=1:1:150
    [L,U]=lu(A);
end
third(5-i,1)=toc/150;

%xronometrisi tis x=U\(L\b)
x=U\(L\b);
tic;
for j=1:1:150
    x=U\(L\b);
end
fourth(5-i,1)=toc/150;

%xronometrisi tis [U,S,V]=svd(A)
[U,S,V]=svd(A);
tic;
for j=1:1:20
   [U,S,V]=svd(A);
end
fifth(5-i,1)=toc/20;

%xronometrisi tis x=fft(A)
x=fft(b);
tic;
for j=1:1:150
    x=fft(b);
end
```

```
    sixth(5-i,1)=toc/150;

  %xronometrisi tis D=A.*A
    D=A.*A;
  for j=1:1:150
    D=A.*A;
  end
  seventh(5-i,1)=toc/150;
end

%emfanisi apotelesmaton me akriveia 3on dekadikon psifion
first_praksi=sprintf('%.3f\t',first)
second_praksi=sprintf('%.3f\t',second)
third_praksi=sprintf('%.3f\t',third)
fourth_praksi=sprintf('%.3f\t',fourth)
fifth_praksi=sprintf('%.3f\t',fifth)
sixth_praksi=sprintf('%.3f\t',sixth)
seventh_praksi=sprintf('%.3f\t',seventh)
```

erotima3meros1.m

Και τα αποτελέσματα που λάβαμε με τον περιορισμό ότι θα παρουσιάσουμε έως και τρία δεκαδικά ψηφία μετα την υποδιαστολή ήταν τα ακόλουθα:

Διαστάσεις	A+b*b'	D = A+A*A	[L,U] = lu(A)	x = U\(L\b)	[U,S,V] = svd(A)	x = fft(b)	D = A.*A
64x64	0.000	0.000	0.000	0.000	0.009	0.000	0.000
128x128	0.000	0.003	0.002	0.000	0.058	0.000	0.000
256x256	0.001	0.026	0.012	0.002	0.427	0.000	0.000
512x512	0.006	0.202	0.087	0.008	7.918	0.000	0.002
1024x1024	0.026	1.574	0.641	0.035	99.479	0.000	0.008

workspace_erotima3.mat (Δεν έχει συμπεριληφθεί στο .zip λόγω του μεγέθους του, αν χρειάζεται μπορώ να το αποστείλω στη διεύθυνση http://students.ceid.upatras.gr/~arabanis/workspace_erotima3.mat)

Από τον παραπάνω πίνακα φαίνεται ότι ο ταχύς μετασχηματισμός Furier διανύσματος είναι η γρηγορότερη πραξη από τις εφτά πράγμα που το αναμέναμε και από την πολυπλοκότητα της πράξης. Ο χειρότερος χρόνος που πήραμε ήταν από την διάσπαση ιδιάζουσων τιμών που πάλι είχαμε την χειρότερη πολυπλοκότητα (βέβαια αυτό δεν ήταν φανερό εξ αρχής αφού εκτός από την 5η πράξη την “ίδια” πολυπλοκότητα O(n^3) έχουν και η 2η αλλά και η 3η). Η 4η και η 1η πράξη έχουν παραπλήσιουσ χρονους πράγμα αναμενόμενο μίας και για τις δύο ο κυρίαρχος όρος είναι ο 2n^2. Τέλος το γινόμενο Hadamard είναι πολύ πιο γρήγορο από τις προαναφερθήσες (1η και 4η πράξη). Λογικό μιας και ο κυρίαρχος όρος εδώ είναι υποδιπλάσιος.

2)Στο σημείο αυτό υπολογίζουμε την επίδοση του περιβάλλοντος υπολογισμού μας σε Mflop/s όσον αφορά τις πράξεις 1, 2, 3, 4 και 7 για όλα μας τα μητρώα.

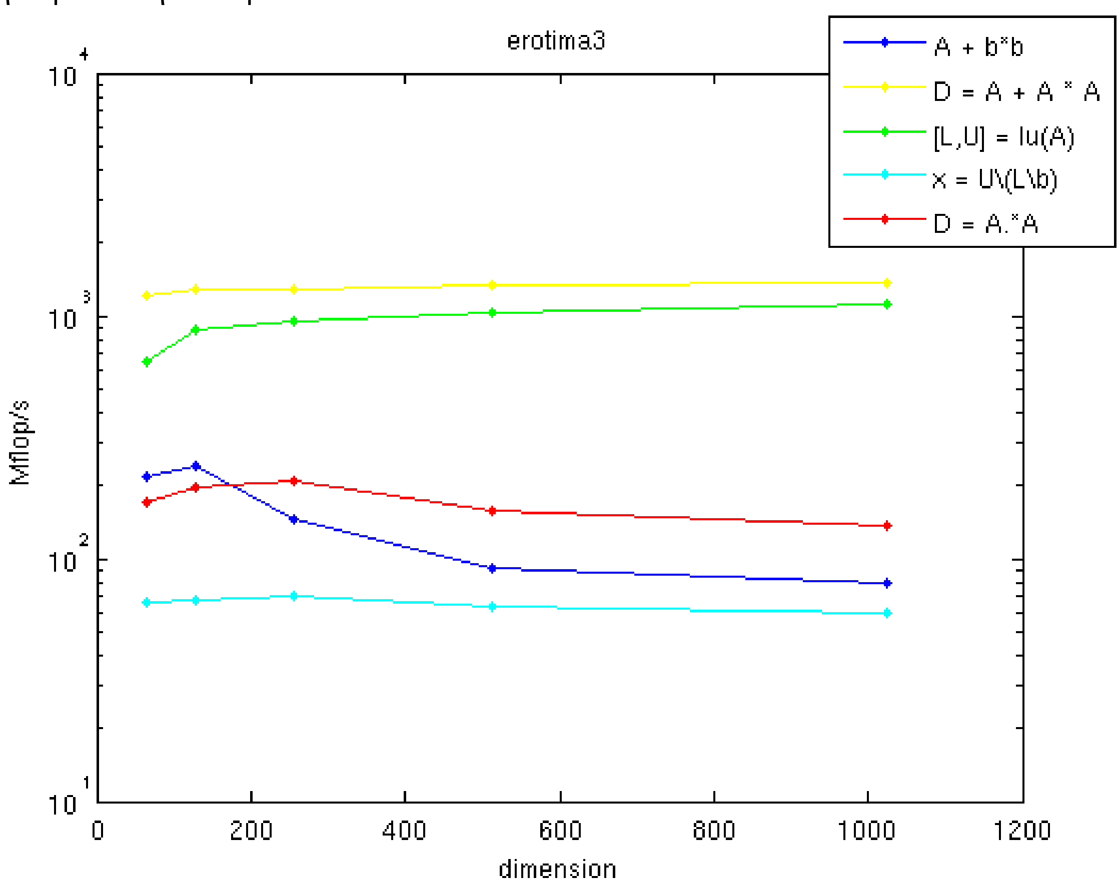
Ο κώδικας που χρησιμοποιήσαμε είναι ο ακόλουθος:

