Hytrac: A Hydrodynamic Front-Tracking Code for the Study of High Energy Density Multi-Material Flows

N. Joiner*, D. Chapman, N. Chaturvedi, T. Edwards, A. Fraser, N. Hawker,
J. Herring, N. Niasse, J. Pecover, M. Read, D. Vassilev and A. Venskus
*nathan.joiner@firstlightfusion.com

first light

Motivation

- First Light Fusion (FLF) is researching nuclear fusion energy production with novel fuel target designs driven by hyper-velocity projectile impact using:
  - M3, a pulsed power facility (Fig. 1), with 8 MA discharge in 1.5 µs, through a flyer plate load, and 2.5 MJ of stored energy at full charge (Aluminium projectile @ >15km/s)
  - A two-stage light gas gun (copper @ ~7km/s)


Fig. 1: Aerial view of M3 pulsed power generator

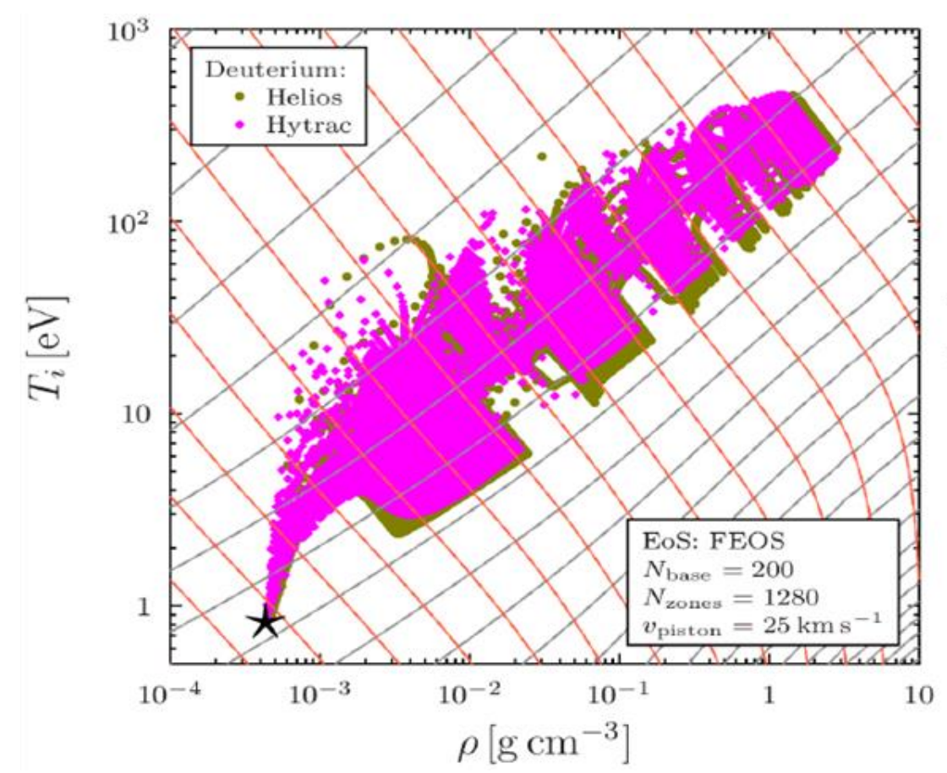

Fig. 2: Hytrac simulation cell fuel state trajectory (1eV ~ 10,000 K)

- This requires hydrodynamic modelling at extreme ranges of density and temperature (Fig. 2)
- EoS and other necessary physics (e.g. conduction, radiation) are highly nonlinear, making code numerical stability challenging
- Accurate capturing of material interface evolution is extremely important for fuel cavity collapse and mixing
- Hytrac has been developed at FLF to address these issues and provide a robust, versatile, high-fidelity fusion target design tool

Hytrac Algorithms

- ***Hytrac* is a 1D/2D *Eulerian AMR radiation-hydrodynamics code* based on the front tracking approach**:
  - Multi-temperature
  - Tabulated EoS (FEOS [1] favoured)
  - Multi-material (***N-fluid***) node tracking
  - Thermal conduction via explicit STS method
  - Emerging radiation transport capability
  - Variable grid resolution with cell-based AMR
  - Parallelised for multi-core computing using HPX
  - Operator splitting methods used for stability

