COMP-40730 HPC

Report for Assignment 2

Author:	Paula Dwan
Due date:	July-2014
Lecturer:	Alexey Lastovetsky
Subject:	COMP-40730 High Performance Computing
College:	University College Dublin

Contents

EXERCISE

Write a parallel Pthreads program computing the norm of the product of two n×n dense matrices on a p-processor SMP so that

- p threads are involved in the parallel computations.
- The 1-dimensional parallel algorithm of matrix multiplication is employed:
  - one of matrices is partitioned in one dimension into p equal slices
  - there is one-to-one mapping between the partitions and threads
  - each thread is responsible for computation of the corresponding slice of the resulting matrix
- Computation of the norm of the resulting matrix employs the mutex synchronization mechanism.

You can use BLAS or ATLAS for local computations.

Experiment with the program and build:

- The dependence of the execution time of the program on the matrix size n.
- The speedup over a serial counterpart of the program.

Explain the results.

Variants of the assignment:

1. Granularity of the program:
   (a) Two successive steps:
      i. Parallel matrix multiplication
      ii. Parallel computation of the norm of the resulting matrix
   (b) One-step algorithm. No intermediate resulting matrix.

2. Partitioning scheme:
   (a) Left matrix is horizontally partitioned
   (b) Right matrix is vertically partitioned

3. Matrix norm to be computed:
   (a) The maximum absolute column sum norm (aka one-norm):

$$||A||_1 = \max_{0 \le j < n} \sum_{i=0}^{n-1} |a_{ij}|$$

   (b) The maximum absolute row sum norm (aka infinity-norm):

$$||A||_\infty = \max_{0 \le i < n} \sum_{j=0}^{n-1} |a_{ij}|$$

Overview of Computations Obtained and How

Assignment 2 basically involved (for me) writing two programmes,

- **A2-pthreads-manual.c**
  one which contained the code for manual straight-forward IJK computation and dgemm to calculate matrix |C| and the infinity norm of |C|.
- **A2-pthreads-solo.c**
  second which contained the code for manual straight-forward IJK computation and pThreads to calculate matrix |C| and the infinity norm of |C|.

I then compared the time taken to calculate matrix |C| for each method using [N]x[N] matrices |A| and |B| of different sizes, allocating values for the |A| and |B| using random numbers (1 to 10 inclusive) or the column value + 1 (so that column 1000 would have a value of 1001).

1. **manual straight-forward IJK computation**

   Implementation of a straight forward matrix nxn multiplication.

   *Present in **A2-pthreads-manual.c** and in **A2-pthreads-solo.c***

   Code

   ```
   for (ni=0 ; ni<rows ; ni++)
   {
      for (nj=0 ; nj<cols ; nj++)
      {
         double sum = 0.0 ;
         for (nk=0 ; nk<rows ; nk++)
         {
            sum+= (A[(ni*rows)+nk]) * (B[(nk*rows)+nj]) ;
         }
         C[(ni*rows)+nj] = sum ;
      }
   }
   ```

2. **Dgemm for straight-forward IJK computation**

   I used dgemm compiled for atlas or for cblas.

   The matrix |C| was calculated using cblas by default otherwise the user could decide to re-build and execute using cblas or atlas).

   *Present in **A2-pthreads-manual.c** only*

   Code

   ```
      int ni, nj ;
   // m, n, k : local integers indicating the size of the matrices for
   // rows x columns :: A : m x  k, B : k x n, C: m x n
   // Here, m = n = k = rows = columns = <nx> = <ny> as supplied
      int lm = rows, ln = rows ;
   // la_offset, lb_offset, lc_offset :
   // Leading dimension of matrix A, B or C respectively, or the number of elements
   // between successive rows for row-major storage or columns for column-major
   // storage.
      int la_offset = rows, lb_offset = cols, lc_offset = rows ;
      int ALPHA=1.0 ;
      int BETA=0.0 ;

      cblas_dgemm( CblasRowMajor, CblasNoTrans, CblasNoTrans, lm, ln, ln, ALPHA, \
         A, la_offset, B, lb_offset, BETA, C, lc_offset) ;
   ```

3. **pThreads calculation**

   This was completed using a data structure, creating a new thread to correspond to the each slice. Infinity norm was mutes'ed to ensure only one thread could access it for updates at any one time.

   *Present in **A2-pthreads-solo.c** only*

   Code

Thus results were obtained for each option 1., 2. and 3. using the same source |A| and |B| for both -i (increment) and also using -r (random) .

Multiple implementation of each .c program was enabled using ***./runAssignment2.sh,*** for example :

```
$ ./runAssignment2.sh -1 -i -v
```

This runs the c. program using cblas and manual straight-forward IJK for incremental column values, using predefined matrix sizes for each implementation.

Single implementation is completed using

- ./**A2-pthreads-manual.c.**

  ```
  $ ./A2-pthreads-manual-cblas -i 10 file.txt file.dat
  ```

  This is compiled using *gcc* for *atlas* and *cblas*, as follows :

  ```
  gcc -I/home/cs/khasanov/libs/CBLAS/src A2-pthreads-manual.c -o A2-pthreads-manual-cblas \
       /home/cs/khasanov/libs/cblas_LINUX.a  /usr/lib/libblas.a -lgfortran
  gcc -o A2-pthreads-manual-atlas A2-pthreads-manual.c  \
       -I/home/cs/khasanov/libs/ATLAS/include/ \
       -L/home/cs/khasanov/libs/ATLAS/lib/Linux_UNKNOWNSSE2_4/ -lcblas -latlas -lm -O3
  ```

- ./**A2-pthreads-solo.c.**

  ```
  $ ./A2-pthreads-solo -i 10 2 file-solo.txt file-solo.dat
  ```

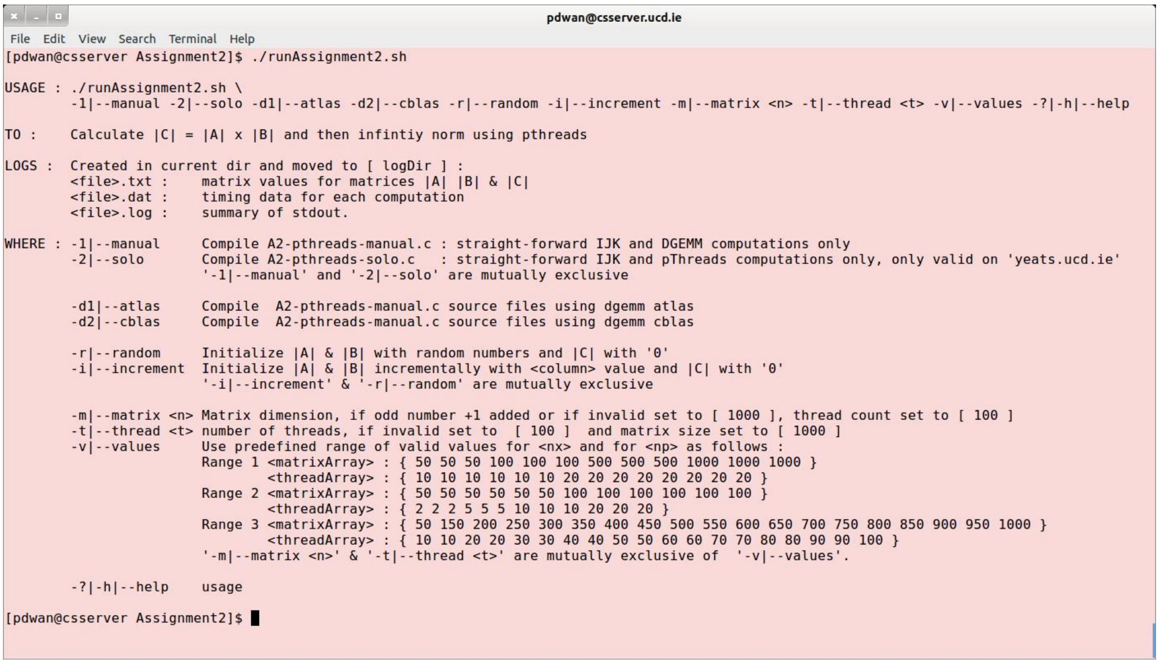
  This is compiled using *gcc* for *pthread*, as follows :

  ```
  gcc -I/home/cs/khasanov/libs/CBLAS/src A2-pthreads-solo.c -o A2-pthreads-solo  \
       -lgfortran -pthread
  ```

Assignment Execution

The compiled .c program ./A2-pthreads-manual-<cblas|atlas> was executed multiple times standalone or using the script *./runAssignment2.sh* to obtain as wide a range of time taken to calculate |C| using each algorithm.

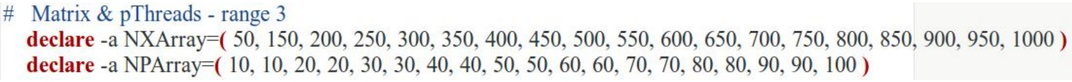
This has multiple options and the syntax and usage follows :



```
[pdwan@csserver Assignment2]$ ./runAssignment2.sh

USAGE : ./runAssignment2.sh \
        -1|--manual -2|--solo -d1|--atlas -d2|--cblas -r|--random -i|--increment -m|--matrix <n> -t|--thread <t> -v|--values -?|-h|--help

TO :    Calculate |C| = |A| x |B| and then infintiy norm using pthreads

LOGS :  Created in current dir and moved to [ logDir ] :
        <file>.txt :    matrix values for matrices |A| |B| & |C|
        <file>.dat :    timing data for each computation
        <file>.log :    summary of stdout.

WHERE : -1|--manual     Compile A2-pthreads-manual.c : straight-forward IJK and DGEMM computations only
        -2|--solo       Compile A2-pthreads-solo.c   : straight-forward IJK and pThreads computations only, only valid on 'yeats.ucd.ie'
                        '-1|--manual' and '-2|--solo' are mutually exclusive

        -d1|--atlas     Compile  A2-pthreads-manual.c source files using dgemm atlas
        -d2|--cblas     Compile  A2-pthreads-manual.c source files using dgemm cblas

        -r|--random     Initialize |A| & |B| with random numbers and |C| with '0'
        -i|--increment  Initialize |A| & |B| incrementally with <column> value and |C| with '0'
                        '-i|--increment' & '-r|--random' are mutually exclusive

        -m|--matrix <n> Matrix dimension, if odd number +1 added or if invalid set to [ 1000 ], thread count set to [ 100 ]
        -t|--thread <t> number of threads, if invalid set to  [ 100 ]  and matrix size set to [ 1000 ]
        -v|--values     Use predefined range of valid values for <nx> and for <np> as follows :
                        Range 1 <matrixArray> : { 50 50 50 100 100 100 500 500 500 1000 1000 1000 }
                                <threadArray> : { 10 10 10 10 10 10 20 20 20 20 20 20 20 20 }
                        Range 2 <matrixArray> : { 50 50 50 50 50 50 100 100 100 100 100 100 }
                                <threadArray> : { 2 2 2 5 5 5 10 10 10 20 20 20 }
                        Range 3 <matrixArray> : { 50 150 200 250 300 350 400 450 500 550 600 650 700 750 800 850 900 950 1000 }
                                <threadArray> : { 10 10 20 20 30 30 40 40 50 50 60 60 70 70 80 80 90 90 100 }
                        '-m|--matrix <n>' & '-t|--thread <t>' are mutually exclusive of  '-v|--values'.

        -?|-h|--help    usage

[pdwan@csserver Assignment2]$ █
```

