IXPUG Workshop at HPC Asia 2021

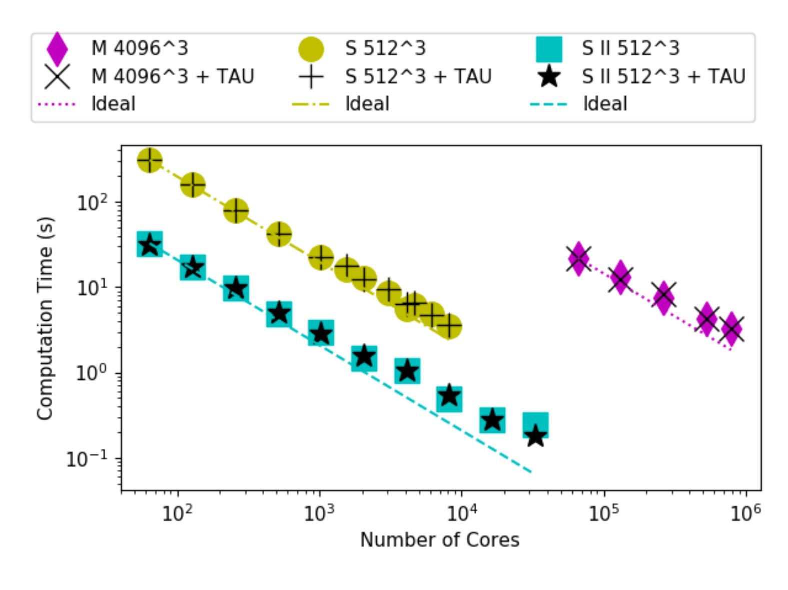
A Comparison of Parallel Profiling Tools for Programs utilizing the FFT

Brian Leu
brian.d.leu@gmail.com
Applied Dynamics International
Ann Arbor, Michigan, USA

Samar Aseeri
King Abdullah University of Science and Technology
Thuwal, Saudi Arabia
samar.aseeri@kaust.edu.sa

Benson K. Muite
Kichakato Kizito
Nairobi, Kenya
benson_muite@emailplus.org

Teaser Image



* Strong scaling and overhead of profiling with TAU for 30 timesteps of the Klein-Gordon equation at spatial resolution of 512^3 and 4096^3 grid points on Mira (M), Shaheen (S), and Shaheen II (S II).

Introduction

- Aim is to see if profiling can help provide an explanation for the differences in scaling behavior of FFTE as compared to 2decomp&FFT.
- The tools, **IPM**, **Scalasca**, **CryaPat**, **mpiP**, **FPMPI** and **TAU** are compared when profiling a Fast Fourier Transform solver for the nonlinear Klein Gordon equation.

Introduction

- The main findings are:
  - Some of these tools can be challenging to setup if not already setup for the user
  - For the number of cores considered here, these tools can be used in a production setting for low overhead profiling
  - Vendor provided tools can allow for well integrated hardware performance monitoring that may not be otherwise easily exposed or integrated in the portable open source tools

Test Program

- Previous work** has compared the performance of a Fast Fourier Transform (FFT) based solver for the three dimensional Klein-Gordon equation using 2dcomp&FFT.
- That study primarily focused on measuring time to solution and strong scaling results.

** S. Aseeri, O. Batrasev, M. Icardi, B. Leu, A. Liu, N. Li, B. K. Muite, E. Müller, B. Palen, M. Quell, H. Servat, P. Sheth, R. Speck, M. Van Moer, and J. Vienne. 2015. Solving the Klein-Gordon Equation Using Fourier Spectral Methods: A Benchmark Test for Computer Performance. In Proceedings of the Symposium on High Performance Computing (Alexandria, Virginia) (HPC '15). Society for Computer Simulation International, San Diego, CA, USA, 182–191. http://dl.acm.org/citation.cfm?id=2872599.2872622

Test Program

- The study also produced a simple performance model. More detailed performance models can be validated using performance measurements.
- In this study the numerical solution of Klein-Gordon equation is again used as an example mini-application for which performance profiling tools can also be compared.

Test Program

- In addition, a solver using the FFT library FFTE is included in the comparison.
- Earlier measurements shown in Teaser Image* showed that TAU, a performance profiling tool can be configured to have low overhead even when profiling at large core counts.
- One of the aims of this work is to quantify the level of overhead and compare TAU to other profiling tools.