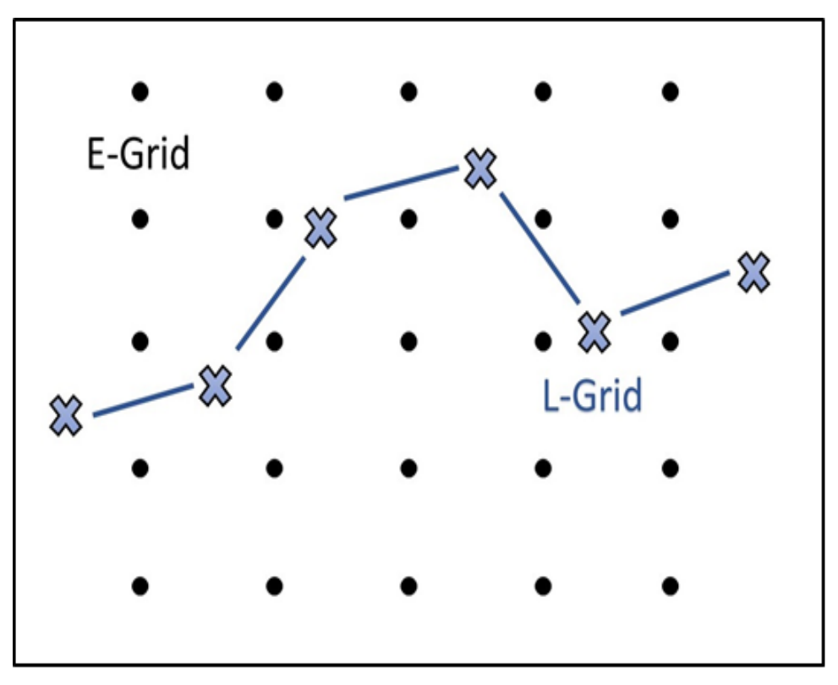

Fig. 3: Front-tracking method – Eulerian (EGrid) and Lagrangian (LGrid) discretisation coupled through ghost-fluid and Riemann solution for the contact states

- ***Front-tracking methodology (Glimm* [2] *based)***:
- Solve Eulerian equations using finite-volumes on an Eulerian grid (***EGrid***) of dimensionality $\mathbb{R}^n$
  - Godunov, slope-limited MUSCL-Hancock and WENO-5th order face reconstructions implemented and order verified
  - Exact Riemann, HLLC or KNP (HLL-like) flux-schemes implemented and verified
  - Multi-material interfaces are treated using the Ghost-Fluid method [2]; extrapolating states on the LGrid
- Propagate material interface using a Lagrangian sub-grid (***LGrid***) of dimensionality $\mathbb{R}^{n-1}$
  - States are tracked at discrete points either side of the interface
  - Interpolated states from the EGrid provide a Riemann solution contact speed to evolve the grid
  - LGrid states used to repopulate Eulerian cells that change material type through interface movement
  - The conservatism of the finite-volume method is broken by front-tracking, but it is expected that the method is asymptotically conservative with increasing resolution (partially verified with Hytrac)

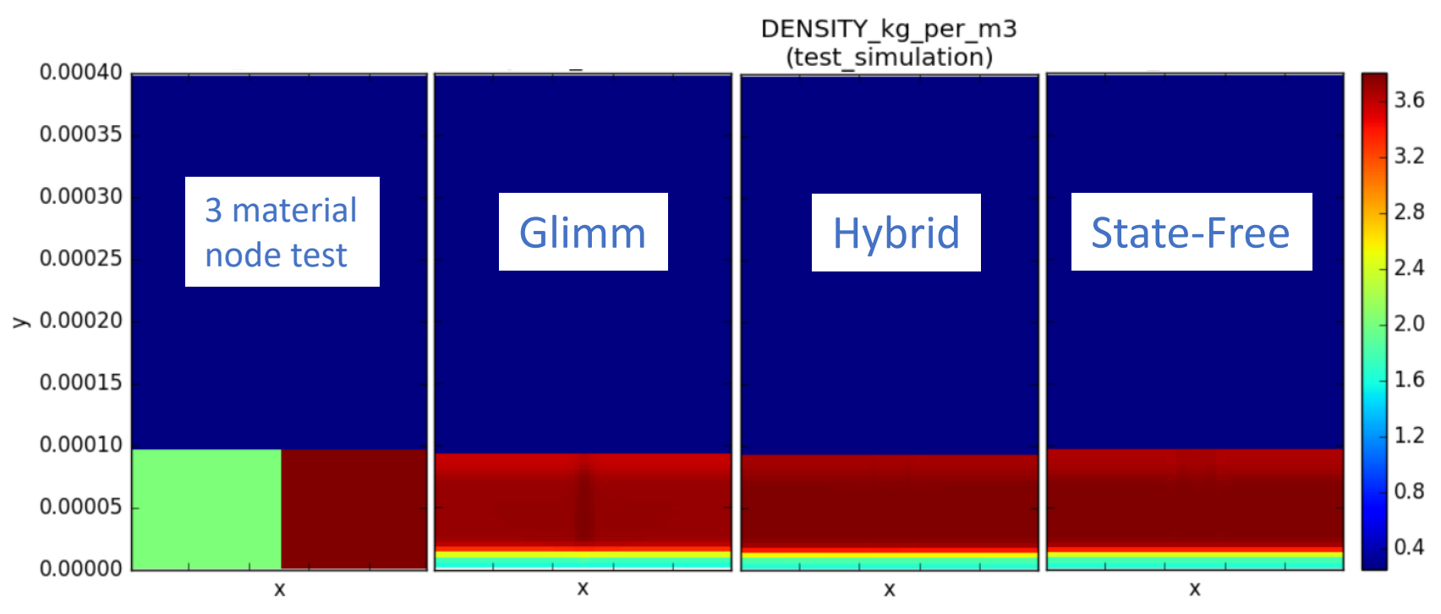

Fig. 4: Simple 2D test of a planar shock with imposed 3-fluid node material interfaces showing differences in front-tracking schemes; Glimm leaves the strongest LGrid imprint, state-free does not capture the interface velocity as accurately, while 'hybrid' displays the best of both methods

