SAND2017-10558R

Advanced Technology and Mitigation (ATDM) SPARC Re-Entry Code

Fiscal Year 2017 Progress and Accomplishments for ECP

The SPARC (Sandia Parallel Aerodynamics and Reentry Code) will provide nuclear weapon qualification evidence for the random vibration and thermal environments created by re-entry of a warhead into the earth's atmosphere. SPARC incorporates the innovative approaches of ATDM projects on several fronts including: effective harnessing of heterogeneous compute nodes using Kokkos, exascale-ready parallel scalability through asynchronous multi-tasking, uncertainty quantification through Sacado integration, implementation of state-of-the-art reentry physics and multiscale models, use of advanced verification and validation methods, and enabling of improved workflows for users. SPARC is being developed primarily for the Department of Energy nuclear weapon program, with additional development and use of the code is being supported by the Department of Defense for conventional weapons programs.

SPARC is a compressible computational fluid dynamics (CFD) code to solve aerodynamics and aerothermodynamics problems. In its present state, SPARC solves the Navier-Stokes and Reynolds-Averaged Navier-Stokes (RANS) turbulence model equations on structured and unstructured grids using a cell-centered finite volume discretization scheme and is targeted towards the transonic flow regime to support gravity bomb analyses and the hypersonic flow regime to support re-entry vehicle analyses. SPARC also solves the transient heat equation and associated equations for non-decomposing and decomposing ablators on unstructured grids using a Galerkin finite element method. One and two-way multiphysics couplings exist between the CFD and ablation solvers within the code. The ATDM program is driving the development of the code for the re-entry application space, which nominally involves hypersonic flows, ablator/thermal response, and structural response of the vehicle under normal and hostile environments.

Hypersonic re-entry vehicles experience severe environments related to heat and vibration generated by aerodynamic loading during re-entry. It is impossible to replicate these environments in a ground test and, given the expense and diagnostic issues associated with flight testing, modeling and simulation is the most viable approach to understand and design to these environments. Validated simulation tools are, therefore, an essential requirement for development of hypersonic flight vehicles and the components they contain. The primary approach to developing these capabilities is to validate them against ground test data, where detailed diagnostic information is available and then apply them to flight scenarios. The end goal for the ATDM Reentry Project is to provide the nuclear weapons program a `virtual flight test' capability from atmospheric pierce point to target that allows weapons engineers to assess vehicle performance and make imperative design decisions.

Fiscal Year 2017 Progress

- Mesh Refinement
    - The inline refinement has been demonstrated for structured meshes in SPARC under MPI parallelism.
    - Geometry projection has been demonstrated but will be integrated in FY18.
- SPARC Performance Portability & Higher Order Methods
    - A Performance portable implementation of the finite volume scheme in SPARC has largely been completed, and detailed performance analyses on several CPU, KNL and GPU platforms has been done. Much of this work was documented in a recent conference paper and presentation.
    - Higher-order stencils for the finite volume scheme are now possible within SPARC. The first implementation of such a method is the subject of work related to the FY18 ATDM L2 Milestone for SPARC.
    - Solver performance has been improved even further with the introduction of SIMD datatypes into the Trilinos Ifpack2 package. The KokkosKernels Team delivered a new Trilinos/Ifpack2 point- and line-implicit solver that is tens of percent faster than SPARC's equivalent optimized native solvers on KNL and Haswell platforms. This new solver has been integrated into SPARC.
- Sensitivity Analysis
    - Verified sensitivities for arc jet problems used for calibration tests.
    - Started working on code refactoring to remove code redundancy and repetitions and clean parts related to sensitivity calculations.
    - Worked on the arc jet calibration problems from a probabilistic standpoint: built surrogate models for SPARC, analyzed global sensitivities, and successfully ran Bayesian calibration. (Due to some run-to-run variability in the SPARC simulations that is currently being investigated, the results will need to be re-run once these issues are resolved.)
- SPARC software redesign implementation
    - Integration of IOSS reader and writer with the new structured and unstructured field and mesh data structures. This activity provides added value by allowing output of any field data. It also allows the mesh database/refinement effort to leverage the IOSS.
    - Conversion of structured algorithms and good progress towards conversion of unstructured algorithms
    - Conversion of mesh discretization line code
    - Conversion of Trilinos/Sacado parameter sensitivity capability to use new fields. This change provided added value by allowing sensitivity of any surface-based field, not just heat flux.

