Numerical Modelling for Analysis of High Enthalpy Impulse Facilities

ISSW35: Shock Waves Down Under, Brisbane 2025

Dr. Nick Gibbons and A/Prof. Rowan Gollan

Centre for Hypersonics, The University of Queensland

Saturday, 5th July

About Us!

Rowan Gollan:

- PhD in hypersonic flow modelling, 2009
- lecturer in School of Mechanicl & Mining Engineering
- researcher at Centre for Hypersonics
- member of core development team for Eilmer

Nick Gibbons:

- PhD in supersonic combustion, 2019
- Started as postdoctoral fellow @ UQ in 2020
- CFD code dev (Eilmer) and HPC sims expert

n.gibbons@uq.edu.au

nicholas-gibbons-67342555

This Session

This session is about numerical modelling of Impulse Facilities:

- Why?
- How?
  - An overview of approaches and (local) tools
  - Scenario: modelling the T4 reflected shock tunnel
    - ESTCn
    - NENZ1d
    - Eilmer
  - Introduction to L1d
  - Introduction to Eilmer
  - Scenario: simulation of a T4 experiment

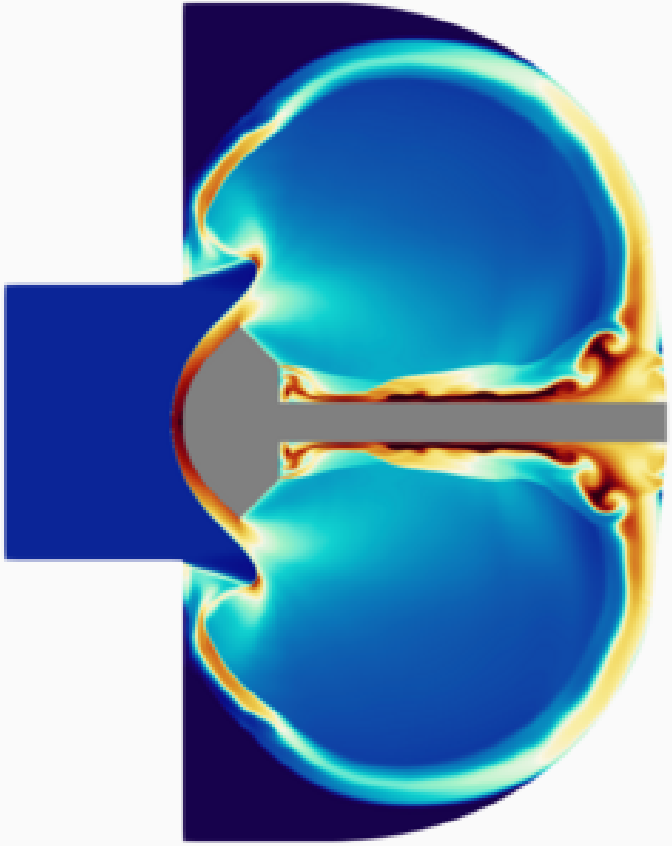
Why model an impulse facility?

Modelling and simulation of impulse facilities can address questions on design and operation.

What is the flow condition coming out of my facility?	$\longrightarrow$	estimation of test conditions
What's going wrong with my facility?	$\longrightarrow$	diagnosis of behaviour
How useful or useable is my test gas?	$\longrightarrow$	ditto
What if I replaced gas A with B? What if I operated in a different mode?	$\longrightarrow$	what-if? exploratory questions
I want to design a new facility	$\longrightarrow$	design and optimisation

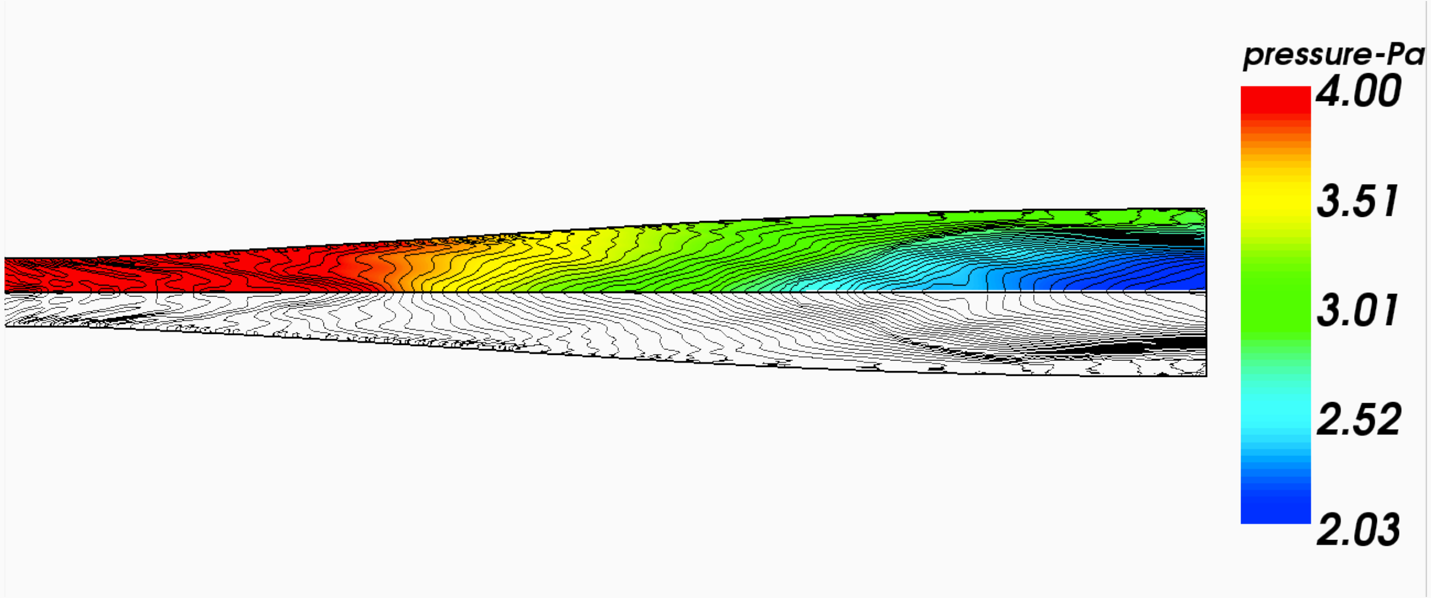
Why model an impulse facility?

What is the flow condition coming out of my facility? $\longrightarrow$ estimation of test conditions



Why model an impulse facility?

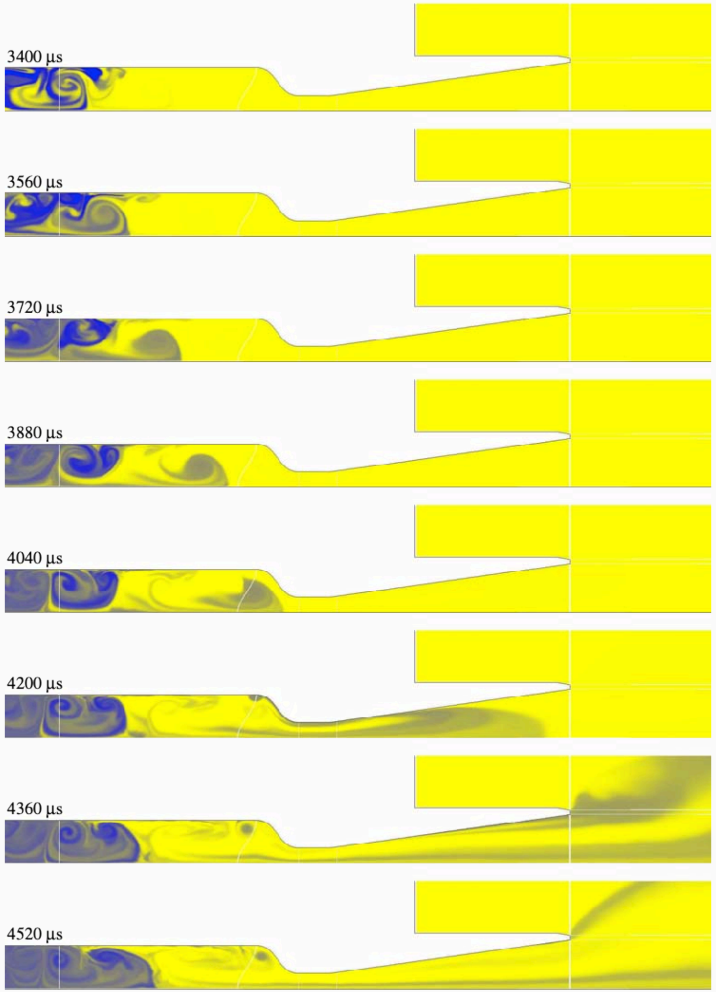
What is the flow condition coming out of my facility?	⟶	estimation of test conditions
What's going wrong with my facility?	⟶	diagnosis of behaviour
How useful or useable is my test gas?	⟶	ditto
What if I replaced gas A with B? What if I operated in a different mode?	⟶	what-if? exploratory questions
I want to design a new facility	⟶	design and optimisation



Why model an impulse facility?

