



DOE

MARKET RESEARCH STUDY

**APPLICATIONS OF
ASCR HPC SOFTWARE**

June 2022

This project was funded by the Department of Energy, an agency of the United States Government, through a contract for Phase 0 Outreach and Assistance for the Small Business Innovation Research/Small Business Technology Transfer Programs Office. Neither the United States Government nor any agency thereof, nor any of its employees, nor the contractor, nor any of their employees, makes any warranty, expressor implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

TABLE OF CONTENTS

Introduction	4
HDF5	5
User Group.....	5
Curated Application List.....	6
Legion	9
User Group.....	9
Curated Application List.....	10
LLVM	12
User Group.....	12
Curated Application List.....	13
MFEM	15
User Group.....	15
Curated Application List.....	16
Open MPI	18
User Group.....	19
Curated Application List.....	19
MPICH	21
User Group.....	21
Curated Application List.....	22
Trilinos	23
User Group.....	24
Curated Application List.....	25
PETSc	27
User Group.....	28
Curated Application List.....	29
SPACK	31
User Group.....	31
Curated Application List.....	32
STRUMPACK	33
User Group.....	33
Curated Application List.....	34
Endnotes	37

INTRODUCTION

The Department of Energy (DOE)'s Advanced Scientific Computing Research (ASCR) program has developed hundreds of software tools for high performance computing (HPC). Although many HPC hardware advances have permeated the market, such as CPUs, GPUs, TPUs and ASICs, the software developed by DOE for different HPC applications has not necessarily been mined. The DOE is interested in making its HPC software portfolio available to the public in order to maximize the value of what has been developed.

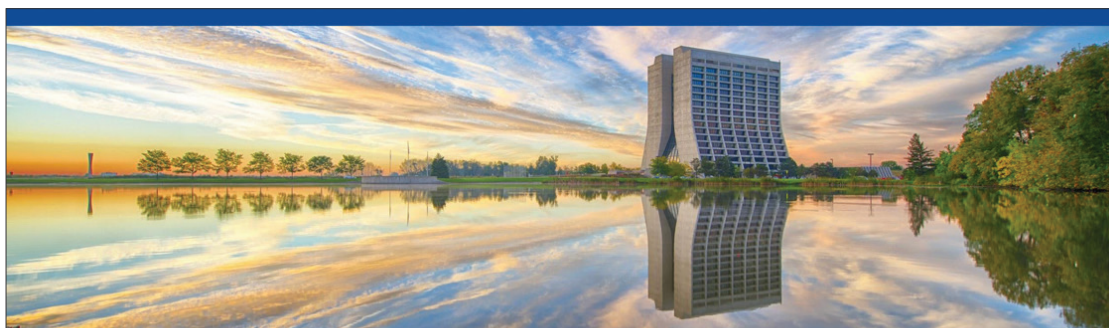
This document, organized by the name of the ASCR-funded software, provides illustrations of how a sample of these tools have been used to advance industry applications. The ASCR-developed HPC software tools are freely available and could save time for the open-source community as programmers develop, enhance and commercialize software tools for use in medical, advanced manufacturing, defense, energy and other applications.

This document highlights ten HPC software tools, each of which has an affiliated User Group. Each section begins with a brief introduction to the genesis of the platform, followed by an introduction to the user group. The primary objective of this document however is to provide those who are unfamiliar with these tools with a curated list of references that demonstrates how others have used these tools to advance knowledge in other fields. The references are recent and included as they represent diverse uses of the HPC software. The references are grouped by sample applications.

HDF HIERARCHICAL DATA FORMAT

Hierarchical Data Format (HDF) is a set of file formats designed to store and organize large amounts of data. HDF was originally developed by the [National Center for Supercomputing Applications \(NCSA\)](#) located at the University of Illinois-Urbana. This is one of the five supercomputing centers originally funded by the National Science Foundation in the 1980s.

HDF5 is a new format first released in beta form in 1998 to meet the increasing demands of scientific computing.¹ It was developed by Quincey Koziol from Lawrence Berkeley National Lab (LBNL) and Dana Robinson from the [HDF Group](#). Details on the differences between HDF and HDF5 are discussed in detail on this [site](#). In a 2021 presentation – the Story of HDF5 in High Energy Physics is presented in this detailed presentation.



The Story of HDF5 in High Energy Physics

Jim Kowalkowski, Marc Paterno and **Saba Sehrish**

HDF User Group Meeting 2021

10/11/2021

[Click to navigate to Source](#)

// HDF5 USER GROUP

The mission of the HDF Group, a not-for-profit organization is to advance the state of the art of open source data management technologies – “ensuring long-term access to the data, and supporting our dedicated and diverse user group”.² The HDF Group [website](#) provides documentation, webinar recordings, an HDF lab, a support portal and a listing of upcoming events.

Astronomy

- (1) Alok Kumar Rai et al. [Experimental investigation to identify galaxy clusters using sparse matrix clustering algorithms](#). Research Square, May 12, 2022.

Abstract: Recent advances in astronomy have shifted observational astronomy toward datadriven astronomy, with an exponential increase in data associated with celestial objects. If we can handle this massive amount of quickly changing data, we will be able to detect galaxy clusters, revealing a wealth of vital information about the evolution of the universe. In this paper, we have proposed a novel clustering technique that can handle massive amounts of rapidly changing data in real time. In this clustering technique we have used 3D sparse matrix where each cell of the 3D matrix can be used to locate/identify a neighbor against the central galactic coordinate. In our investigation, we also used HDF5 to store and organize the discovered galaxy clusters so that we didn't have to re-run our algorithm every time a new coordinate was encountered. Following preparation, the astronomical data containing the galactic coordinates RA, DEC, and radial-distance obtained from redshift is input into our developed algorithm to identify, store, and depict the galaxy clusters.

Biomedical

- (1) J. Rodrigues, H. Gamboa, V. Kublanov and A. Dolganov, "[Storage of Biomedical Signals: Comparative Review of Formats and Databases](#)," 2019 International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON), 2019, pp. 0652-0656, doi: 10.1109/SIBIRCON48586.2019.8958222.

Abstract: Current trends in healthcare digitalization make the task of collecting and storing the data of biomedical signals of the subjects relevant. In this article will be described: (i) the existing standard formats of biomedical data files and their comparison in terms of several technical features; (ii) the public databases where biomedical data is accessible, the type of data stored in these databases, how the data is structured. The comparative analysis revealed that **the HDF5 file format** has the best properties, including compression, speed and ease of access, and the availability of metadata. The most common datasets available online are related to electrocardiography and electroencephalography studies. The comparative review showed that there is a significant reserve that can be used by novice researchers to evaluate the processing algorithms of biomedical signals.

- (2) Janek Gröhl, Lina Hacker, Ben T. Cox, Kris K. Dreher, Stefan Morscher, Avotra Rakotondrainibe, François Varray, Lawrence C.M. Yip, William C. Vogt, Sarah E. Bohndiek, The IPASC data format: A consensus data format for photoacoustic imaging. Photoacoustics, Volume 26, June 2022.

Abstract: [Photoacoustic imaging](#) (PAI) is an emerging modality that has shown promise for improving patient management in a range of applications.

Unfortunately, the current lack of uniformity in PAI [data formats](#) compromises inter-user data exchange and comparison, which impedes: technological progress; effective research collaboration; and efforts to deliver multi-centre clinical trials. To overcome this challenge, the International Photoacoustic Standardisation Consortium (IPASC) has established a data format with a defined consensus [metadata structure](#) and developed an open-source software [application programming interface](#) (API) to enable conversion from proprietary file formats into the IPASC format. The format is based on **Hierarchical Data Format 5 (HDF5)** and designed to store photoacoustic raw [time series data](#). Internal quality control mechanisms are included to ensure completeness and consistency of the converted data. By unifying the variety of proprietary data and metadata definitions into a consensus format, IPASC hopes to facilitate the exchange and comparison of PAI data.

Energy

- (1) Wei Lu et al. [Research and Application of Nuclear Reactor Computational Data Framework Based on HDF5](#). Communications in Computer and Information Science. June 29, 2021.

Abstract: Nuclear reactor computing software need to process and maintain complex and massive data sets. In order to meet the requirement of mass data storage and processing in software, the data storage model and I/O method and programming interface of **HDF5 (Hierarchical Data Format v5)** are deeply studied. According to the data storage and processing requirement of typical nuclear reactor computing data such as assembly or core neutronics computing data and core thermal-hydraulic computing data, RCDF-H5 (Reactor Compute Data Framework based on HDF5) is proposed. KYMRES (KYlin-2 Main RESULTS databank) and COMRES (COre Main RESULTS databank) are designed and implemented based on RCDF-H5. The performance tests show that RCDF-H5 has a higher I/O efficiency than conventional storage solutions. A new data storage and management solution for nuclear reactor computing software is provided.

