Propel: A Hybrid-Parallel Reacting Flow Code For Combustion Research

Brian D. Taylor, Douglas A. Schwer, Andrew Corrigan
The Laboratories for Computational Physics and Fluid Dynamics
Naval Research Laboratory, Washington, DC

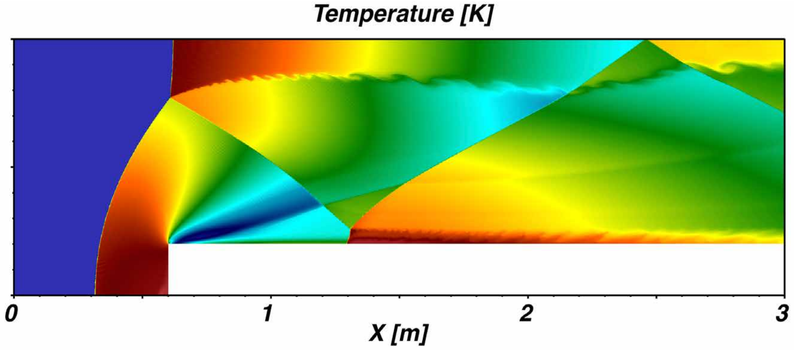
Abstract
Propel is a high-performance hybrid-parallel simulation code for unsteady, compressible reacting flows developed at the Naval Research Laboratory. It is currently being used to simulate combustion flow fields in complex propulsion system domains using reaction models of varying levels of cost and complexity. In this work we describe how Propel has been used to investigate Rotating Detonation Engines (RDEs), a novel combustor design for propulsion and power generation that harnesses the high thermodynamic efficiency of detonation. We also describe the implementation of detailed thermochemistry and chemical kinetics in Propel, along with performance and scalability results for simulations of two-dimensional hydrogen-air detonations using a 10 species, 34 reaction mechanism.

Propel Code Description
Developed using C++ template programming
- **Physics, grid, solution algorithms all specified via templates**
- **These constructs are combined at compile time to ensure efficiency**

Solver features
- **Finite volume and finite-element spatial discretizations**
- **Structured, unstructured and hybrid grids**
- **Multi-dimensional Flux Corrected Transport (FCT) flux limiter for capturing solution discontinuities**
- **Taylor-Galerkin and Runge-Kutta time integration schemes**
- **Several gas models: Combinations of ideal gas equation of state and various chemical reaction models**
- **Chemical reaction source term integrated in operator-split manner**

Parallelization
- **Coarse-grained, distributed memory parallelism using MPI:**
  - **Domain decomposition using ParMETIS**
- **Fine-grained, shared memory parallelism using Thrust library:**
  - **Serial, CUDA, OpenMP, Thread Building Blocks backends**
  - **Design follows Standard Template Library structure**

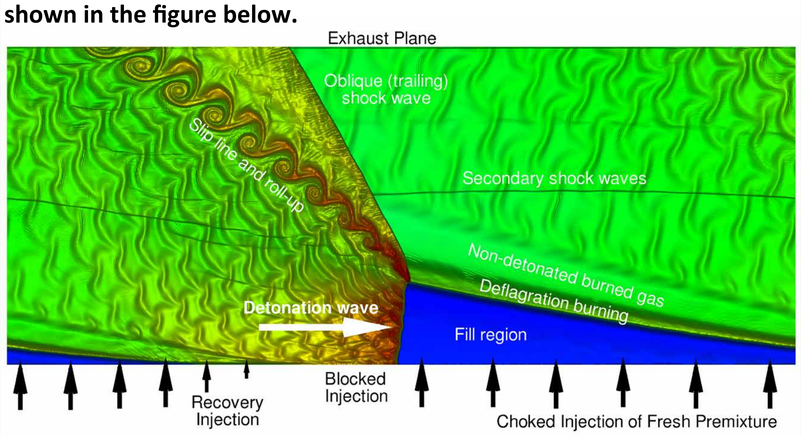
***Below*: Mach 3 flow of hydrogen-air mixture over a forward-facing step**
Temperature [K]



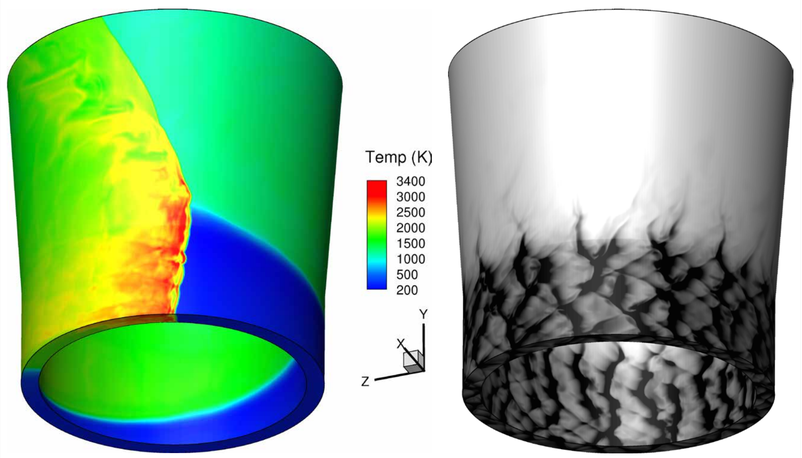
Acknowledgements
This work was supported by the Office of Naval Research through the NRL 6.1 Computational Physics Task Area