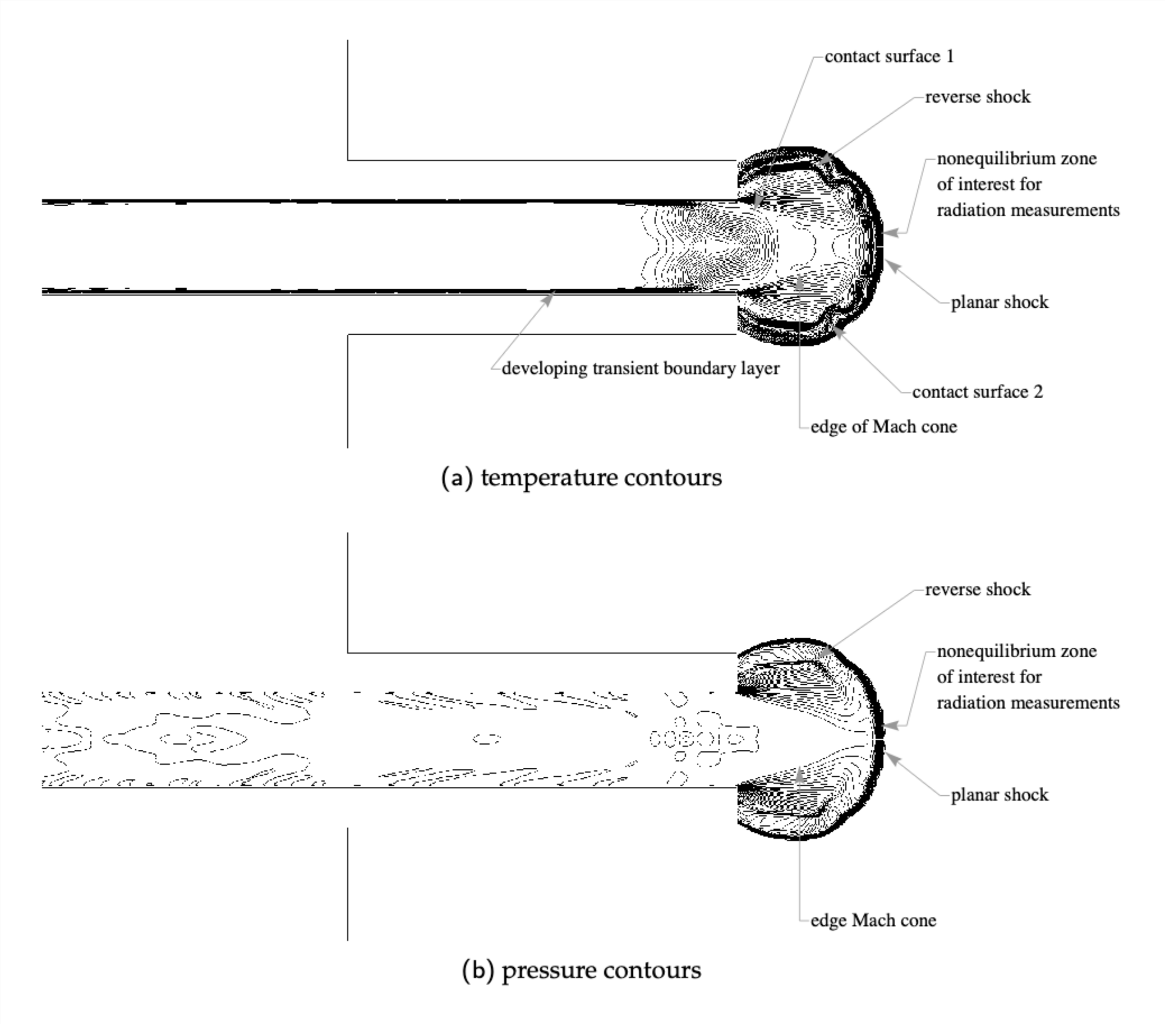
What is the flow condition coming out of my facility?	$\longrightarrow$	estimation of test conditions
What's going wrong with my facility?	$\longrightarrow$	diagnosis of behaviour
How useful or useable is my test gas?	$\longrightarrow$	ditto
What if I replaced gas A with B? What if I operated in a different mode?	$\longrightarrow$	what-if? exploratory questions
I want to design a new facility	$\longrightarrow$	design and optimisation

Why model an impulse facility? Estimating test gas contamination



source: Goozée et al. (2006) Shock Waves, 15(3) [1]

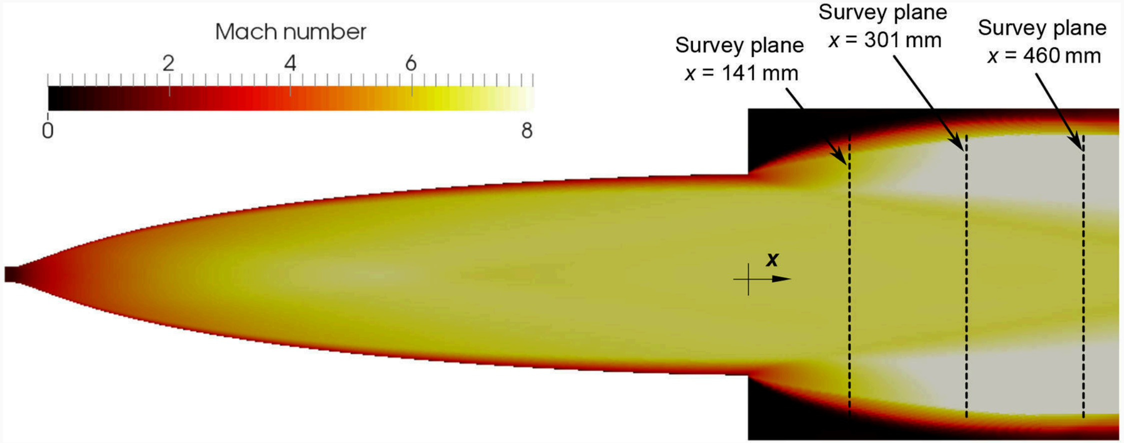
Why model an impulse facility? Estimating test gas structure



source: Gollan (2008), PhD thesis [2]

Why model an impulse facility?

What is the flow condition coming out of my facility?	→	estimation of test conditions
What's going wrong with my facility?	→	diagnosis of behaviour
How useful or useable is my test gas?	→	ditto
What if I replaced gas A with B? What if I operated in a different mode?	→	what-if? exploratory questions
I want to design a new facility	→	design and optimisation



source: Chan et al. (2018), JPP, 34(6) [3]

How can we analyse impulse facilities? a categorisation of modelling approaches

Physical fidelity

- state-to-state estimates built from *simple/ idealised* flow processes
- space-time discretisations of the governing equations

Time to answer

- fractions of sections on a desktop
- 1000s of CPU hours on a parallel-processing high-performance computer

Geometry representation

- abstracted: lengths, area ratios
- quasi-one-dimensional
- two dimensionsal: planar and *axisymmetric*
- three dimensional

Information inputs

- inputs on nominal information
- inputs on measured information (*a priori*)
- data fusion approaches (*a posteriori*)

Applicability

- specialist tool
- general-purpose tool

How can we analyse impulse facilities? Overview of our toolkit

The Gasdynamics Toolkit (GDTk) is a collection of software tools for analysing hypersonic flow

- Developed over 30 years here in Australia, at UQ and UniSQ
- Open-source and built with transparency and reproducibility in mind

GDTk tools for analysing impulse facilities

ESTCN	RST shot processing. State-to-state eq chemistry.
NENZF1d	RST shot processing. Q1D nozzle calc with full nonequilibrium
PITOT3	XT and RST full facility calc with eq chemistry.
L1d	1D Lagrangian full facility sim with noneq and viscous losses
Eilmer	2D/3D CFD with full nonequilibrium and boundary layers

Project Website: [4]

ESTCn and Friends

Lots of activity in Australia over the years modelling Reflected Shock Tunnels

- 1960's: ESTC, NENZF (McIntosh @ ANU and Lordi et al. @ Cornell)
- 1990's: STN (P. Jacobs and Krek @ UQ)
- 2010's: ESTCj, NENZFr (P. Jacobs and many others @ UQ)
- Current: ESTCN, NENZF1d, Eilmer

ESTCn and Friends

Lots of activity in Australia over the years modelling Reflected Shock Tunnels

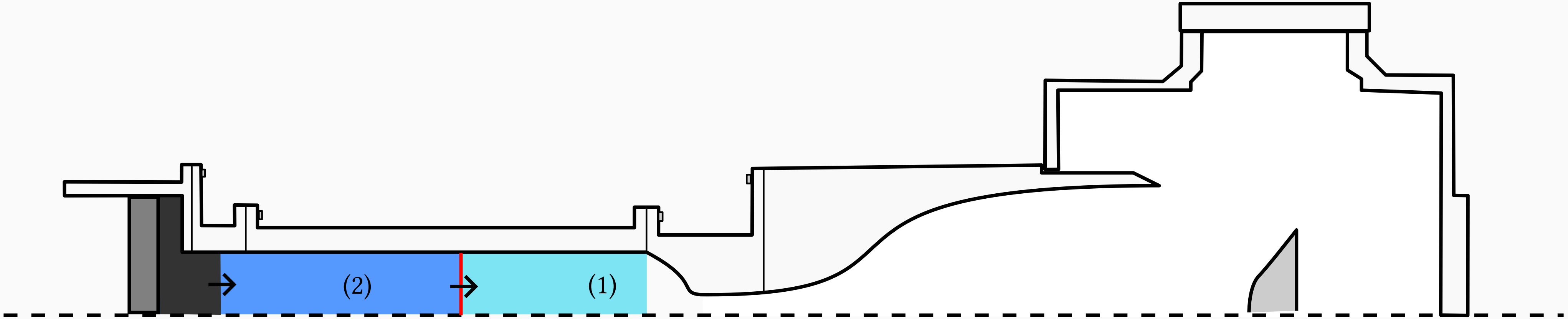
- 1960′s: ESTC, NENZF (McIntosh @ ANU and Lordi et al. @ Cornell)
- 1990′s: STN (P. Jacobs and Krek @ UQ)
- 2010′s: ESTCj, NENZFr (P. Jacobs and many others @ UQ)
- Current: ESTCN, NENZF1d, Eilmer

