

Επιστημονικός Υπολογισμός

Project 1

Λουδάρος Ιωάννης (1067400)



Μπορείτε να δείτε την τελευταία έκδοση του Project [εδώ](#) ή σκανάροντας τον κωδικό QR που βρίσκεται στην επικεφαλίδα.

Περιγραφή Αναφοράς

Παρακάτω παραθέτω τις απαντήσεις μου στην “Εργαστηριακή Άσκηση (Μέρος 1)” του μαθήματος “Επιστημονικός Υπολογισμός” καθώς και σχόλια τα οποία προέκυψαν κατά την εκπόνηση του.

Περιεχόμενα

1. Εισαγωγικά	2
1.1.Στοιχεία Υπολογιστικού Συστήματος	2
Ερωτήματα (i) και (ii)	2
Ερώτημα (iii)	3
2. Αραιές Αναπαραστάσεις και Κατασκευές Μητρώων	3
2.1.Η Συνάρτηση: sp_mat2latex	3
2.2.Η Συνάρτηση: blkToeplitzTrid	4
2.3.Η Συνάρτηση: sp_mx2bccs	5

Απαντήσεις

1. Εισαγωγικά

1.1. Στοιχεία Υπολογιστικού Συστήματος

Ερωτήματα (i) και (ii)

Χαρακτηριστικό	Απάντηση
Έναρξη/Λήξη Εργασίας	10/01/22 - 16/01/22
Model	Προσωπικό Λάπτοπ MacBook Pro
O/S	macOS Big Sur 11.6.1 — System Report
Processor Name	Dual-Core Intel Core i5 — System Report
Processor Speed	2,4 GHz — System Report
Number of Processors	1 — System Report
Total # of Cores	2 — System Report
Total # of Threads	4 — System Report
FMA Instruction	yes — running: sysctl -n hw.optional.fma
L1 Cache	32 KB (per Core) — running: sysctl -a grep hw.l1
L2 Cache	256 KB (per Core) — System Report
L3 Cache	3 MB — System Report
Gflop/s	4 GFlop/s — PassMark Performance Test
Memory	8 GB — System Report
Memory Bandwidth	9978 / 6477 (Read /Write) MB/s — PassMark Performance Test
MATLAB Version	MATLAB Web (Equivalent to MATLAB 9.11.0.1811744 (R2021b) Update 1) — bench
BLAS	Intel(R) Math Kernel Library Version 2019.0.3 Product Build 20190125 for Intel(R) 64 architecture applications, CNR branch AVX512 — Μέσω του version('blas')
LAPACK	Intel(R) Math Kernel Library Version 2019.0.3 Product Build 20190125 for Intel(R) 64 architecture applications, CNR branch AVX512, supporting Linear Algebra PACKage (LAPACK 3.7.0) — Μέσω του version('lapack')

Προσοχή, οι τρόποι εύρεσης των ζητούμενων πληροφοριών αναγράφονται στο πεδίο της απάντησης μετά το χαρακτηριστικό “—”.

Ερώτημα (iii)

Όπως μας γίνεται αντιληπτό, το MATLAB Web δεν χρησιμοποιεί τους πόρους του προσωπικού μου λάπτοπ, όπως αναφέρονται παραπάνω, αλλά εκτελείται σε μια εικονική μηχανή της Mathworks.