Traffic Patterns

- (1) Scholtes, Maïke et al. [OMEGAFormat: A Comprehensive Format of Traffic Recordings for Scenario Extraction](#). Workshop Fahrerassistenz und automatisiertes Fahren

Abstract: The most crucial aspects influencing the behavior of Automated Vehicles (AVs) are static environment, dynamic objects and weather. Traffic recordings that include these aspects allow the derivation of comprehensive test scenarios of AVs. However, existing recording descriptions are either not designed for test scenario derivation or use separate formats for the mentioned aspects. Therefore, in this paper, we present an **HDF5-based recording** format that unifies the data basis for scenario derivation by covering all layers of the 6-Layer Model (6LM). By open-sourcing the format specification along with a library† and converters and enrichers §, we hope to facilitate research on scenario generation considering all relevant aspects of traffic.

LEGION

Legion is a data-centric parallel programming system for writing portable high performance programs targeted at distributed heterogeneous architectures. It is used with parallel, accelerated, distributed machines.³ Legion was developed as an open source project with contributors from Stanford University, Los Alamos National Laboratory (LANL)⁴, Stanford Linear Accelerator (SLAC), Nvidia and funding from the Office of Advanced Scientific Computing Research (ASCR).⁵



The Legion Programming Model, Wonchan Lee, Stanford University

[Click to Navigate to video](#)

// LEGION USER GROUP

The URL for the [Legion User Group](#) is at Stanford. The website has documentation and tutorials for new and experienced Legion users. The events listed on the site always have an assortment of useful videos and presentations to review.

Biomedical

- (1) Chang, Hsing-Yin et al. [Scaling and Acceleration of Three-dimensional Structure Determination for Single-Particle Imaging Experiments with SpiniFEL](#). arXiv:2109.05339v1, 2021

Abstract: The Linac Coherent Light Source (LCLS) is an X-ray free electron laser (XFEL) facility enabling the study of the structure and dynamics of single macromolecules. A major upgrade will bring the repetition rate of the X-ray source from 120 to 1 million pulses per second. Exascale high performance computing (HPC) capabilities will be required to process the corresponding data rates. We present SpiniFEL, an application used for structure determination of proteins from single-particle imaging (SPI) experiments. An emerging technique for imaging individual proteins and other large molecular complexes by outrunning radiation damage, SPI breaks free from the need for crystallization (which is difficult for some proteins) and allows for imaging molecular dynamics at near ambient conditions. SpiniFEL is being developed to run on supercomputers in near real-time while an experiment is taking place, so that the feedback about the data can guide the data collection strategy. We describe here how we reformulated the mathematical framework for parallelizable implementation and accelerated the most compute intensive parts of the application. We also describe the use of Pygion, a Python interface for the **Legion task-based programming model** and compare to our existing MPI+GPU implementation.

Concentrated Solar Power

- (1) Torres, Hilario Cardenas. [Simulations of Particle Laden Turbulence in a Radiation Environment Using Heterogeneous High Performance Computing Systems](#). Stanford University ProQuest Dissertation Publishing, 2021.

Abstract: The Predictive Science Academic Alliance Program (PSAAP) II at Stanford University investigated particle-laden turbulent flows in a radiation environment for concentrated solar power (CSP) applications. Over the course of the PSAAP project an extensive series of laboratory scale experiments and corresponding high fidelity simulations were designed and conducted for a conceptual volumetric solar energy receiver design. The configuration of interest involved the coupling of three distinct physical phenomena: fluid turbulence, particle dynamics, and the transport of thermal radiation.

The work presented in this thesis is primarily focused on computational thermal radiation transport and multi-physics simulations of this conceptual solar energy receiver. The governing equations and numerical methods used to model the three major physical processes relevant to the PSAAP application (fluid, particles, and radiation) are presented.... An overview of the multi-physics solver that was developed to simulate the physical processes relevant to the PSAAP application, Soleil-X, is given. Soleil-X used the **Legion programming system**, through the Regent programming language, in order to effectively use high performance computing systems with a variety of diverse architectures.

Hypersonics

- (1) Mario Di Renzo, Sergio Pirozzoli, HTR-1.2 solver: [Hypersonic Task-based Research solver version 1.2](#), Computer Physics Communications, Volume 261, 2021.

Abstract: We present an updated version of the open-source hypersonics Task-based Research (HTR) solver for hypersonic aerothermodynamics. The solver, whose first version was presented in Di Renzo et al. (2020), is designed for direct numerical simulation (DNS) of canonical hypersonic flows at high Reynolds numbers in which thermo-chemical effects induced by high temperatures are relevant. The solver relies on high-order spatial discretization on structured meshes and efficient time integrators for stiff systems within the Regent/**Legion software stack**, which makes the code highly portable and scalable in CPU and GPU-based supercomputers. The new version herein presented includes several optimizations and new tools for data analysis, along with novel user option for hybrid skew-symmetric/targeted essentially non-oscillatory numerics, to offer higher computational efficiency and lower numerical dissipation at moderate Mach numbers, inclusion of a new combustion mechanism for methane and oxygen, and new recycling-rescaling inlet boundary conditions targeted to the simulation of fully developed turbulent boundary layers.

Rocket Engines

- (1) Kazuki, Maeda et al. [A task-based parallel framework for ensemble simulations of rocket ignition](#). APS Division of Fluid Dynamics (Fall) 2021.

Abstract: We present an integrated, parallel computational framework for exascale-oriented ensemble simulations of laser-induced ignition in a methane-oxygen rocket combustor. The framework employs the reacting flow solver HTR (Di Renzo et al., Comp. Phys. Comm. 2020). The solver uses the task-based programming model built on the **Legion runtime system** to achieve scalable simulation on supercomputers with heterogeneous architectures. The compressible, multi-species Navier-Stokes equations with finite-rate combustion chemistry are discretized on curvilinear grids using a low-dissipation conservative formulation. Laser-induced ignition is modeled by rapid, intense energy-deposition. This solver is integrated in a continuous development environment to manage a large software development team. The framework also leverages **Legion's** mapper to efficiently perform the execution of ensembles simultaneously on GPUs and CPUs across multiple fidelities to carry out reliability and uncertainty quantification studies. We show verification examples as well as demonstrate the framework through combustor simulations using representative parameters.

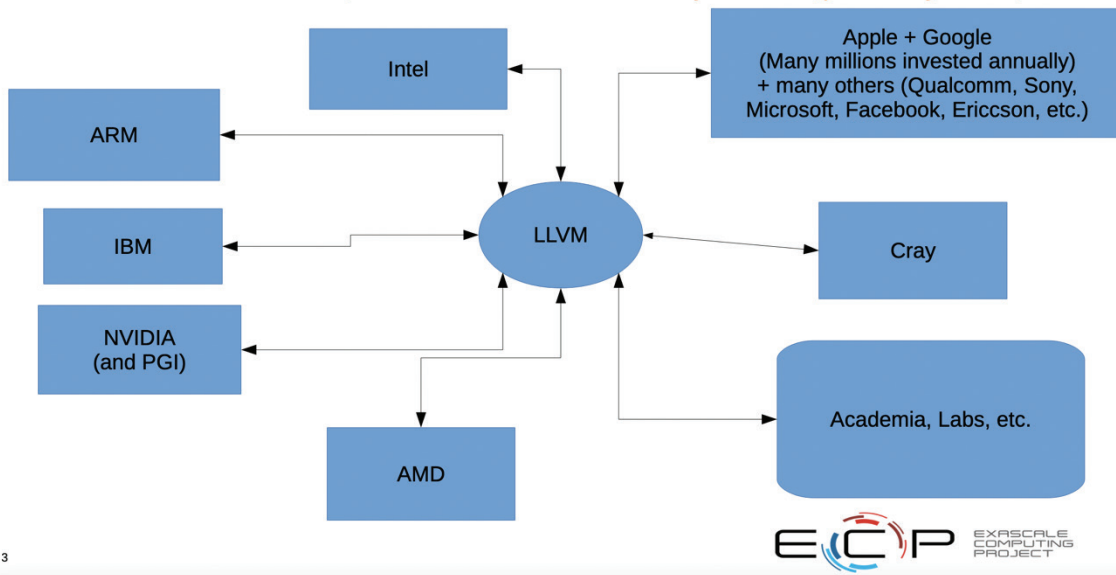
LLVM

LLVM is a collection of modular and reusable compiler and toolchain technologies. LLVM is not an acronym, but the full name of the project. LLVM began at the University of Illinois as a research project by [Chris Lattner](#) and **Vikram Adve**. Its goal was to provide a modern, SSA-based compilation strategy capable of supporting both static and dynamic compilation of arbitrary programming languages. According to the LLVM user group “LLVM has grown into an umbrella project consisting of a number of subprojects, many of which are being used in production by a wide variety of commercial and open source projects as well as being used in academic research.”

// LLVM USER GROUP

LLVM has a large, diverse and active group of users. The LLVM User Group website provides a listing of the primary sub-projects, documentation, a list of universities and commercial entities and others actively involved with LLVM, proceedings from annual meetings and many developer resources. The Department of Energy has made extensive use of LLVM with the Exascale Computing Project (ECP).^{6,7}

A role in exascale? Current/Future HPC vendors are already involved (plus many others)...