Fiscal Year 2017 Accomplishments

Inline mesh refinement in SPARC. Staff at Sandia National Laboratories have implemented an inline mesh refinement capability within the SPARC compressible fluids code. Analysts running SPARC have a need for high resolution studies (>1Billion grid cells) as well as for mesh refinement studies to quantify numerical error. Because of the high costs with using such large meshes (generation, decomposition, file transfer) it is desirable to read in a much smaller mesh and refine the mesh internally in the code to the desired scale. In FY17 we demonstrated an integrated inline refinement capability in SPARC for structured meshes with support for shared memory (OpenMP/GPU using Kokkos) as well for distributed memory (using MPI). Projection of the new surfaces nodes to the geometry was also implemented but will be integrated in FY18. A remaining challenge is to incorporate a rebalance step to enable the refinement and solve in SPARC to use different number of processors.

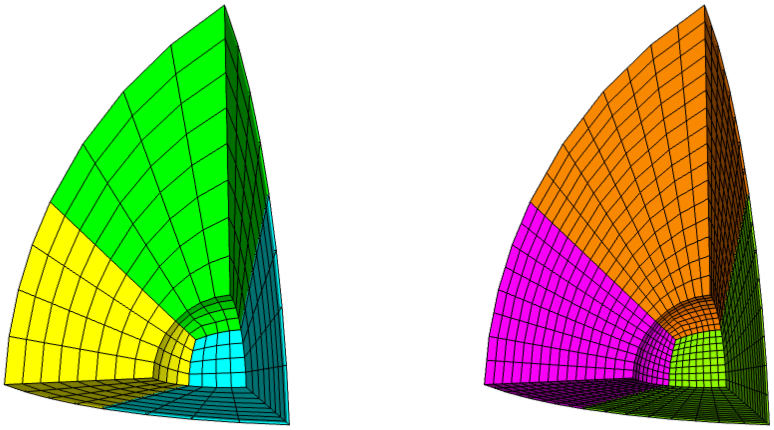

Figure: This plot shows a refinement of a multi-block structured mesh. Here the boundary nodes are not projected to the geometry.

ATDM Application and Component Integration. The ATDM re-entry application code team has integrated the IOSS (Input/Output SubSystem) component into SPARC. This integration, which includes new features such as automatic parallel decomposition of mesh databases, and support for the CGNS (Computational Fluid Dynamics General Notation System) database formats, not only enables access to the Exodus and CGNS database formats, but also integrates SPARC with other key ATDM components because IOSS acts as an interface to these services. These other components include Catalyst, which provides in-situ visualization; Data Warehouse for accessing high performance IO hardware; and Percept, which will provide in-line refinement of mesh databases. A remaining challenge is to exercise the capabilities provided by these components on large-scale datasets.