Profiling Tools, FFT Library and Test Platforms

- The Profiling Tools:
  - IPM
  - Cryapat
  - FPMPI
  - mpiP
  - Sclalsca
  - TAU

Profiling Tools, FFT Library and Test Platforms

Table 1: An overview of parallel performance tools.

Tool	Profiling	Tracing	Open Source	Development	Support	Forum
CrayPat[12]	Yes	Yes	No	Yes	Paid	No
Extrae[6]	Yes	Yes	Yes	Active	Yes	No
FPMPI[21]	Yes	Yes	Yes	No	No	No
Hpctoolkit[1]	Yes	Yes	Yes	Active	Paid	Open
IPM[16, 48]	Yes	No	Yes	Maintain	No	No
mpiP[47]	Yes	No	Yes	Very little	No	Open
Openspeedshop[27]	Yes	Yes	Yes	Yes	Paid	Closed
PAPIex[33]	Yes	No	Yes	Yes	No	No
Scalasca[17]	Yes	Yes	Yes	Active	Free best effort	No
TAU[7, 30, 31, 42]	Yes	Yes	Yes	Active	Paid	Open
Vtune[23]	Yes	Yes	No	Yes	Paid	Open

Profiling Tools, FFT Library and Test Platforms

- The Parallel FFT Libraries
  - 2decomp&FFT
  - FFTE

Profiling Tools, FFT Library and Test Platforms

- The Platforms
  - Shaheen
  - Mira
  - Shaheen II

Profiling Overhead Measurements

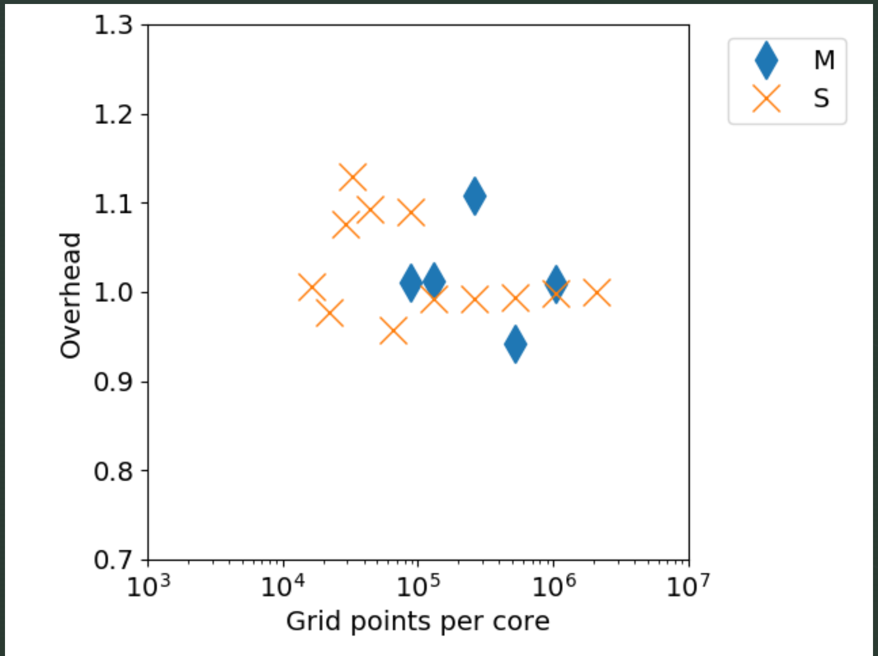
- The supplementary materials at https://zenodo.org/record/4032591 includes procedures for installing IPM, TAU, mpiP, FPMPI and Sclalasca on Shaheen II, some profiler outputs and code for reproducing the results.
- In the appendix, there are typical profiler outputs from TAU, Scalasca, Craypat, mpiP and FPMPI for both FFTE and 2decomp&FFT.