3

Current/Future HPC vendors using LLVM

[Click to see slide deck](#)⁸

Internet of Things

- (1) Fernando A. Teixeira, Fernando M.Q. Pereira, Hao-Chi Wong, José M.S. Nogueira, Leonardo B. Oliveira, [SloT: Securing Internet of Things through distributed systems analysis](#), Future Generation Computer Systems, Volume 92, 2019, Pages 1172-1186

Abstract: The Internet of Things (IoT) is increasingly more relevant. This growing importance calls for tools able to provide users with correct, reliable and secure systems. In this paper, we claim that traditional approaches to analyze distributed systems are not expressive enough to address this challenge. As a solution to this problem, we present SloT, a tool to analyze security aspects of distributed IoT programs and thus protect them against buffer overflow attacks. Our key insight is to look at a distributed system as a single body, and not as separate programs that exchange messages. We then can crosscheck information inferred from different nodes. To construct this global view of a distributed system, we introduce a novel algorithm that discovers inter-program links efficiently. Such links let us build an inter-program view, a knowledge that we can thus forward to a traditional buffer overflow static analysis tool. We prove that our algorithm always terminates and it correctly models the semantics of a distributed system. We have implemented our solution on top of the **LLVM compiler**, and have used it to secure five ContikiOS applications against buffer overflow attacks. Our solution produces code as safe as the code secured by more traditional analyses; however, applications instrumented by our solution have less than 6% of runtime and program size overhead on average.

Machine Learning

- (1) Trofin, M. et al. [MLGO: a Machine Learning Guided Compiler Optimizations Framework](#). arXiv January 13, 2021.

Abstract: Leveraging machine-learning (ML) techniques for compiler optimizations has been widely studied and explored in academia. However, the adoption of ML in general-purpose, industry strength compilers has yet to happen. We propose MLGO, a framework for integrating ML techniques systematically in an industrial compiler -- **LLVM**. As a case study, we present the details and results of replacing the heuristics-based inlining-for-size optimization in **LLVM** with machine learned models. To the best of our knowledge, this work is the first full integration of ML in a complex compiler pass in a real-world setting. It is available in the main LLVM repository. We use two different ML algorithms: Policy Gradient and Evolution Strategies, to train the inlining-for-size model, and achieve up to 7% size reduction, when compared to state of the art LLVM -Oz. The same model, trained on one corpus, generalizes well to a diversity of real-world targets, as well as to the same set of targets after months of active development. This property of the trained models is beneficial to deploy ML techniques in real-world settings.

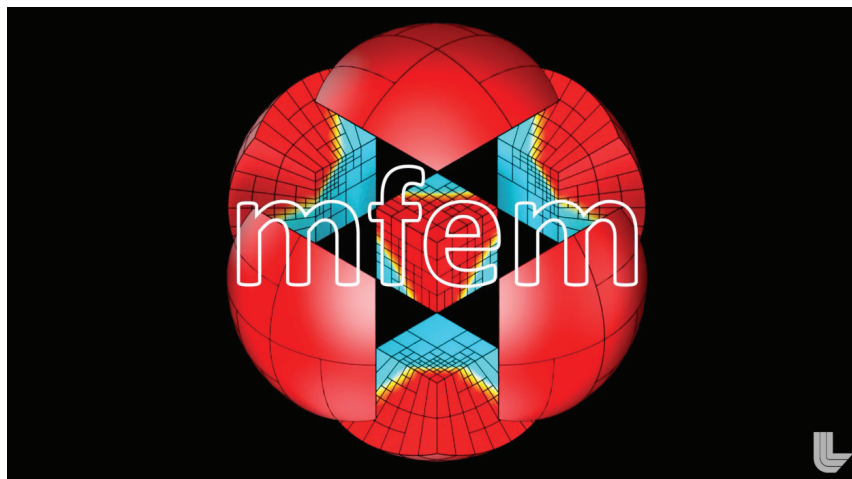
Translation Validation (TV) Systems

- (1) Kasampalis, Theodoros et al. [Language-parametric compiler validation with application to LLVM](#). ASPLO 2021: Proceedings of the 26th ACM International Conference on Architectural Support for Programming Languages and Operating Systems. April 2021

Abstract: We propose a new design for a Translation Validation (TV) system geared towards practical use with modern optimizing compilers, such as LLVM. Unlike existing TV systems, which are custom-tailored for a particular sequence of transformations and a specific, common language for input and output programs, our design clearly separates the transformation-specific components from the rest of the system, and generalizes the transformation-independent components. Specifically, we present Keq, the first program equivalence checker that is parametric to the input and output language semantics and has no dependence on the transformation between the input and output programs. The Keq algorithm is based on a rigorous formalization, namely cut-bisimulation, and is proven correct. We have prototyped a TV system for the Instruction Selection pass of **LLVM**, being able to automatically prove equivalence for translations from **LLVM** IR to the MachineIR used in compiling to x86-64. This transformation uses different input and output languages, and as such has not been previously addressed by the state of the art. An experimental evaluation shows that Keq successfully proves correct the translation of over 90% of 4732 supported functions in GCC from SPEC 2006.

MFEM

MFEM is an open-source C++ library for solving partial differential equations using the finite element method, developed and maintained by researchers at the Lawrence Livermore National Laboratory (LLNL) and the [MFEM](#) open-source community. The MFEM mathematical library enables application scientists to quickly prototype parallel physics application codes based on partial differential equations (PDEs) discretized with high-order finite elements.⁹



What is MFEM?
[Click to see video](#)

MFEM, which stands for Mixed Finite Element Formulation provides high-order mathematical algorithms for large-scale scientific simulations. It can conceptually be viewed as a finite element toolbox that provides the building blocks for developing finite element algorithms in a manner similar to that of MATLAB for linear algebra methods. “The software features many linear and nonlinear solvers, analysis of 2D and 3D finite element spaces, advanced meshing capabilities, and parallel scalability for different hardware architectures including heterogeneous CPU/GPU machines.”¹⁰ MFEM enables high-performance scalable finite element discretization research and application development on a wide variety of platforms, ranging from laptops to supercomputers. MFEM is used in many projects, including [BLAST](#), [Cardioid](#), [VisIt](#), [RF-SciDAC](#), [FASTMath](#), [xSDK](#), and [CEED](#) in the [Exascale Computing Project \(ECP\)](#).¹¹

// MFEM USER GROUP

[MFEM](#) has an active user group. There’s a rich documentation tab and numerous examples of how MFEM has been used in numerous applications. The Community section contains news, videos, seminars and workshops. The MFEM site also contains a Gallery of screenshots of various simulations based on MFEM

Advanced Manufacturing

Advanced Manufacturing technologies aim to support emerging innovations in manufacturing, and include additive manufacturing (AM), bio-inspired manufacturing, quantum manufacturing, modeling and simulation, and others. Researchers at the [Oak Ridge National Laboratory \(ORNL\)](#) and [Lawrence Livermore National Laboratory \(LLNL\)](#) are collaborating on the Exascale Additive Manufacturing project ([ExaAM](#)), which uses exascale simulation to enable the design of AM components with location-specific properties and the acceleration of performance certification. The project has resulted in the creation of several new exascale-ready open-source simulation capabilities for modeling the AM process including polycrystal plasticity (ExaConstit) built on MFEM.¹²

- (1) Hong Wang. An Optimal-Order [Error Estimate for a Family of ELLAM-MFEM Approximations to Porous Medium Flow](#). Society for Industrial and Applied Mathematics. Volume 46, Issue 4

Abstract: Mathematical models used to describe porous medium flow lead to coupled systems of time-dependent nonlinear partial differential equations, which present serious mathematical and numerical difficulties ... The **ELLAM-MFEM** time-stepping procedure, in which an Eulerian–Lagrangian localized adjoint method (ELLAM) is used to solve the transport equation and a mixed finite element method (**MFEM**) is used for the pressure equation, simulates porous medium flow accurately even if large spatial grids and time steps are used. In this paper we prove an optimal-order error estimate for a family of ELLAM-MFEM approximations.

- (2) Atif Kutlu. [A novel mixed finite element formulation based on the refined zigzag theory for the stress analysis of laminated composite plates](#). Composite Structures. Volume 267, 1 July 2021

Abstract: In this study, a Mixed Finite Element Formulation (**MFEM**) for the stress analysis of laminated composite plates within the realm of the Hellinger-Reissner principle and Refined Zigzag Theory is presented. The present approach, MRZT, introduces the stress and moment resultants as well as the shear forces as additional degrees of freedom at each node.”

Energy

A research team from North Carolina State University used MFEM in neutron transport modeling for the design of microreactors, a new class of compact reactor with relatively small electrical output. MFEM satisfies their need for a finite element framework with GPU support and rapid prototyping, and enabled the team to discretize a neutron transport equation with six independent variables in space, direction, and energy. The team is modeling the MARVEL reactor, part of the Exascale Computing Project (ECP), which is planned for construction at Idaho National Laboratory (INL).¹³

Shiraiwa, S et al. [RF wave simulation for cold edge plasmas using the MFEM library](#). EPG Web Conf. Volume 157, 2017

Abstract: A newly developed generic electro-magnetic (EM) simulation tool for modeling RF wave propagation in SOL plasmas is presented. The primary motivation of this development is to extend the domain partitioning approach for incorporating arbitrarily shaped SOL plasmas and antenna to the TORIC core ICRF solver, which was previously demonstrated in the 2D geometry [S. Shiraiwa, et. al., "HISTORIC: extending core ICRF wave simulation to include realistic SOL plasmas", Nucl. Fusion in press], to larger and more complicated simulations by including a 3D realistic antenna and integrating RF rectified sheath potential model."