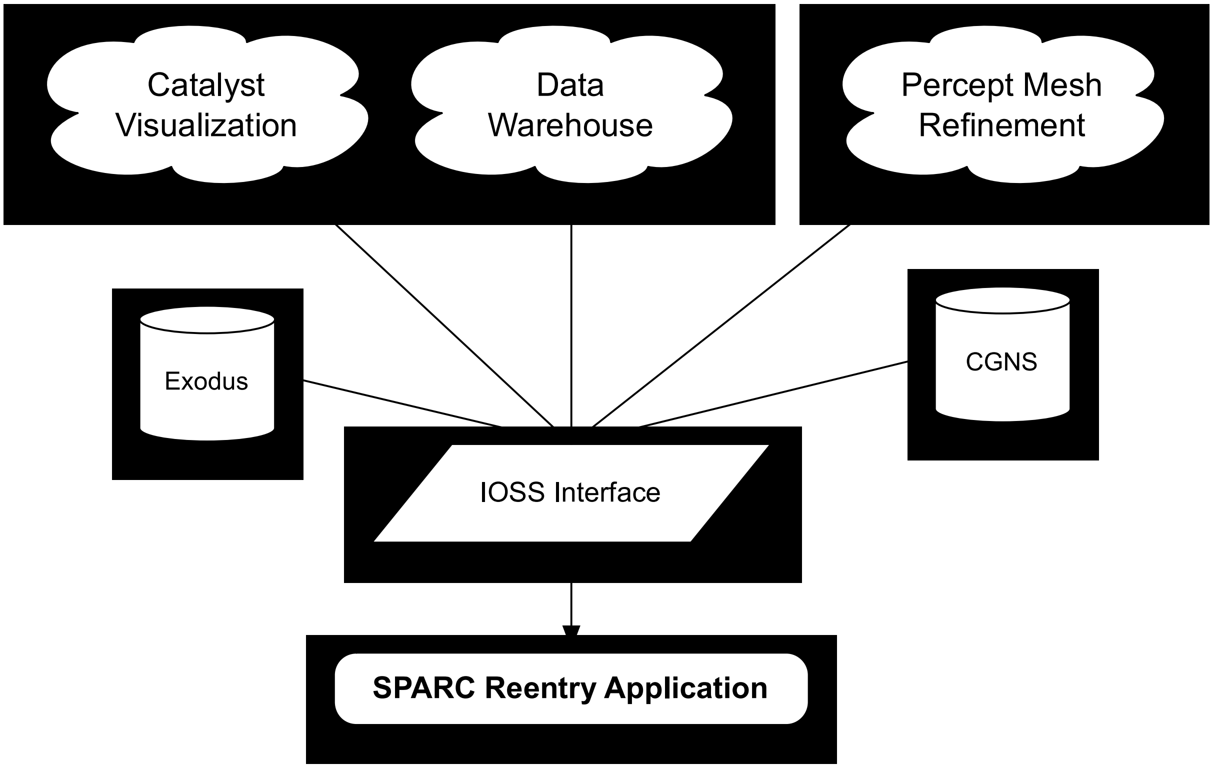

Figure: The integration with the IOSS interface acts as a hub and provides access to several ATDM components and database formats.

Reentry App Verification using the Method of Manufactured Solutions. Researchers at Sandia National Laboratories have begun rigorous work to verify the correctness of the ATDM reentry app, SPARC. In verifying SPARC, it was discovered that most of the boundary condition implementations caused degradation to only first-order accuracy, instead of the expected second-order accuracy. These implementations are being modified to ensure the expected second-order accuracy in the reentry app prior to its deployment. One- and two-dimensional heavily non-uniform domains have been tested for second-order spatial accuracy to ensure errors decrease twice as fast as the mesh is refined. Accuracy is tested by manufacturing solutions, and inserting the solutions into the governing equations to obtain source terms. With the added source terms, the solver approximates the manufactured solution. Comparing the solver solution with the manufactured solution provides insight into the error and how it scales with mesh refinement.

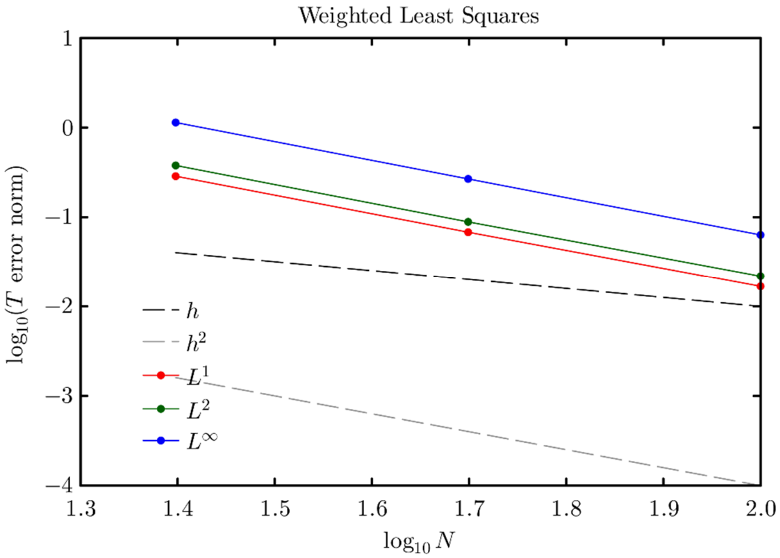

Figure: This plot shows an example of second-order accurate convergence for a non-uniform, two-dimensional mesh used for verification.