```
%ASKISI 1 ERWTIMA 3 IPOEROTIMA 2
mflops=zeros(5,5);
```

```
%arxikopoioume enan pinaka me tis diastaseis ton mitroon
x=[64;128;256;512;1024];

%upologismos ton mflops gia kathe praksi se kathe diastasis mitroo
for i=1:1:5
   mflops(i,1)=2*x(i)^2/(first(i,1)*10^6);
   mflops(i,2)=2*x(i)^3/(second(i,1)*10^6);
   mflops(i,3)=((2/3)*x(i)^3+x(i)^2)/(third(i,1)*10^6);
   mflops(i,4)=2*x(i)^2/(fourth(i,1)*10^6);
   mflops(i,5)=x(i)^2/(seventh(i,1)*10^6);
end

%grafiki parastasi ton mflops gia kathe mia apo tis prakseis 1, 2, 3, 4
%kai 7 sinartisei ton diaforon megethon mitroon
semilogy(x, mflops(:,1), '.-');
% kanoume hold gia  na emfanistoun se koini grafiki parastasi
hold on;
semilogy(x, mflops(:,2), 'y.-');
semilogy(x, mflops(:,3), 'g.-');
semilogy(x, mflops(:,4), 'c.-');
semilogy(x, mflops(:,5), 'r.-')
legend('A  + b*b', 'D = A + A * A', '[L,U] = lu(A)', 'x = U\\(L\\b)', 'D = A.*A')
hold;
```

erotima3meros2.m

Η γραφική παράσταση που προκύπτει από τον κώδικα αυτό είναι:

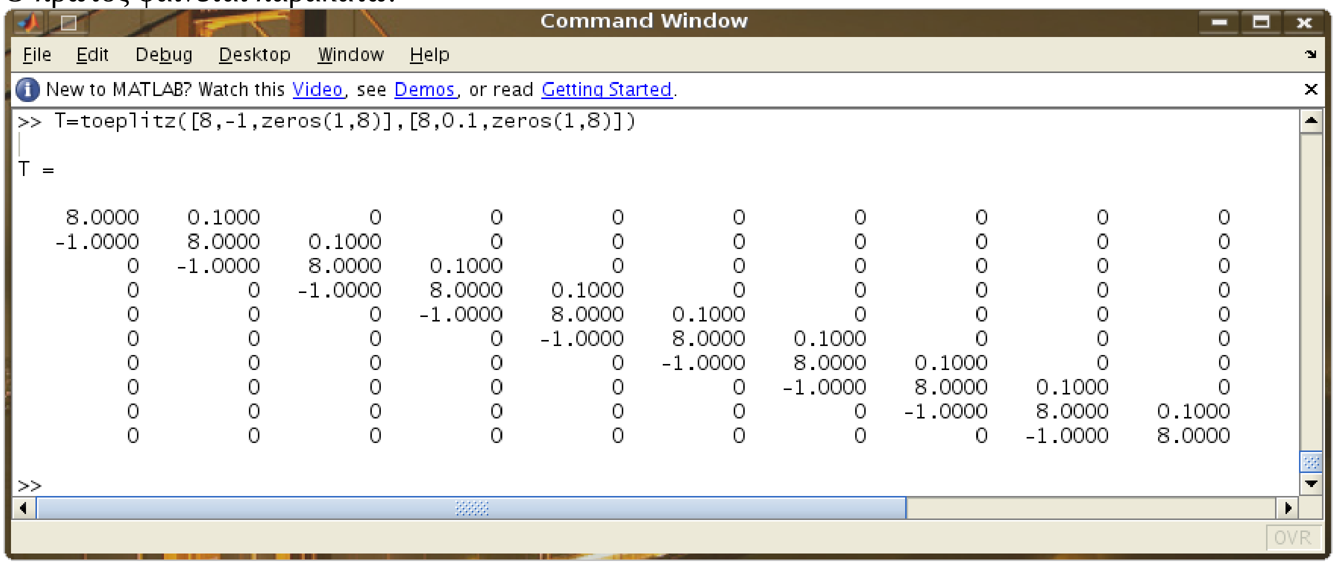


erotima3.png

3) Από την προηγούμενη γραφική παράσταση θα μπορούσαμε να κατατάξουμε τις πέντε πράξεις που βλέπουμε με βάση τις επιδόσεις τους από την καλύτερη προς την χειρότερη ως εξης:

- ✓ πολλαπλασιασμός μητρώων
- ✓ παραγοντοποίηση LU
- ✓ γινόμενο Hadamard
- ✓ ανανέωση 1ης τάξης
- ✓ πίσω/εμπρός αντικατάσταση

IV)0) Στο σημείο αυτό κατ'όπιν υπόδειξης της άσκησης φτιάξαμε και εκτυπώσαμε πίανακες toeplitz T=toeplitz([μ,-1,zeros(1,n)],[μ,0.1,zeros(1,n)]) με μ=8 (γιατι το ΑΜ μου είναι το 3628) και n=8 και 18. Ο πρώτος φαίνεται παρακάτω:



snapshot5.png

Τον δεύτερο δεν τον τυπώσαμε λόγω του ότι ήταν πολύ μεγαλύτερος και ήταν ψιλοαδύνατον να τον αναπαραστήσουμε σε μορφή εύκολα κατανοητη. Παρόλαυτα στο Command Window γράψαμε το ακόλουθο:

```
>> T=toeplitz([8,-1,zeros(1,18)],[8,0.1,zeros(1,18)])
```

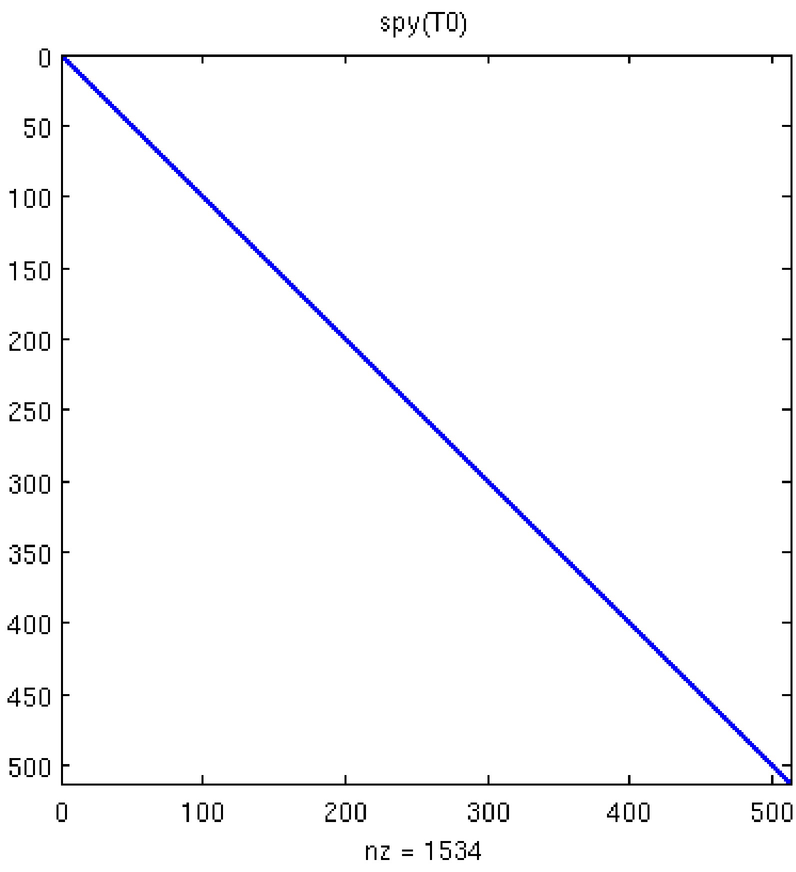
1) Στο ερώτημα αυτό μας ζητήθηκε να δημιουργήσουμε τρία συγκεκριμένα μητρώα T0, T1, T2 και να τα οπτικοποιήσουμε με τη βοήθεια της spy. Επίσης να δούμε το πλήθος των μη μηδενικών στοιχείων τους και να μετρήσουμε το χώρο που χρειάζονται για την αποθήκευσή τους. Αξίζει να σημειώσουμε εδώ οτι λόγω του ότι, όπως ξέραμε εκ των προταίρων, τα μητρώα μας είχαν μεγάλο πληθος μηδενικών στοιχείων επιλέχθηκε να τα αποθηκεύσουμε και σε full μορφή αλλά και σε sparse για να δούμε τις διαφορές τους.

Ο κώδικας που συντάξαμε ήταν ο ακόλουθος:

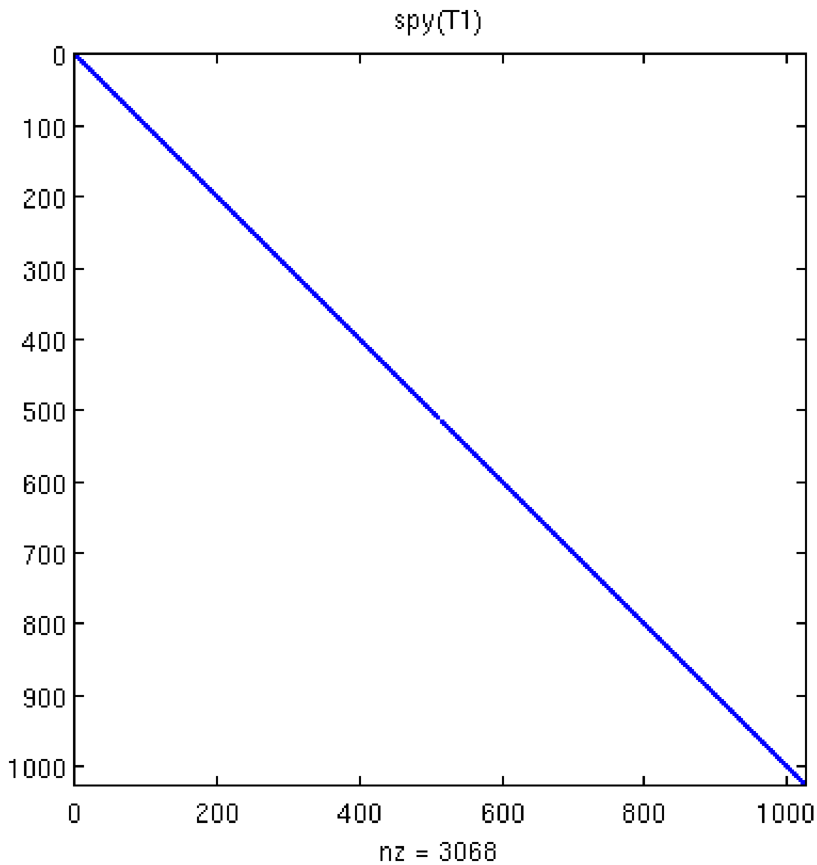
```
%ASKISI1 EROTIMA4 MEROS1

%dimiourgia T0
T0=toeplitz([8,-1,zeros(1,510)],[8,0.1,zeros(1,510)]);
%optikopoiisi to T0
spy(T0)
```

```
%dimiourgia T1
T1=kron(eye(2,2),T0);
%optikopoiisi to T1  se diaforetiko figure
figure
spy(T1)

%dimiourgia T2
T2=kron([1,-1;-1,1],T1);
%optikopoiisi to T2 se diaforetiko figure
figure
spy(T2)

%plithos mi midenikon stoixeion gia ta T0, T1 kai T2 antistoixa
nnzT0=nnz(T0)
nnzT1=nnz(T1)
nnzT2=nnz(T2)

%anaparastasi se sparse morfi ton T0, T1 kai T2 antistoixa
T0sparse=sparse(T0);
T1sparse=sparse(T1);
T2sparse=sparse(T2);