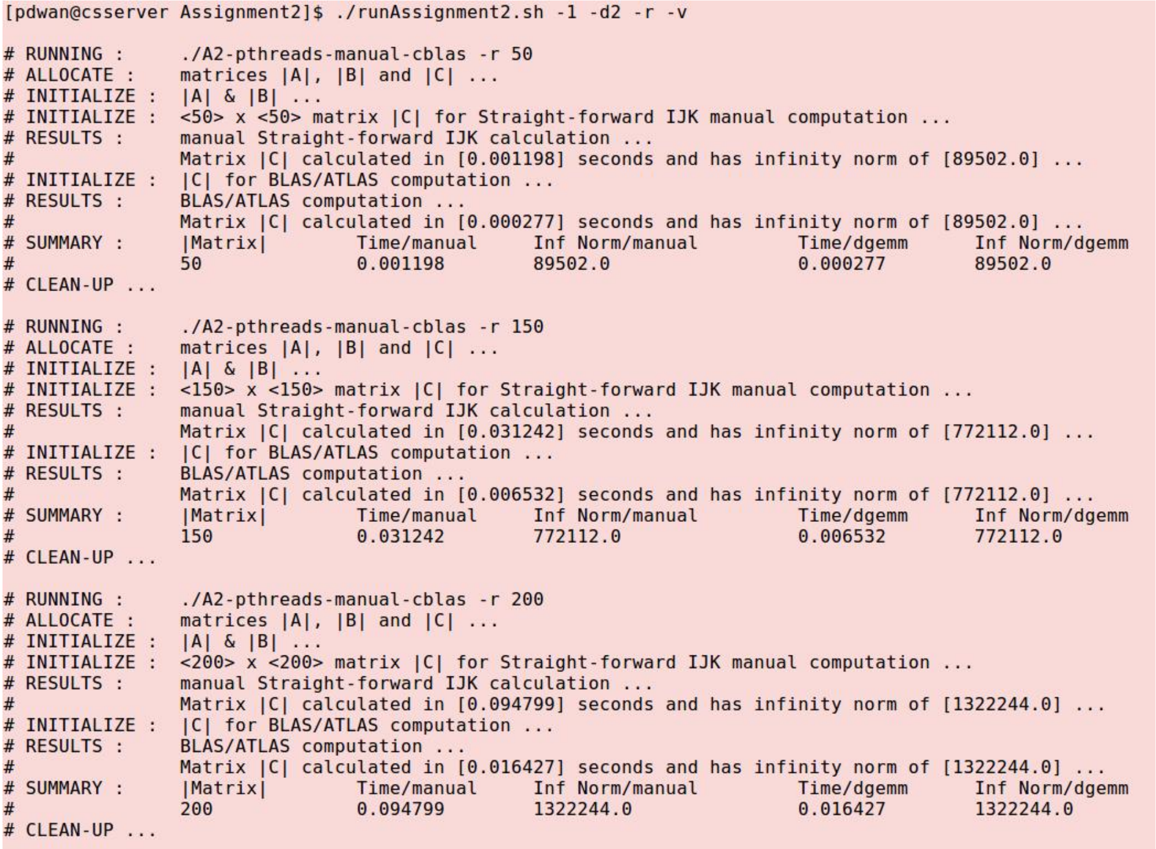
Execute this script in the home directory of Assignment 2.

Sample execution follows for range 3 :



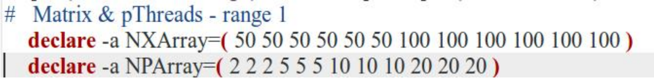
```
#  Matrix & pThreads - range 3
   declare -a NXArray=( 50, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800, 850, 900, 950, 1000 )
   declare -a NPArray=( 10, 10, 20, 20, 30, 30, 40, 40, 50, 50, 60, 60, 70, 70, 80, 80, 90, 90, 100 )
```

```
$ ./runAssignment2.sh -1 -d2 -r -v
```



```
[pdwan@csserver Assignment2]$ ./runAssignment2.sh -1 -d2 -r -v

# RUNNING :     ./A2-pthreads-manual-cblas -r 50
# ALLOCATE :    matrices |A|, |B| and |C| ...
# INITIALIZE :  |A| & |B| ...
# INITIALIZE :  <50> x <50> matrix |C| for Straight-forward IJK manual computation ...
# RESULTS :     manual Straight-forward IJK calculation ...
#               Matrix |C| calculated in [0.001198] seconds and has infinity norm of [89502.0] ...
# INITIALIZE :  |C| for BLAS/ATLAS computation ...
# RESULTS :     BLAS/ATLAS computation ...
#               Matrix |C| calculated in [0.000277] seconds and has infinity norm of [89502.0] ...
# SUMMARY :     |Matrix|        Time/manual     Inf Norm/manual         Time/dgemm      Inf Norm/dgemm
#               50              0.001198        89502.0                 0.000277        89502.0
# CLEAN-UP ...

# RUNNING :     ./A2-pthreads-manual-cblas -r 150
# ALLOCATE :    matrices |A|, |B| and |C| ...
# INITIALIZE :  |A| & |B| ...
# INITIALIZE :  <150> x <150> matrix |C| for Straight-forward IJK manual computation ...
# RESULTS :     manual Straight-forward IJK calculation ...
#               Matrix |C| calculated in [0.031242] seconds and has infinity norm of [772112.0] ...
# INITIALIZE :  |C| for BLAS/ATLAS computation ...
# RESULTS :     BLAS/ATLAS computation ...
#               Matrix |C| calculated in [0.006532] seconds and has infinity norm of [772112.0] ...
# SUMMARY :     |Matrix|        Time/manual     Inf Norm/manual         Time/dgemm      Inf Norm/dgemm
#               150             0.031242        772112.0                0.006532        772112.0
# CLEAN-UP ...

# RUNNING :     ./A2-pthreads-manual-cblas -r 200
# ALLOCATE :    matrices |A|, |B| and |C| ...
# INITIALIZE :  |A| & |B| ...
# INITIALIZE :  <200> x <200> matrix |C| for Straight-forward IJK manual computation ...
# RESULTS :     manual Straight-forward IJK calculation ...
#               Matrix |C| calculated in [0.094799] seconds and has infinity norm of [1322244.0] ...
# INITIALIZE :  |C| for BLAS/ATLAS computation ...
# RESULTS :     BLAS/ATLAS computation ...
#               Matrix |C| calculated in [0.016427] seconds and has infinity norm of [1322244.0] ...
# SUMMARY :     |Matrix|        Time/manual     Inf Norm/manual         Time/dgemm      Inf Norm/dgemm
#               200             0.094799        1322244.0               0.016427        1322244.0
# CLEAN-UP ...
```

Sample execution follows for range 1 for pThreads:



```
#  Matrix & pThreads - range 1
   declare -a NXArray=( 50 50 50 50 50 50 100 100 100 100 100 100 )
   declare -a NPArray=( 2 2 2 5 5 5 10 10 10 20 20 20 )
```

```
$ ./runAssignment2.sh -2 -r -v
```

Please retain the overall directory structure when unzipping.

Note that the script `./runAssignment2.sh` allows two types of implementation

- Multiple iteration
  use the switch <-v | --values>, when a predefined range applies for [N] : matrix size and [T] : Thread size.
- Single iteration
  use the switch <-m | --matrix> [N] where the user specifies value for [N] : matrix size
  and the switch <-t | --thread> [T] where the user specifies value for [T] : Thread size.

RUNNING A2-PTHREADS-MANUAL-<ATLAS | CBLAS> : STANDALONE

The compiled .c program may also be run standalone. Usage and sample execution follows :

RUNNING A2-PTHREADS-SOLO : STANDALONE

The compiled .c program may also be run standalone. Usage and sample execution follows :

Log Files Obtained

Data text files suitable containing the values of the computation used for matrices |A| and |B| and the results stored in |C| are saved in the appropriate log files. File naming convention via the script is :

<data log file name>	`Values-<time>-A2-pthreads-<iteration>.txt`
example:	`Values-20140715.170928-A2-pthreads-0.txt` `file-10txt`

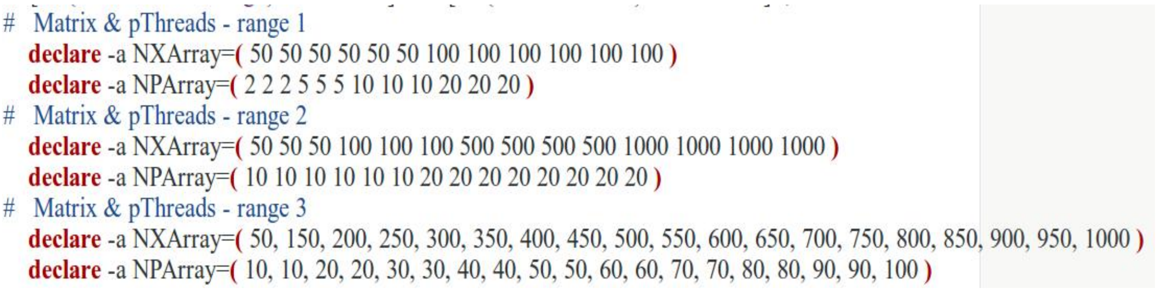
Single iteration also applies where the user enters arbitrary, valid values for matrix size and does not use the scripts and the other required parameters. Each new matrices |A| and |B| and the results in |C| were saved to the data file, thus simple validation using *LibreOffice Calc*.

A summary file containing processing time for each computation (manual & DGEMM and manual & pThreads) is also saved. This is in a format suitable for us with GNUplot.

<timing log file name>	`Data-<time>-A2-pthreads-1D.dat`

example:	`Data-20140715.171337-A2-pthreads-1D.dat`
	`Data-20140715.172124-A2-pthreads-1D.dat`
	`file-10dat`

I did not save a separate .dat file for each run of the script for each algorithm. Instead each .dat file contains the time taken for each matrix size for the preset range of values. **./runAssignment2.sh** may be updated with more if needed but the following are those in use at the moment.



```
#  Matrix & pThreads - range 1
   declare -a NXArray=( 50 50 50 50 50 50 100 100 100 100 100 100 )
   declare -a NPArray=( 2 2 2 5 5 5 10 10 10 20 20 20 )
#  Matrix & pThreads - range 2
   declare -a NXArray=( 50 50 50 100 100 100 500 500 500 500 1000 1000 1000 1000 )
   declare -a NPArray=( 10 10 10 10 10 10 20 20 20 20 20 20 20 20 )
#  Matrix & pThreads - range 3
   declare -a NXArray=( 50, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800, 850, 900, 950, 1000 )
   declare -a NPArray=( 10, 10, 20, 20, 30, 30, 40, 40, 50, 50, 60, 60, 70, 70, 80, 80, 90, 90, 100 )
```

For compilation using the script, a suffix of ***-atlas*** indicates compilation for atlas and a suffix of ***-cblas*** indicates that the c program was compiled via cblas. No suffix indicates compilation using mpicc for pThreads.

Finally a log file containing a listing of each algorithm used for that iteration.

After each run, all .log, .txt, .dat and .bup files are copied to the directory *logDir/.*

If either compiled file is used without the script then .dat and .txt files may be named whatever the user wishes and no .log file applies.

I wished to keep each .c program as clean as possible and so all production setup was completed in the script for each assignment. Thus file creation and validation for each iteration was completed before the .c program was even called. Simple validation of the arguments passed to each .c program is also completed if ran standalone.

I also spot-checked the results as practical. Results spot-check are detailed in Appendix I – Validate Results.

