The Journal of Open Source Software

rgfrosh: A Python package for calculating shock conditions using real gas equations of state

Cory Kinney[1] **and Subith Vasu**[1]

1 Department of Mechanical and Aerospace Engineering, University of Central Florida, Orlando, FL USA

DOI: 10.21105/joss.06855

Software

- Review
- Repository
- Archive

Editor: Kyle Niemeyer

Reviewers:

- @cartemic
- @blackspur

Submitted: 29 April 2024
Published: 11 July 2024

Summary

`rgfrosh` solves the reflected shock equations for an arbitrary equation of state, allowing for accurate calculation of the temperature and pressure behind a reflected shock accounting for real gas effects. Forward and inverse solvers are provided to calculate experimental conditions based on measurements and to plan experiments based on desired conditions, respectively. `rgfrosh` is designed to be lightweight and extensible, with a simple interface for users to utilize existing thermodynamic libraries.

Statement of need

Combustion is a complex process that is influenced by numerous factors, including temperature, pressure, and chemical composition. Combustion modeling relies on chemical kinetics mechanisms that detail how chemical reaction rates – involving many intermediate species – vary with temperature and pressure. Experimental data is essential for measuring reaction rates, refining mechanisms, and validating predictions from these models for certain fuels and conditions.

Shock tubes are ideal experimental facilities for performing fundamental research in combustion because they use shock waves to impart step changes in temperature and pressure to a test gas mixture. The gas behind the reflected shock is relatively stagnant, and the system is considered adiabatic on the relevant time-scales; therefore, the shock tube closely resembles a constant volume zero-dimensional reactor, allowing for comparison between experimental measurements and model predictions. Accurate calculation of the reflected shock conditions is essential for correctly interpreting experimental results. This is accomplished by solving the reflected shock equations under an assumed equation of state using the known initial state of the gas and the measured shock velocity.

Fundamental research into combustion at extremely high-pressure conditions, such as for rocket engines or direct-fire supercritical CO_2 power cycles (Kinney, 2022), requires consideration of real gas effects in experimental measurements and model simulations.

State of the field

A shock solver supporting real gas equations of state called RGFROSH was previously developed in FORTRAN by Davidson & Hanson (1996); however, there existed a need for a modernized open-source version, as the original is not generally available nor would it be readily compatible with modern tools. The extensive SDToolbox (Browne et al., 2021) has functionality for computing the postshock state for a chemically frozen shock; however, SDToolbox only supports Cantera (Goodwin et al., 2023) for thermodynamic properties and does not provide a convenient interface to obtain the full solution for a reflected shock for both the forward and inverse problems. Thus, the present `rgfrosh` was developed in Python as a solution that can utilize any modern thermodynamic library to solve for the full reflected shock solution for an arbitrary equation of state to enable further research into high-pressure combustion.

Features

`rgfrosh` provides a simple interface for solving the reflected shock equations, allowing the user to obtain the full solution with a single function call. Two models are provided - `IdealShock` for calorically perfect gases, and `FrozenShock` for arbitrary equations of state. The former is primarily included for comparison and validation purposes, while the latter is the primary focus of the package as it allows for accurate calculation of the reflected shock conditions for real gases.

To remain as lightweight and extensible as possible, `rgfrosh` relies on external packages for the key thermodynamic functions required by the `FrozenShock` solver. The required interface is defined by the `ThermoInterface` protocol class, which was written to provide native support for Cantera (Goodwin et al., 2023). Additionally, an interface is provided to wrap CoolProp (Bell et al., 2014), which itself has backend support for NIST REFPROP (Lemmon et al., 2002), for use with the solver. These two compatible packages enable support for a wide range of equations of state which should cover the majority of use cases; however, any user-defined class that implements the simple `ThermoInterface` protocol can be used with the solver.

The primary use cases for `rgfrosh` are experiment postprocessing and experiment planning. The `solve_incident` and `solve_reflected` methods implement the Newton-Raphson solver detailed by Davidson & Hanson (1996) for calculating the incident and reflected shock conditions, respectively, from the initial conditions and the experimentally measured shock velocity. The `solve_initial` method implements the algorithm derived by the author (see Kinney, 2022, A.3) for calculating the initial pressure and incident shock velocity, temperature, and pressure from the initial temperature and target reflected shock temperature and pressure for an experiment.

Future work

Future work includes the consideration of vibrational non-equilibrium in the shock solvers. Current solver functionality would be classified as equilibrium-equilibrium (EE) mode, referring to the incident and reflected shock, respectively; frozen-equilibrium and frozen-frozen modes are planned.

Acknowledgements

We would like to acknowledge D. F. Davidson and R. K. Hanson for authoring the original software this work is based on, granting permission to use the package name, and for providing validation data for comparison. Additionally, the authors appreciate funding from the University of Central Florida.

References

Bell, I. H., Wronski, J., Quoilin, S., & Lemort, V. (2014). Pure and pseudo-pure fluid thermophysical property evaluation and the open-source thermophysical property library CoolProp. *Industrial & Engineering Chemistry Research*, *53*(6), 2498–2508. https://doi.org/10.1021/ie4033999

Browne, S., Ziegler, J., Bitter, N., Schmidt, B., Lawson, J., & Shepherd, J. E. (2021). *SDToolbox - numerical tools for shock and detonation wave modeling*. California Institute of Technology, Pasadena, CA. https://shepherd.caltech.edu/EDL/PublicResources/sdt/

Davidson, D. F., & Hanson, R. K. (1996). Real gas corrections in shock tube studies at high pressures. *Israel Journal of Chemistry*, *36*(3), 321–326. https://doi.org/10.1002/ijch.199600044

Goodwin, D. G., Moffat, H. K., Schoegl, I., Speth, R. L., & Weber, B. W. (2023). *Cantera: An object-oriented software toolkit for chemical kinetics, thermodynamics, and transport processes*. https://www.cantera.org. https://doi.org/10.5281/zenodo.8137090

Kinney, C. (2022). *Extreme-pressure ignition studies of methane and natural gas with CO2 with applications in rockets and gas turbines*. https://stars.library.ucf.edu/etd2020/1033

Lemmon, E. W., Huber, M. L., & McLinden, M. O. (2002). NIST reference fluid thermodynamic and transport properties–REFPROP. *NIST Standard Reference Database*, *23*(2002), v7.