

ADVANCED PROGRAMMING TECHNIQUES

ECE 6122

FINAL PROJECT REPORT

GROUP MEMBERS

ANISH RAMACHANDRA HEGDE (GTID# 903390048)
GEORGE JACOB (GTID# 903261077)
HITESH SUBRAMANYA BHAT (GTID# 903382625)
NARESH HARISH PRABHU (GTID# 903389386)
SRAVANI MADDALA (GTID# 903307338)

IMPLEMENTATION INFORMATION

1. Our group has used a **single CMakeLists.txt file**
2. The modules to be loaded are:
 - module load gcc/4.9.0
 - module load cmake/3.9.1
 - module load openmpi
 - module load cuda
3. Run times considered for plotting are done for cases of excluding and including the writing to output file.
4. **Reverse DFT** has been implemented to satisfy graduate student criterion

NODE CONFIGURATIONS

Node = 1

PPN = 8

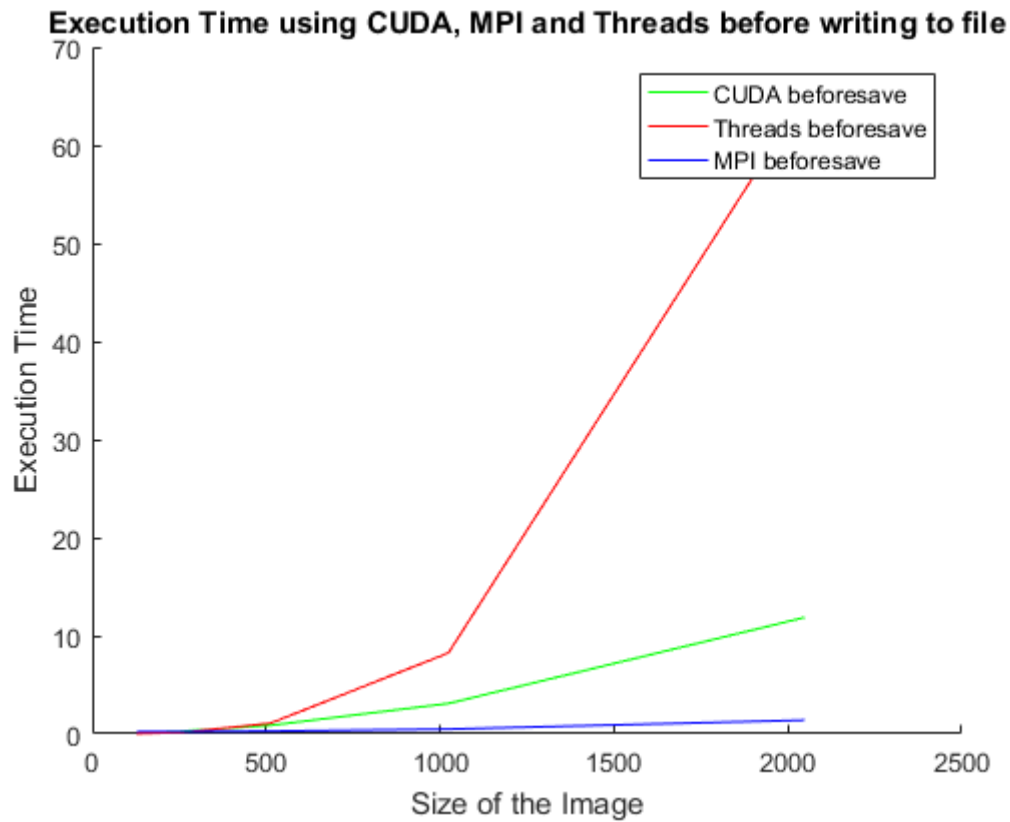
Nvidia Tesla p100 GPU

RUN TIME GRAPHS

1 - Run times excluding Writing to File

N		128	256	512	1024	2048
Threads	DFT	0.02316	0.1454	1.0766	8.2502	64.9472
MPI	FFT	0.2170	0.2298	0.28008	0.48155	1.3980
CUDA	DFT	0.1834	0.2698	0.8644	3.0976	11.863