GNUplot Execution

I followed the same structure for each .dat file as produced, an example follows :

Sample .dat file	

If wished, the .txt file contains the matrices |A| and |B| used to calculate |C| and the type of computation applicable and the time taken to complete. The .dat file is just a summary of the matrix sizes (when the later is applicable) as well as time taken for each type of computation.

The contents of each .dat was then presented in graphical format using GNUplot, comparing times taken for manual and for BLAS/ATLAS computations.

Sample GNUplot program execution

```
# To execute, launch GNUplot and run :
# gnuplot> load <filename.gp>
# making sure that the data file name used is updated if needed.
# -----------------------------------------------------------------------
# Paula Dwan : Assignment 2
reset
set size 1,1
set grid
set title 'pThreads : Matrix size -v- Time taken'
set ylabel 'Time taken / seconds'
set xlabel 'Matrix size'
#
set xrange [0:1000]
set yrange [0:25]
set xtics (0,100,200,300,400,500,600,700,800,900,1000)
set ytics (0,5,10,15,20,25)
set origin 0,0
set key outside
#
plot 'logDir/Data-pthreads.dat' u 1:2 t 'manual' w l lw 0.8 lc rgb 'blue',
'logDir/Data-pthreads.dat' u 1:4 t 'dgemm' w l lw 0.8 lc rgb 'red'
#
pause -1
```

Thankfully for Linux (Ubuntu) – I could install and run GNUplot locally.

Screen shots of each were taken and added to the section Summary Results.

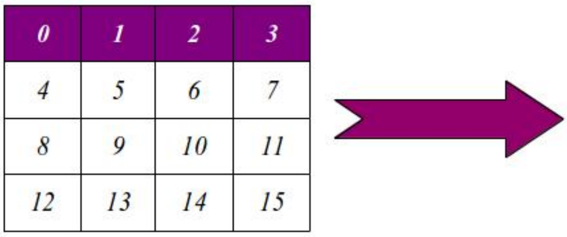
Matrix Sizes Evaluated

When applying the three options I used matrices |A| and |B| of varying sizes from [10x10] to [1000x1000]. The values in each matrix were dependent on the switch

- **-r**
  random from 1 to 10, so each cell regardless of matrix size had a value of 1 to 10 inclusive. This reduced computation time based on large cell values.
- **-i**
  increment based on column index + 1, so all cells in column 1000 had a value of 1001. This could increase computation time of larger matrices as the cell would have comparatively larger values also.

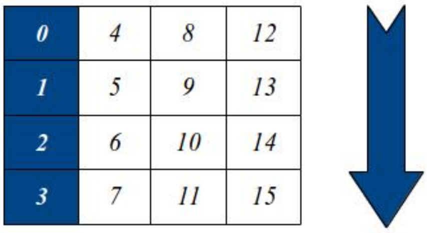
Row-Major as applied

For the straight-forward algorithm of $|\mathbf{C}|_{[ij]} \mathrel{+}= |\mathbf{A}|_{[ik]} * |\mathbf{B}|_{[kj]}$ uses row major as follows (e.g.: [4x4] matrix:



0	*1*	*2*	*3*
4	*5*	*6*	*7*
8	*9*	*10*	*11*
12	*13*	*14*	*15*

Whereas column-major is as follows :



0	*4*	*8*	*12*
1	*5*	*9*	*13*
2	*6*	*10*	*14*
3	*7*	*11*	*15*

Summary Results

Build/plot:	• The dependence of the execution time of the program on the matrix size n. • The speedup over a serial counterpart of the program.
Variant :	• *One-step algorithm. No intermediate resulting matrix.* • *Left matrix is horizontally partitioned* • *The maximum absolute row sum norm (aka infinity-norm):* $\|\|A\|\|_\infty = \max_{0 \le i < n} \sum_{j=0}^{n-1} \|a_{ij}\|$
Infinity norm	*Sum the absolute values along each row and then take the largest value as the answer.* *Example: A = | 1 −7 |* *| −2 −3 |* *then matrix norm of A = max (1 + |-7|, |-2| + |-3|) = max (8, 5) = __**8**__*

GNUPlot Graphs – matrix range 50 → 1000 : <increment>

$ ./runAssignment2.sh -i -v : manual – sample run

Data

Graph

$ ./runAssignment2.sh -i -v : solo | pThreads – sample run

Data

Graph

GNUPlot Graphs – matrix range 50 → 100 : <increment>

$ ./runAssignment2.sh -i -v : manual – sample run

Data

Graph

$ ./runAssignment2.sh -i -v : solo | pThreads – sample run

Data

Graph

GNUPlot Graphs – Matrix Range 50 → 1000 : <random 1 → 10>

$ ./runAssignment2.sh -r -v : manual – sample run

Data

Graph

$ ./runAssignment2.sh -r -v : solo | pThreads – sample run

Data

Graph

CONCLUSIONS

Manual straight-forward IJK computation seems to be consistent in time taken for one or multiple processors with a slight increase when using multiple processors. However this could be perceived to be due to the fact that the cell values were not fully incremental (e.g.: matrix column ha1000 having value if 1001, but instead based on the segment value also → [N] / [P]). This reduced the cell value somewhat but the time taken overall was consistent averaging about 18 to 19 seconds for multiple processors (A2-pthreads-solo.c) and 20 to 22 seconds for single processors (A2-pthreads-manual.c). Dgemm (cblas only) was computed on single processor only in order to complete the results obtained for large and smaller matrix sizes with the time taken for straight-forward IJK computations. It seemed to remain reasonably consistent for smaller matrices of time taken for dgemm : manual of 1 : 5 and larger matrices of 1 : 4.

For matrices of [100] x [100], pThreads was slower than a manual straight-forward IJK computation. However, not surprisingly for large matrices of [1000] x [1000], pThreads was usually quicker to compute results by a ratio of pThreads : manual of 1 : 2.5. This was the case for both random and incremental cell value initialization for |A| and |B|.

I was restricted to a maximum of five processor on csicluster (csicluster02 still appears to be inoperable). It would be interesting to calculate the results for the large matrices using more processors and confirm if the trend continues, whereby computation time is reduced over that on a single processor : csserver. Or if the amalgamation of the results over a larger number of processors for a similarly large matrix would negate the speed of computing the individual segments.

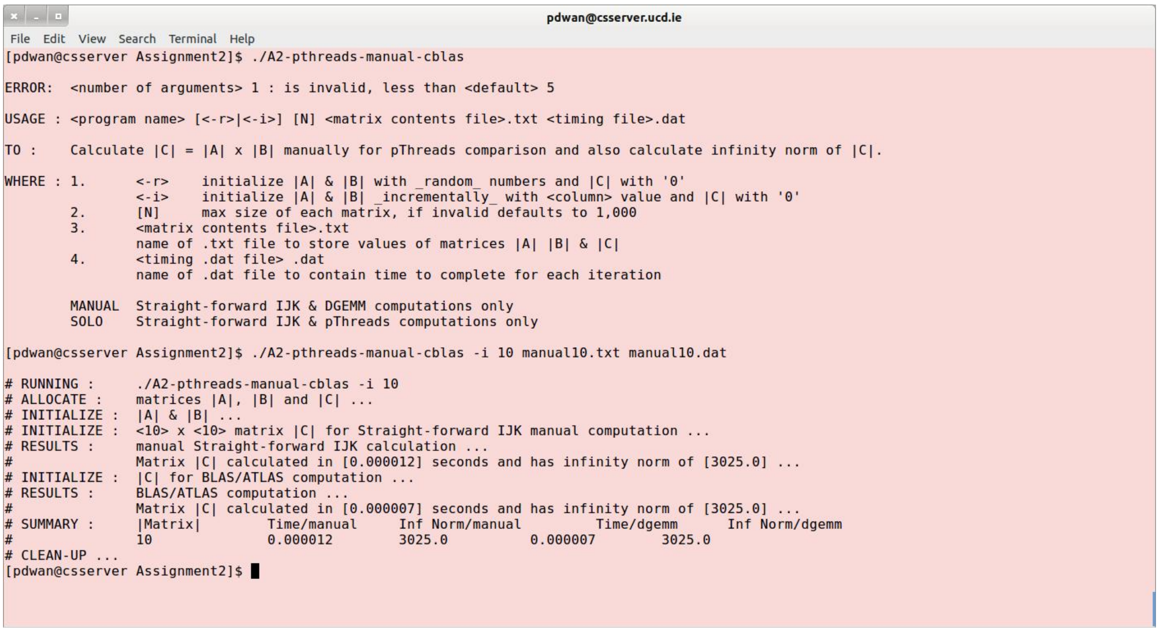
APPENDICES

APPENDIX I – VALIDATE RESULTS

Spot check only using 10x10 matrices, initializing matrices |A| and |B| using successive column values.

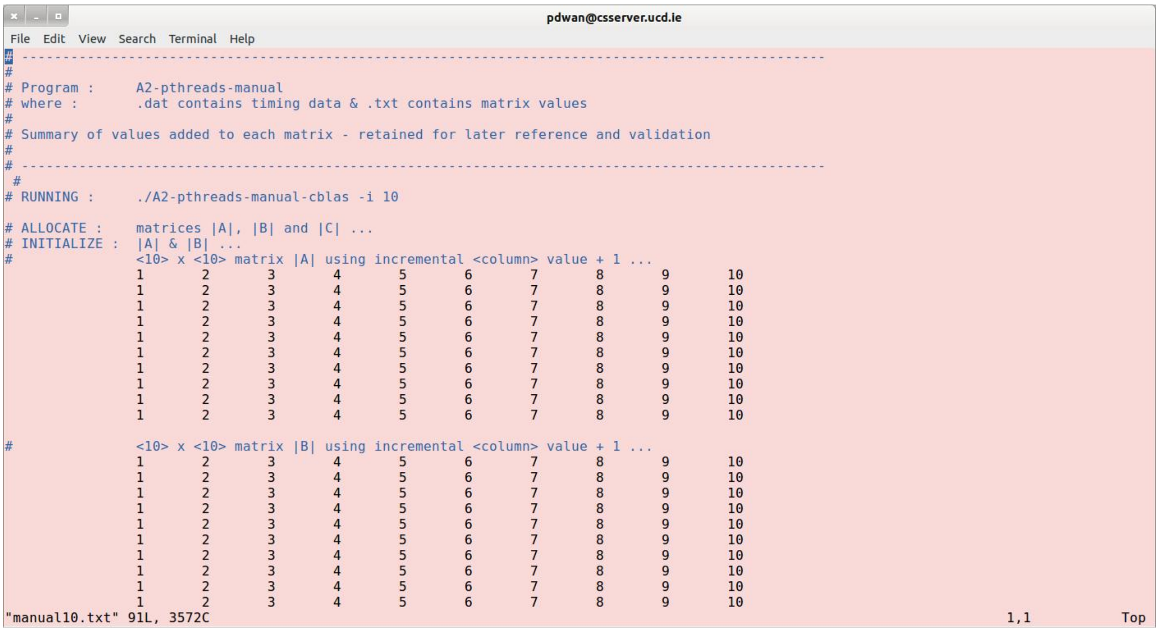
A2-pthreads-manual-cblas

Executed using :