"Equilibrium Shock Tube Code" is really an *approach* that keeps getting reimplemented:

- There's an up to date version in the GDTk called ESTCN
- Let's look at how it works

The Important Stuff begins with an incident shock in the shock tube:

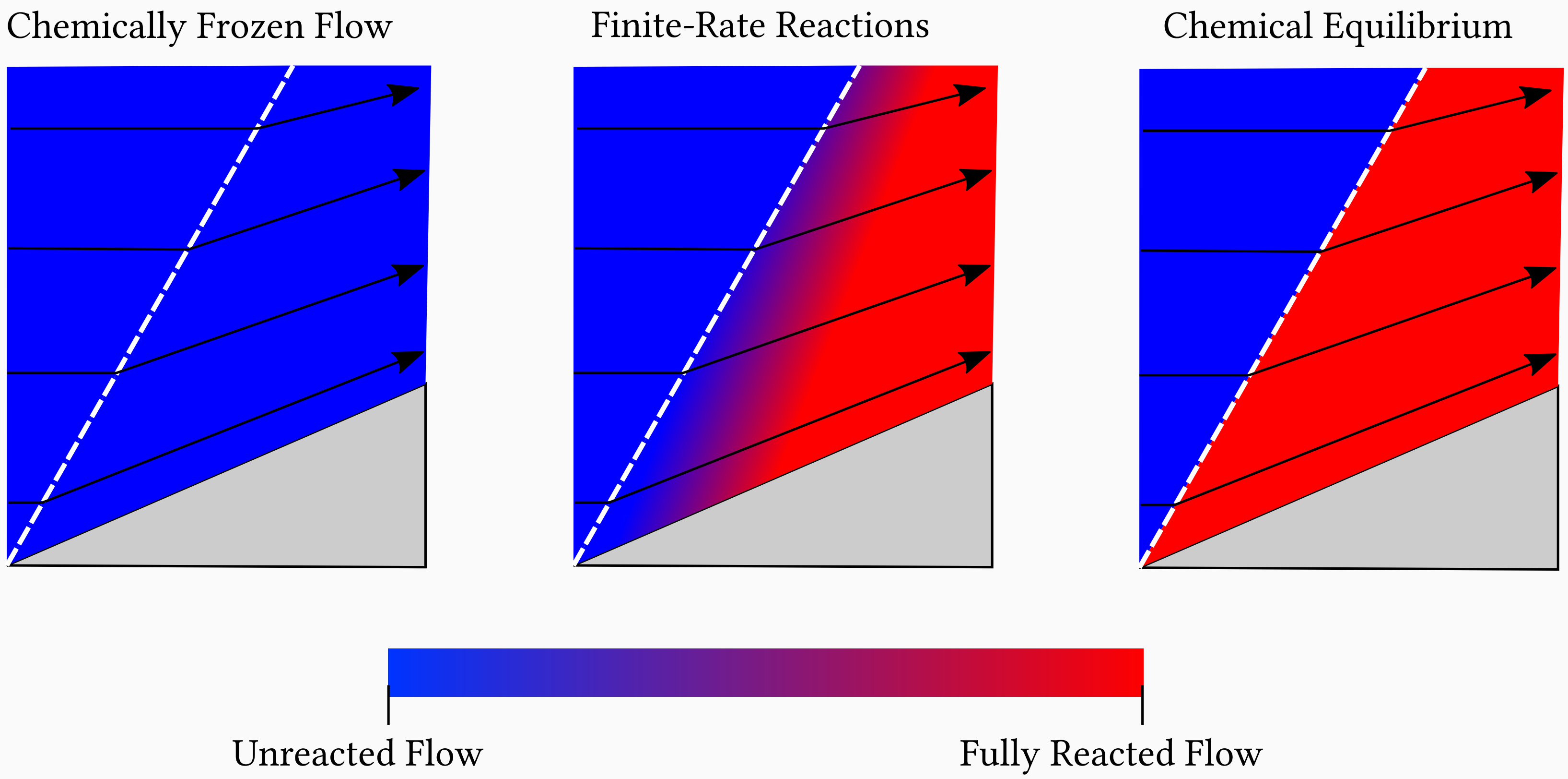
- Experimentalists measure the incident shock speed
- Initial fill condition of state 1 is known
- Equilibrium shock jump relations solved to get state (2)



Interlude: Equilibrium Shock Jump Relations

Impulse facility shocks are often quite strong:

- Flow chemistry occurs behind the shock, affecting its properties



Interlude: Equilibrium Shock Jump Relations

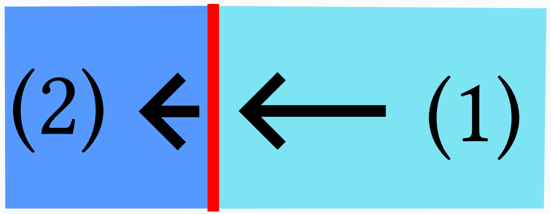
The Rankine-Hugoniot relations are still valid, but they need coupling to chemistry:

- Conservation of mass, momentum and energy and the shock:
- Remember that these are in the shock centered reference frame!

$$\rho_2 V_2 = \rho_1 V_1$$

$$\rho_2 V_2^2 + p_2 = \rho_1 V_1^2 + p_1$$

$$h_2 + \frac{1}{2} V_2^2 = h_1 + \frac{1}{2} V_1^2$$



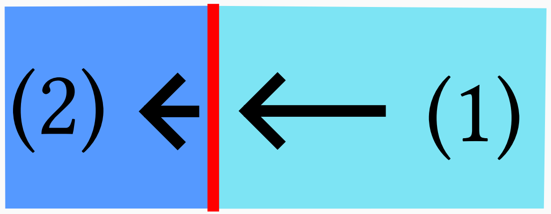
The Rankine-Hugoniot relations are still valid, but they need coupling to chemistry:

- Solve iteratively by first guessing V_2 the postshock speed

$$\rho_2 = \frac{\rho_1 V_1}{V_2}$$

$$p_2 = \left(\rho_1 V_1^2 + p_1\right) - \rho_2 V_2^2$$

$$e_2 = \frac{\left(h_1 + \frac{1}{2} V_1^2\right) - \frac{1}{2}\rho_2 V_2^3 - p_2 V_2}{\rho_1 v_1}$$



Interlude: Equilibrium Shock Jump Relations

The Rankine-Hugoniot relations are still valid, but they need coupling to chemistry:

- Solve iteratively by first guessing V_2 the postshock speed
- Rankine-Hugoniot equations give a speculative ρ_2, p_2, and e_2
- Now we need an equilibrium chemistry solver...

Interlude: Equilibrium Shock Jump Relations

The Rankine-Hugoniot relations are still valid, but they need coupling to chemistry:

- Solve iteratively by first guessing V_2 the postshock speed
- Rankine-Hugoniot equations give a speculative ρ_2, p_2, and e_2
- Now we need an equilibrium chemistry solver…

In the past we have used NASA CEA for this, but now we have a built in tool

- equilibrium-c, written by me! [5]
- My talk at the ISSW35 is all about it: (Monday at 1040am, Hawken 50-C207)

Interlude: Equilibrium Shock Jump Relations

The Rankine-Hugoniot relations are still valid, but they need coupling to chemistry:

- Solve iteratively by first guessing V_2 the postshock speed
- Rankine-Hugoniot equations give a speculative ρ_2, p_2, and e_2
- Now we need an equilibrium chemistry solver…

In the past we have used NASA CEA for this, but now we have a built in tool

- equilibrium-c, written by me! [5]
- My talk at the ISSW35 is all about it: (Monday at 1040am, Hawken 50-C207)

Use internal energy e_2 and density ρ_2 to evaluate the equilibrium mole fractions:

$$X_2^s(\rho_2, e_2) = [X^{N_2}, X^{O_2}, X^N, ...]$$

Interlude: Equilibrium Shock Jump Relations

The Rankine-Hugoniot relations are still valid, but they need coupling to chemistry:

- Solve iteratively by first guessing V_2 the postshock speed
- Rankine-Hugoniot equations give a speculative ρ_2, p_2, and e_2
- Consult equilibrium chemistry solver for $X_2^s(\rho_2, e_2)$

Now we calculate *another* pressure:

$$p_{\text{eos}} = \rho_2 \sum_s \frac{R_u}{M_s X_2^s} T_2$$

The Rankine-Hugoniot relations are still valid, but they need coupling to chemistry:

- Solve iteratively by first guessing V_2 the postshock speed
- RH equations give a speculative ρ_2, p_2, and e_2
- Consult equilibrium chemistry solver for $X_2^s(\rho_2, e_2)$
- Equation of state gives a second pressure p_{eos}