%emfanisi tou xorou pou xreiazetai gia tin apothikeusi ton mitroon mas eite
%einai se fuul eite se sparse anaparastasi
whos
```
erotima4meros1.m

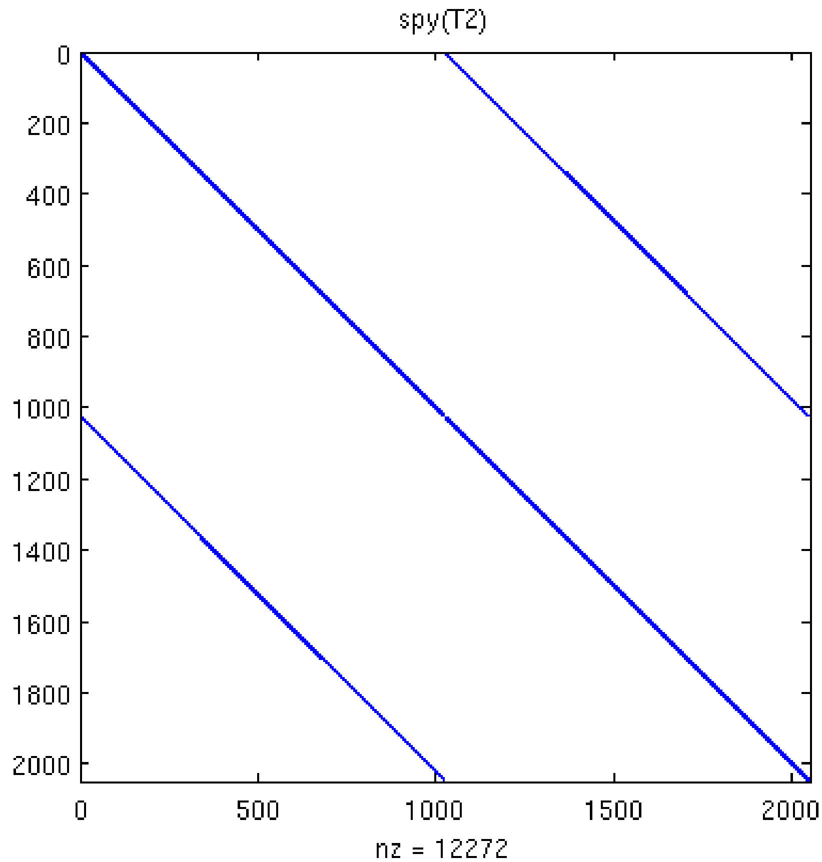
Και τα αποτελέσματα που λάβαμε ήταν τα ακόλουθα:



erotima4_1.png

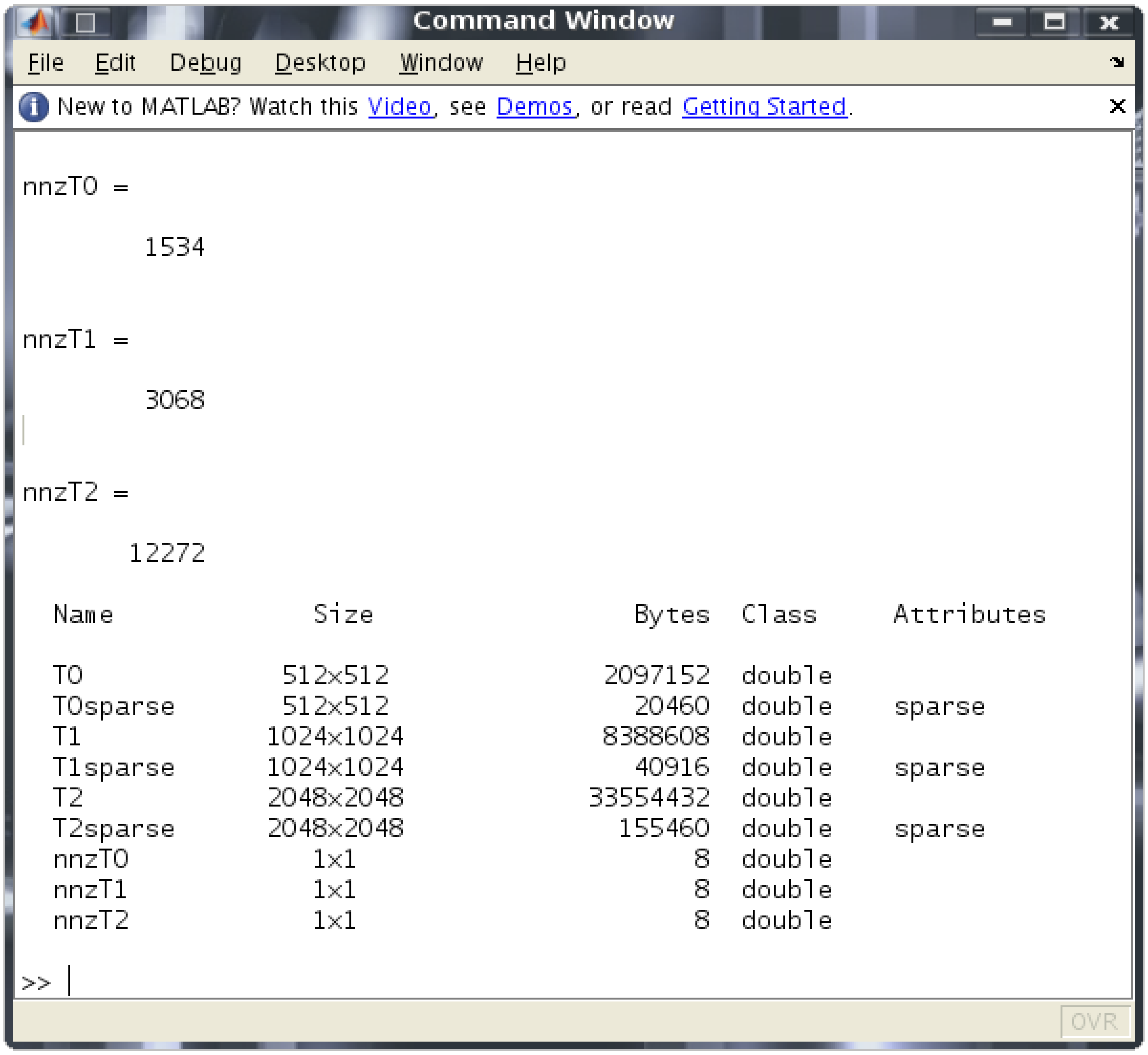


erotima4_2.png



erotima4_3.png

Οι οπτικοποιήσεις ήταν αναμενόμενες μιας και ξέραμε ότι τα σημεία στα οποία υπάρχουν μη μηδενικά στοιχεια είναι αυτά που φαίνονται από τις μπλέ γραμμές.



snapshot5.png

Απο το snapshot5.png μπορόυμε να δούμε ότι το T2 έχει περισσότερα μη μηδενικά στοιχεία από το T1 το οποίο έχει με τη σειρά του περισσότερα από το T0. Επίσης ισχύει ότι το T0 είναι μικρότερων διαστάσεων από το T1 το οποίο με τη σειρά του είναι μικρότερο από το T2. Από την δεύτερη παρατήρηση μπορούμε να καταλάβουμε το λόγο για τον οποίο τα full μητρώα απαιτούν το χώρο που απαιτούν απο τη μνήμη για την αποθηκευση τους, μιας και όσο μεγαλύτερα ειναι τοσο περισσότερο χώρο χρειάζονται. Απο την άλλη για τα sparse δεν ισχει κάτι τέτοιο μιας και ο χώρος που χρειάζεται εξαρτάται από τα μη μηδενικά στοιχεία προς αποθήκευση, πράγμα που δικαιολογήτε και απο την πρώτη παρατήρηση που κάναμε αν δούμε σαφώς και το χώρο που θέλουν τα αντιστοιχα sparse για την αποθηκευσή τους.

2) Στο σημείο αυτό μας ζητήθηκε να φτιάξουμε συγκεκριμάνα T0, T1 και T2 μητρώα και να υπολογίσουμε το διανυσμα AAAAAAAb με b = rand(n,1) και A τα T0, T1 και T2 κάθε φορά. Ο υπολόγισμός έγινε και προς τα αριστερά και προς τα δεξιά, και για full μητρώα αλλά και για sparse.

Ο κώδικας που τρέξαμε ήταν ο ακόλουθος:

```
%ASKISI 1 ERWTIMA 4o MEROS2
```

```
n=510;
%m: to ligotero simantiko psifio tis dekadikis anaparastasis tou AM mou pou
%einai 3628
m=8;
%dimiourgia mitroou T0
T0=toeplitz([m,-1,zeros(1,n)],[m,0.1,zeros(1,n)]);
%dimiourgia mitroou T1
T1=kron(eye(2,2),T0);
%dimiourgia mitroou T2
J=[1,-1;-1,1];
T2=kron(J,T1);
%dimiourgia tuxaiou dianismatos b
b0=rand(length(T0),1);
b1=rand(length(T1),1);
b2=rand(length(T2),1);

%arxikopoiisei ton dianismaton deksia kai aristera pou prokeitai na filane
%tous xronous tis pros deksia kai aristera praksis me ti kathe grammi tous
%na anaparista to xrono gia kathe ena apo ta dianismata T0, T1 kai T2 gia
%ta opoia egine i metrisi antistoixa.
deksia=zeros(3,2);
aristera=zeros(3,2);

for i=1:1:2
   %xronomerisi tis zitoumenis praksis (me dio tropous) gia ta zitoumena mitroa
   %xronometrisi tis pros ta deksia me dedomeno T0
   c=T0*T0*T0*T0*T0*T0*T0*T0*b0;
   tic;
   for j=1:1:50
      c=T0*T0*T0*T0*T0*T0*T0*T0*b0;
   end
   deksia(1,i)=toc/50;
   %xronometrisi tis pros ta deksia me dedomeno T1
   c=T1*T1*T1*T1*T1*T1*T1*T1*b1;
   tic;
   for j=1:1:20
      c=T1*T1*T1*T1*T1*T1*T1*T1*b1;
   end
   deksia(2,i)=toc/20;
   %xronometrisi tis pros ta deksia me dedomeno T2
   c=T2*T2*T2*T2*T2*T2*T2*T2*b2;
   tic;
   for j=1:1:5
      c=T2*T2*T2*T2*T2*T2*T2*T2*b2;
   end
      deksia(3,i)=toc/5;

   %xronometrisi tis pros ta aristera me dedomeno T0
      c=T0*(T0*(T0*(T0*(T0*(T0*(T0*(T0*b0)))))));
      tic;
   for j=1:1:50
      c=T0*(T0*(T0*(T0*(T0*(T0*(T0*(T0*b0)))))));
   end
   aristera(1,i)=toc/50;
   %xronometrisi tis pros ta aristera me dedomeno T1
   c=T1*(T1*(T1*(T1*(T1*(T1*(T1*(T1*b1)))))));
   tic;
```