Rotating Detonation Engines
Rotating detonation engines (RDEs) represent a novel approach to harnessing the high efficiency thermodynamic cycle of detonation for propulsion and power generation. An RDE combustor consists of an annular channel through which detonations propagate azimuthally. Fluidic valves inject fuel and oxidizer at the head end of the RDE. Detonation waves propagate azimuthally through this mixture, compressing and rapidly burning it. Combustion products are then exhausted axially from the combustor.

The primary features of the azimuthally-unrolled RDE flow field are shown in the figure below.



Although the basic operation of an RDE is reasonably well understood, much remains to be learned about its performance, stability of operation, thermal characteristics, and other aspects of operation.



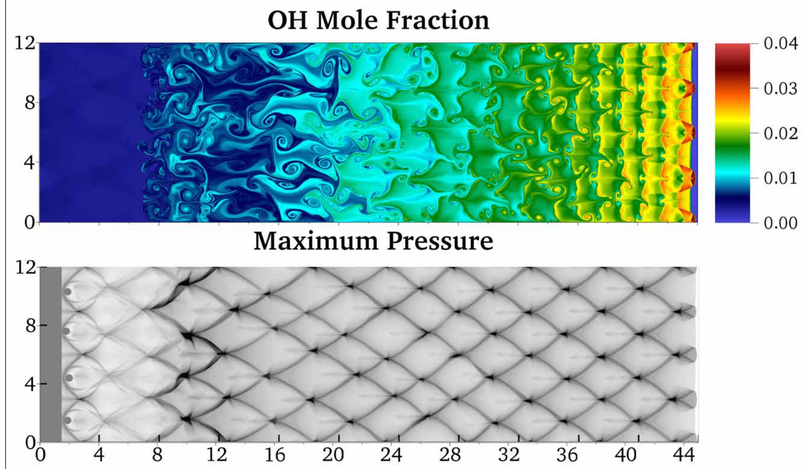
Propel has been used to investigate several aspects of RDEs:
- **Effect of mean radius and varying axial cross-sectional area on expansion flow**
- **Differences in flow conditions on the inner and outer combustor walls in three-dimensional simulations**
- **Effect of expanding exhaust through a converging-diverging nozzle**
- **Effect of detailed chemistry for non-detonation burning and chemical recombination in the expanding exhaust gas**

Schwer and Kailasanath, 53rd AIAA Aerospace Sciences Meeting, Paper 1602 (2015)
Nordeen, Schwer and Corrigan, 50th AIAA Joint Propulsion Conference, Paper 3900 (2014)
Schwer, Corrigan and Kailasanath, 52nd AIAA Aerospace Sciences Meeting, Paper 1014 (2014)

Detailed Gas-Phase Thermochemistry and Chemical Kinetics
The ability to use detailed thermochemistry and chemical kinetic reaction models has recently been added to Propel. Such models accurately reflect elementary chemical reactions, at the price of increased complexity, size, and computational cost, relative to simpler phenomenological models typically used in combustion simulations.

In our implementation, the chemical model specification is read from a Chemkin input file by a preprocessor that generates optimized C++ code for the thermodynamic functions and chemical kinetic rate expressions needed by Propel. Equilibrium reversible reactions are converted to pairs of irreversible reactions via least-squares fits. The generated source code has all thermodynamic and kinetic rate constants inline.

Chemical rates are integrated in Propel using the stiff ODE solver CHEMEQ2, a 2nd-order, α-QSS predictor-corrector method that is *O(n)* in storage and computation. As an explicit method, it is easy to implement and does not require Jacobian evaluations or linear solvers.



Propel's detailed chemistry solver was benchmarked on a 2D detonation problem (*similar to above*) with 3.6 million grid cells using a 10 species hydrogen-air mechanism with 34 irreversible reactions. CPUs are 8 core 2.6GHz Intel Xeon E5-2650 v2. GPUs are Nvidia GeForce GTX TITAN Black. The maximum speedup of GPUs relative to CPUs is 6.7. Scalability of GPUs on this problem is limited by the fixed problem size.

# of Compute Units	Millions of grid cell updates / second / compute unit	
	GPUs	CPUs
1	9.04	N/A
2	8.64	1.33
4	8.45	1.32
8	7.81	1.27
16	7.06	1.24
24	5.73	1.23

Taylor, Schwer and Corrigan, 53rd AIAA Aerospace Sciences Meeting, Paper 0842 (2015)

Future Work
- **Dynamic load balancing of chemical source term across MPI ranks**
- **Explore ways to reduce resource usage in chemical source term**