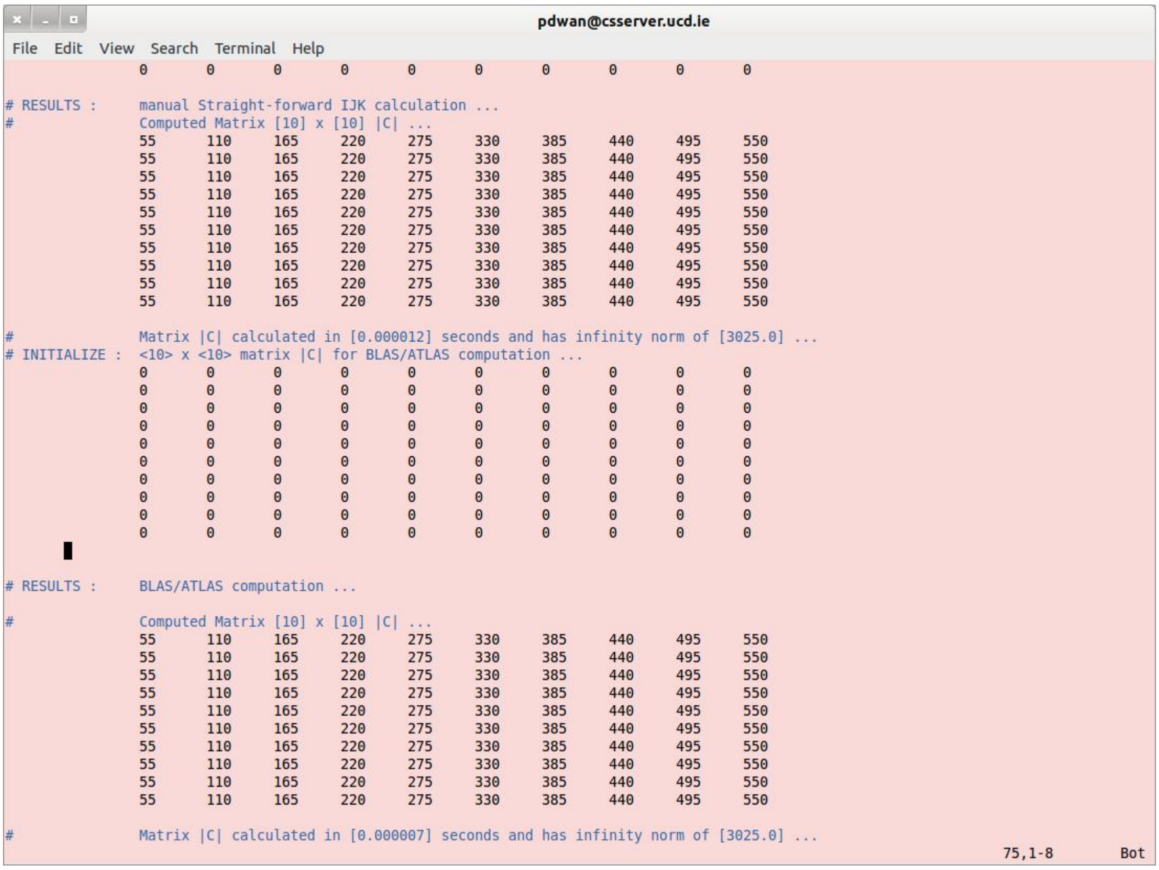


```
[pdwan@csserver Assignment2]$ ./A2-pthreads-manual-cblas

ERROR:  <number of arguments> 1 : is invalid, less than <default> 5

USAGE : <program name> [<-r>|<-i>] [N] <matrix contents file>.txt <timing file>.dat

TO :    Calculate |C| = |A| x |B| manually for pThreads comparison and also calculate infinity norm of |C|.

WHERE : 1.      <-r>    initialize |A| & |B| with _random_ numbers and |C| with '0'
                <-i>    initialize |A| & |B| _incrementally_ with <column> value and |C| with '0'
        2.      [N]     max size of each matrix, if invalid defaults to 1,000
        3.      <matrix contents file>.txt
                name of .txt file to store values of matrices |A| |B| & |C|
        4.      <timing .dat file> .dat
                name of .dat file to contain time to complete for each iteration

        MANUAL  Straight-forward IJK & DGEMM computations only
        SOLO    Straight-forward IJK & pThreads computations only

[pdwan@csserver Assignment2]$ ./A2-pthreads-manual-cblas -i 10 manual10.txt manual10.dat

# RUNNING :     ./A2-pthreads-manual-cblas -i 10
# ALLOCATE :    matrices |A|, |B| and |C| ...
# INITIALIZE :  |A| & |B| ...
# INITIALIZE :  <10> x <10> matrix |C| for Straight-forward IJK manual computation ...
# RESULTS :     manual Straight-forward IJK calculation ...
#               Matrix |C| calculated in [0.000012] seconds and has infinity norm of [3025.0] ...
# INITIALIZE :  |C| for BLAS/ATLAS computation ...
# RESULTS :     BLAS/ATLAS computation ...
#               Matrix |C| calculated in [0.000007] seconds and has infinity norm of [3025.0] ...
# SUMMARY :     |Matrix|        Time/manual     Inf Norm/manual         Time/dgemm      Inf Norm/dgemm
#               10              0.000012        3025.0          0.000007        3025.0
# CLEAN-UP ...
[pdwan@csserver Assignment2]$ █
```

Resulting sample Matrix .txt file contains :

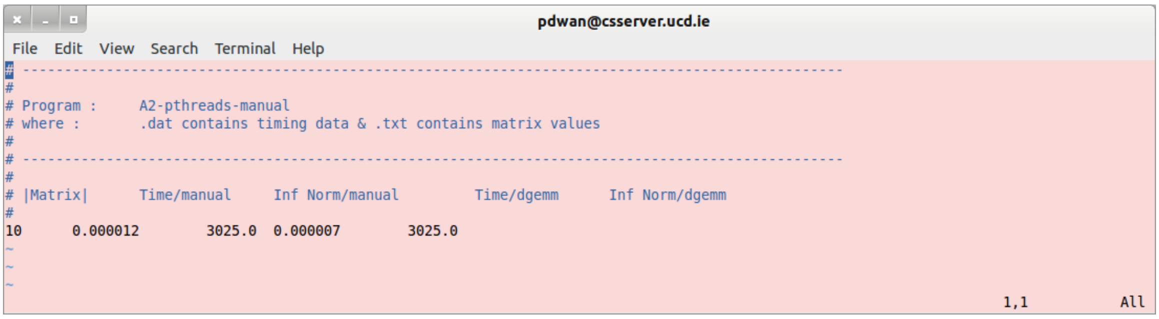


```
# -------------------------------------------------------------------------------------------------
#
# Program :     A2-pthreads-manual
# where :       .dat contains timing data & .txt contains matrix values
#
# Summary of values added to each matrix - retained for later reference and validation
#
# -------------------------------------------------------------------------------------------------
 #
# RUNNING :     ./A2-pthreads-manual-cblas -i 10

# ALLOCATE :    matrices |A|, |B| and |C| ...
# INITIALIZE :  |A| & |B| ...
#               <10> x <10> matrix |A| using incremental <column> value + 1 ...
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10

#               <10> x <10> matrix |B| using incremental <column> value + 1 ...
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
"manual10.txt" 91L, 3572C                                                                                        1,1           Top
```



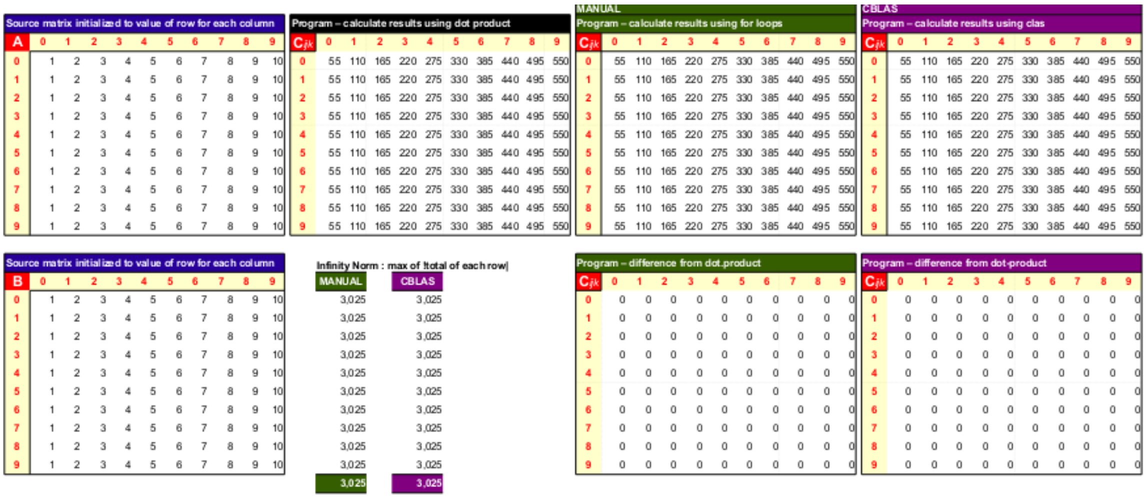
```
                0       0       0       0       0       0       0       0       0       0

# RESULTS :     manual Straight-forward IJK calculation ...
#               Computed Matrix [10] x [10] |C| ...
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550

#               Matrix |C| calculated in [0.000012] seconds and has infinity norm of [3025.0] ...
# INITIALIZE :  <10> x <10> matrix |C| for BLAS/ATLAS computation ...
                0       0       0       0       0       0       0       0       0       0
                0       0       0       0       0       0       0       0       0       0
                0       0       0       0       0       0       0       0       0       0
                0       0       0       0       0       0       0       0       0       0
                0       0       0       0       0       0       0       0       0       0
                0       0       0       0       0       0       0       0       0       0
                0       0       0       0       0       0       0       0       0       0
                0       0       0       0       0       0       0       0       0       0
                0       0       0       0       0       0       0       0       0       0
                0       0       0       0       0       0       0       0       0       0


# RESULTS :     BLAS/ATLAS computation ...

#               Computed Matrix [10] x [10] |C| ...
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550
                55      110     165     220     275     330     385     440     495     550

#               Matrix |C| calculated in [0.000007] seconds and has infinity norm of [3025.0] ...
                                                                                      75,1-8        Bot
```

Resulting sample summary timing data file contains :



```
# -----------------------------------------------------------------------------------------
#
# Program :     A2-pthreads-manual
# where :       .dat contains timing data & .txt contains matrix values
#
# -----------------------------------------------------------------------------------------
#
# |Matrix|      Time/manual     Inf Norm/manual         Time/dgemm      Inf Norm/dgemm
#
10      0.000012        3025.0  0.000007        3025.0
~
~
~
                                                                                      1,1           All
```

Validating results for manual computation gives :



Source matrix initialized to value of row for each column

A	0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	1	2	3	4	5	6	7	8	9	10
3	1	2	3	4	5	6	7	8	9	10
4	1	2	3	4	5	6	7	8	9	10
5	1	2	3	4	5	6	7	8	9	10
6	1	2	3	4	5	6	7	8	9	10
7	1	2	3	4	5	6	7	8	9	10
8	1	2	3	4	5	6	7	8	9	10
9	1	2	3	4	5	6	7	8	9	10

Program – calculate results using dot product

C_{jk}	0	1	2	3	4	5	6	7	8	9
0	55	110	165	220	275	330	385	440	495	550
1	55	110	165	220	275	330	385	440	495	550
2	55	110	165	220	275	330	385	440	495	550
3	55	110	165	220	275	330	385	440	495	550
4	55	110	165	220	275	330	385	440	495	550
5	55	110	165	220	275	330	385	440	495	550
6	55	110	165	220	275	330	385	440	495	550
7	55	110	165	220	275	330	385	440	495	550
8	55	110	165	220	275	330	385	440	495	550
9	55	110	165	220	275	330	385	440	495	550

