Kernel-Based and Total Performance Analysis of CGYRO on 4 Leadership Systems

Igor Sfiligoi
General Atomics
San Diego, CA, USA
sfiligoii@fusion.gat.com

Jeff Candy
General Atomics
San Diego, CA, USA
candy@fusion.gat.com

Emily Belli
General Atomics
San Diego, CA, USA
bellie@fusion.gat.com

ABSTRACT

We present the results of an exhaustive performance analysis of the CGYRO code on 4 leadership systems spanning 5 different configurations. CGYRO is a relatively new fusion plasma simulator designed from the ground up to operate efficiently on multicore and GPU-accelerated systems. The gyrokinetic equations specify a 5-dimensional distribution function for each species, with species coupled through both the Maxwell equations and collision operator. For the cross-machine performance analysis, we report and compare timings for 4 computational and 4 communication kernels, illustrating the strengths and weaknesses of the floating-point and communication architectures of the respective systems.

ACM Reference Format:
Igor Sfiligoi, Jeff Candy, and Emily Belli. 2018. Kernel-Based and Total Performance Analysis of CGYRO on 4 Leadership Systems. In *Proceedings of SC18*. ACM, New York, NY, USA, 3 pages.

1 FUSION PLASMA AS A LOW-CARBON ENERGY SOURCE

The US and global economies increasingly depend on reliable sources of energy. In coming decades, these sources must become increasingly low-carbon to mitigate the risks of climate change. Thus, the challenge to harness the virtually inexhaustible potential of fusion energy is being pursued in a coordinated worldwide effort. In parallel with a vibrant global research program, numerical simulation serves as a powerful tool for accelerating progress. Simulations are used to validate basic theory, plan experiments, interpret results on present devices, and ultimately to design future devices. While the *long-term* goal of fusion simulation is to provide the scientific basis for a demonstration reactor, a *near-term* goal is to refine our understanding of physics issues associated with burning plasmas. This simulation capability relies on high-performance computing, enabling researchers to obtain key insights from fundamental physical models.

2 CGYRO: A MULTISCALE-OPTIMIZED FUSION PLASMA SOLVER

CGYRO[1, 2, 4, 5] is an Eulerian gyrokinetic solver designed and optimized for collisional, electromagnetic, multiscale fusion plasma simulation. It is written in Fortran 2008 and was designed to be suitable for next-generation computational systems that require high levels of parallel concurrency. The implementation combines 15 years of algorithmic lessons learned from GYRO[3, 6–8], together with an array distribution scheme and loop structure that targets modern multicore and accelerated (GPU) architectures. CGYRO was designed for operation on Petascale systems, and employs MPI to split the problem over a potentially very large number of compute processes. Moreover, the in-process parallelization scheme for most kernels employs cache-aligned data arrays, OpenMP and OpenACC parallelized loops and vectorization-friendly operations.

CGYRO operates on a 6 dimensional grid (3D space + 2 D velocity + 1 D species). The numerical discretization is spectral in the radial and binormal dimensions (k_x, k_y), pseudospectral in both velocity dimensions, and uses a unique 5th-order conservative upwind scheme in θ, the coordinate along the field lines. The grid is split using two orthogonal MPI communicators. The first MPI communicator size is fixed by the problem size; it is always the value of N_TOROIDAL (i.e. $k_y/2$). All the other dimensions are then lumped together, and can be distributed along the other MPI communicator.

The CGYRO code can be logically split in four logical kernels: `str`, `nl`, `field` and `coll`. Each simulation performs many iterations of the above, with MPI collective communication required to maintain a global view. The problem has been formulated to intentionally push most of the computation to the FFT-based `nl` kernel.

More details about the data structures and communication patterns are presented in the poster. The poster also includes the detailed mathematical formulation governing gyrokinetic fusion plasma simulations.

3 CROSS-PLATFORM BENCHMARK SETUP

CGYRO performance testing was carried out on four leadership systems: TACC Stampede 2, NERSC Cori, CSCS Piz Daint and OLCF Titan. Two different partitions were used on Stampede 2, resulting in 5 different configurations being tested; 2 KNL-based, one Skylake-based and 2 hybrid CPU-GPU architectures. An overview of the key features of each architecture is given in Tables 1 (CPU systems) and 2 (CPU/GPU systems).

For the KNL systems, only 64 (of 68) cores per chip are used in order to facilitate problem splitting (i.e., array distribution). Moreover, we configure the runtime environment to use 2 out of 4 possible hyperthreads. Although in some cases we see a small performance improvement with the maximum 4 hyperthreads, the improvement is usually *insignificant*. All KNL nodes were configured to use the cache memory mode and quadrant cluster mode.

CGYRO can be used for both small-scale simulations of the plasma core and large-scale, multiscale simulations required for

---

SC18, November 2018, Dallas, Texas USA

Table 1: Architecture overview of CPU-only systems including theoretical peak.

	Stampede2		Cori
	Skylake	KNL	
CPU Model	2 x Xeon Plat 8160	Xeon Phi 7250	Xeon Phi 7250
Threads/node	96	272 (128 used)	272 (128 used)
TFLOPS/node	3.5	3.0	3.0
Nodes	1736	4200	9668
Interconnect	Intel Omni-Path		Cray Aries
Net. Topology	Fat Tree		Dragonfly
Compiler	Intel Fort 17		Intel Fort 17
FFT library	Intel MKL		FFTW v3.3.6

Table 2: Architecture overview of hybrid CPU/GPU systems including theoretical peak.