Shale-Gas Transport

- (1) El-Amin, M.F., Kou, J. & Sun, S. [Numerical Modeling and Simulation of Shale-Gas Transport with Geomechanical Effect](#). *Transp Porous Med* **126**, 779–806 (2019)

Abstract: Throughout this study, we present a dual-continuum model of transport of the natural gas in shale formations. The model includes several physical mechanisms such as diffusion, adsorption and rock stress sensitivity. The slippage has a clear effect in the low-permeability formations which can be described by the apparent permeability. The adsorption mechanism has been modeled by the Langmuir isotherm. The porosity-stress model has been used to describe stress state of the rocks. The thermodynamics deviation factor is calculated using the equation of state of Peng–Robinson. The governing differential system has been solved numerically using the mixed finite element method (**MFEM**). The stability of the MFEM has been investigated theoretically and numerically. A semi-implicit scheme is employed to solve the two coupled pressure equations, while the thermodynamic calculations are conducted explicitly. Moreover, numerical experiments are performed under the corresponding physical parameters of the model. Some represented results are shown in graphs including the rates of production as well as the pressures and the apparent permeability profiles.

OPEN MPI

The Open Message Passing Interface (MPI) Project is a standardized application programming interface (API) used for parallel and/or distributed computing. As an open source implementation, Open MPI offers advantages for system and software vendors, application developers, and computer science researchers.^{14,15} Features already implemented or currently in short-term development for Open MPI include full MPI-3.1 standards conformance; thread safety and concurrency; dynamic process spawning; network and process fault tolerance; support network heterogeneity; single library supports all networks; run-time instrumentation; job scheduler and operating systems support; production quality software; high performance on all platforms; portable and maintainable; tunable by installers and end-users; component-based design; documented APIs; active and responsive mailing list; and open source license based on the Berkeley Source Distribution (BSD) license.¹⁶

Open MPI is developed and maintained by a consortium of academic, research, and industry partners, combining expertise, technologies, and resources throughout the High Performance Computing community.¹⁷

DOE-SUPPORTED COMPUTING TECHNOLOGIES THAT MADE A DIFFERENCE

MPI AND MPICH

RUSTY LUSK

Mathematics and Computer Science Division
Argonne National Laboratory

September 26, 2017

[Click to see slide deck](#)

// OPEN MPI USER GROUP

According to the presentation from Argonne National Laboratory, MPI is not specifically an ASCR project, but rather is funded indirectly. Further, the “most significant contribution from ASCR was the creation of a flexible research environment from which MPI emerged, followed by continued support for MPI-related research and for MPICH, its most applicable product, as well as the MPI standardization effort in general.”¹⁸ The Open MPI project has many members, contributors, and partners. “Contributors” provide resources to the project such as code, testing, and hosting services, among others. “Members” are contributors who have voting rights, which entitles them to help determine the direction of the project, participate in release processes, etc. “Partners” provide services to the Open MPI project.¹⁹

The user group offers downloads, documentation, source code access, bug tracking, regression testing, and version information, as well as information on sub-projects which currently include:

- **Hardware Locality:** The Portable Hardware Locality (hwloc) software package aims to help applications with gathering information about increasingly complex parallel computing platforms in order to optimize use and efficiency. It does so by providing a portable abstraction across the hierarchical topology of modern architectures while also gathering various system attributes as well as the locality of I/O devices.²⁰
- **MPI Testing Tool:** The MPI Testing Tool (MTT) is a fully-automated general infrastructure for testing MPI implementations and running performance benchmarks. MTT can potentially be distributed across multiple clusters/environments/organizations, from which it gathers all results back to a central database for analysis. MTT tests whether the MPI can be successfully installed, as well as whether MPI test programs can be compiled and linked against the MPI installation, and if those test programs run successfully with valid performance results.²¹
- **Open Tool for Parameter Optimization (OTPO):** OTPO is a new framework designed to aid in the optimization of the MCA parameters to determine the best set for a given platform through systematically testing a large number of combinations of Open MPI’s run-time tunable parameters based on a user input file.²²

Machine Learning

- (1) S. Alemany, J. Nucciarone and N. Pissinou, "Jespipe: A Plugin-Based, Open MPI Framework for Adversarial Machine Learning Analysis," *2021 IEEE International Conference on Big Data (Big Data)*, 2021, pp. 3663-3670, doi: 10.1109/BigData52589.2021.9671385.

Abstract: Research is increasingly showing the tremendous vulnerability in machine learning models to seemingly undetectable adversarial inputs. One of the current limitations in adversarial machine learning research is the incredibly time-consuming testing of novel defenses against various attacks and across multiple datasets, even with high computing power. To address this limitation, we have developed Jespipe as a new plugin-based, parallel-by-design **Open MPI** framework that aids in evaluating the robustness of machine learning models. The plugin-based nature of this framework enables researchers to specify any pre-training data manipulations, machine learning models, adversarial models, and analysis or visualization metrics with their input Python files. Because this framework is plugin-based, a researcher can easily incorporate model implementations using popular deep learning libraries such as PyTorch, Keras, TensorFlow, Theano, or MXNet, or adversarial robustness tools such as IBM's Adversarial Robustness Toolbox or Foolbox. The parallelized nature of this framework also enables researchers to evaluate various learning or attack models with multiple datasets simultaneously by specifying all the models and datasets they would like to test with our XML control file template. Overall, Jespipe shows promising results by reducing latency in adversarial machine learning algorithm development and testing compared to traditional Jupyter notebook workflows.

Satellite imagery

A. Goyal, Z. Li and H. Kimm, "[Comparative Study on Edge Detection Algorithms Using OpenACC and OpenMPI on Multicore Systems](#)," *2017 IEEE 11th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoc)*, 2017, pp. 67-74, doi: 10.1109/MCSoc.2017.16.

Abstract: In this paper, we present a comparative study on parallel edge detection algorithms upon high-resolution satellite images, implemented on **OpenACC, Hybrid OpenMP/MPI, OpenMP, and MPI models** on which the Sobel, Prewitt and Canny algorithms were developed using C++ language and OpenCV. The performance of these computing models were measured in terms of speedup and execution time by implementing the edge detection algorithms using various sized images and programming constructs. The program implementations using OpenACC display the largest speedup in CPU time, which is followed by the Hybrid **OpenMP/MPI model**. The parallel detection algorithms using OpenACC obtain the greatest speedup of around 6.5 over OpenMP model. The parallel Sobel and Prewitt algorithms are relatively 2 times faster than the Canny in all respects."

MPICH

MPICH is a high-performance, freely available, widely portable implementation of the Message Passing Interface (MPI) standard. Its purpose is to provide an MPI implementation that efficiently support different computation and communication platforms including commodity clusters, high-speed networks and proprietary high-end computing systems, while enabling next generation research in MPI through an easily extended modular framework for other derived implementations.

Originally developed during the MPI standards process as a means of providing feedback to the MPI Forum on implementation and usability, MPICH has gone through several implementation iterations was primarily developed at Argonne National Laboratory (ANL) with contributions from external collaborators.²³ and is now distributed as a source and has been tested on several platforms.²⁴

// MPICH USER GROUP

The MPICH User group [website](#) provides downloads, documentation and support.

Cloud services

- (1) Choi, HyeonSeong; Kim, Youngrang; Lee, Jaehwan; Kim, Yoonhee. [Empirical Performance Evaluation of Communication Libraries for Multi-GPU based Distributed Deep Learning in a Container Environment](#). Korean Society for Internet Information, 2021.

Abstract: Recently, most cloud services use Docker container environment to provide their services. However, there are no researches to evaluate the performance of communication libraries for multi-GPU based distributed deep learning in a Docker container environment. In this paper, we propose an efficient communication architecture for multi-GPU based deep learning in a Docker container environment by evaluating the performances of various communication libraries. We compare the performances of the parameter server architecture and the All-reduce architecture, which are typical distributed deep learning architectures. Further, we analyze the performances of two separate multi-GPU resource allocation policies - allocating a single GPU to each Docker container and allocating multiple GPUs to each Docker container. We also experiment with the scalability of collective communication by increasing the number of GPUs from one to four. Through experiments, we compare OpenMPI and MPICH, which are representative open source MPI libraries, and NCCL, which is NVIDIA's collective communication library for the multi-GPU setting. In the parameter server architecture, we show that using CUDA-aware OpenMPI with multi-GPU per Docker container environment reduces communication latency by up to 75%. Also, we show that using NCCL in All-reduce architecture reduces communication latency by up to 93% compared to other libraries.