- A '***state-free***' method [3] has been implemented that reduces the coupling of the grids in an effort to provide greater code robustness
- A **hybrid approach** that utilises Glimm's propagation method, with the *state-free* Ghost-Fluid method, is proving to be the most reliable in terms of stability and accuracy in many cases
- Significant development of LGrid redistribution and untangling algorithms was required for robust code operation in general geometries

HPX Parallelism

- Multi-core parallelisation using the HPX (High Performance ParalleX) runtime environment[4]
  - HPX has been designed for systems of any scale, from hand-held devices to large-scale HPC
  - HPX exposes a uniform, standards-oriented (C++) API for ease of programming parallel and distributed applications
  - HPX has unified syntax and semantics for local and remote operations
  - Automated HPX load balancing capability is attractive for AMR based CFD
  - Easy to develop an API that allows physics modules to be added and automatically parallelised

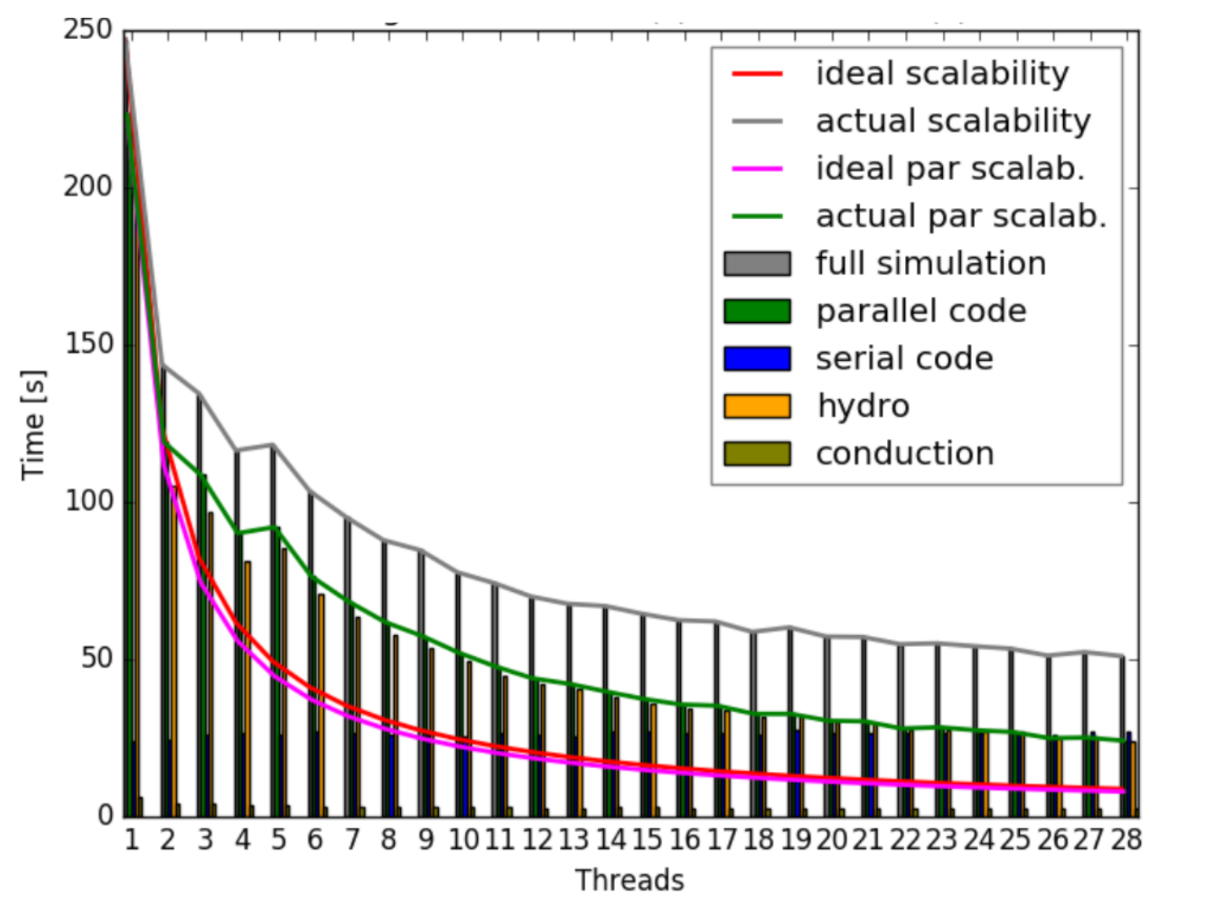

Fig. 5: Parallel scalability using HPX with Hytrac for a 2D full-physics fusion target simulation

- Operator based thread parallelism implemented in Hytrac
  - HPX allows asynchronous code execution using hundreds of millions of threads
  - Concurrency manageable with HPX dataflow and future based synchronization
- We plan to leverage the HPX Active Global Address Space (AGAS) across remote localities, in future work, to expand code CPU scalability
  - This gives the ability to rebalance data and access/modify data classes from another locality directly