MANUAL

Program – calculate results using for loops

C_{jk}	0	1	2	3	4	5	6	7	8	9
0	55	110	165	220	275	330	385	440	495	550
1	55	110	165	220	275	330	385	440	495	550
2	55	110	165	220	275	330	385	440	495	550
3	55	110	165	220	275	330	385	440	495	550
4	55	110	165	220	275	330	385	440	495	550
5	55	110	165	220	275	330	385	440	495	550
6	55	110	165	220	275	330	385	440	495	550
7	55	110	165	220	275	330	385	440	495	550
8	55	110	165	220	275	330	385	440	495	550
9	55	110	165	220	275	330	385	440	495	550

CBLAS

Program – calculate results using clas

C_{jk}	0	1	2	3	4	5	6	7	8	9
0	55	110	165	220	275	330	385	440	495	550
1	55	110	165	220	275	330	385	440	495	550
2	55	110	165	220	275	330	385	440	495	550
3	55	110	165	220	275	330	385	440	495	550
4	55	110	165	220	275	330	385	440	495	550
5	55	110	165	220	275	330	385	440	495	550
6	55	110	165	220	275	330	385	440	495	550
7	55	110	165	220	275	330	385	440	495	550
8	55	110	165	220	275	330	385	440	495	550
9	55	110	165	220	275	330	385	440	495	550

Source matrix initialized to value of row for each column

B	0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	1	2	3	4	5	6	7	8	9	10
3	1	2	3	4	5	6	7	8	9	10
4	1	2	3	4	5	6	7	8	9	10
5	1	2	3	4	5	6	7	8	9	10
6	1	2	3	4	5	6	7	8	9	10
7	1	2	3	4	5	6	7	8	9	10
8	1	2	3	4	5	6	7	8	9	10
9	1	2	3	4	5	6	7	8	9	10

Infinity Norm : max of !total of each row|

MANUAL	CBLAS
3,025	3,025
3,025	3,025
3,025	3,025
3,025	3,025
3,025	3,025
3,025	3,025
3,025	3,025
3,025	3,025
3,025	3,025
3,025	3,025
3,025	3,025

Program – difference from dot.product

C_{jk}	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0

Program – difference from dot-product

C_{jk}	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0

A2-pthreads-solo

Executed using :

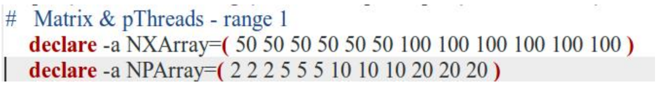
Resulting matrix file contains :

Resulting sample summary timing data file contains :

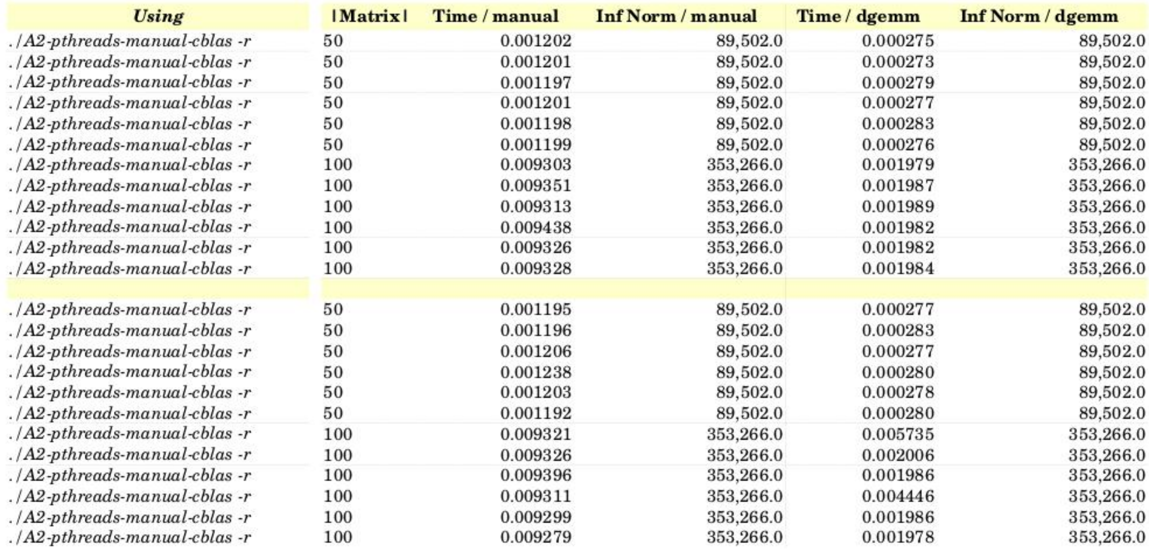
Validating results for pthreads computation gives :

APPENDIX II – SUMMARY OF RESULTS OBTAINED

./A2-pthreads-manual-cblas -r 50 → 100 : range 1

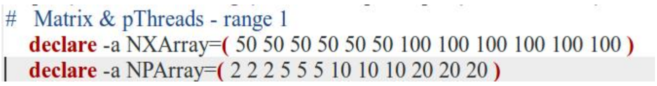


```
#  Matrix & pThreads - range 1
   declare -a NXArray=( 50 50 50 50 50 50 100 100 100 100 100 100 )
   declare -a NPArray=( 2 2 2 5 5 5 10 10 10 20 20 20 )
```

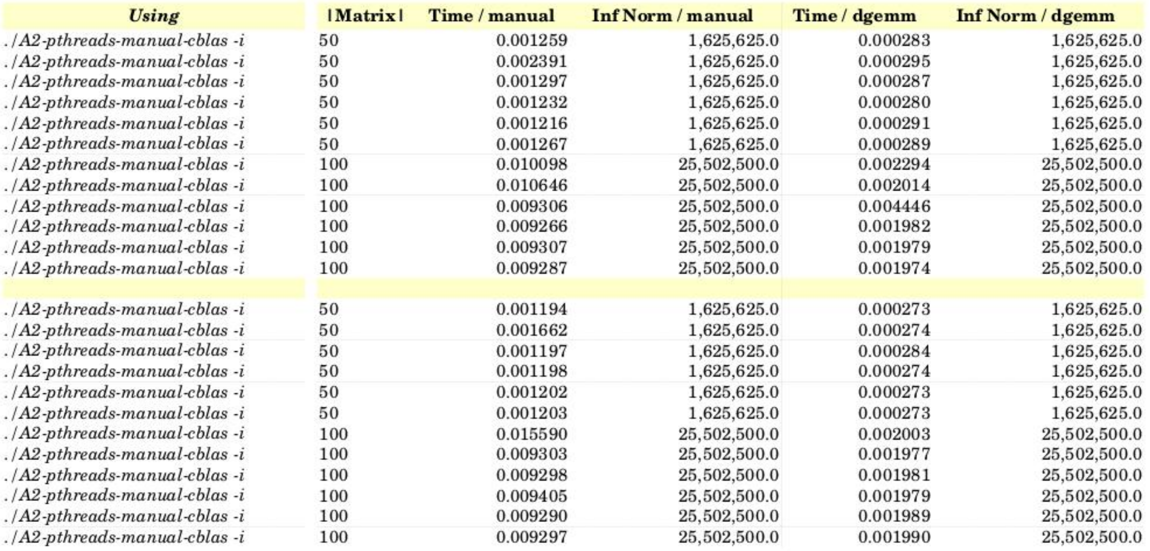


Using	|Matrix|	Time / manual	Inf Norm / manual	Time / dgemm	Inf Norm / dgemm
./A2-pthreads-manual-cblas -r	50	0.001202	89,502.0	0.000275	89,502.0
./A2-pthreads-manual-cblas -r	50	0.001201	89,502.0	0.000273	89,502.0
./A2-pthreads-manual-cblas -r	50	0.001197	89,502.0	0.000279	89,502.0
./A2-pthreads-manual-cblas -r	50	0.001201	89,502.0	0.000277	89,502.0
./A2-pthreads-manual-cblas -r	50	0.001198	89,502.0	0.000283	89,502.0
./A2-pthreads-manual-cblas -r	50	0.001199	89,502.0	0.000276	89,502.0
./A2-pthreads-manual-cblas -r	100	0.009303	353,266.0	0.001979	353,266.0
./A2-pthreads-manual-cblas -r	100	0.009351	353,266.0	0.001987	353,266.0
./A2-pthreads-manual-cblas -r	100	0.009313	353,266.0	0.001989	353,266.0
./A2-pthreads-manual-cblas -r	100	0.009438	353,266.0	0.001982	353,266.0
./A2-pthreads-manual-cblas -r	100	0.009326	353,266.0	0.001982	353,266.0
./A2-pthreads-manual-cblas -r	100	0.009328	353,266.0	0.001984	353,266.0
./A2-pthreads-manual-cblas -r	50	0.001195	89,502.0	0.000277	89,502.0
./A2-pthreads-manual-cblas -r	50	0.001196	89,502.0	0.000283	89,502.0
./A2-pthreads-manual-cblas -r	50	0.001206	89,502.0	0.000277	89,502.0
./A2-pthreads-manual-cblas -r	50	0.001238	89,502.0	0.000280	89,502.0
./A2-pthreads-manual-cblas -r	50	0.001203	89,502.0	0.000278	89,502.0
./A2-pthreads-manual-cblas -r	50	0.001192	89,502.0	0.000280	89,502.0
./A2-pthreads-manual-cblas -r	100	0.009321	353,266.0	0.005735	353,266.0
./A2-pthreads-manual-cblas -r	100	0.009326	353,266.0	0.002006	353,266.0
./A2-pthreads-manual-cblas -r	100	0.009396	353,266.0	0.001986	353,266.0
./A2-pthreads-manual-cblas -r	100	0.009311	353,266.0	0.004446	353,266.0
./A2-pthreads-manual-cblas -r	100	0.009299	353,266.0	0.001986	353,266.0
./A2-pthreads-manual-cblas -r	100	0.009279	353,266.0	0.001978	353,266.0

./A2-pthreads-manual-cblas -i 50 → 100 : range 1

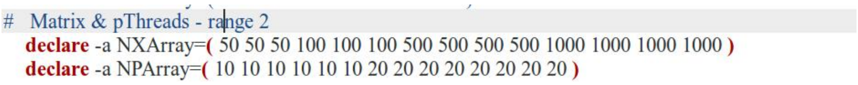


```
#  Matrix & pThreads - range 1
   declare -a NXArray=( 50 50 50 50 50 50 100 100 100 100 100 100 )
   declare -a NPArray=( 2 2 2 5 5 5 10 10 10 20 20 20 )
```