Generic

- (1) William Gropp, Rajeev Thakur, Pavan Balaji, [Translational research in the MPICH project](#), Journal of Computational Science, Volume 52, 2021,

Abstract: The **MPICH** project is an example of translational research in computer science before that term was well known or even coined. The project began in 1992 as an effort to develop a portable, high-performance implementation of the emerging Message-Passing Interface (MPI) Standard. It has enabled the widespread adoption of MPI as a way to write scalable parallel applications on systems of all sizes including upcoming exascale [supercomputers](#). In this paper, we describe how the translational research process was used in MPICH, how that led to its success, the challenges encountered and lessons learned, and how the process could be applied to other similar projects.

TRILINOS

A scientific and mathematical library developed at Sandia National Laboratories, [Trilinos](#) is a collection of open-source reusable scientific software libraries, referred to as packages, intended to be used as building blocks for the development of scientific applications. It provides an object-oriented software framework for the solution of large-scale, complex multi-physics engineering and scientific problems on new and emerging high-performance computing (HPC) architectures. Trilinos is known in particular for its linear solvers, non-linear solvers, transient solvers, optimization solvers, and uncertainty quantification (UQ) solvers. Each Trilinos package is a self-contained, independent piece of software with its own set of requirements, its own development team, and group of users.²⁵ The list of packages within Trilinos can be found [here](#). A frequently cited package is Kokkos, mentioned in the figure below.

This section provides information on the [Trilinos](#) project, a community effort to develop scientific and mathematical libraries that enable the solution of large-scale multi-physics, multi-scale, and related modeling and simulation problems. This section also includes examples/use cases to illustrate applications in which it is being used.

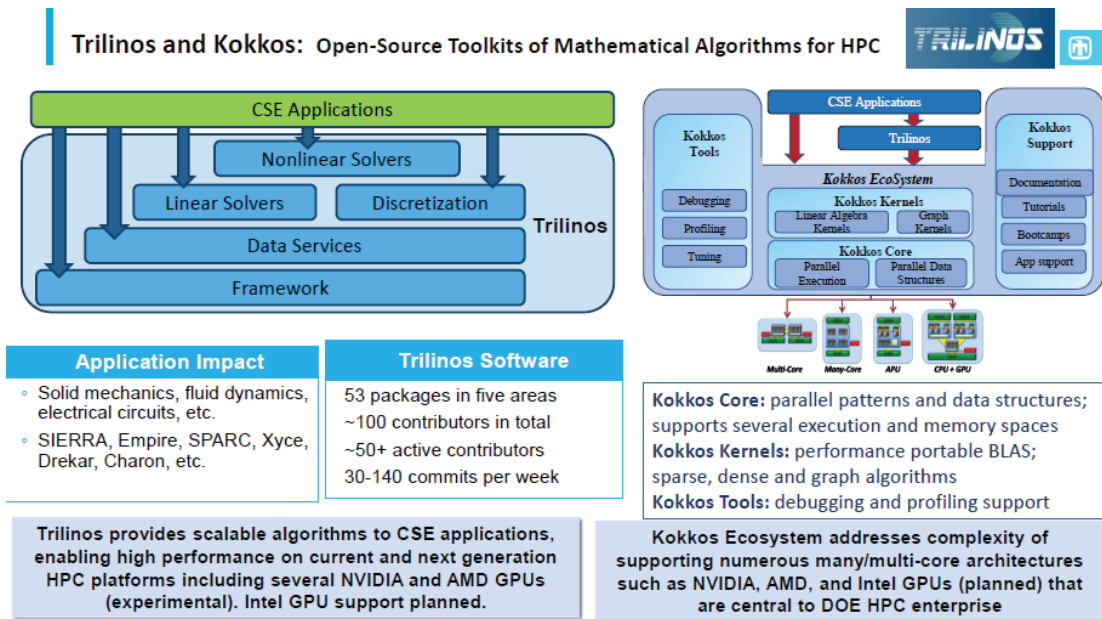


Figure 1: Trilinos and Kokkos
 Source: [Exascale Computing Project, 2021](#)

// TRILINOS USER GROUP

[Trilinos](#) has a very active user group. It is a community-driven project, with developers at several organizations. Sandia National Laboratories is the project lead, with other major participants including Oak Ridge National Laboratory and Technical University of Berlin. This website features tutorials; documentation, and access to an active community.

// APPLICATION AREAS

Energy

One key challenge hindering the wide-scale deployment of wind energy in the utility grid without subsidies, is the capability to predict and minimize plant-level energy losses, currently estimated at 20% in relatively flat areas, and significantly higher in regions of complex terrain.

Trilinos has been utilized for modeling wind plant performance in the ECP's ExaWind project to advance the fundamental understanding of the flow physics governing whole wind plant performance, including wake formation, complex-terrain impacts, and turbine-turbine interaction effects. One of the primary applications codes in the ExaWind environment is Nalu-Wind, an unstructured-grid, acoustically incompressible computational fluid dynamics (CFD) code. Nalu-Wind contains the infrastructure for unstructured-grid discretization of the underlying models, and heavily uses the Trilinos Sierra Toolkit, which provides an unstructured mesh, in-memory, parallel-distributed database. Predictive, physics-based, high-fidelity computational models validated with targeted experiments will drive innovation in the blade, turbine, and wind plant design processes by providing a validated "ground truth" foundation for new turbine design models, wind plant siting, operational controls, and the reliable integration of wind energy into the grid.^{26, 27}

Atmospheric Research

Spotz, William et al. [Aeras: A Next Generation Global Atmosphere Model](#). Sandia Abstract: Sandia National Laboratories is developing a new global atmosphere model named Aeras that is performance portable and supports the quantification of uncertainties. These next-generation capabilities are enabled by building Aeras on top of Albany, a code base that supports the rapid development of scientific application codes while leveraging Sandia's foundational mathematics and computer science packages in Trilinos and Dakota. Embedded uncertainty quantification (UQ) is an original design capability of Albany, and performance portability is a recent upgrade. Other required features, such as shell-type elements, spectral elements, efficient explicit and semi-implicit time-stepping, transient sensitivity analysis, and concurrent ensembles, were not components of Albany as the project began, and have been (or are being) added by the Aeras team. We present early UQ and performance portability results for the shallow water equations.

Earth Systems

- (1) Jerry Watkins, Max Carlson, Kyle Shan, Irina Tezaur, Mauro Perego, Luca Bertagna, Carolyn Kao, Matthew J. Hoffman, Stephen F. Price. [Performance portable ice-sheet modeling with MALI](#). Cornell University, 2022.

Abstract: High resolution simulations of polar ice-sheets play a crucial role in the ongoing effort to develop more accurate and reliable Earth-system models for probabilistic sea-level projections. These simulations often require a massive amount of memory and computation from large supercomputing clusters to provide sufficient accuracy and resolution. The latest exascale machines poised to come online contain a diverse set of computing architectures. In an effort to avoid architecture specific programming and maintain productivity across platforms, the ice-sheet modeling code known as MALI uses high level abstractions to integrate Trilinos libraries and the Kokkos programming model for performance portable code across a variety of different architectures. In this paper, we analyze the performance portable features of MALI via a performance analysis on current CPU-based and GPU-based supercomputers. The analysis highlights performance portable improvements made in finite element assembly and multigrid preconditioning within MALI with speedups between 1.26-1.82x across CPU and GPU architectures but also identifies the need to further improve performance in software coupling and preconditioning on GPUs. We also perform a weak scalability study and show that simulations on GPU-based machines perform 1.24-1.92x faster when utilizing the GPUs. The best performance is found in finite element assembly which achieved a speedup of up to 8.65x and a weak scaling efficiency of 82.9% with GPUs. We additionally describe an automated performance testing framework developed for this code base using a changepoint detection method. The framework is used to make actionable decisions about performance within MALI. We provide several concrete examples of scenarios in which

the framework has identified performance regressions, improvements, and algorithm differences over the course of two years of development.

- (2) Alexander Heinlein and Jonas Thies. [Project: Overlapping Schwarz Domain Decomposition Methods for Implicit Ocean Models](#). Delft University of Technology, 2021.

Abstract: Earth system models (ESMs) are one of the key players in making projections for the future development of the Earth’s climate. An important component of these models are ocean models that simulate the flow of sea water based on physical conservation laws (conservation of mass, momentum and energy). These laws are formulated as partial differential equations known as the ‘primitive equations’: they describe fluid flow in a rotating, spherical domain under certain assumptions (e.g. an incompressible fluid). Practically all ocean models in use today rely on explicit time integration schemes. For increasing mesh resolutions, time step constraints due to the Courant–Friedrich–Levy (CFL) condition become increasingly strict. Therefore, for simulating long time spans or to compute a (quasi) stationary state, it may become desirable to use implicit time integration schemes, such that efficient numerical solvers for large linear equations systems are needed. Possible applications are scientific discovery (e.g. when simulating the climate of past times in paleo-climate research) or computing a starting configuration for virtual experiments (‘spin-up problem’). In the past, we have shown that efficient simulations with implicit time stepping without restrictions on the step size can be performed using tailored preconditioning techniques and high performance computing (HPC). Here, nested iterative methods were employed to split the problem into simpler linear systems. This leads to a complex convergence behavior and large overall iteration counts. The goal of this project is to take a new route and employ a fully coupled preconditioning approach based on monolithic overlapping Schwarz domain decomposition methods using the FROSch (Fast and Robust Overlapping Schwarz) solver framework, which is part of **Trilinos**. As the basis for the simulations, the **Trilinos-based** implementation of the implicit Earth system model of intermediate complexity (I-EMIC)¹ is used; see [1, 5] for the model equations. The project builds on the current state of the art in implicit Earth system modeling and solver technology. It deals with an open research question with potentially significant impact. Depending on the findings and the candidate’s enthusiasm, a subsequent publication of the results in a peer-reviewed journal is an option. During the project, a direct collaboration with climate scientists who developed the I-EMIC model at the Institute for Marine and Atmospheric Modeling at Utrecht University is possible and is also aspired.