Profiling Overhead Measurements

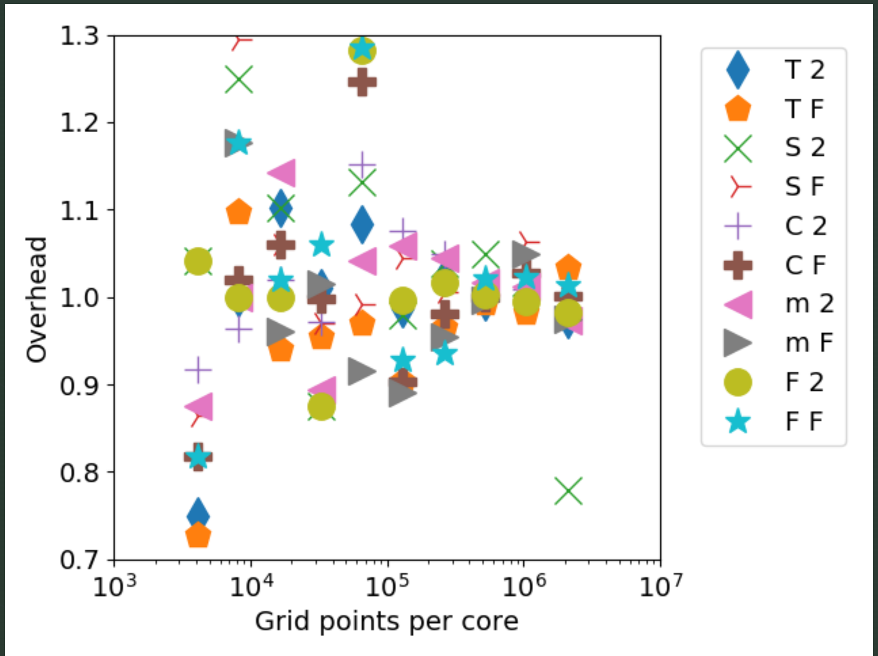
Table 2: Table comparing execution times for 30 time steps of FFT based Klein Gordon equation solvers for a 512^3 problem size using 2decomp&FFT with FFTW and using FFTE libraries. Instrumentation has been done with the Craypat-lite, Scalasca, TAU, mpiP and FPMPI tools on Shaheen II. The mean and standard deviation from 3 runs are shown.

Cores	FFT	No profiling	Craypat-lite	Scalasca	TAU	mpiP	FPMPI
64	2decomp&FFT	32.54 ± 0.10s	31.92 ± 0.38s	25.35 ± 0.02s	31.72 ± 0.50s	31.67 ± 0.10s	31.97 ± 0.41s
64	FFTE	30.68 ± 0.05s	30.69 ± 0.03s	30.67 ± 0.01s	30.45 ± 0.36s	29.90 ± 0.33s	31.11 ± 0.05s
128	2decomp&FFT	17.43 ± 0.17s	17.57 ± 0.24s	17.72 ± 0.94s	17.45 ± 0.17s	17.64 ± 0.03s	17.33 ± 0.07s
128	FFTE	17.76 ± 0.03s	18.26 ± 0.08s	18.89 ± 0.001s	18.06 ± 0.16s	18.62 ± 0.02s	18.17 ± 0.09s
256	2decomp&FFT	9.72 ± 0.26s	9.71 ± 0.03s	10.20 ± 0.07s	9.67 ± 0.01s	9.88 ± 0.23s	9.75 ± 0.06s
256	FFTE	9.73 ± 0.02s	9.73 ± 0.01s	9.81 ± 0.01s	9.78 ± 0.06s	9.70 ± 0.01s	9.93 ± 0.22s
512	2decomp&FFT	4.93 ± 0.07s	5.17 ± 0.26s	5.12 ± 0.04s	5.04 ± 0.13s	5.15 ± 0.17s	5.01 ± 0.03s
512	FFTE	5.09 ± 0.03s	4.99 ± 0.21s	4.97 ± 0.03s	4.91 ± 0.04s	4.86 ± 0.13s	4.76 ± 0.01s
1024	2decomp&FFT	2.93 ± 0.02s	3.15 ± 0.29s	2.87 ± 0.06s	2.89 ± 0.03s	3.10 ± 0.10s	2.92 ± 0.03s
1024	FFTE	2.48 ± 0.03s	2.24 ± 0.02s	2.59 ± 0.29s	2.24 ± 0.01s	2.21 ± 0.05s	2.30 ± 0.08s
2048	2decomp&FFT	1.45 ± 0.017s	1.67 ± 0.05s	1.64 ± 0.02s	1.57 ± 0.03s	1.51 ± 0.13s	1.86 ± 0.20s
2048	FFTE	1.30 ± 0.10s	1.62 ± 0.45s	1.29 ± 0.07s	1.26 ± 0.02s	1.19 ± 0.03s	1.67 ± 0.51s
4096	2decomp&FFT	1.04 ± 0.03s	1.01 ± 0.04s	0.91 ± 0.03s	1.05 ± 0.08s	0.93 ± 0.06s	0.91 ± 0.04s
4096	FFTE	0.66 ± 0.03s	0.659 ± 0.002s	0.64 ± 0.02s	0.63 ± 0.01s	0.67 ± 0.03s	0.70 ± 0.01s
8192	2decomp&FFT	0.487 ± 0.002s	0.50 ± 0.02s	0.54 ± 0.02s	0.54 ± 0.02s	0.56 ± 0.11s	0.49 ± 0.02s
8192	FFTE	0.50 ± 0.01s	0.53 ± 0.05s	0.53 ± 0.10s	0.47 ± 0.02s	0.48 ± 0.03s	0.51 ± 0.01s
16384	2decomp&FFT	0.28 ± 0.01s	0.27 ± 0.01s	0.35 ± 0.08s	0.28 ± 0.01s	0.28 ± 0.01s	0.28 ± 0.02s
16384	FFTE	0.255 ± 0.002s	0.26 ± 0.02	0.33 ± 0.06s	0.28 ± 0.03s	0.30 ± 0.10s	0.30 ± 0.04s
32768	2decomp&FFT	0.24 ± 0.04s	0.22 ± 0.04s	0.25 ± 0.06s	0.18 ± 0.01s	0.21 ± 0.03s	0.25 ± 0.06s
32768	FFTE	0.22 ± 0.04s	0.18 ± 0.02	0.19 ± 0.01s	0.16 ± 0.01s	0.31 ± 0.21s	0.18 ± 0.02s