Verification and Validation

- We have built an extensive simulation test-suite that includes
  - Standard compressible flow problems (e.g. Sod shocktubes, double Mach reflection, Shafranov[5])
  - US Trilab (LLNL, LANL, SNL) test-suite [6,7] cases for compressible hydrocodes (this includes challenging tests such as Sedov blast waves, Noh's wall overheating test, RMIV, RMI validation case for Air-SF6 interface etc.)
  - Simple self-consistency and analytic checks
  - Cross code benchmarks
  - Method of Manufactured Exact Solutions
  - Validation against literature based experimental data
  - Validation against in-house experimental data
- The test suite is run as part of the code continuous integration procedure at all stages of development

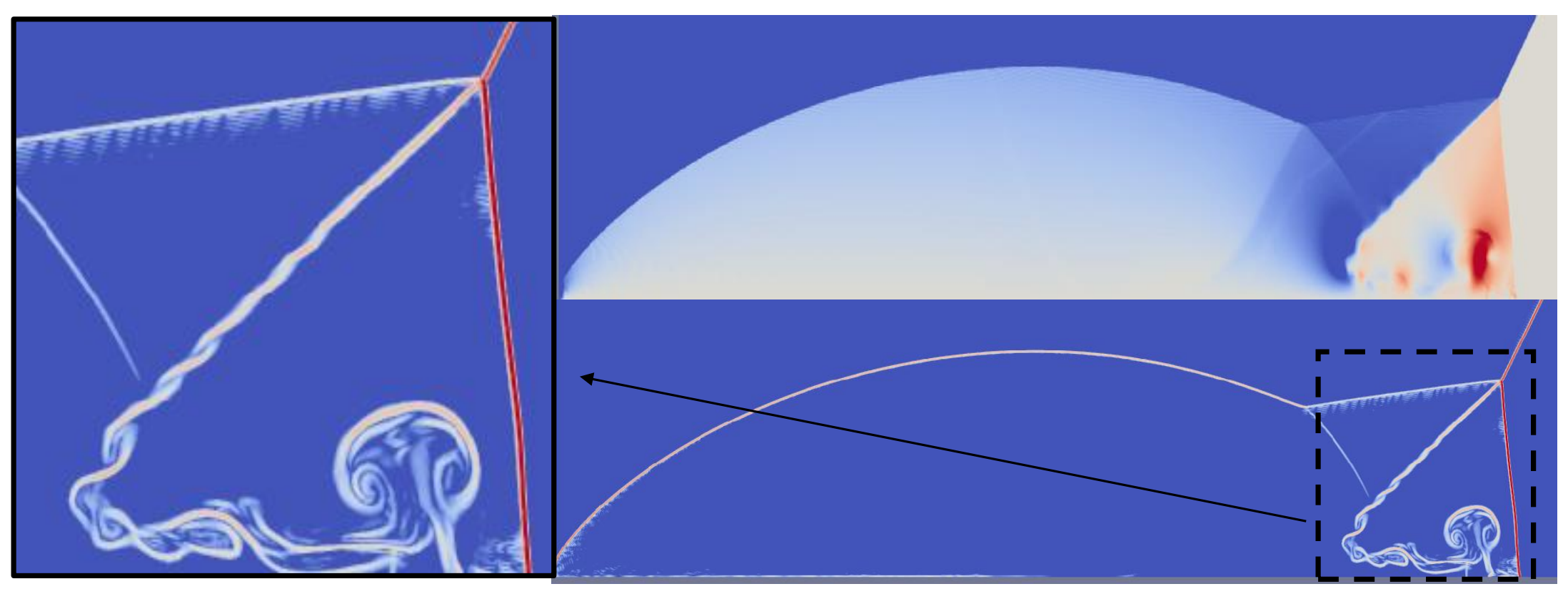
Fig. 6: Hytrac simulation of double Mach reflection – plot of ρv_y (top) and ∇T (bottom) visualising incident/reflected shocks, slip stream and KHI