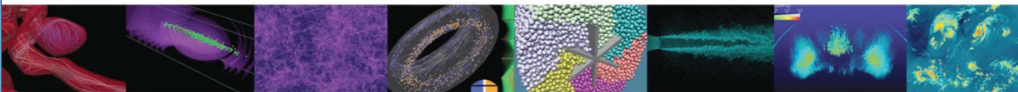
PETSc

PORTABLE, EXTENSIBLE
TOOLKIT FOR SCIENTIFIC
COMPUTATION

Pronounced PET-see, the Portable, Extensible Toolkit for Scientific Computation (PETSc) is a suite of data structures and routines developed by Argonne National Laboratory (ANL) Mathematics and Computer Science (MCS) Division. PETSc is used for “the scalable (parallel) solution of scientific applications modeled by partial differential equations.”²⁸ The basic mission of Argonne’s Mathematics and Computer Science (MCS) Division has long been to accelerate scientific productivity by offering intellectual and technical leadership in computing sciences. As far back as the ‘70s, Argonne has launched a series of software engineering projects that resulted in their release. Argonne MCS researchers continue this tradition of innovation, present day, with an added emphasis on portability and scalability.²⁹

ATPESC 2015

ARGONNE TRAINING PROGRAM ON EXTREME-SCALE COMPUTING



PETSc: Portable, Extensible Toolkit for
Scientific Computing
Barry Smith, Argonne National Laboratory

Argonne Leadership Computing Facility

[Click to see video](#)

A major thrust of Argonne’s MCS Division is “applied mathematics and the incorporation of new numerical methods into portable, high-performance, open source software. MCS computer scientists design robust optimization algorithms, multiscale solvers for linear and nonlinear systems, and automatic differentiation techniques for sensitivity analysis. The new techniques are then incorporated into robust numerical toolkits for use by application scientists in solving large-scale problems. For example, **PETSc** (Portable, Extensible Toolkit for Scientific Computation) is widely used for applications including transonic flow, modeling vortex dynamics in high-temperature superconductors, parallelization of a 3D magnetostatics code, and study of compressible flows at low and transonic Mach number.”³⁰

Regarding its reach and popularity, “It [PETSc] is the world’s most widely used parallel numerical software library for partial differential equations and sparse matrix computations.”³¹

// PETSc USER GROUP

The [PETSc User Group](#) offers introductory information, an FAQ compilation, download and installation guidelines, documentation, tutorials, information on developers/documentation and more.

PETSc has been used for modeling “in all” of the following areas (listed alphabetically): Acoustics, Aerodynamics, Air Pollution, Arterial Flow, Bone Fractures, Brain Surgery, Cancer Surgery, Cancer Treatment, Carbon Sequestration, Cardiology, Cells, CFD, Combustion, Concrete, Corrosion, Data Mining, Dentistry, Earth Quakes, Economics, Esophagus, Fission, Fusion, Glaciers, Ground Water Flow, Linguistics, Mantel Convection, Magnetic Films, Material Science, Medical Imaging, Ocean Dynamics, Oil Recovery, PageRank, Polymer Injection Molding, Polymeric Membranes, Quantum computing, Seismology, Semiconductors, Rockets, Relativity, Surface Water Flow.³²

Artificial Tissues

- (1) Cristea, Artur et al. [Lattice Boltzmann Simulations of Morphogenesis in Artificial Tissues](#). Romanian Journal of Physics. December 30, 2020.

Abstract: Understanding the principles of morphogenesis is indispensable for developing effective strategies to build living tissues in the laboratory. Tissue fusion is essential in bioprinting, an emergent technique based on computer-controlled delivery of multicellular building blocks and supportive hydrogels. The objective of this work was to predict the time course of the fusion of multicellular systems. Computational methods, including Lattice Boltzmann models, proved to be able to simulate in vitro morphogenesis. Here, a Hermite Lattice Boltzmann (HLB) model was used to simulate the sidewise fusion of contiguous cylinders made of cohesive cells in a hydrogel. Fusion eventually gave rise to a tubular construct, in qualitative agreement with 3D tissue printing experiments. Also, the HLB model accounted for the experimentally observed spontaneous rounding of irregular tissue fragments. Our numerical code was based on parallel computing, implemented using the Compute Unified Device Architecture (CUDA) libraries on nVidia M2090 Graphics Processing Unit blades, on lattices of up to 4096×4096 nodes. Importantly, the CUDA implementation was about 20 times faster than the parallel computation code written using the **Portable, Extensible Toolkit for Scientific Computation (PETSc)**, running on a cluster of 32 central processing units. In conclusion, the CUDA version of the HLB model can be employed to simulate the post-printing evolution of bioprinted tissue constructs.

Geophysics

- (1) Vaclav Hapla et al. [Fully Parallel Mesh I/O Using PETSc DMplex with an Application to Waveform Modeling](#). SIAM Journal on Scientific Computing, v43, Issue 2, 2021.

Abstract: Large-scale PDE simulations using high-order finite-element methods on unstructured meshes are an indispensable tool in science and engineering. The widely used open-source **PETSc** library offers an efficient representation of generic unstructured meshes within its DMplex module. This paper details our recent implementation of parallel mesh reading and topological interpolation (computation of edges and faces from a cell-vertex mesh) into DMplex. We apply these developments to seismic wave propagation scenarios on Mars as an example application. The principal motivation is to overcome single-node memory limits and reach mesh sizes which were impossible before. Moreover, we demonstrate that scalability of I/O and topological interpolation goes beyond 12,000 cores, and memory-imposed limits on mesh size vanish.³³

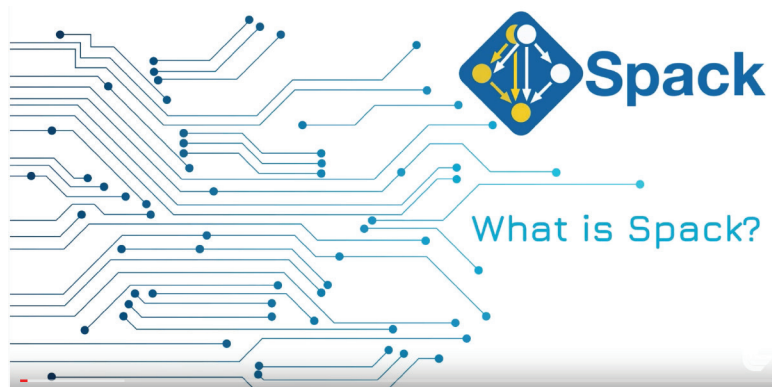
Wind Energy: Layout Optimization

- (1) Jeffrey Allen et al. [Progress on Optimizing Wind Farms and Rotor Designs Using Adjoints](#). 2021

Abstract: Modern wind plants are increasingly tasked with multiple performance objectives. In addition to designing plants that maximize power output and minimize the levelized cost of energy (LCOE), the design and operation of wind plants is increasingly influenced by challenges regarding grid integration of variable generation renewables. This places a growing emphasis on making wind plants more controllable and predictable. WindSE is a Reynolds-averaged Navier-Stokes (RANS) model designed around analytical gradient and adjoint methods, with the ability to capture terrain-induced effects, as shown in Figure 1. The recent addition of an unsteady solver with an actuator line method (ALM) and ongoing work to enable massively parallel optimizations gives it a unique niche to explore coupled plant-level controls and design problems. This code is an open source python package built on the FEniCS framework that utilizes fast, parallel **PETSc** solvers to model fluid flow throughout wind-farm scale domains. Two recent studies performed using WindSE demonstrate the capability to optimize under a wide variety of flow conditions and objective functions. In the first, we present an optimization focused on modifying the layout of a wind farm with a fixed number of turbines for maximum total power output [1]. This study highlights the ability to quickly perform simulations using the steady Navier-Stokes solver combined with rotors represented as actuator disks while also stressing the importance of capturing terrain-induced effects. Gradient-based optimization using the RANS equations is viable due to the inclusion of efficiently computed adjoint derivatives. We interpret the physical results of the optimal layout and also discuss the computational cost of scaling to larger problems. In the second study, we present the capabilities of the unsteady Navier-Stokes solver, where rotor-blade profiles represented by actuator lines are optimized to enhance wake steering effects and overall power production [2]. We quantify the wind plant performance gains obtained from this type of simultaneous control co-design optimization as compared to optimizing the blade design and yaw independently. Figure 2 shows the differences between a baseline two-turbine system and an optimized system where we fine-tune the blade chord profile. Results and challenges from each study are quickly summarized and used to motivate the current development efforts within WindSE. Current and future work is focused on enabling higher-resolution studies with more degrees of freedom through parallelization of both the simulation and optimization algorithms. We present benchmarking results to show that WindSE performs well in both weak- and strong-scaling tests and further demonstrate that the optimizer obtains the same convergence rates in both shared- and distributed-memory environments. Using larger wind farms, we can study deep-array effects within an optimization context, allowing the use of objective functions that have been previously unstudied. As an example, we present ongoing work on a blockage metric which characterizes the loss of available kinetic energy due to wake effects from multiple upstream turbines.³⁴