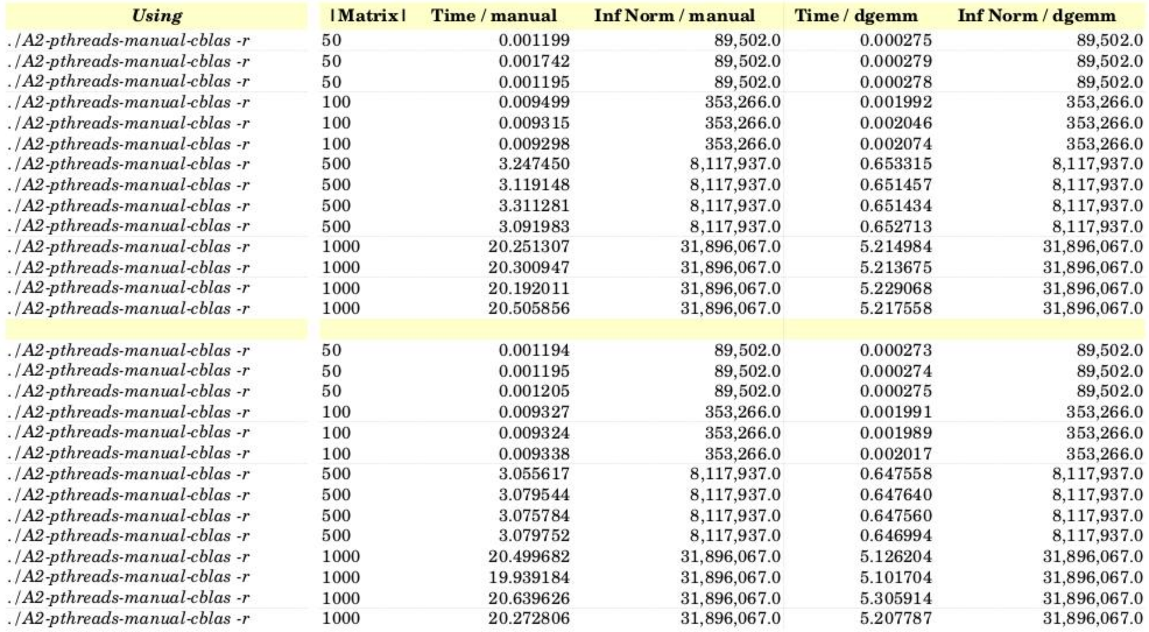


Using	|Matrix|	Time / manual	Inf Norm / manual	Time / dgemm	Inf Norm / dgemm
./A2-pthreads-manual-cblas -i	50	0.001259	1,625,625.0	0.000283	1,625,625.0
./A2-pthreads-manual-cblas -i	50	0.002391	1,625,625.0	0.000295	1,625,625.0
./A2-pthreads-manual-cblas -i	50	0.001297	1,625,625.0	0.000287	1,625,625.0
./A2-pthreads-manual-cblas -i	50	0.001232	1,625,625.0	0.000280	1,625,625.0
./A2-pthreads-manual-cblas -i	50	0.001216	1,625,625.0	0.000291	1,625,625.0
./A2-pthreads-manual-cblas -i	50	0.001267	1,625,625.0	0.000289	1,625,625.0
./A2-pthreads-manual-cblas -i	100	0.010098	25,502,500.0	0.002294	25,502,500.0
./A2-pthreads-manual-cblas -i	100	0.010646	25,502,500.0	0.002014	25,502,500.0
./A2-pthreads-manual-cblas -i	100	0.009306	25,502,500.0	0.004446	25,502,500.0
./A2-pthreads-manual-cblas -i	100	0.009266	25,502,500.0	0.001982	25,502,500.0
./A2-pthreads-manual-cblas -i	100	0.009307	25,502,500.0	0.001979	25,502,500.0
./A2-pthreads-manual-cblas -i	100	0.009287	25,502,500.0	0.001974	25,502,500.0
./A2-pthreads-manual-cblas -i	50	0.001194	1,625,625.0	0.000273	1,625,625.0
./A2-pthreads-manual-cblas -i	50	0.001662	1,625,625.0	0.000274	1,625,625.0
./A2-pthreads-manual-cblas -i	50	0.001197	1,625,625.0	0.000284	1,625,625.0
./A2-pthreads-manual-cblas -i	50	0.001198	1,625,625.0	0.000274	1,625,625.0
./A2-pthreads-manual-cblas -i	50	0.001202	1,625,625.0	0.000273	1,625,625.0
./A2-pthreads-manual-cblas -i	50	0.001203	1,625,625.0	0.000273	1,625,625.0
./A2-pthreads-manual-cblas -i	100	0.015590	25,502,500.0	0.002003	25,502,500.0
./A2-pthreads-manual-cblas -i	100	0.009303	25,502,500.0	0.001977	25,502,500.0
./A2-pthreads-manual-cblas -i	100	0.009298	25,502,500.0	0.001981	25,502,500.0
./A2-pthreads-manual-cblas -i	100	0.009405	25,502,500.0	0.001979	25,502,500.0
./A2-pthreads-manual-cblas -i	100	0.009290	25,502,500.0	0.001989	25,502,500.0
./A2-pthreads-manual-cblas -i	100	0.009297	25,502,500.0	0.001990	25,502,500.0

./A2-pthreads-manual-cblas -r 50 → 1000 : range 2

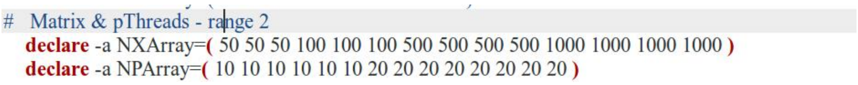


```
#  Matrix & pThreads - range 2
   declare -a NXArray=( 50 50 50 100 100 100 500 500 500 500 1000 1000 1000 1000 )
   declare -a NPArray=( 10 10 10 10 10 10 20 20 20 20 20 20 20 20 )
```

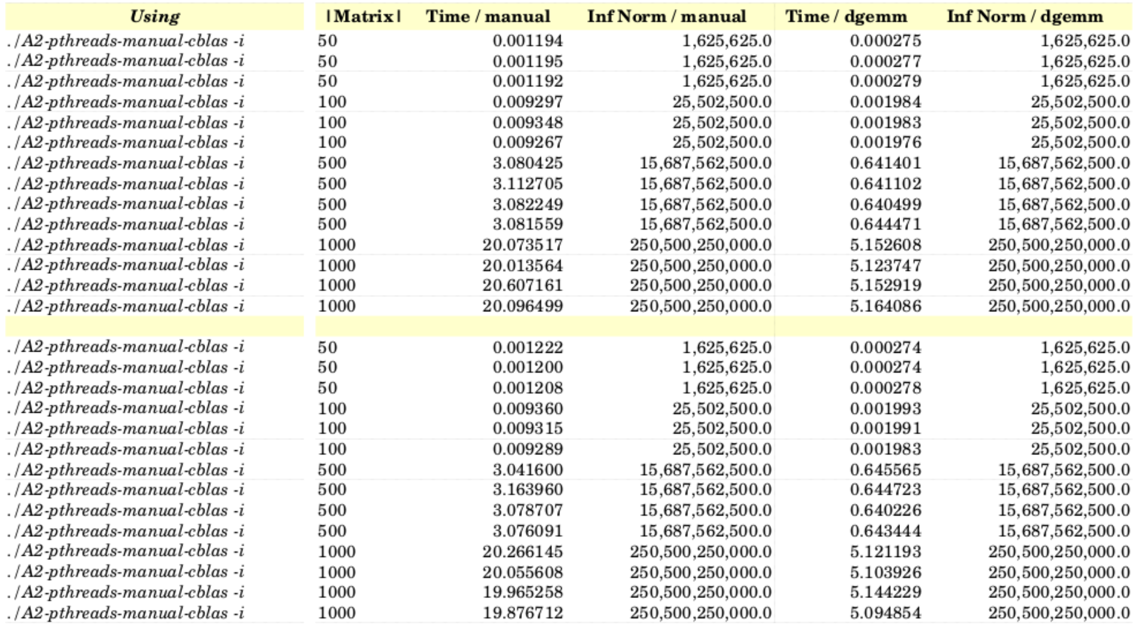


Using	|Matrix|	Time / manual	Inf Norm / manual	Time / dgemm	Inf Norm / dgemm
./A2-pthreads-manual-cblas -r	50	0.001199	89,502.0	0.000275	89,502.0
./A2-pthreads-manual-cblas -r	50	0.001742	89,502.0	0.000279	89,502.0
./A2-pthreads-manual-cblas -r	50	0.001195	89,502.0	0.000278	89,502.0
./A2-pthreads-manual-cblas -r	100	0.009499	353,266.0	0.001992	353,266.0
./A2-pthreads-manual-cblas -r	100	0.009315	353,266.0	0.002046	353,266.0
./A2-pthreads-manual-cblas -r	100	0.009298	353,266.0	0.002074	353,266.0
./A2-pthreads-manual-cblas -r	500	3.247450	8,117,937.0	0.653315	8,117,937.0
./A2-pthreads-manual-cblas -r	500	3.119148	8,117,937.0	0.651457	8,117,937.0
./A2-pthreads-manual-cblas -r	500	3.311281	8,117,937.0	0.651434	8,117,937.0
./A2-pthreads-manual-cblas -r	500	3.091983	8,117,937.0	0.652713	8,117,937.0
./A2-pthreads-manual-cblas -r	1000	20.251307	31,896,067.0	5.214984	31,896,067.0
./A2-pthreads-manual-cblas -r	1000	20.300947	31,896,067.0	5.213675	31,896,067.0
./A2-pthreads-manual-cblas -r	1000	20.192011	31,896,067.0	5.229068	31,896,067.0
./A2-pthreads-manual-cblas -r	1000	20.505856	31,896,067.0	5.217558	31,896,067.0
./A2-pthreads-manual-cblas -r	50	0.001194	89,502.0	0.000273	89,502.0
./A2-pthreads-manual-cblas -r	50	0.001195	89,502.0	0.000274	89,502.0
./A2-pthreads-manual-cblas -r	50	0.001205	89,502.0	0.000275	89,502.0
./A2-pthreads-manual-cblas -r	100	0.009327	353,266.0	0.001991	353,266.0
./A2-pthreads-manual-cblas -r	100	0.009324	353,266.0	0.001989	353,266.0
./A2-pthreads-manual-cblas -r	100	0.009338	353,266.0	0.002017	353,266.0
./A2-pthreads-manual-cblas -r	500	3.055617	8,117,937.0	0.647558	8,117,937.0
./A2-pthreads-manual-cblas -r	500	3.079544	8,117,937.0	0.647640	8,117,937.0
./A2-pthreads-manual-cblas -r	500	3.075784	8,117,937.0	0.647560	8,117,937.0
./A2-pthreads-manual-cblas -r	500	3.079752	8,117,937.0	0.646994	8,117,937.0
./A2-pthreads-manual-cblas -r	1000	20.499682	31,896,067.0	5.126204	31,896,067.0
./A2-pthreads-manual-cblas -r	1000	19.939184	31,896,067.0	5.101704	31,896,067.0
./A2-pthreads-manual-cblas -r	1000	20.639626	31,896,067.0	5.305914	31,896,067.0
./A2-pthreads-manual-cblas -r	1000	20.272806	31,896,067.0	5.207787	31,896,067.0

./A2-pthreads-manual-cblas -i 50 → 1000 : range 2

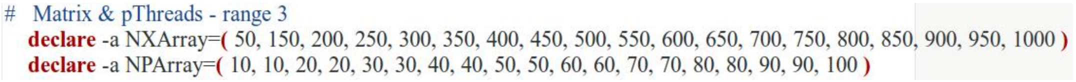


```
#  Matrix & pThreads - range 2
   declare -a NXArray=( 50 50 50 100 100 100 500 500 500 500 1000 1000 1000 1000 )
   declare -a NPArray=( 10 10 10 10 10 10 20 20 20 20 20 20 20 20 )
```