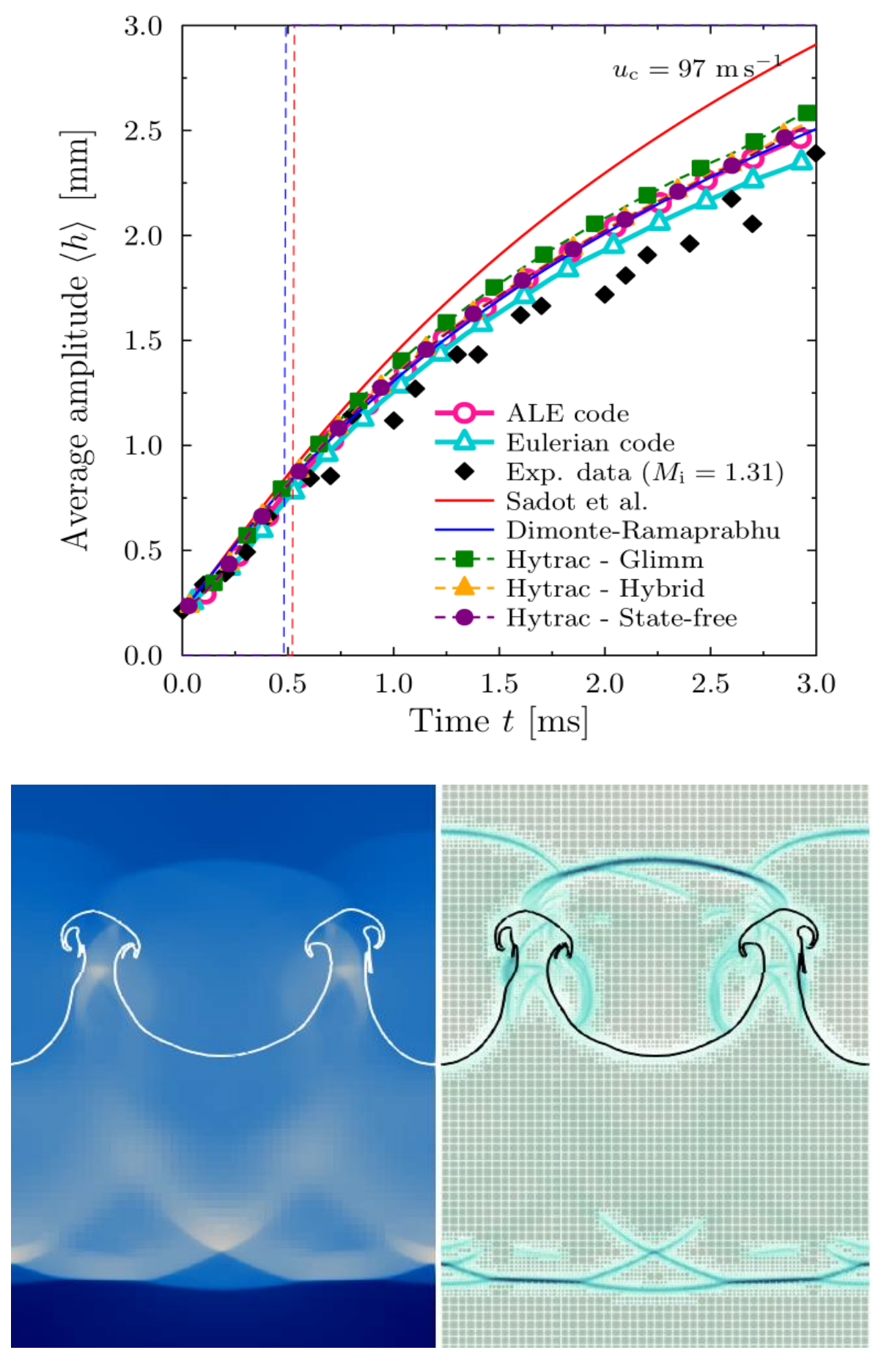

Fig. 7: Validation of the Richtmeyer-Meshkov instability (RMI). (Top)time-dependent 'spike' amplitude from simulations compared to literature. (Bottom) double RMI simulation from Hytrac, left is pseudocolour plot of ion pressure with LGrid overlay, right shows AMR grid.