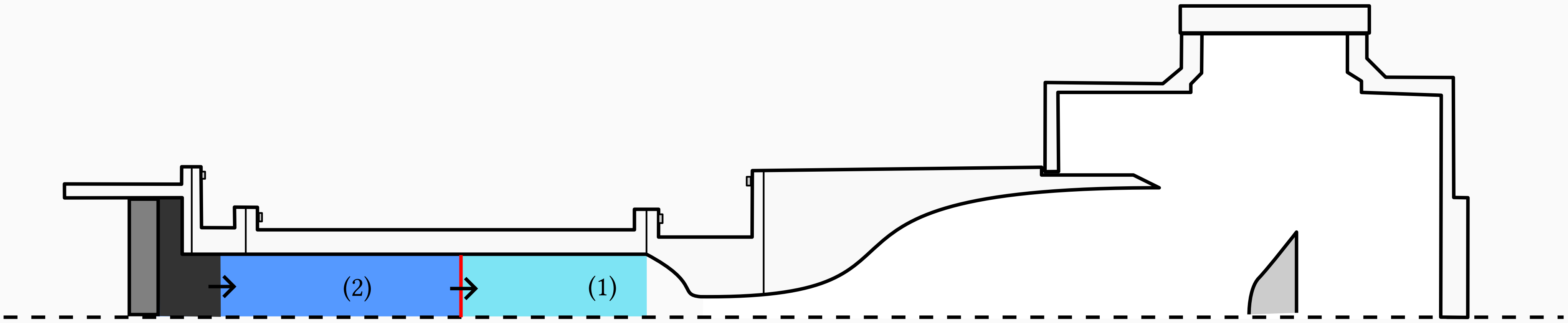
Iteratively adjust the guess V_2 until p_2 and p_{eos} are the same:

$$\frac{p_2 - p_{\text{eos}}}{p_2} \to 0$$

Step 1: Incident Shock Calculation

Example, T4 shot number 11760:

- Shock tube filled 194 kPa at T=300 K, shock speed measured at 2241 m/s
- Enthalpy: 4.6 MJ/kg, Flight Mach Number: $\approx$ 9.7, Flight Alt $\approx$ 33 km

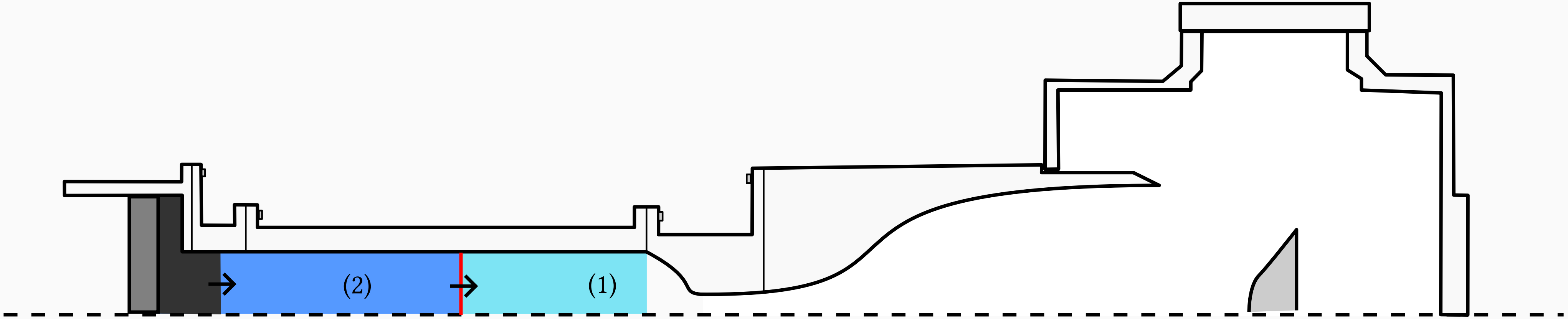


Step 1: Incident Shock Calculation

Example, T4 shot number 11760:

- Shock tube filled 194 kPa at T=300 K, shock speed measured at 2241 m/s
- Enthalpy: 4.6 MJ/kg, Flight Mach Number: ≈ 9.7, Flight Alt ≈ 33 km

State 2:	p (kPa)	T (K)	v (m/s)	NOX %	O %
Equilibrium	9701	2337	349.3	1.73	0.15
Ideal Gas	9359	2695	417.4	0.0	0.0
error	3.5%	15.0%	19.5%	100%	100%

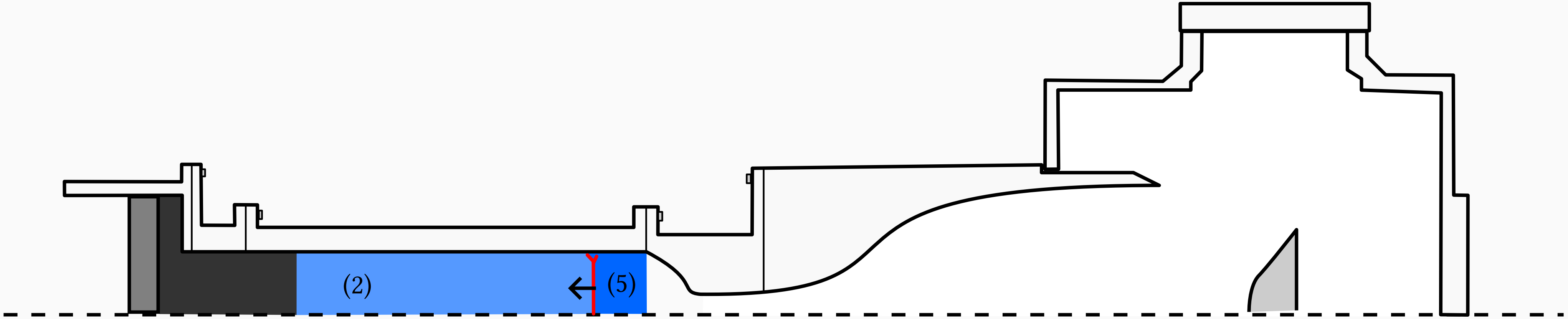


Step 2: Shock Reflection

This bit is a little more complicated

- Similar algorithm to the incident shock, but with fixed V_2=0 in the lab frame

	p (kPa)	T (K)	v (m/s)	NOX %	O %
State 1	194	300	0	0	0
State 2	9712	2338	349	1.75	0.15
State 5	76,469	4126	0	10.8	1.4



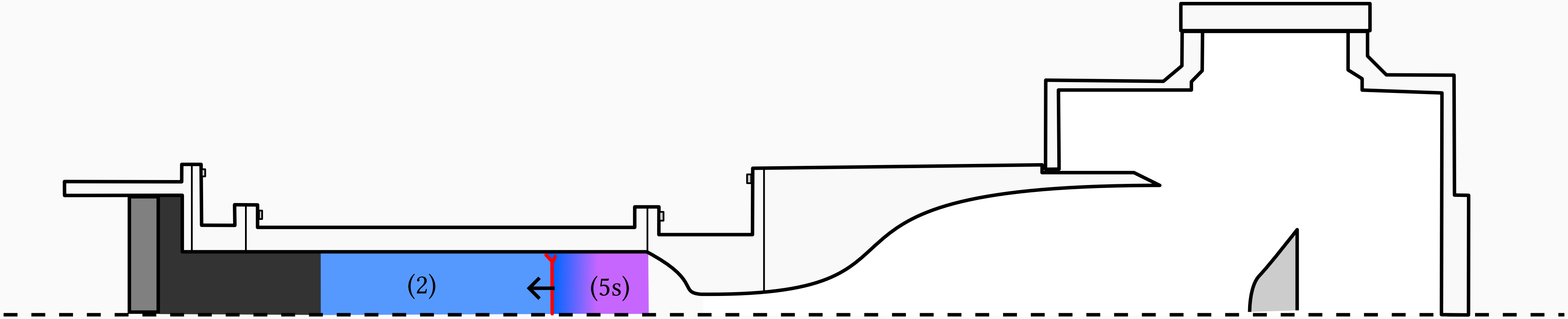
Step 3: Correct Stagnation Pressure

T4 and other Shock Tunnels have pressure sensors in the stagnation region

- We need to correct the perfectly stagnated state 5 using an isentropic expansion
- Most equilibrium solvers have a known pressure and entropy mode
- The cause of this discrepancy is debated (Any ideas from people in the room?)

$$\text{Known: } p_{5s} = p_e \qquad s_{5s} = s_5$$

$$\text{Compute: } T_{5s}(p_{5s}, s_5) \qquad X_{5s}(p_{5s}, s_5)$$

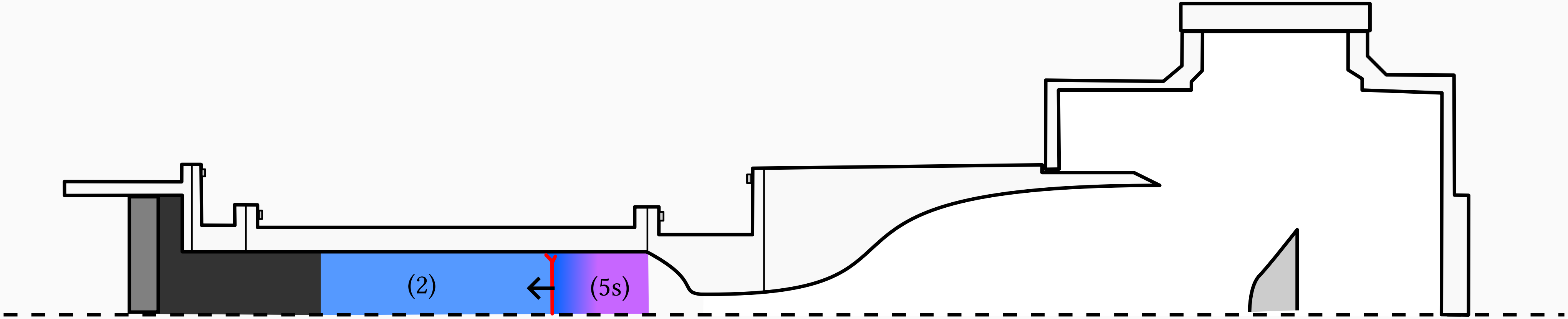


Step 3: Correct Stagnation Pressure

T4 and other Shock Tunnels have pressure sensors in the stagnation region

- We need to correct the perfectly stagnated state 5 using an isentropic expansion

	p (kPa)	T (K)	v (m/s)	NOX %	O %
State 1	194	300	0	0	0
State 2	9712	2338	349	1.75	0.15
State 5	76,469	4126	0	10.8	1.4
State 5s	45,000	3768	0	8.9	0.9