Discussion



(a) Calculated overhead of profiling with TAU for 30 timesteps of the Klein-Gordon equation on Mira (M) and Shaheen (S) for 40963 and 5123 grids respectively.



(b) Calculated overhead of profiling with Klein Gordon solver for 30 timesteps on Shaheen II. The first character in the legend indicates the profiling tool, C: Craypat-lite, F: FPMPI, m: mpiP and T: TAU. The second character in the legend indicates the FFT library, F: FFTE and 2: 2decomp&FFT

Discussion

- Craypat-lite and FPMPI automatically provide short summary text files with aggregate information at the end of each profiling experiment.
- mpiP also provides a summary text file, however this contains node level information, which while useful for understanding performance imbalance, can become quite lengthy when using thousands of cores.
- Sclalsca, TUA and Craypat-lite also produce comprehensive measurement files that can be postprocessed to obtain further information.

Discussion

- For the current experiments, the summary file provided by FPMPI was the simplest to use as it give the number of MPI_ALLTOALL calls which are most useful when understanding the performance of communication in the parallel FFT, which dominates the wall clock time.
- TAU, Scalasca and Craypat-lite also provide useful information on MPI_ALLTOALL to allow one to determine the reasons for the performance differences between the solver using FFTE and the solver using 2decomp&FFT.
- The summary information from mpiP is less useful for this as it is aggregated by MPI rank, rather than by MPI call.

Summary

- Craypat-lite, Sclasca, mpiP, FPMPI and TAU offer reasonable summarized default lightweight profiling options for parallel programs which can be obtained by compiling the programs with appropriate wrappers.
- Reports produced using Craypat-lite have the most comprehensive information, though it is also possible to obtain more comprehensive information from TAU and Scalasca by changing the runtime configuration.
- On the core counts used here, all these programs have low enough overhead to be used in a production setting, allowing for monitoring of program performance.

Summary

- In parallel programs, a call graph of function information or timing of MPI routines is provided by most of the tools. Learning to use these is a transferable skill.
- The vendor provided performance tools may be more comprehensive and are automatically configured for the user, but since users typically use more than one kind of computer, the open source tools are preferable for initial training.
- For the beginning user, it is helpful to pre-install a lightweight low overhead configuration of a productivity tool for production setting use.

Future work

- Further work involves profiling other parallel FFT libraries, such as heFFTe, AccFFT, nb3dfft, FluidFFT, P3DFFT, PFFT, fftmpi, SWFFT.
- Additional profilig and tracing tools such as Extrae, PapiEx, Intel Vtune and OpenSpeedshop will be included in a more detailed comparison.
- Finally, there are other numerical methods that can be used to solve the Klein Gordon equation, it would be interesting to used these other methods to aid in performance prediction and optimal matching of algorithm to hardware architecture.

Acknowledgments

- The computational resources used to build and test the programs were:
  - Shaheen and Shaheen II operated by the KAUST Supercomputing Laboratory.
  - Hazelhen at HLRS.
  - K computer that was operated by RIKEN
  - Rocket and Vedur operated by the University of Tartu HPC center
  - Mira that was operated by the Argonne Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC02-06CH11357.