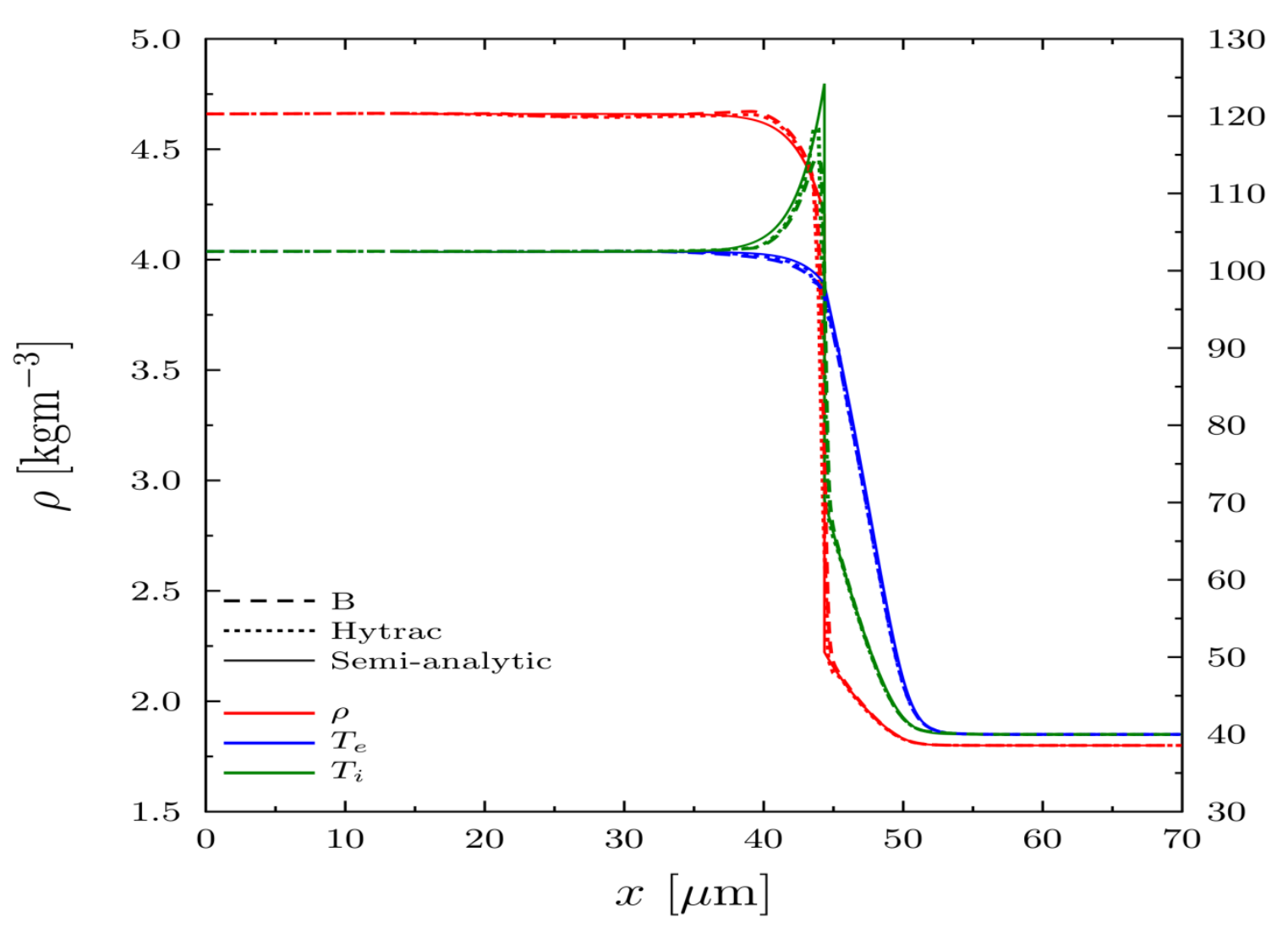

Fig. 8: Shafranov test-case: a self similar 2T shock with thermal diffusion; comparison of Hytrac vs. analytic solution and FLFs MHD code 'B'

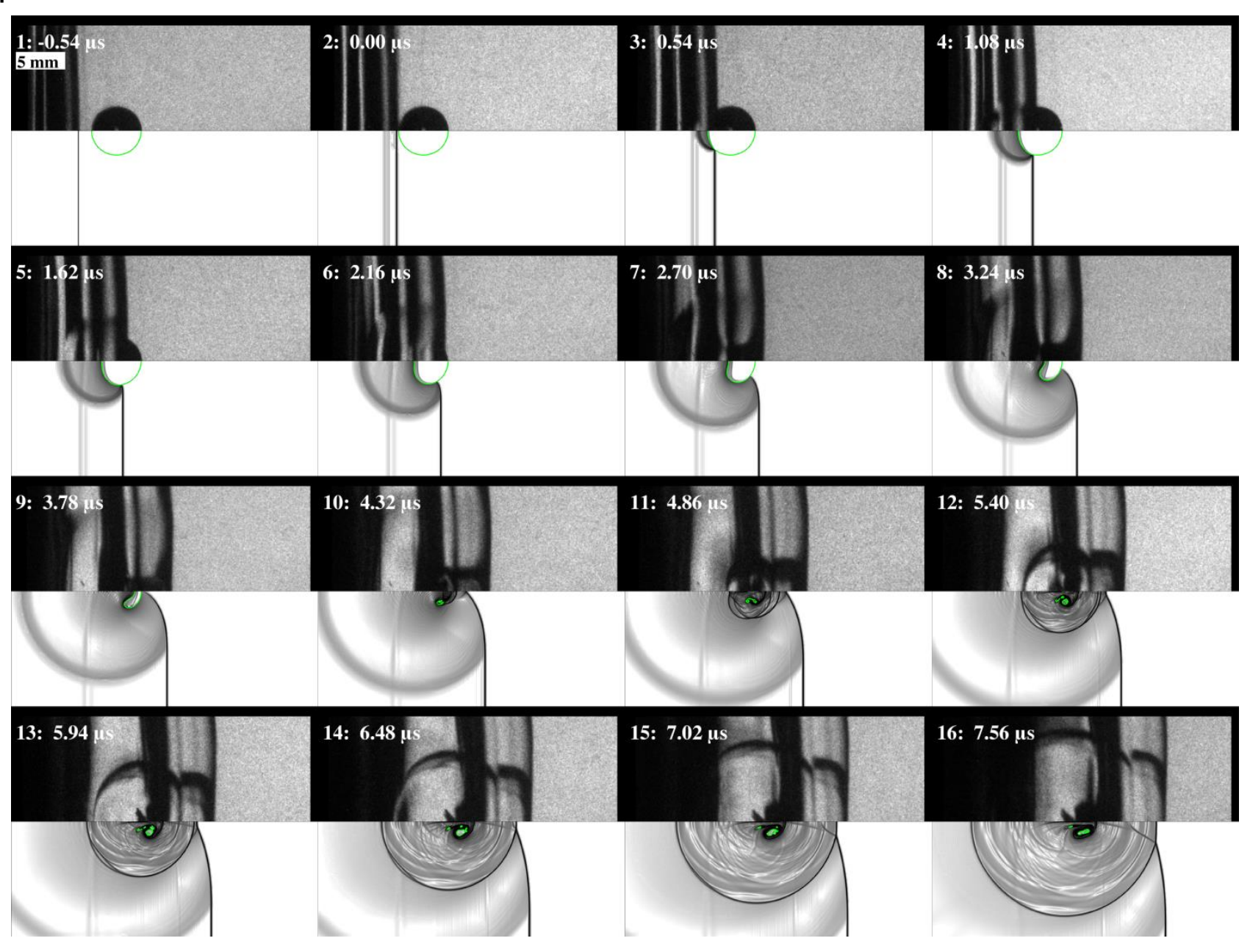
Fig. 9: Shock induced spherical cavity collapse; numerical and experimental Schlieren are compared