	Piz Daint	Titan
CPU Model	Xeon E5-2690 v3	Opteron 6274
GPU Model	Tesla P100	Tesla K20X
Threads/node	24 (12 used) + 3584	16 + 2688
TFLOPS/node	4.5 (0.5+4.0)	1.5 (0.2+1.3)
Nodes	5320	18688
Interconnect	Cray Aries	Cray Gemini
Net. Topology	Dragonfly	3D Torus
Compiler	PGI Fort 17	PGI Fort 17
FFT library	cuFFT	cuFFT

accurately describing the edge region. CGYRO performance testing was carried out using two test cases, `nl03` and `nl04`, that are broadly representative of small multiscale simulations. A more detailed description of each of the two cases is presented in the poster.

4 PERFORMANCE ANALYSIS

To make inter-machine comparisons, one must have a meaningful *equal performance metric*. Because it can be misleading to compare multicore CPU systems to hybrid CPU-GPU systems using a thread-to-thread comparison, the benchmark results presented in the poster use either node-by-node comparisons, or comparisons using vendor peak-performance claims. This means, for example, that two Titan nodes = one Cori KNL node = 3.0 peak TFLOPS.

Analyzing the strong scaling results of the total application runtime, we observe that CPU systems perform closer to peak PFLOPS than GPU systems, but scale less. On the flip side, GPU systems are severely limited on the lower end, due their small on-board memory size.

Understanding the per-system performance in more detail requires a deeper, kernel-level analysis. The benchmark results of the `nl03` test case at fixed 128 nodes are given in Fig. 1. Additional benchmark results are presented in the poster.

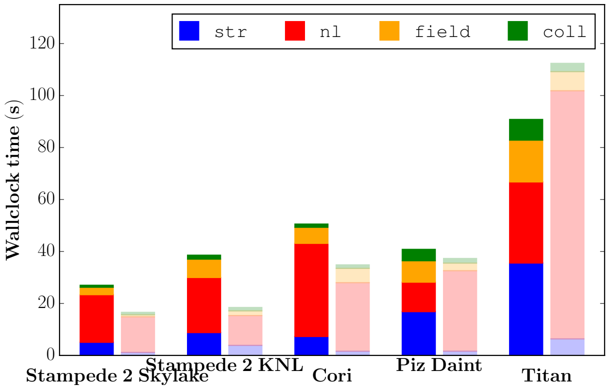

Figure 1: Kernel-level analysis for `nl03` with data taken at fixed 128 nodes. For each hardware configuration, left bars indicate compute time, and right (slightly faded) bars are the corresponding communication times.

First, on the CPU systems, the compute time is highly dominated by the nonlinear step. This is indeed a feature of the spectral algorithm that simpifies the linear dynamics, pushing the computational burden to the nonlinear term which is evaluated with a series of 2D FFTs. On the GPU systems, the extremely high performance of cuFFT gives rise to a relatively short time spent in the `nl` kernel, especially for the smaller `nl03` test case. On the larger `nl04` test case, the difference is much less noticeable. Moreover, the MKL FFT library, used on Stampede 2, significantly outperforms FFTW, which is used on Cori.

Regarding interconnects, we find that Intel Omni-Path outperforms Cray Aries, giving a very good balance with the floating-point performance on the Stampede2 systems. Moreover, the nearly 8-year-old Cray Gemini interconnect on Titan shows its age and is the poorest performer of the group.

5 SUMMARY

In this poster we describe the mathematical formulation, numerical discretization, and performance/scaling results for the new CGYRO gyrokinetic code. The performance data was collected on 4 current leadership systems, spanning both CPU-only and hybrid-GPU configurations. For the cross-machine performance analysis, we compare timings for 4 computational and 4 communication kernels kernels, thereby illustrating the strengths and weaknesses of the floating-point and communication architectures of the respective systems. Excellent strong scaling results are observed on both multicore-CPU and CPU/GPU systems.

ACKNOWLEDGMENTS

This material is based upon work supported by the U.S. Department of Energy, Office of Science, through the AToM project under Grant DE-SC0017992. The research used resources of the Oak Ridge Leadership Computing Facility under Contract DE-AC05-00OR22725 and of the National Energy Research Scientific Computing Center under Contract No. DE-AC02-05CH11231. We also acknowledge support by the Swiss National Supercomputing Centre (CSCS) under project ID s819, and support by Texas Advanced Computing Center (TACC) under project ID TG-PHY170044 that contributed to this paper.

REFERENCES

[1] E.A. Belli and J. Candy. 2017. Implications of advanced collision operators for gyrokinetic simulation. *Plasma Phys. Control. Fusion* 59 (2017), 045005.

[2] E.A. Belli and J. Candy. 2018. Impact of centrifugal drifts on ion turbulent transport. *Phys. Plasmas* 25 (2018), 032301.

[3] J. Candy and E. Belli. 2010. GYRO Technical Guide. *General Atomics Technical Report* GA-A26818 (2010).

[4] J. Candy and E.A. Belli. 2018. Spectral treatment of gyrokinetic shear flow. *J. Comput. Phys.* 356 (2018), 448.

[5] J. Candy, E.A. Belli, and R.V. Bravenec. 2016. A high-accuracy Eulerian gyrokinetic solver for collisional plasmas. *J. Comput. Phys.* 324 (2016), 73.

[6] J. Candy and R.E. Waltz. 2003. An Eulerian gyrokinetic-Maxwell solver. *J. Comput. Phys.* 186 (2003), 545.

[7] J. Candy and R.E. Waltz. 2003. Anomalous Transport in the DIII-D Tokamak Matched by Supercomputer Simulation. *Phys. Rev. Lett.* 91 (2003), 045001–1.

[8] J. Candy, R.E. Waltz, and W. Dorland. 2004. The local limit of global gyrokinetic simulations. *Phys. Plasmas* 11 (2004), L25.