Computer Type	LU	FFT	ODE	Sparse	2-D	3-D
Windows 10, AMD Ryzen Threadripper(TM) 3970x @ 3.50 GHz	0.2008	0.1881	0.3469	0.4396	0.2029	0.1117
Debian 10(R), AMD Ryzen Threadripper 2950x @ 3.50 GHz	0.3122	0.2377	0.3219	0.5047	0.5941	0.1631
iMac, macOS 11.2.3, Intel Core i9 @3.6 GHz	0.3278	0.2648	0.2674	0.2763	0.6898	0.3946
Windows 10, Intel Xeon(R) W-2133 @ 3.60 GHz	0.4154	0.2991	0.4348	0.4574	0.3167	0.2184
Windows 10, Intel Xeon CPU E5-1650 v3 @ 3.50 GHz	0.4614	0.3030	0.4455	0.4433	0.3559	0.2623
Windows 10, AMD Ryzen(TM) 7 1700 @ 3.00 GHz	0.7507	0.5163	0.4884	0.5441	0.3397	0.1849
Windows 10, Intel Core i7-10610 @ 1.8 GHz	0.9218	0.4394	0.3666	0.3844	0.7386	0.6251
This machine	0.5014	0.3309	0.5527	0.6299	2.5464	0.2621
Surface Pro 3, Windows(R) 10, Intel(R) Core(TM) i7-5600U @ 2.6 GHz	1.7475	0.9090	0.6178	0.5711	0.5713	0.3623
MacBook Pro, macOS 10.15.2, Intel Core i5 @ 2.6 GHz	1.6237	0.9786	0.5446	0.6173	2.5214	2.0229

2. Αραιές Αναπαραστάσεις και Κατασκευές Μητρώων

Στον φάκελο “Code” μπορείτε να βρείτε όλα όσα αναφέρονται παρακάτω και να τα τρέξετε τοπικά στην δική σας εγκατάσταση. Το αρχείο “Project.mlx” προσφέρει μια ολική εικόνα των απαντήσεων ώστε να διευκολύνει τον διορθωτή, ενώ για το κάθε ερώτημα μπορείτε να βρείτε το αντίστοιχο αρχείο .m που περιλαμβάνει την συνάρτηση που ζητείται μαζί με εκτενή σχολιασμό ώστε να γίνεται κατανοητή με την πρώτη ανάγνωση.

2.1. Η Συνάρτηση: `sp_mat2latex`

Η συνάρτηση `sp_mat2latex` δέχεται ως όρισμα ένα αραιό μητρώο και παράγει τον κατάλληλο κώδικα LaTeX ώστε να το παρουσιάσει σε μορφή ‘`csr`’ ή ‘`csc`’ ανάλογα με την επιλογή του χρήστη. Ο κώδικας αυτός εκτυπώνεται στην οθόνη του χρήστη και αποθηκεύεται στο αρχείο “out.tex”.

Μπορείτε να δείτε ένα παράδειγμα εκτέλεσης της `sp_mat2latex` παρακάτω:

```
A = 4x4
    0.3984    0.1895    0.8423    0
         0    0.5458         0    0
    0.9416    0.4122    0.1788    0
         0         0    0.7134    0
```

```
[val,row_ptr,col_idx] = sp_mat2latex(A,'csr');

$$ val = \begin{tabular}{|1|1|1|1|1|1|1|1|}\hline
0.3984 & 0.1895 & 0.8423 & 0.5458 & 0.9416 & 0.4122 & 0.1788 & 0.7134\\ \hline
\end{tabular}$$
$$ IA = \begin{tabular}{|1|1|1|1|1|1|1|1|}\hline
1 & 62 & 63 & 62 & 61 & 62 & 63 & 63\\ \hline
\end{tabular}$$
$$ JA = \begin{tabular}{|1|1|1|1|1|1|1|1|}\hline
1 & 64 & 65 & 68 & 69\\ \hline
\end{tabular}$$
```

2.2. Η Συνάρτηση: blkToeplitzTrid

Τώρα θα εξετάσουμε την συνάρτηση `blkToeplitzTrid`, η οποία δέχεται ως ορίσματα έναν ακέραιο αριθμό n και τρία τετραγωνικά μητρώα ίδιου μεγέθους (A , B , C). Επιστρέφει σε αραιή μορφή το μπλοκ Toeplitz τριδιαγώνιο μητρώο:

$$\begin{pmatrix} A & C & 0 & \dots & 0 \\ B & A & C & \ddots & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & B & A & C \\ 0 & \dots & 0 & B & A \end{pmatrix}$$

Εδώ μπορείτε να δείτε παράδειγμα εκτέλεσης για $n=3$:

<pre>A = 3x3 0.2259 0.4357 0.4302 0.1707 0.3111 0.1848 0.2277 0.9234 0.9049 B = 3x3 0.9797 0.2581 0.2622 0.4389 0.4087 0.6028 0.1111 0.5949 0.7112 C = 3x3 0.2217 0.3188 0.0855 0.1174 0.4242 0.2625 0.2967 0.5079 0.8010</pre>	<pre>R = blkToeplitzTrid(n,B,A,C); full(R) ans = 9x9 0.2259 0.4357 0.4302 0.2217 0.3188 0.0855 0 0 0 0.1707 0.3111 0.1848 0.1174 0.4242 0.2625 0 0 0 0.2277 0.9234 0.9049 0.2967 0.5079 0.8010 0 0 0 0.9797 0.2581 0.2622 0.2259 0.4357 0.4302 0.2217 0.3188 0.0855 0.4389 0.4087 0.6028 0.1707 0.3111 0.1848 0.1174 0.4242 0.2625 0.1111 0.5949 0.7112 0.2277 0.9234 0.9049 0.2967 0.5079 0.8010 0 0 0 0.9797 0.2581 0.2622 0.2259 0.4357 0.4302 0 0 0 0.4389 0.4087 0.6028 0.1707 0.3111 0.1848 0 0 0 0.1111 0.5949 0.7112 0.2277 0.9234 0.9049</pre>
---	--

```
1 %Testing the Matrices
2 if( ~(size(A,1)==size(A,2)) || ~(size(B,1)==size(B,2)) || ~(size(C,1)==size(C,2)) )
3     error("There are non-square matrices in the arguments.");
4
5 else
6     if ( ~( (size(A,1)==size(B,1)) && (size(B,1)==size(C,1)) ) )
7         error("The matrices are not of the same size.")
8     end
```

Για την υλοποίηση της συνάρτησης χρειάστηκαν 7 γραμμές για να εξετάσουμε αν τα δεδομένα είναι στην κατάλληλη μορφή.

Και 13 γραμμές για την δημιουργία του ζητούμενου τριδιαγώνιου μητρώου.

```
1 m=size(A,1); % Κρατάμε το μήκος (ή το πλάτος) των μητρώων
2 Z=zeros(m); % Προετοιμάζουμε ένα μητρώο στο μέγεθος των υπολοίπων με μηδενικά
3 T={Z,A,B,C}; % Οργανώνουμε τα μητρώα μας,
4 % τώρα μπορούμε να τους αναφερόμαστε εύκολα μέσω δεικτών στο T
5
6 % Προετοιμάζουμε την κλήση της toeplitz
7 col = ones(1,n);
8 col(1)=2;
9 col(2)=3;
10 row = ones(1,n);
11 row(1)=2;
12 row(2)=4;
13 Indexes = toeplitz(col,row); % Δημιουργήσαμε ένα μητρώο που σε κάθε του θέση
14 % είναι ο δείκτης που αντιστοιχεί σε ένα μητρώο του T
15
16 R = cell2mat(T(Indexes));
17 R = sparse(R);
18 end
```

2.3.Η Συνάρτηση: `sp_mx2bccs`

Η `sp_mx2bccs` μετατρέπει κανονικά μητρώα σε BCCS (block compressed column storage). Λαμβάνει ως είσοδο το μητρώο προς μετατροπή και έναν ακέραιο `nb` ο οποίος καθορίζει το μέγεθος των blocks.

A2 = 6x6

0	0	0	0	4	2
0	0	1	1	2	2
6	3	0	0	0	0
3	0	0	0	0	0
2	1	0	0	2	1
1	1	0	0	1	1

```
[val,brow_idx,bcol_ptr] = sp_mx2bccs(A2,2)
```

val = 1x20

1 1 1 0 1 0 1 4 2 2 2 2 1 1 1

Columns: 6:20

brow_idx = 1x5

2 3 1 1 3

bcol_ptr = 1x4

1 3 4 6