```
    for j=1:1:50
        c=T1*(T1*(T1*(T1*(T1*(T1*(T1*(T1*b1)))))));
    end
    aristera(2,i)=toc/50;
    %xronometrisi tis pros ta aristera me dedomeno T2
    c=T2*(T2*(T2*(T2*(T2*(T2*(T2*(T2*b2)))))));
    tic;
    for j=1:1:50
        c=T2*(T2*(T2*(T2*(T2*(T2*(T2*(T2*b2)))))));
    end
    aristera(3,i)=toc/50;

    %xronometrisi tora ton idion prakseon gia ta sparse
    T0=sparse(T0);
    T1=sparse(T1);
    T2=sparse(T2);
end

%ektiposi ton xronon ton pros ta deksia prakseon gia full mitroa
deksia_full=sprintf('%.3f\t',deksia(:,1))
%ektiposi ton xronon ton pros ta deksia prakseon gia sparse mitroa
deksia_sparse=sprintf('%.3f\t',deksia(:,2))
%ektiposi ton xronon ton pros ta aristera prakseon gia full mitroa
aristera_full=sprintf('%.3f\t',aristera(:,1))
%ektiposi ton xronon ton pros ta aristera prakseon gia sparse mitroa
aristera_sparse=sprintf('%.3f\t',aristera(:,2))
```
erotima4meros2.m

Και τα αποτελέσματα που πήραμε φαίνονται στον παρακάτω πίνακα:

	Υπολογισμός προς τα δεξιά	Υπολογισμός προς τα αριστερά
T0	1.394	0.004
T1	10.954	0.023
T2	86.958	0.085
sparse(T0)	0.002	0.000
sparse(T1)	0.005	0.000
sparse(T2)	0.030	0.001

workspace_erotima4meros2.mat/snapshot7.png

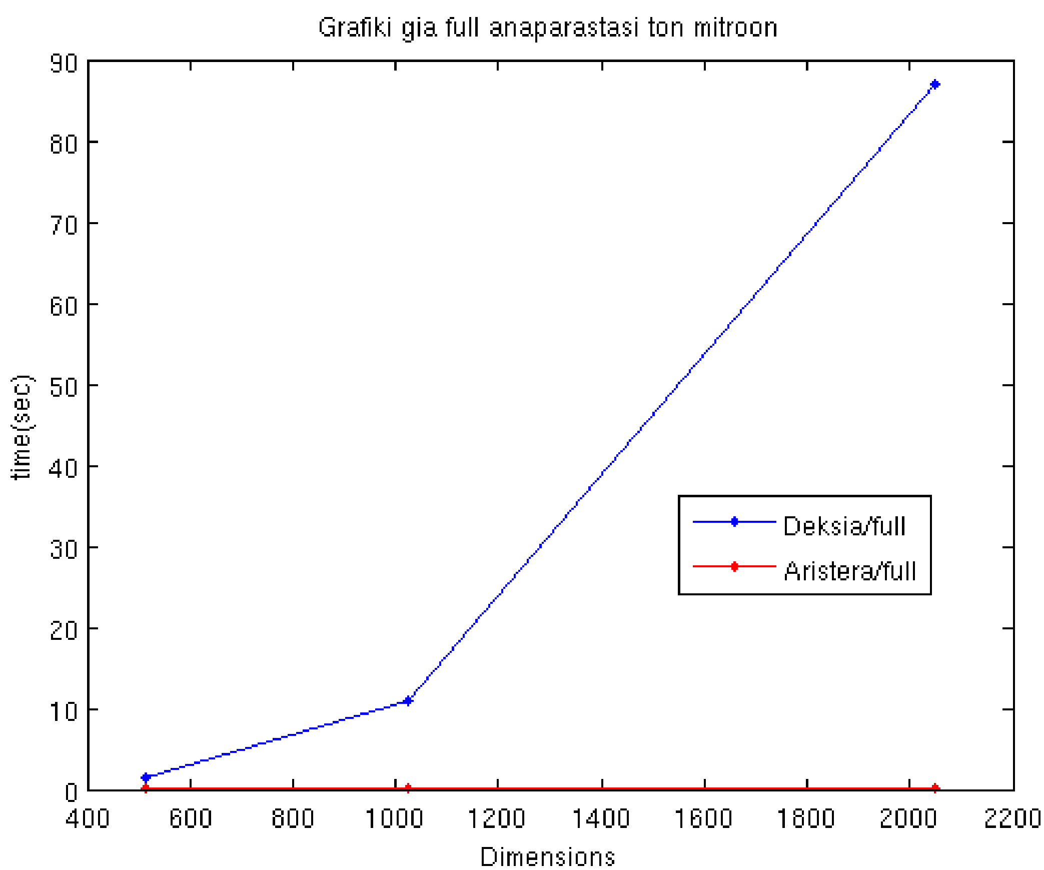
Όπως εύκολα μπορεί να παρατηρήσει κανείς οι πράξεις με τα sparse μητρώα είναι είναι “υπερβολικά” πιο γρήγορες. Επίσης όσο μεγαλύτερο είναι το μητρώο και όσο πιο πολλά μη μηδενικά στοιχεία έχει τόσο πιο πολύ αργεί να γίνει η πράξη. Ακόμα ευκολα φαίνεται ότι ο υπολογισμός του διανύσματος προς τα αριστερά ειναι πιο γρήγορος από τον ίδιο στα δεξιά πράγμα λογικό μιας και στον αριστερό γίνονται 8 πράξεις μητρώου nxn με nx1, ενώ στον δεξιά 7 nxn μητρώα πολλαπλασιαζονται με 7 nxn μητρώα και τέλος ένα nxn με ένα nx1. Με άλλα λόγια στην προς δεξιά απαιτούνται περισσότερες πράξεις.

Για την εμφάνισει των παραπάνω αποτελεσμάτων σε γραφικές παραστάσεις τρέξαμε τον ακόλουθο κώδικα:

```
%ASKISI1 EROTIMA4 MEROS3 - ANAPARASTASI APOTELESMATON SE GRAFIKI PARASTASI
```

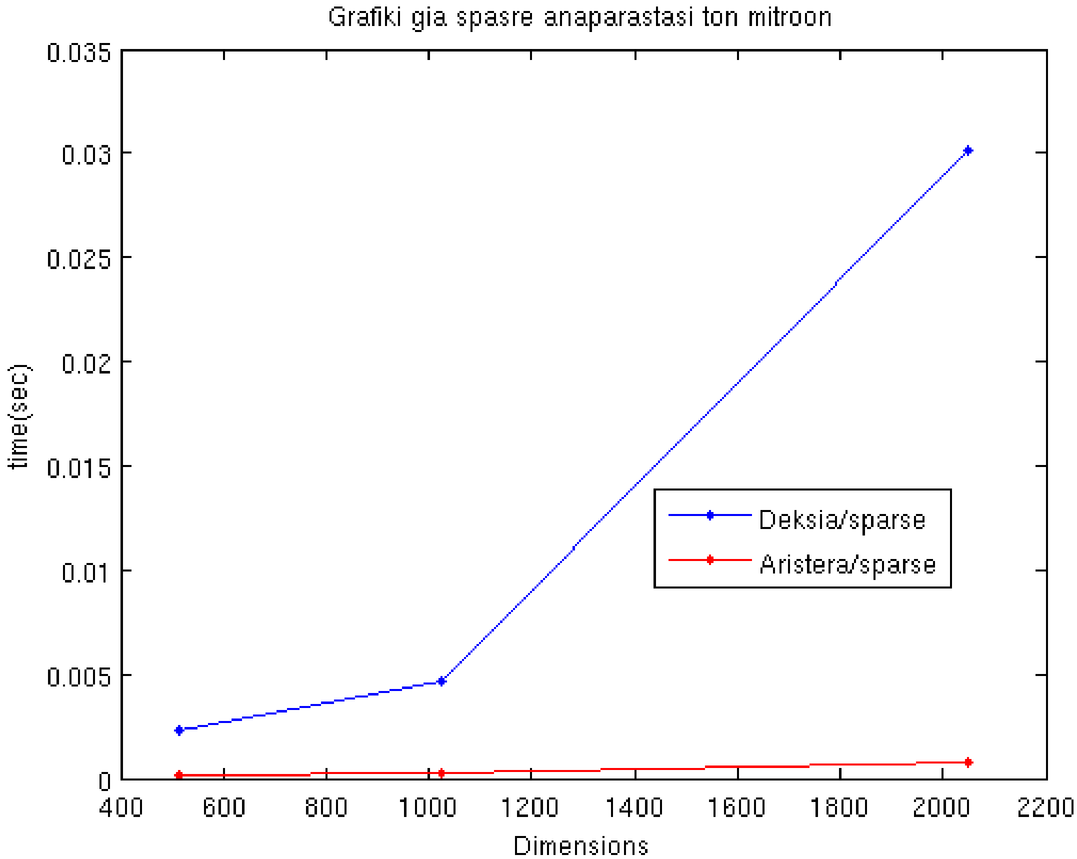
```
%arxikopoioume enan pinaka me tis diastaseis ton mitroon
x=[length(T0);length(T1);length(T2)];

%grafiki parastasi gia ta full mitroa
plot(x, deksia(:,1), '.-');
% kanoume hold gia  na emfanistoun se koini grafiki parastasi
hold on;
plot(x, aristera(:,1), 'r.-')
legend('Deksia/full', 'Aristera/full')
hold;


%grafiki parastasi gia ta sparse mitroa se allo figure
figure
plot(x, deksia(:,2), '.-');
% kanoume hold gia  na emfanistoun se koini grafiki parastasi
hold on;
plot(x, aristera(:,2), 'r.-')
legend('Deksia/sparse', 'Aristera/sparse')
hold;
```
erotima4meros3.m

Και πήραμε τα ακόλουθα αποτελέσματα:



erotima4_4full.png

erotima4_4sparse.png



Και στις δύο περιπτώσεις όπως άλλωστε αναφεραμε παραπάνω ο χρόνος αυξάνεται με την αύξηση των διαστάσεων (αυξουσα γραφική παράσταση) και πάντα ο προς αριστερά υπολογισμός είναι εκπληκτικά ταχύτερος απο τον προς τα δεξιά για το συγκεκριμένο διάνυσμα.