UNSTRUCTURED-GRID ALGORITHMS FOR A MANY-CORE LANDSCAPE

AARON WALDEN1,2, ERIC J. NIELSEN1, MOHAMMAD ZUBAIR2, JOHN C. LINFORD3, JUSTIN P. LUITJENS4, JASON ORENDE2, IZAAK BEEKMAN3, SAMUEL KHUVIS3 AND SAMEER S. SHENDE3

1 NASA Langley Research Center
   Hampton, VA, 23666, USA
   {aaron.walden,eric.j.nielsen}@nasa.gov
2 Old Dominion University
   Norfolk, VA, 23529, USA
   {zubair,jorender}@cs.odu.edu
3 ParaTools, Inc.
   Eugene, OR, 97405, USA
   {jlinford,ibeekman,skhuvis,sameer}@paratools.com
4 NVIDIA Corp.
   Salt Lake City, UT, 84101, USA
   jluitjens@nvidia.com

Key words: Parallel CFD applications, Heterogeneous computing, High-performance computing, Optimization

Abstract. We explore the transition of a legacy, MPI-only, domain-decomposed unstructured-grid code highly optimized for multi-core systems to shared-memory MPI+OpenMP and MPI+CUDA models more suitable for a future high-performance computing landscape dominated by heterogeneous many-core architectures. We study node-level performance characteristics of compute-intensive kernels hand-optimized using CUDA and AVX512 vector intrinsics. Strong scaling results are presented which contrast the scalability of the original MPI-only model with that of the hybrid models.

1 INTRODUCTION

NASA Langley Research Center’s FUN3D is an unstructured-grid computational fluid dynamics software suite used to tackle complex aerodynamics problems [1]. The toolset enables multidisciplinary capabilities through coupling to variable fidelity models encompassing structural effects, multibody dynamics, acoustics, radiation, optics, propulsion, and ablation. FUN3D provides advanced adjoint-based design capability, enabling formal optimization of time-dependent moving-body simulations involving turbulent flows. The adjoint formulation is also used to perform rigorous mesh adaptation and error estimation.

FUN3D currently relies on explicit domain decomposition and a coarse-grained MPI-based communication paradigm optimized for multi-core systems. Practical considera-
tions guiding the development of future exascale-class systems impose severe power constraints on their design [2]. This next generation of high-performance computing systems is likely to rely on many-core architectures offering substantially improved energy efficiency through higher degrees of concurrency and lower clock rates. These systems are also expected to lower the amount of memory available per core. Ultimately, these and other factors are likely to render legacy MPI-based paradigms inefficient, and it is widely recognized that such applications will require a transition to some form of hybrid parallelism to effectively leverage future exascale-class computing platforms.

In this work, we describe the transition of FUN3D from a pure MPI model to shared-memory hybrid MPI+OpenMP and MPI+CUDA models with support for heterogeneous architectures. We detail the optimization of the two most computationally intensive kernels with particular emphasis on the development of custom CUDA and AVX512 vector intrinsic implementations. Race condition avoidance, vectorization, and memory bandwidth optimization are also discussed. We study performance at the node level for the following architectures: Intel Xeon Broadwell and Skylake, Intel Xeon Phi Knights Landing, and NVIDIA Pascal P100 and Volta V100 GPUs.

We perform strong scaling studies of both conventional FUN3D and our hybrid shared-memory implementations across a variety of high-performance computing systems. In our strong scaling studies, we focus on the scalability of FUN3D’s multicolor point implicit linear solver, which performs the bulk of inter-node communication. In particular, the issue of MPI performance scaling with the number of ranks per node is studied. We also study load balancing in the context of mixed-element grids and the performance of intra-node and inter-node data movement in a heterogeneous CPU+GPU setting.

REFERENCES