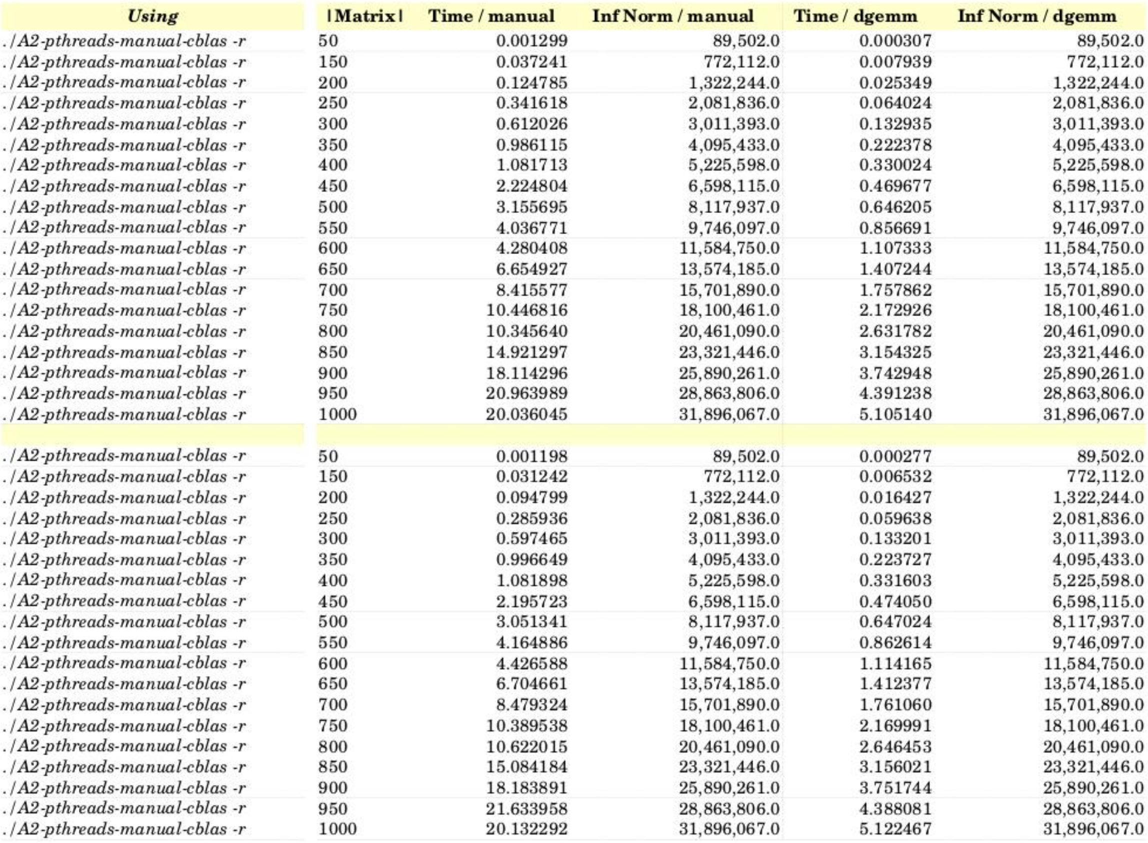


Using	|Matrix|	Time / manual	Inf Norm / manual	Time / dgemm	Inf Norm / dgemm
./A2-pthreads-manual-cblas -i	50	0.001194	1,625,625.0	0.000275	1,625,625.0
./A2-pthreads-manual-cblas -i	50	0.001195	1,625,625.0	0.000277	1,625,625.0
./A2-pthreads-manual-cblas -i	50	0.001192	1,625,625.0	0.000279	1,625,625.0
./A2-pthreads-manual-cblas -i	100	0.009297	25,502,500.0	0.001984	25,502,500.0
./A2-pthreads-manual-cblas -i	100	0.009348	25,502,500.0	0.001983	25,502,500.0
./A2-pthreads-manual-cblas -i	100	0.009267	25,502,500.0	0.001976	25,502,500.0
./A2-pthreads-manual-cblas -i	500	3.080425	15,687,562,500.0	0.641401	15,687,562,500.0
./A2-pthreads-manual-cblas -i	500	3.112705	15,687,562,500.0	0.641102	15,687,562,500.0
./A2-pthreads-manual-cblas -i	500	3.082249	15,687,562,500.0	0.640499	15,687,562,500.0
./A2-pthreads-manual-cblas -i	500	3.081559	15,687,562,500.0	0.644471	15,687,562,500.0
./A2-pthreads-manual-cblas -i	1000	20.073517	250,500,250,000.0	5.152608	250,500,250,000.0
./A2-pthreads-manual-cblas -i	1000	20.013564	250,500,250,000.0	5.123747	250,500,250,000.0
./A2-pthreads-manual-cblas -i	1000	20.607161	250,500,250,000.0	5.152919	250,500,250,000.0
./A2-pthreads-manual-cblas -i	1000	20.096499	250,500,250,000.0	5.164086	250,500,250,000.0
./A2-pthreads-manual-cblas -i	50	0.001222	1,625,625.0	0.000274	1,625,625.0
./A2-pthreads-manual-cblas -i	50	0.001200	1,625,625.0	0.000274	1,625,625.0
./A2-pthreads-manual-cblas -i	50	0.001208	1,625,625.0	0.000278	1,625,625.0
./A2-pthreads-manual-cblas -i	100	0.009360	25,502,500.0	0.001993	25,502,500.0
./A2-pthreads-manual-cblas -i	100	0.009315	25,502,500.0	0.001991	25,502,500.0
./A2-pthreads-manual-cblas -i	100	0.009289	25,502,500.0	0.001983	25,502,500.0
./A2-pthreads-manual-cblas -i	500	3.041600	15,687,562,500.0	0.645565	15,687,562,500.0
./A2-pthreads-manual-cblas -i	500	3.163960	15,687,562,500.0	0.644723	15,687,562,500.0
./A2-pthreads-manual-cblas -i	500	3.078707	15,687,562,500.0	0.640226	15,687,562,500.0
./A2-pthreads-manual-cblas -i	500	3.076091	15,687,562,500.0	0.643444	15,687,562,500.0
./A2-pthreads-manual-cblas -i	1000	20.266145	250,500,250,000.0	5.121193	250,500,250,000.0
./A2-pthreads-manual-cblas -i	1000	20.055608	250,500,250,000.0	5.103926	250,500,250,000.0
./A2-pthreads-manual-cblas -i	1000	19.965258	250,500,250,000.0	5.144229	250,500,250,000.0
./A2-pthreads-manual-cblas -i	1000	19.876712	250,500,250,000.0	5.094854	250,500,250,000.0

./A2-pthreads-manual-cblas -r 50 → 1000 : range 3

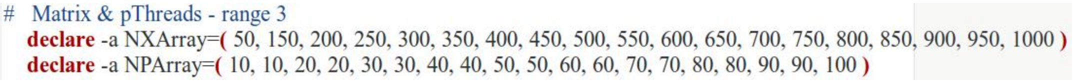


```
#  Matrix & pThreads - range 3
   declare -a NXArray=( 50, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800, 850, 900, 950, 1000 )
   declare -a NPArray=( 10, 10, 20, 20, 30, 30, 40, 40, 50, 50, 60, 60, 70, 70, 80, 80, 90, 90, 100 )
```