N-fluid node interface propagation

- An N-fluid node is the point at which three or more materials are coincident
  - The node class is implemented as a data-structure that links N LGrids
  - Each LGrid is propagated independently then the node is propagated (reformed) using averaging and/or geometric relations between free ends of the LGrid
- Simple test-cases (see e.g. Fig. 4) perform reasonably but the solution can degrade with increasing N
- Cross code triple-point benchmark case found[8], but experimental validation data needed, as currently only the correct qualitative behaviour (e.g. Fig. 10 & Fig. 11) has been confirmed in these simple test cases

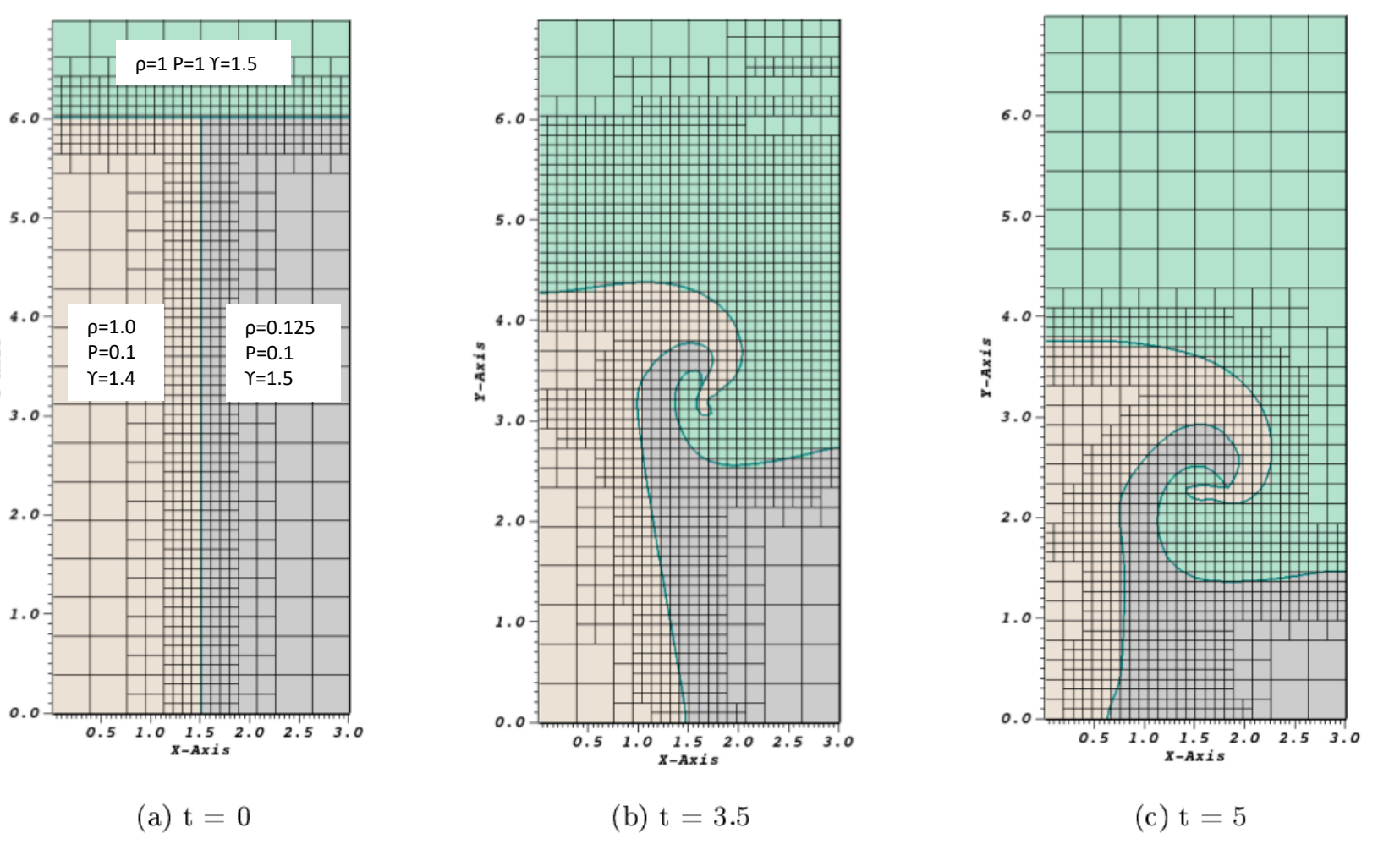

Fig. 10: Material triple point test case. Three ideal gas materials initialised with varying density, pressure and adiabatic index induce shock and material roll-up