SPACK

Spack (Supercomputer PACKage manager) first appeared as a prototype package manager in 2013, developed and named by Lawrence Livermore National Laboratory's Todd Gamblin. Six years later, in 2019 Spack received an [R&D 100](#) award in the Software/Services category and also was an R&D Special Recognition medalist in the Market Disruptor–Services category.³⁵ Spack was released as an open-source project in 2014, and its developer Gamblin presented it at the Supercomputing 2015 (SC15) conference.



[Click here to navigate to video](#)

Spack is the software deployment tool of the Exascale Computing Project ([ECP](#)), which is a joint effort involving the DOE Office of Science and the NNSA (National Nuclear Security Administration) that brings together several national labs. Combining forces, these labs are tasked with addressing the hardware, software, and application challenges native to the DOE/NNSA's scientific and national security missions.

Its significance involves large scale supercomputing simulations that contain millions of lines of code and have to rely on hundreds of external software libraries (or packages.) These “packages” can be problematic in these scenarios since “Users of the same high performance computing (HPC) system frequently need different versions and configurations of these packages to test the performance and compatibility of their code.”³⁶ Consequently, packaging tools are critical in HPC environments, yet traditional package managers are restrictive since they are unable to manage simultaneous installations of multiple versions and configurations, the outcome of which is that users, developers, and HPC support staff have to spend many hours building codes and libraries by hand.³⁷

// SPACK USER GROUP

The SPACK [website](#) contains many useful tools for those who want to use SPACK. GitHub, support documentation and discussion threads can be readily accessed from this site.

Atmospheric Research

- (1) Martin, Randall et al. [Improved Advection, Resolution, Performance, and Community Access in the New Generation \(Version 13\) of the High Performance GEOS-Chem Global Atmospheric Chemistry Model \(GCHP\)](#). European Geosciences Union, 2022.

We describe a new generation of the high-performance GEOS-Chem (GCHP) global model of atmospheric composition developed as part of the GEOS-Chem version 13 series. GEOS-Chem is an open-source grid-independent model that can be used online within a meteorological simulation or off-line using archived meteorological data. GCHP is an offline implementation of GEOS-Chem driven by NASA Goddard Earth Observing System (GEOS) meteorological data for massively parallel simulations. Version 13 offers transformational advances in GCHP for ease of use, computational performance, versatility, resolution, and accuracy. Specific improvements include (a) stretched-grid capability for higher resolution in user-selected regions, (b) easier build with a build system generator (CMake) and a **package manager (Spack)**, (c) software containers to enable immediate model download and configuration on local computing clusters, (d) better parallelization to enable simulation on thousands of cores, (e) multi-node cloud capability, and (f) more accurate transport with new native cubed-sphere GEOS meteorological archives including air mass fluxes at hourly temporal resolution with spatial resolution up to C720 (~12 km). The C720 data are now part of the operational GEOS Forward Processing (GEOS-FP) output stream, and a C180 (~50 km) consistent archive for 1998–present is now being generated as part of a new GEOS-IT data stream. Both of these data streams are continuously being archived by the GEOS-Chem Support Team for access by GCHP users. Directly using horizontal air mass fluxes rather than inferring from wind data significantly reduces global mean error in calculated surface pressure and vertical advection.

Scientific Software Package Management in Research Universities with Diverse User Bases

- (1) Qiyang Hu and Shao-Ching Huang [UCLA]. [A Hybrid Approach to Scientific Software Package Management on a HPC Cluster](#). SC20, 2020.

Abstract: We [both authors are affiliated with UCLA] present a practical approach for managing the scientific packages in the HPC cluster environment of a research university environment that has a diverse user base. The primary goal is to minimize the HPC operational team’s burden of installing and maintaining a large number of software packages to support the broadband spectrum of the university’s computational research. We propose a hybridizing management method that can harness the power of modern software management frameworks and the flexibility of in-house, or “manual”, installation sources, and at the same time present a coherent view to the end users. The **Spack** [1] framework was used in this work. Our hybrid approach is applicable to using other framework tools. The manipulation of Spack-generated environment module files and the typical workflow are illustrated and discussed based on our use case.³⁸

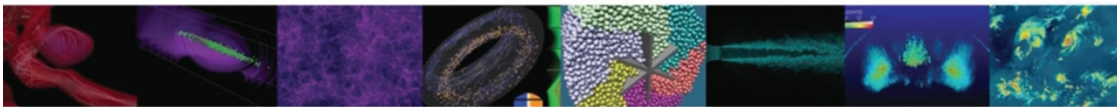
STRUMPACK

The Structured Matrix Package (STRUMPACK) is open-source software, started in 2014 and was developed by Lawrence Berkeley National Laboratory researchers. It has been supported by the FASTMath SciDAC Institute funded by the Department of Energy and by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration.

STRUMPACK is a portable software library that “provides linear algebra routines and linear system solvers for sparse and dense rank-structured linear systems that runs well on many-core processors and GPUs. In short, STRUMPACK focuses on efficiently performing dense linear algebra on relatively small submatrices in the global matrix. Portable programming paradigms that can run well on both CPUs and GPUs are crucial to the efficient use of modern large-scale compute platforms. The STRUMPACK library is closely related to the SuperLU library, both of which provide direct solvers for sparse linear systems. The former targets symmetric-patterned sparse linear systems, whereas the latter targets nonsymmetric linear systems.”³⁹

ATPESC 2016

ARGONNE TRAINING PROGRAM ON EXTREME-SCALE COMPUTING



SuperLU and STRUMPACK
Sparse Direct Solver and Preconditioner

X. Sherry Li,
Lawrence Berkeley National Laboratory

Pause (k)

Argonne Leadership Computing Facility

[Click to see video](#)

// STRUMPACK USER GROUP

[The STRUMPACK User](#) Group provides downloads, documentation, publications and help.

Engineering Problems

- (1) Anton, J. et al. [Using low-rank approximation techniques for engineering problems](#). Livermore Software Technology Corporation

Abstract: Matrices that appear in the boundary element methods and finite element methods are often structured (or low-rank, or data-sparse) [3, 5]. This means that they exhibit rank-deficient blocks, typically the blocks corresponding to far range interactions in the physical space. Identifying and compressing these low-rank blocks, e.g., using SVD or a rank-revealing factorization, is the key to reducing the storage and computational requirements of many matrix operations, such as performing matrix-vector products, computing eigenvalues, and solving linear systems. In this talk, we will focus on the latter, for both dense and sparse matrices. For sparse matrices, the low-rank property is usually not found in the input matrix but at intermediate steps of the factorization algorithms used to solve linear systems. Many different techniques, referred to as low-rank representations, have been proposed in the literature. Among others, the Hierarchically Semi-Separable (HSS) matrices [8] and Block Low-Rank representations [2] have been widely studied and have recently been implemented in parallel solvers. However very few comparison results can be found in the literature; usually they are restricted to model problems, or to comparing a single low-rank algorithm against a non-low-rank one. Our goal is to compare the performance of the HSS and BLR approaches for dense and sparse matrices arising from engineering applications. LS-DYNA [6] is a highly advanced nonlinear finite element code. It allows implicit and explicit simulations of multi-physics problems, such as mechanics, fluid dynamics, acoustics, electromagnetism. . . It is widely used by the automotive, aerospace, and construction industries among others. The matrices that we will consider for this presentation all arise from implicit simulations performed with LS-DYNA for real world applications. We will compare the HSS and BLR techniques using multiple high-performance implementations. The HSS-based solver we will use is the **STRUMPACK** code [4] [7], that can be used as a preconditioner or as a direct solver for both dense and sparse problems. For BLR we will use MUMPS [1] [2], a sparse direct solver that has recently gained Block Low-Rank features

Machine Learning [Kernel Methods]

- (1) Elizaveta Rebrova et al. [A Study of Clustering Techniques and Hierarchical Matrix Formats for Kernel Ridge Regression](#). 2018 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), 2018.