Step 4: Isentropic Expansion to Mach 1

From here we need to compute the nozzle, starting with the converging section

- Once again we use ps mode of our equilibrium solver
- Guess p_6 and iterate until $M_6 = 1$

Step 4: Isentropic Expansion to Mach 1

From here we need to compute the nozzle, starting with the converging section

- Once again we use ps mode of our equilibrium solver
- Guess p_6 and iterate until $M_6 = 1$

$$X_6^s = \text{eqsolve}(p_6, s_{5s}) \qquad T_6 = \text{eqsolve}(p_6, s_{5s})$$

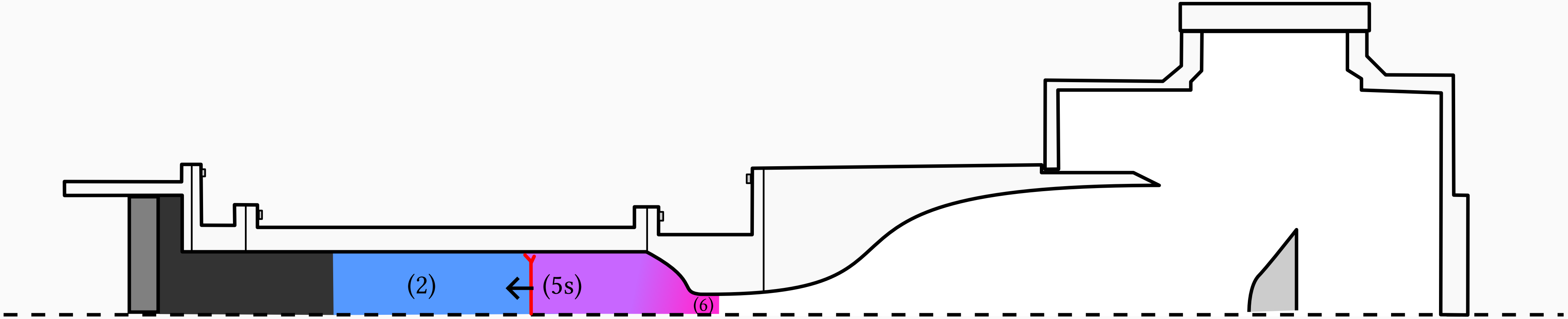
$$M_6 = \sqrt{\frac{2h_{5s} + V_{5s}^2 - 2h(X_6, T_6)}{\gamma(X_6)R(X_6)T_6}}$$

$$M_6(p_6) - 1 = 0$$

Step 5: Isentropic Expansion to Mach 1

From here we need to compute the nozzle, starting with the converging section:

	p (kPa)	T (K)	v (m/s)	NOX %	O %
State 1	194	300	0	0	0
State 2	9712	2338	349	1.75	0.15
State 5	76,469	4126	0	10.8	1.4
State 5s	45,000	3768	0	8.9	0.9
State 6	24,531	3391	1122	6.8	0.7

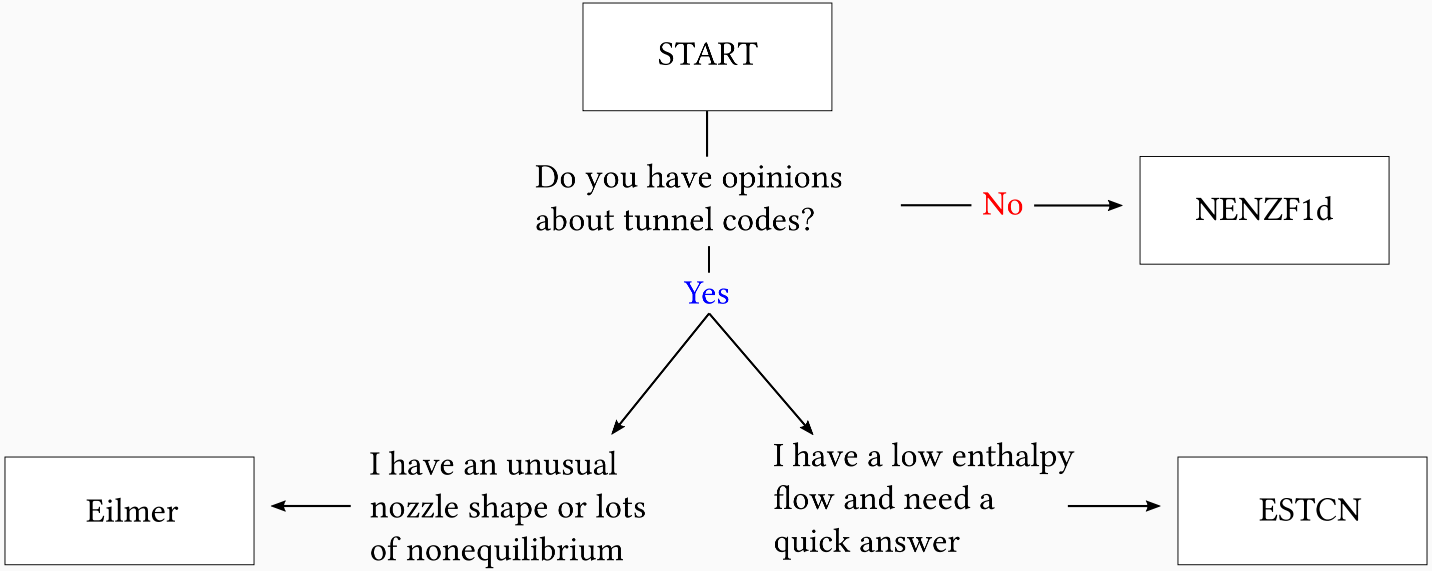


Step 6: Supersonic Expansion

This is where the calculation starts to get interesting:

- Expanding supersonic flows can have chemical and thermal nonequilibrium
- Viscous boundary layers and turbulence on the nozzle wall
- Some nozzle shapes can induce significant non-uniformity

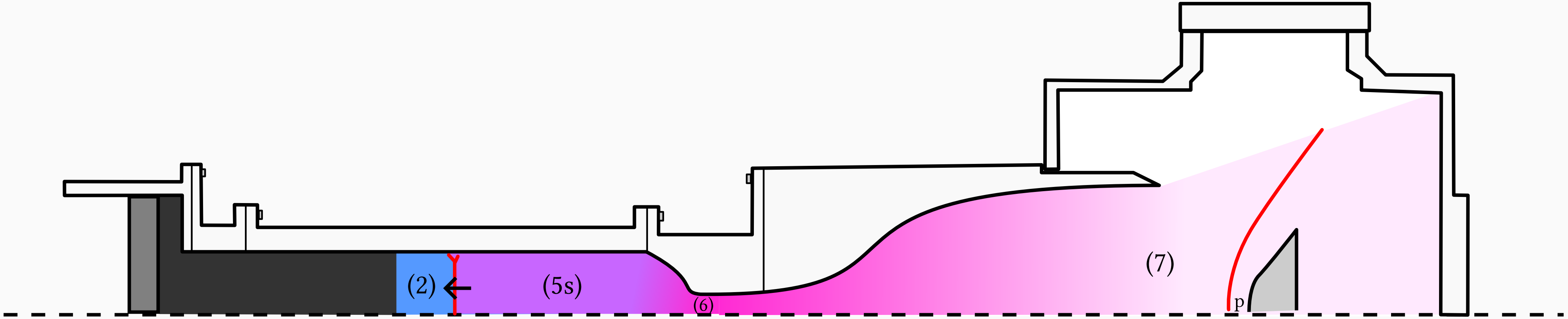
We have several different tools, with different levels of fidelity and run time:



Step 6: Isentropic Expansion with Known Pitot/Stag Ratio

If we have an actual pitot probe we can extend our current approach:

- Calculate the pitot/stag ratio and expand the flow until we reach it



Step 6: Isentropic Expansion with Known Pitot/Stag Ratio

If we have an actual pitot probe we can extend our current approach:

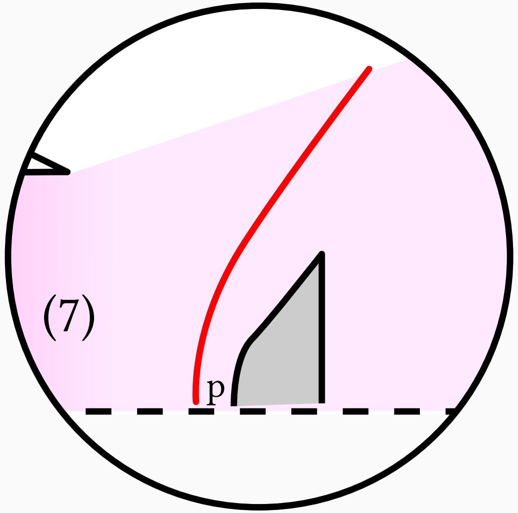
- Calculate the pitot/stag ratio and expand the flow until we reach it
- Once again we guess at p_7 and iterate

$$X_7^s = \text{eqsolve}(p_7, s_6) \qquad T_7 = \text{eqsolve}(p_7, s_6)$$

$$M_7 = \sqrt{\frac{2h_{5s} + V_{5s}^2 - 2h(X_7, T_7)}{\gamma(X_7)R(X_7)T_7}}$$

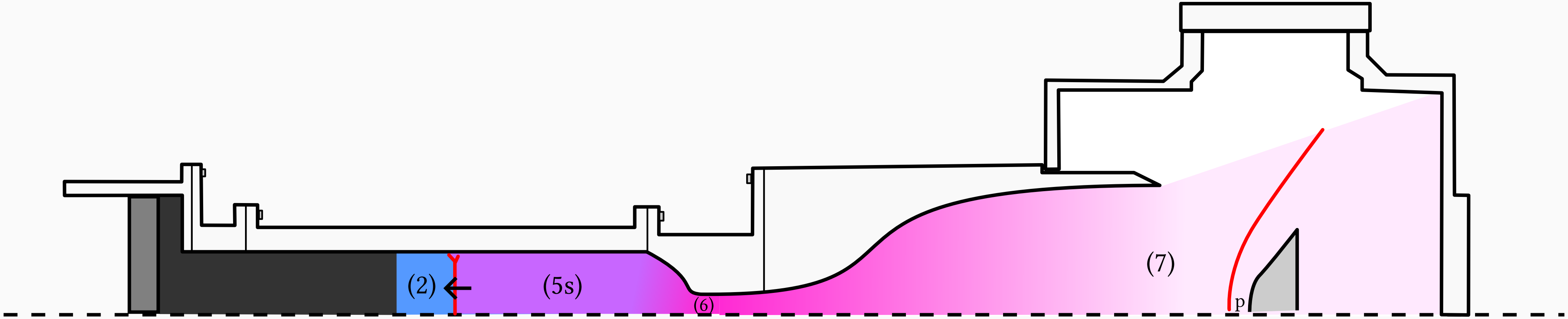
$$p_p = \text{normalshock}(M_7, p_7, T_7)$$

$$\frac{p_p(p_7)}{p_{5s}} - 7.01 \times 10^{-3} = 0$$



Step 6: Isentropic Expansion with Known Pitot/Stag Ratio

	p (kPa)	T (K)	v (m/s)	NOX %	O %
State 1	194	300	0	0	0
State 2	9712	2338	349	1.75	0.15
State 5	76,469	4126	0	10.8	1.4
State 5s	45,000	3768	0	8.9	0.9
State 6	24,531	3391	1122	6.8	0.7
State 7	4.956	466.8	3005	0	0



Alternate Step 6: Quasi-1D Space Marching with NENZF1d

NENZF1d is a program for fast Reflected Shock Tunnel Analysis:

- Includes thermochemical nonequilibrium in the expanding part of the nozzle
- Solves in seconds or less using a 1D space marching approach
- Accuracy comparable to CFD in a good nozzle

See Peter Jacobs' 2020 CfH Seminar: [6]

Calculation of Test Flow Conditions for Reflected-Shock Tunnels

Peter Jacobs, Wilson Chan, Tamara Sopek, Fabian Zander, Kyle Damm, Nick Gibbons, Rowan Gollan

The University of Queensland (at one time or another)

03 Dec 2020

Link:

Alternate Step 6: Quasi-1D Space Marching with NENZF1d

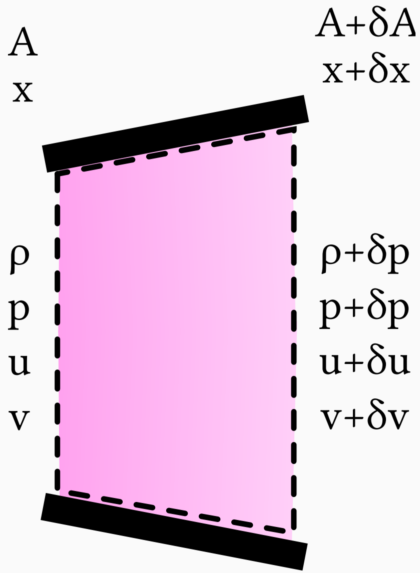
NENZF1d uses the state-to-state equilibrium up to the nozzle throat

- From there, choose a timestep and march in small increments along the nozzle
- Each increment has a quasi-1D control volume

$$0 = \rho A \delta v + v A \delta \rho + \rho v \delta A$$

$$0 = \rho v \delta v + \delta p$$

$$0 = v E A \delta \rho + (\rho E + p) A \delta v + \rho v A \delta e + (\rho E + p) v \delta A$$



Where: $E = e + \frac{1}{2} v^2$ and $\delta E = \delta e + v \delta v$

Alternate Step 6: Quasi-1D Space Marching with NENZF1d

These 3 equations are linear in the 4 increments δ:

- Add in the perfect gas equation of state for another equation
- We can solve a linear system for each small control volume

$$\begin{pmatrix} vA & \rho A & 0 & 0 \\ 0\rho v & 1 & 0 & \\ vEA & \rho(E+p)A & 0 & \rho v A \\ RT & 0 & -1 & \frac{\rho R}{c_v} \end{pmatrix} \cdot \begin{pmatrix} \delta\rho \\ \delta v \\ \delta p \\ \delta e \end{pmatrix} = \begin{pmatrix} -\rho v \delta A \\ 0 \\ -v(\rho E + p)\delta A \\ 0 \end{pmatrix}$$

Alternate Step 6: Quasi-1D Space Marching with NENZF1d

But wait! Aren't we trying to include thermochemical nonequilibrium?

- Run an adiabatic-fixed mass reactor for a short amount of time dt
- Split the pressure increment into $\delta p = \delta p_{\text{chem}} + \delta p_{\text{gas}}$
- Tv and chemical composition changes are loosely coupled into the marching

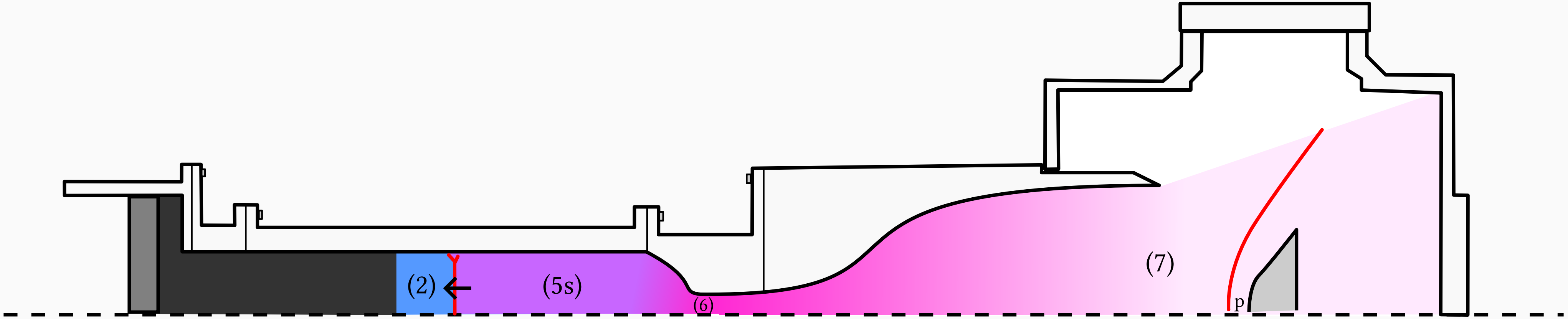
$$\begin{pmatrix} vA & \rho A & 0 & 0 \\ 0\rho v & 1 & 0 & \\ vEA & \rho(E+p)A & 0 & \rho v A \\ RT & 0 & -1 & \frac{\rho R}{c_v} \end{pmatrix} \cdot \begin{pmatrix} \delta\rho \\ \delta v \\ \delta p_{\text{gas}} \\ \delta e \end{pmatrix} = \begin{pmatrix} -\rho v \delta A \\ -\delta p_{\text{chem}} \\ -v(\rho E + p)\delta A \\ 0 \end{pmatrix}$$

Alternate Step 6: Quasi-1D Space Marching with NENZF1d

Example, T4 shot number 11760:

- Shock tube filled 194 kPa at T=300 K, shock speed measured at 2241 m/s
- Enthalpy: 4.6 MJ/kg, Flight Mach Number: ≈ 9.7, Flight Alt ≈ 33 km

State 7:	p (kPa)	T (K)	v (m/s)	NOX %	O %
ESTCN	4.956	466.8	3005	0	0
NENZF1d	4.65	426.5	2955	6	4e-11
error	6.6%	9.5%	1.7%	100%	100%

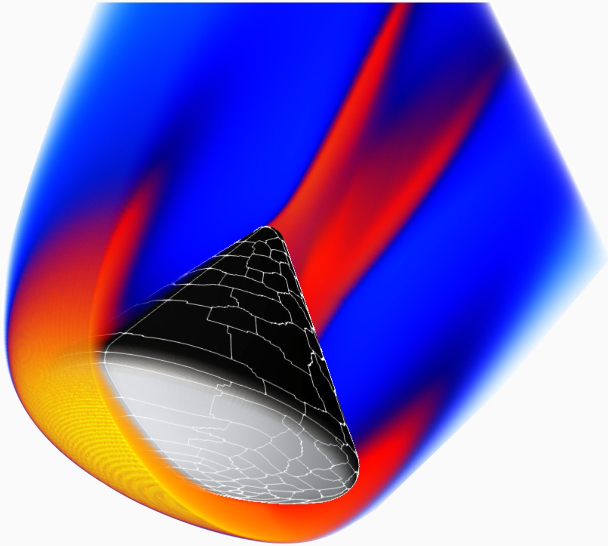


Final Alternate Step 6: 2D Axisymmetric Simulation using Eilmer

Eilmer is our general purpose compressible flow solver:

- Free and open-source
- Structured and Unstructured grids in 2D/3D
- Very capable nonequilibrium thermochemistry models
- Extensively validated against decades of experiments
- Scales on thousands of cores for LES/DNS

Link to CPC Paper: [7]



Apollo capsule at 18.6° angle of attack

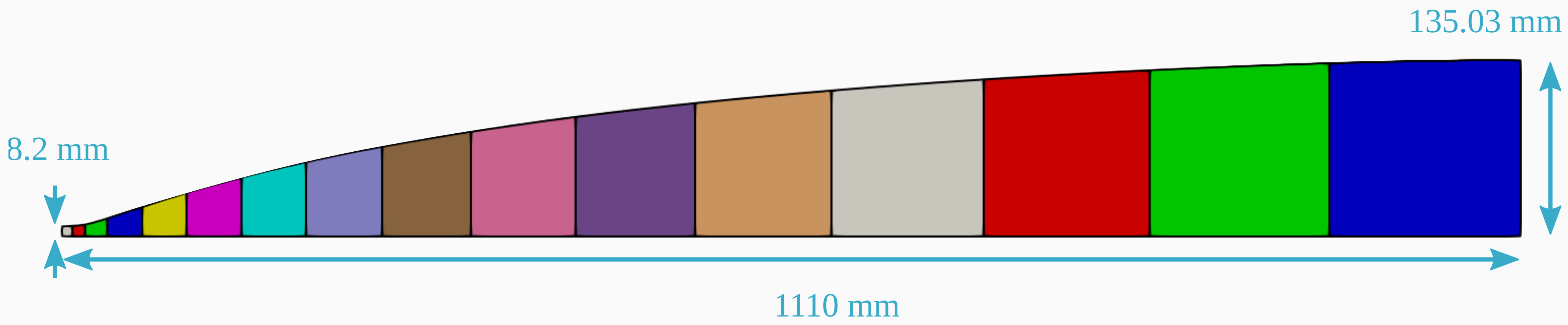
Final Alternate Step 6: 2D Axisymmetric Simulation using Eilmer

Shot 11760 used T4's Mach 8 Nozzle:

- There's a demo simulation of this in the Eilmer examples
- Let's try it!

Simulation Details:

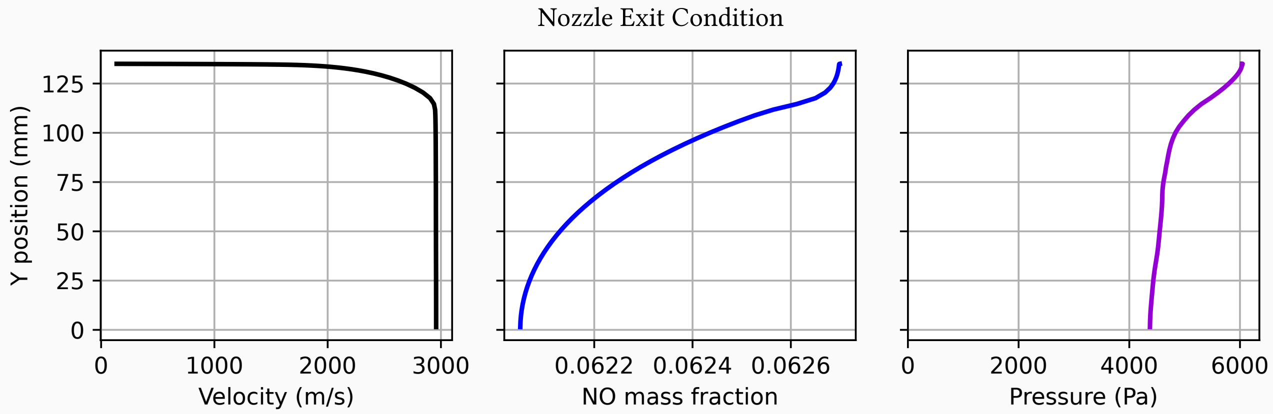
Solver Mode	Gas Model	Turbulence	Grid	n Blocks
Steady Axi-2D	6sp Air 1T	Spalart-Allmaras	191x71	16

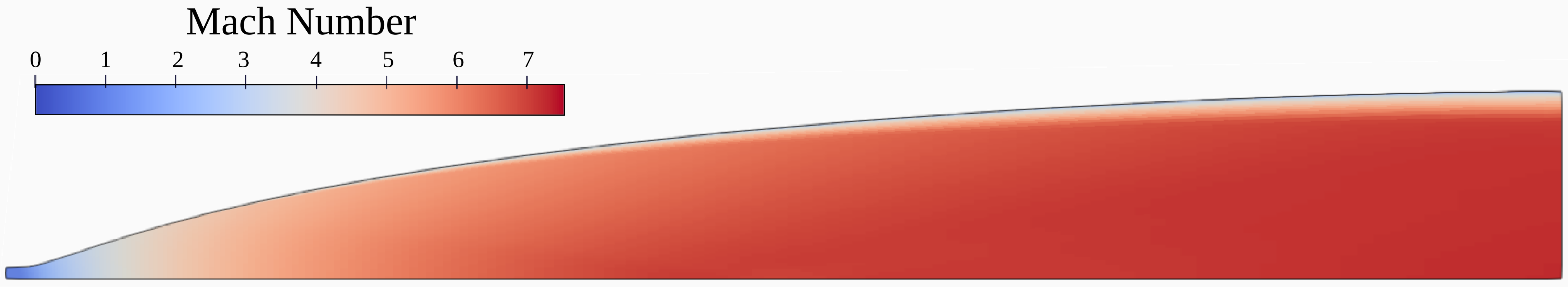


Another Alternate Step 6: 2D Axisymmetric Simulation using Eilmer

Runtime Details:

- Sim Time: 1992 seconds
- CPU: AMD PRO 3955WX x16
- Termination GRR: 1e-12





Final Comparison

We can cut out the core flow and 1Dify it to compare to NENZF1d

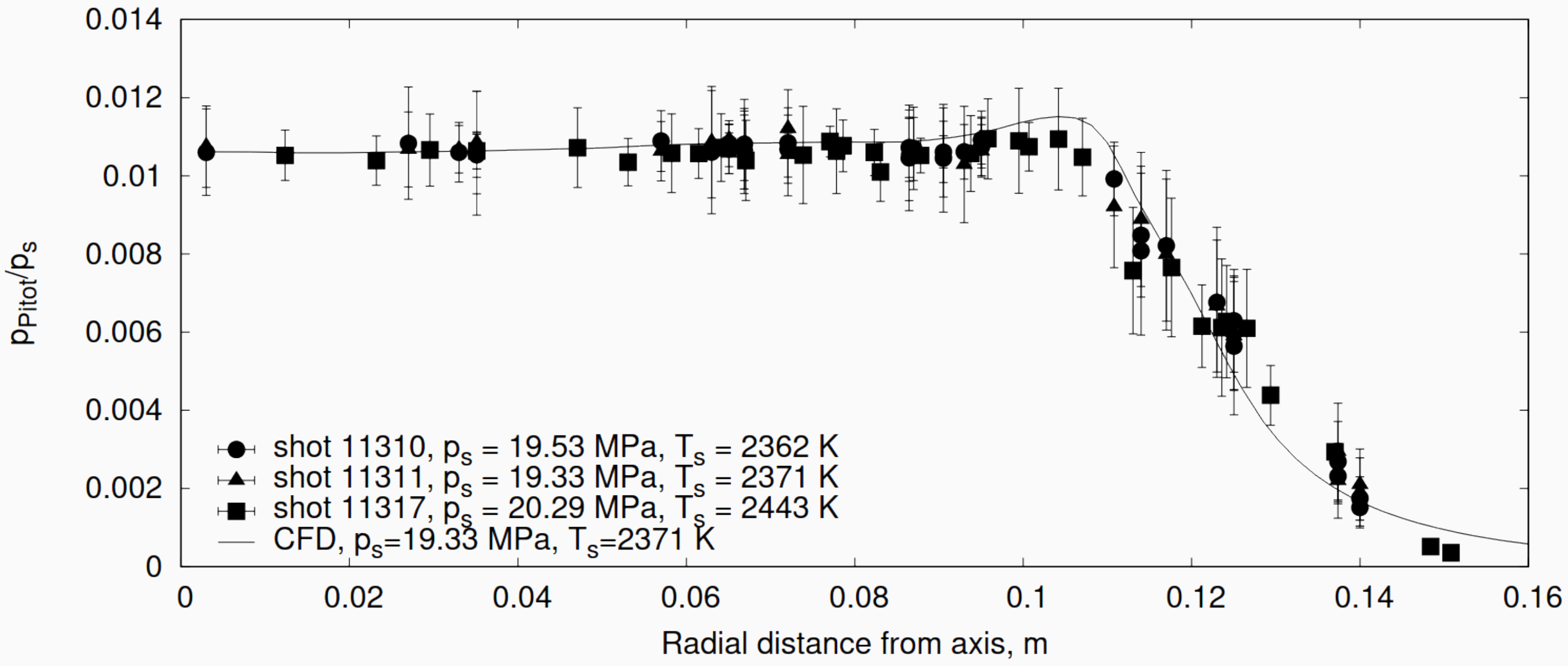
- Matches pretty nicely
- Disclaimer: I did use the predicted pitot/stag from the CFD

State 7:	p (kPa)	T (K)	v (m/s)	NOX %	O %
Eilmer	4.67	427.0	2955.0	6.21	4.27e-11
NENZF1d	4.65	426.5	2955.4	6.19	4.24e-11
error	0.4%	0.1%	0.0%	0.3%	0.7%
ESTCN	4.96	466.8	3005	0	0
error	6.1%	9.3%	1.7%	100%	100%

Does it match experiments?