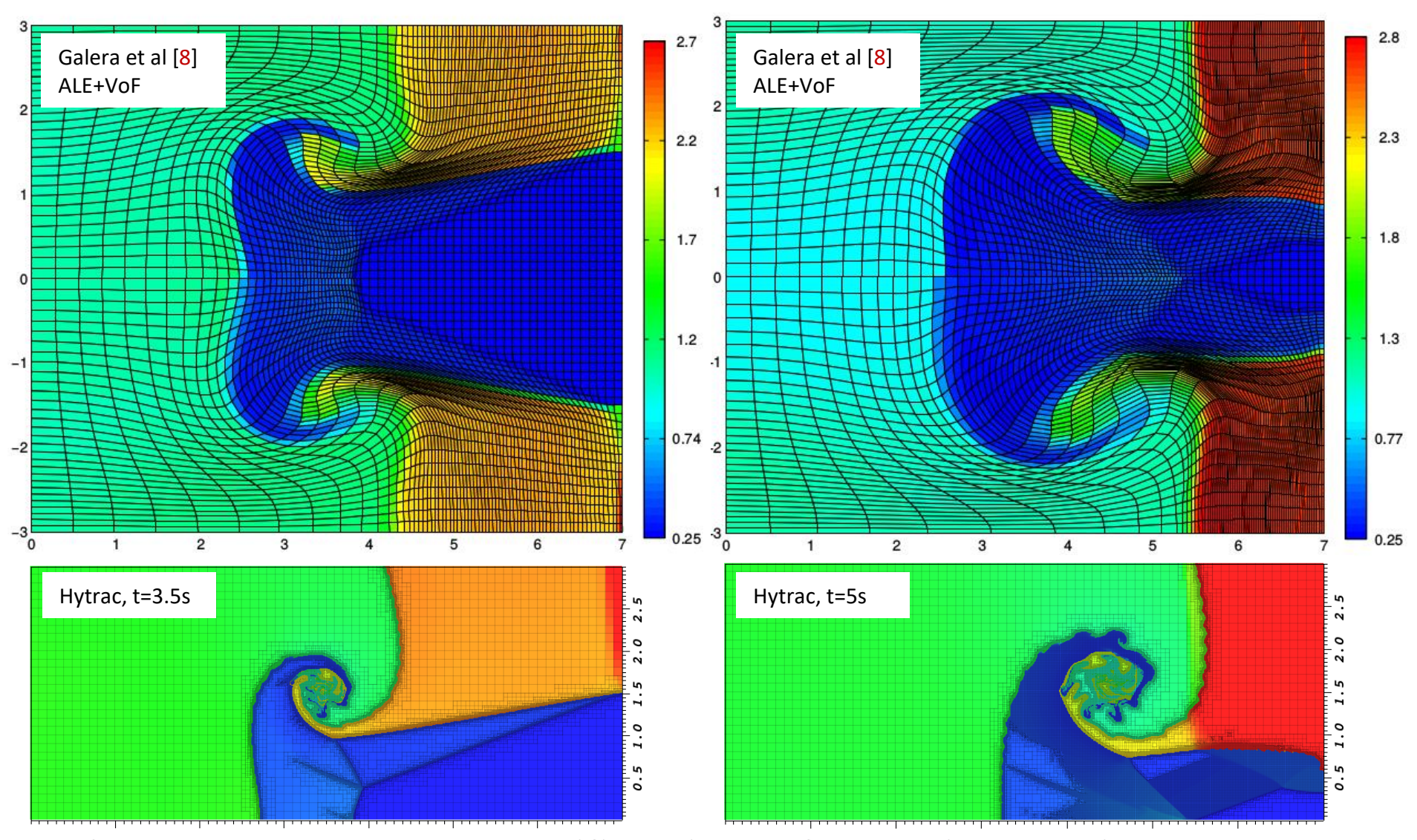

Fig. 11: Cross code verification of the triple point test-case. Specific internal energy is compared for Hytrac and the ALE+VoF method of Galera et al [8]

Summary

- ***Hytrac status:***
- We have developed a flexible, robust and performant hydrocode aimed at high energy density studies and fusion target design
- Benchmarking and standard test cases believe us to consider Hytrac comparable with the state-of-the-art fusion system design tools, both in terms of accuracy and performance
- ***Planned future developments:***
- Extension to 3D
- Full Radiation-Hydrodynamics
- Inclusion of other physics models (e.g. viscosity, alpha particle heating)
- Distributed HPC capability (domain decomposition)
- Validation of complex interface dynamics and multi-fluid node behaviour
- ***We are open to starting new collaborations for benchmarking, model development, and validation***

References

1. Faik et al, HEDP 8, 349-359 (2012)
2. Glimm et al., SIAM JSC, 24, 208-236 (2002)
3. Bempedelis et al., JCP, (2018)
4. http://stellar-group.org/libraries/hpx
5. Shafranov, JETP 5, 6, 1183-1187 (1957)
6. Kamm et al., LLNL-TR-411291, (2009)
7. Brock et al., UCRL-TR-226984, (2006)
8. Galera et al. , https://hal.inria.fr/inria-00453534 (2010)