Abstract: We present memory-efficient and scalable algorithms for kernel methods used in machine learning. Using hierarchical matrix approximations for the kernel matrix the memory requirements, the number of floating point operations, and the execution time are drastically reduced compared to standard dense linear algebra routines. We consider both the general H matrix hierarchical format as well as Hierarchically Semi-Separable (HSS) matrices. Furthermore, we investigate the impact of several preprocessing and clustering

techniques on the hierarchical matrix compression. Effective clustering of the input leads to a ten-fold increase in efficiency of the compression. The algorithms are implemented using the **STRUMPACK** solver library. These results confirm that – with correct tuning of the hyperparameters – classification using kernel ridge regression with the compressed matrix does not lose prediction accuracy compared to the exact – not compressed – kernel matrix and that our approach can be extended to $O(1M)$ datasets, for which computation with the full kernel matrix becomes prohibitively expensive. We present numerical experiments in a distributed memory environment up to 1,024 processors of the NERSC’s Cori supercomputer using well-known datasets to the machine learning community that range from dimension 8 up to 784.⁴⁰ Published in: 2018 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW).⁴¹

Multi-physics and Multi-scale Simulations

- (1) Hartwig Antz et al. [Preparing Sparse Solvers for Exascale Computing](https://royalsocietypublishing.org/doi/full/10.1098/rsta.2019.0053). Philosophical Transactions of the Royal Society, A: Mathematical, Physical and Engineering Sciences, March 6, 2020 | <https://royalsocietypublishing.org/doi/full/10.1098/rsta.2019.0053>

Abstract: Sparse solvers provide essential functionality for a wide variety of scientific applications. Highly parallel sparse solvers are essential for continuing advances in high-fidelity, multi-physics and multi-scale simulations, especially as we target exascale platforms. This paper describes the challenges, strategies and progress of the US Department of Energy Exascale Computing project towards providing sparse solvers for exascale computing platforms. We address the demands of systems with thousands of high-performance node devices where exposing concurrency, hiding latency and creating alternative algorithms become essential. The efforts described here are works in progress, highlighting current success and upcoming challenges.

This article is part of a discussion meeting issue ‘Numerical algorithms for high-performance computational science.’

Libraries overview: In this paper, we present the efforts of five solver development teams to provide capabilities needed by the high-performance computing (HPC) community, especially the ECP. Two of the teams, PETSc (led from Argonne National Laboratory) and Trilinos (led from Sandia National Laboratories), provide specific solver components and infrastructure for coordinating component use. Ginkgo (led from the University of Tennessee), hypre (led from Lawrence Livermore National Laboratory) and **SuperLU/STRUMPACK** (led from Lawrence Berkeley National Laboratory) provide key solver capabilities that can be used independently or as part of the PETSc and Trilinos infrastructures.

The hypre package (§3) provides distributed memory parallel sparse iterative solvers and is particularly well known for its multigrid preconditioners. PETSc

(§4) and Trilinos (§6) provide comprehensive solver capabilities widely used in the community. SuperLU (§5) provides sparse parallel direct solvers. All of these projects have been available to the HPC community for many years and have evolved over time to support a variety of computing platforms. Ginkgo (§2), **STRUMPACK (also in §5)** and KokkosKernels (also in §6) are newer efforts that have emerged to address the particular challenges that modern node architectures present ... All of the solvers discussed in this paper are freely available as open-source software.⁴²

ENDNOTES

- 1 [Introduction to HDF5](#)
- 2 [About Us. HFD Group](#)
- 3 Aiken, Alex and McCormick, Pat. [The Legion Programming System](#)
- 4 [Legion Programming System](#). Los Alamos National Laboratory
- 5 [Legion](#)
- 6 Finkel, Hal et al [DOE Proxy Apps-Clang/LLVM vs. the World!](#) EuroLLVM, 2018
- 7 Finkel Hal et al. [What is LLVM? And a Status Update](#)
- 8 IBID
- 9 LLNL Computing: MFEM: Scalable Finite Element Discretization Library: <https://computing.llnl.gov/projects/mfem-scalable-finite-element-discretization-library>
- 10 “MFEM team hosts first community workshop”, Lawrence Livermore National Lab news release (Nov. 12, 2021): <https://computing.llnl.gov/about/newsroom/mfem-team-hosts-first-community-workshop>
- 11 MFEM website: <https://mfem.org/>
- 12 Shiraiwa, S et al. [RF wave simulation for cold edge plasmas using the MFEM library](#). EPG Web Conf. Volume 157, 2017
- 13 MFEM team hosts first community workshop”, Lawrence Livermore National Lab news release (Nov. 12, 2021): <https://computing.llnl.gov/about/newsroom/mfem-team-hosts-first-community-workshop>
- 14 “What is Open MPI?” Open MPI, <https://www.open-mpi.org/faq/?category=general#what-is-mpi>
- 15 “A High Performance Message Passing Interface,” [Open MPI](#), <https://www.open-mpi.org/>
- 16 “A High Performance Message Passing Interface,” [Open MPI](#), <https://www.open-mpi.org/>
- 17 “A High Performance Message Passing Interface,” [Open MPI](#), <https://www.open-mpi.org/>
- 18 “DOE-SUPPORTED COMPUTING TECHNOLOGIES THAT MADE A DIFFERENCE MPI AND MPICH,” Argonne National Laboratory, 9/26/17 https://science.osti.gov/-/media/ascr/ascac/pdf/meetings/201709/Rusty_Lusk_ASCAC-201709.pdf
- 19 “The Open MPI Development Team,” Open MPI, <https://www.open-mpi.org/about/members/>
- 20 “hwloc,” Open MPI, <https://www.open-mpi.org/projects/hwloc/>
- 21 “MPI Testing Tool (MTT),” Open MPI, <https://www.open-mpi.org/projects/mtt/>
- 22 “Open Tool for Parameter Optimization (OTPO),” Open MPI, <https://www.open-mpi.org/projects/otpo/>
- 23 Farber, Rob. [ECP brings a host of hardware-accelerated and GPU-friendly optimizations to the MPICH library](#). February 18,2022
- 24 “MPICH Overview,” MPICH, <https://www.mpich.org/about/overview/>
- 25 Trilinos website: <https://trilinos.github.io/>

- 26 ECP: Map Applications to Target Exascale Architecture with Machine-Specific Performance Analysis, Including Challenges and Projections (ECP-U-AD-RPT_2021_00208), March 31, 2021: https://www.exascaleproject.org/wp-content/uploads/2022/02/ECP_AD_Milestone_Map-Applications-to-Target-Exascale-Architecture_v1.0_20210331_FINAL.pdf
- 27 Watkins, Jerry et al. [Evaluating, improving and maintaining a performance portable land-ice model using Albany, Trilinos and Kokkos](#). March 4, 2021
- 28 <https://www.anl.gov/mcs/petsc-portable-extensible-toolkit-for-scientific-computation>
- 29 <https://www.anl.gov/mcs/petsc-portable-extensible-toolkit-for-scientific-computation>
- 30 <https://archive.cra.org/CRN/articles/sept06/pieper.html>
- 31 <https://help.rc.ufl.edu/doc/PETSc>
- 32 https://petsc.org/release/miscellaneous/applications_publications/
- 33 Vaclav Hapla et al. Fully Parallel Mesh I/O Using PETSc DMplex with an Application to Waveform Modeling. SIAM Journal on Scientific Computing, v43, Issue 2, 2021 | <https://epubs.siam.org/doi/10.1137/20M1332748>
- 34 Allen, Jeffery et al. "Progress on Optimizing Wind Farms and Rotor Designs Using Adjoints." Presented at the Wind Energy Science Conference, 25-28 May 2021 | <https://www.osti.gov/servlets/purl/1823435> | <https://www.osti.gov/biblio/1823435>
- 35 <https://computing.llnl.gov/projects/spack-hpc-package-manager>
- 36 <https://computing.llnl.gov/projects/spack-hpc-package-manager>
- 37 <https://computing.llnl.gov/projects/spack-hpc-package-manager>
- 38 Qiyang Hu and Shao-Ching Huang [UCLA]. [A Hybrid Approach to Scientific Software Package Management on a HPC Cluster](#). SC20, 2020 | https://sc20.supercomputing.org/proceedings/sotp/sotp_files/sotp110s2-file1.pdf
- 39 [STRUMPACK Speeds Sparse Algorithms on CPUs and GPUs](#). Exascaleproject.org. January 15, 2021 | <https://portal.nersc.gov/project/sparse/strumpack/>
- 40 Elizaveta Rebrova et al. [A Study of Clustering Techniques and Hierarchical Matrix Formats for Kernel Ridge Regression](#). 2018 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), 2018 | <https://ieeexplore.ieee.org/document/8425507/authors#authors>
- 41 (Elizaveta Rebrova et al. [A Study of Clustering Techniques and Hierarchical Matrix Formats for Kernel Ridge Regression](#). 2018 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), 2018 | <https://ieeexplore.ieee.org/document/8425507/authors#authors>
- 42 Hartwig Antz et al. [Preparing Sparse Solvers for Exascale Computing](#). Philosophical Transactions of the Royal Society, A: Mathematical, Physical and Engineering Sciences, March 6, 2020 | <https://royalsocietypublishing.org/doi/full/10.1098/rsta.2019.0053>