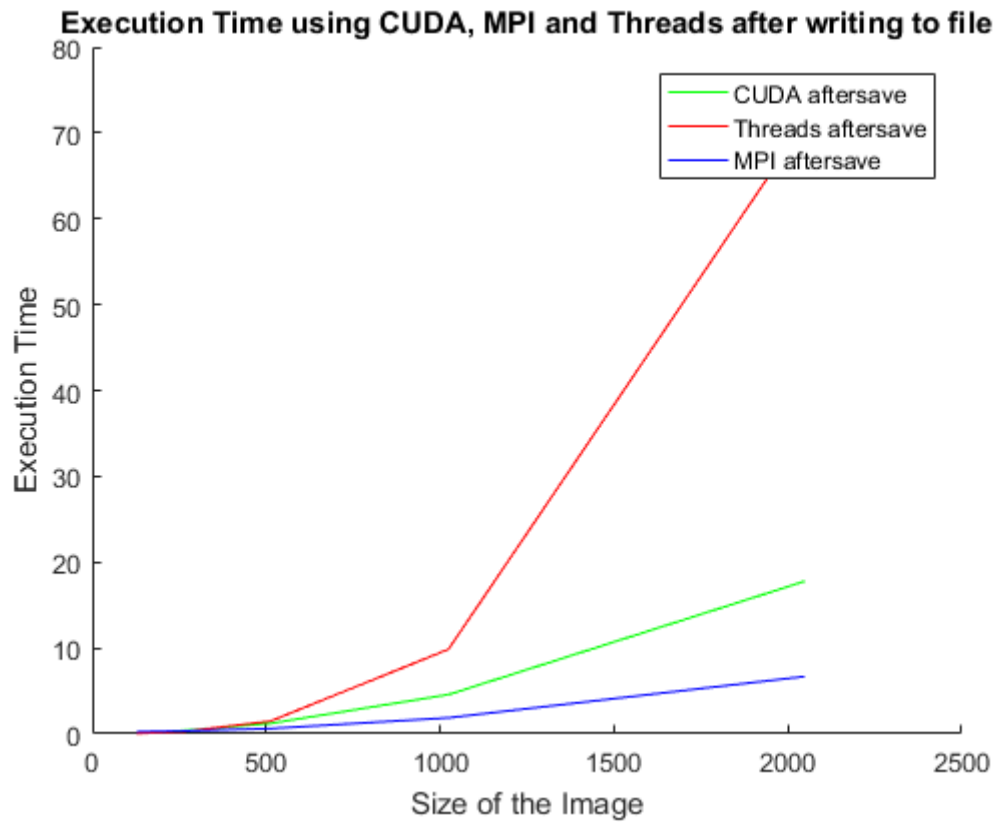
(Time units in seconds)



2 - Run times including Writing to File

N		128	256	512	1024	2048
Threads	DFT	0.0489	0.25334	1.4768	9.8465	71.1266
MPI	FFT	0.2549	0.3328	0.6172	1.8701	6.68
CUDA	DFT	0.20498	0.36308	1.2124	4.5724	17.7696

(Time units in seconds)



INSIGHTS

- The general observation is that MPI implementation is the fastest, followed closely by CUDA implementation, with C++ thread implementation in last place.
- MPI implementation is the fastest as it uses FFT algorithm which reduces complexity from $O(N^2)$ to $O(N\log N)$.
- For $N = 128$, we observe that C++ thread implementation has faster time performance than both CUDA and MPI implementations. This can be attributed to the fact that sending and receiving data between processors in MPI, and transfer of control from device to host in CUDA, take up comparable amount of time as to that of the DFT computation.
- The FFT(Cooley Tukey) implementation in MPI has been done using the iterative method rather than recursive method. No significant time improvements are seen in using recursive method as compared to the iterative method. Furthermore, recursive method uses stack space, whereas iterative method has $O(1)$ space complexity and uses no stack space.