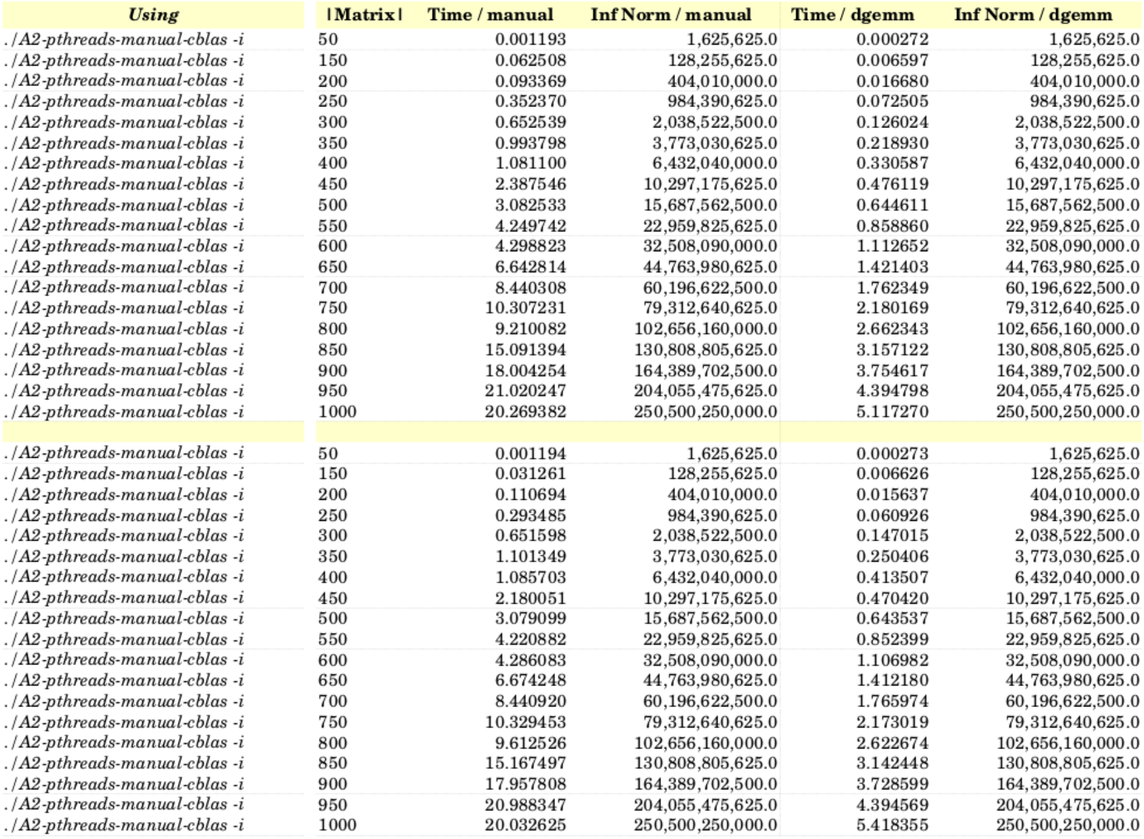


Using	|Matrix|	Time / manual	Inf Norm / manual	Time / dgemm	Inf Norm / dgemm
./A2-pthreads-manual-cblas -r	50	0.001299	89,502.0	0.000307	89,502.0
./A2-pthreads-manual-cblas -r	150	0.037241	772,112.0	0.007939	772,112.0
./A2-pthreads-manual-cblas -r	200	0.124785	1,322,244.0	0.025349	1,322,244.0
./A2-pthreads-manual-cblas -r	250	0.341618	2,081,836.0	0.064024	2,081,836.0
./A2-pthreads-manual-cblas -r	300	0.612026	3,011,393.0	0.132935	3,011,393.0
./A2-pthreads-manual-cblas -r	350	0.986115	4,095,433.0	0.222378	4,095,433.0
./A2-pthreads-manual-cblas -r	400	1.081713	5,225,598.0	0.330024	5,225,598.0
./A2-pthreads-manual-cblas -r	450	2.224804	6,598,115.0	0.469677	6,598,115.0
./A2-pthreads-manual-cblas -r	500	3.155695	8,117,937.0	0.646205	8,117,937.0
./A2-pthreads-manual-cblas -r	550	4.036771	9,746,097.0	0.856691	9,746,097.0
./A2-pthreads-manual-cblas -r	600	4.280408	11,584,750.0	1.107333	11,584,750.0
./A2-pthreads-manual-cblas -r	650	6.654927	13,574,185.0	1.407244	13,574,185.0
./A2-pthreads-manual-cblas -r	700	8.415577	15,701,890.0	1.757862	15,701,890.0
./A2-pthreads-manual-cblas -r	750	10.446816	18,100,461.0	2.172926	18,100,461.0
./A2-pthreads-manual-cblas -r	800	10.345640	20,461,090.0	2.631782	20,461,090.0
./A2-pthreads-manual-cblas -r	850	14.921297	23,321,446.0	3.154325	23,321,446.0
./A2-pthreads-manual-cblas -r	900	18.114296	25,890,261.0	3.742948	25,890,261.0
./A2-pthreads-manual-cblas -r	950	20.963989	28,863,806.0	4.391238	28,863,806.0
./A2-pthreads-manual-cblas -r	1000	20.036045	31,896,067.0	5.105140	31,896,067.0
./A2-pthreads-manual-cblas -r	50	0.001198	89,502.0	0.000277	89,502.0
./A2-pthreads-manual-cblas -r	150	0.031242	772,112.0	0.006532	772,112.0
./A2-pthreads-manual-cblas -r	200	0.094799	1,322,244.0	0.016427	1,322,244.0
./A2-pthreads-manual-cblas -r	250	0.285936	2,081,836.0	0.059638	2,081,836.0
./A2-pthreads-manual-cblas -r	300	0.597465	3,011,393.0	0.133201	3,011,393.0
./A2-pthreads-manual-cblas -r	350	0.996649	4,095,433.0	0.223727	4,095,433.0
./A2-pthreads-manual-cblas -r	400	1.081898	5,225,598.0	0.331603	5,225,598.0
./A2-pthreads-manual-cblas -r	450	2.195723	6,598,115.0	0.474050	6,598,115.0
./A2-pthreads-manual-cblas -r	500	3.051341	8,117,937.0	0.647024	8,117,937.0
./A2-pthreads-manual-cblas -r	550	4.164886	9,746,097.0	0.862614	9,746,097.0
./A2-pthreads-manual-cblas -r	600	4.426588	11,584,750.0	1.114165	11,584,750.0
./A2-pthreads-manual-cblas -r	650	6.704661	13,574,185.0	1.412377	13,574,185.0
./A2-pthreads-manual-cblas -r	700	8.479324	15,701,890.0	1.761060	15,701,890.0
./A2-pthreads-manual-cblas -r	750	10.389538	18,100,461.0	2.169991	18,100,461.0
./A2-pthreads-manual-cblas -r	800	10.622015	20,461,090.0	2.646453	20,461,090.0
./A2-pthreads-manual-cblas -r	850	15.084184	23,321,446.0	3.156021	23,321,446.0
./A2-pthreads-manual-cblas -r	900	18.183891	25,890,261.0	3.751744	25,890,261.0
./A2-pthreads-manual-cblas -r	950	21.633958	28,863,806.0	4.388081	28,863,806.0
./A2-pthreads-manual-cblas -r	1000	20.132292	31,896,067.0	5.122467	31,896,067.0

./A2-pthreads-manual-cblas -i 50 → 1000 : range 3

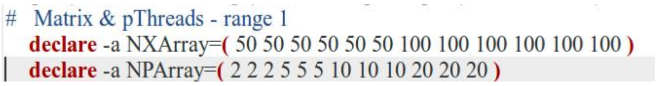


```
#   Matrix & pThreads - range 3
    declare -a NXArray=( 50, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800, 850, 900, 950, 1000 )
    declare -a NPArray=( 10, 10, 20, 20, 30, 30, 40, 40, 50, 50, 60, 60, 70, 70, 80, 80, 90, 90, 100 )
```



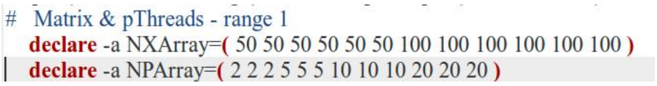
Using	|Matrix|	Time / manual	Inf Norm / manual	Time / dgemm	Inf Norm / dgemm
./A2-pthreads-manual-cblas -i	50	0.001193	1,625,625.0	0.000272	1,625,625.0
./A2-pthreads-manual-cblas -i	150	0.062508	128,255,625.0	0.006597	128,255,625.0
./A2-pthreads-manual-cblas -i	200	0.093369	404,010,000.0	0.016680	404,010,000.0
./A2-pthreads-manual-cblas -i	250	0.352370	984,390,625.0	0.072505	984,390,625.0
./A2-pthreads-manual-cblas -i	300	0.652539	2,038,522,500.0	0.126024	2,038,522,500.0
./A2-pthreads-manual-cblas -i	350	0.993798	3,773,030,625.0	0.218930	3,773,030,625.0
./A2-pthreads-manual-cblas -i	400	1.081100	6,432,040,000.0	0.330587	6,432,040,000.0
./A2-pthreads-manual-cblas -i	450	2.387546	10,297,175,625.0	0.476119	10,297,175,625.0
./A2-pthreads-manual-cblas -i	500	3.082533	15,687,562,500.0	0.644611	15,687,562,500.0
./A2-pthreads-manual-cblas -i	550	4.249742	22,959,825,625.0	0.858860	22,959,825,625.0
./A2-pthreads-manual-cblas -i	600	4.298823	32,508,090,000.0	1.112652	32,508,090,000.0
./A2-pthreads-manual-cblas -i	650	6.642814	44,763,980,625.0	1.421403	44,763,980,625.0
./A2-pthreads-manual-cblas -i	700	8.440308	60,196,622,500.0	1.762349	60,196,622,500.0
./A2-pthreads-manual-cblas -i	750	10.307231	79,312,640,625.0	2.180169	79,312,640,625.0
./A2-pthreads-manual-cblas -i	800	9.210082	102,656,160,000.0	2.662343	102,656,160,000.0
./A2-pthreads-manual-cblas -i	850	15.091394	130,808,805,625.0	3.157122	130,808,805,625.0
./A2-pthreads-manual-cblas -i	900	18.004254	164,389,702,500.0	3.754617	164,389,702,500.0
./A2-pthreads-manual-cblas -i	950	21.020247	204,055,475,625.0	4.394798	204,055,475,625.0
./A2-pthreads-manual-cblas -i	1000	20.269382	250,500,250,000.0	5.117270	250,500,250,000.0
./A2-pthreads-manual-cblas -i	50	0.001194	1,625,625.0	0.000273	1,625,625.0
./A2-pthreads-manual-cblas -i	150	0.031261	128,255,625.0	0.006626	128,255,625.0
./A2-pthreads-manual-cblas -i	200	0.110694	404,010,000.0	0.015637	404,010,000.0
./A2-pthreads-manual-cblas -i	250	0.293485	984,390,625.0	0.060926	984,390,625.0
./A2-pthreads-manual-cblas -i	300	0.651598	2,038,522,500.0	0.147015	2,038,522,500.0
./A2-pthreads-manual-cblas -i	350	1.101349	3,773,030,625.0	0.250406	3,773,030,625.0
./A2-pthreads-manual-cblas -i	400	1.085703	6,432,040,000.0	0.413507	6,432,040,000.0
./A2-pthreads-manual-cblas -i	450	2.180051	10,297,175,625.0	0.470420	10,297,175,625.0
./A2-pthreads-manual-cblas -i	500	3.079099	15,687,562,500.0	0.643537	15,687,562,500.0
./A2-pthreads-manual-cblas -i	550	4.220882	22,959,825,625.0	0.852399	22,959,825,625.0
./A2-pthreads-manual-cblas -i	600	4.286083	32,508,090,000.0	1.106982	32,508,090,000.0
./A2-pthreads-manual-cblas -i	650	6.674248	44,763,980,625.0	1.412180	44,763,980,625.0
./A2-pthreads-manual-cblas -i	700	8.440920	60,196,622,500.0	1.765974	60,196,622,500.0
./A2-pthreads-manual-cblas -i	750	10.329453	79,312,640,625.0	2.173019	79,312,640,625.0
./A2-pthreads-manual-cblas -i	800	9.612526	102,656,160,000.0	2.622674	102,656,160,000.0
./A2-pthreads-manual-cblas -i	850	15.167497	130,808,805,625.0	3.142448	130,808,805,625.0
./A2-pthreads-manual-cblas -i	900	17.957808	164,389,702,500.0	3.728599	164,389,702,500.0
./A2-pthreads-manual-cblas -i	950	20.988347	204,055,475,625.0	4.394569	204,055,475,625.0
./A2-pthreads-manual-cblas -i	1000	20.032625	250,500,250,000.0	5.418355	250,500,250,000.0

./A2-pthreads-solo -i 50 → 100 : range 1



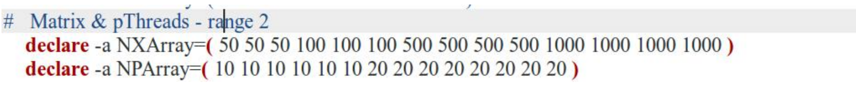
```
#  Matrix & pThreads - range 1
   declare -a NXArray=( 50 50 50 50 50 50 100 100 100 100 100 100 )
   declare -a NPArray=( 2 2 2 5 5 5 10 10 10 20 20 20 )
```

./A2-pthreads-solo -r 50 → 100 : range 1



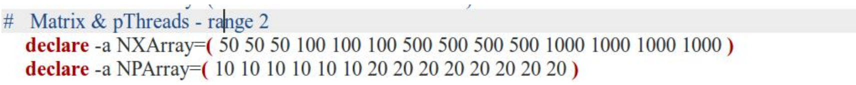
```
#  Matrix & pThreads - range 1
   declare -a NXArray=( 50 50 50 50 50 50 100 100 100 100 100 100 )
   declare -a NPArray=( 2 2 2 5 5 5 10 10 10 20 20 20 )
```

./A2-pthreads-solo -i 50 → 1000 : range 2



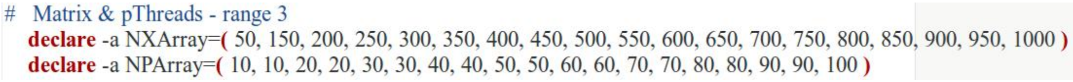
```
#  Matrix & pThreads - range 2
   declare -a NXArray=( 50 50 50 100 100 100 500 500 500 500 1000 1000 1000 1000 )
   declare -a NPArray=( 10 10 10 10 10 10 20 20 20 20 20 20 20 20 )
```

./A2-pthreads-solo -r 50 → 1000 : range 2



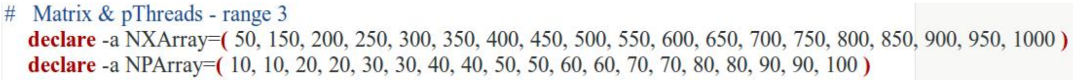
```
#  Matrix & pThreads - range 2
   declare -a NXArray=( 50 50 50 100 100 100 500 500 500 500 1000 1000 1000 1000 )
   declare -a NPArray=( 10 10 10 10 10 10 20 20 20 20 20 20 20 20 )
```

./A2-pthreads-solo -i 50 → 1000 : range 3



```
#  Matrix & pThreads - range 3
   declare -a NXArray=( 50, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800, 850, 900, 950, 1000 )
   declare -a NPArray=( 10, 10, 20, 20, 30, 30, 40, 40, 50, 50, 60, 60, 70, 70, 80, 80, 90, 90, 100 )
```

./A2-pthreads-solo -r 50 → 1000 : range 3



```
#  Matrix & pThreads - range 3
   declare -a NXArray=( 50, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800, 850, 900, 950, 1000 )
   declare -a NPArray=( 10, 10, 20, 20, 30, 30, 40, 40, 50, 50, 60, 60, 70, 70, 80, 80, 90, 90, 100 )
```

Appendix III – References / Acknowledgements

- www.stackoverflow.com : general queries on pThreads functionality not working and possible workarounds / solutions
- http://en.wikipedia.org/wiki/Basic_Linear_Algebra_Subprograms
- www.ucd.ie : COMP40700 High Performance Computing – Notes 2014 “pThreads”
- http://blog.speedgocomputing.com/search/label/PThread