Some testing of the nozzle CFD by Wilson Chan and PJ [3] shows good agreement:

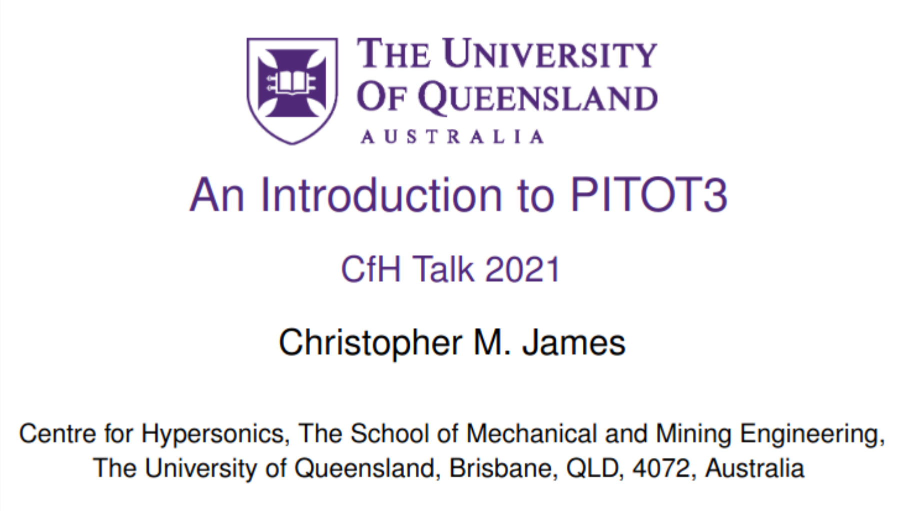
- Pitot rake measurements have pretty large error bars



PITOT is a more flexible state-to-state calculator written and maintained by Dr. Chris James:

- Python code with composable class structure
- Shock Tubes, Expansion Tubes, Diaphragms and Pistons
- Very useful for developing new conditions and other off-design behaviour

See Chris's CfH talk on the website: [8]



PDF Link:

Overview of L1d

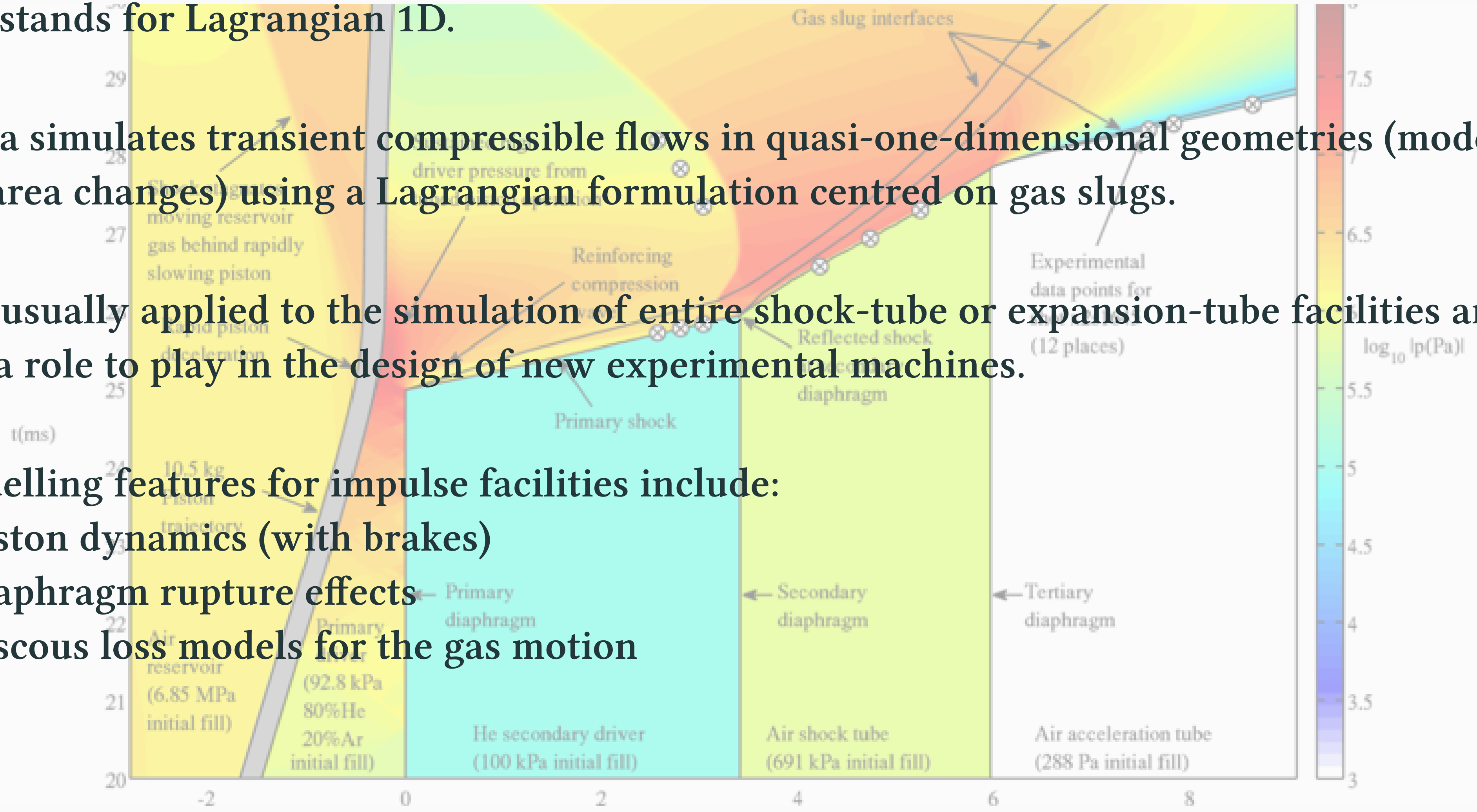
L1d stands for Lagrangian 1D.

It is a simulates transient compressible flows in quasi-one-dimensional geometries (moderate area changes) using a Lagrangian formulation centred on gas slugs.

It is usually applied to the simulation of entire shock-tube or expansion-tube facilities and has a role to play in the design of new experimental machines.

Modelling features for impulse facilities include:

- piston dynamics (with brakes)
- diaphragm rupture effects
- viscous loss models for the gas motion



L1d: input file demo

```
# exptube.py
config.title = 'Expansion-tube demo, 2020-05-20'
my_gm = add_gas_model('ideal-air-gas-model.lua')

# Define the tube wall.
add_break_point(0.0, 0.02)
add_break_point(3.0, 0.02)

# Create the gas-path.
left_wall = VelocityEnd(x0=0.0, vel=0.0)
driver_gas = GasSlug(p=1.0e6, vel=0.0, T=3000.0, gmodel_id=my_gm, ncells=200)
interface = GasInterface(x0=0.5)
driven_gas = GasSlug(p=5.0e3, vel=0.0, T=300.0, gmodel_id=my_gm, ncells=100)
dia = Diaphragm(x0=1.0, p_burst=10.0e3)
accel_gas = GasSlug(p=100.0, vel=0.0, T=300.0, gmodel_id=my_gm, ncells=300)
free_end = FreeEnd(x0=2.5)
assemble_gas_path(left_wall, driver_gas, interface, driven_gas,
                  dia, accel_gas, free_end)

# Set some time-stepping parameters
config.dt_init = 1.0e-8
config.max_time = 1.0e-3
config.max_step = 9000
add_dt_plot(0.0, 10.0e-6, 1.0e-6)
add_history_loc(0.90)
add_history_loc(2.45)
```

Eilmer: a brief history

- cns4u began life in 1990 (Peter Jacobs, ICASE @ NASA Langley)
- mb_cns started in mid 1990s (Peter Jacobs + grad students @ UQ)
- mbcns2 in early 2000s in C++, generalised thermochemistry modelling
- elmer/elmer2 in early 2000s, 3D code in C with Python
- merge mbcns2 and elmer2 ⟶ Eilmer v3 in Nov 2008, C++ core
- Eilmer v4 written in D programming language, began in 2014
- 2024: (soft) release of Eilmer v5

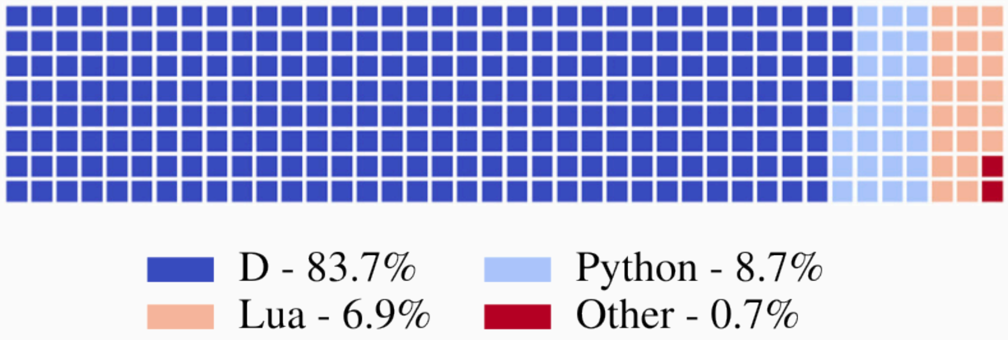
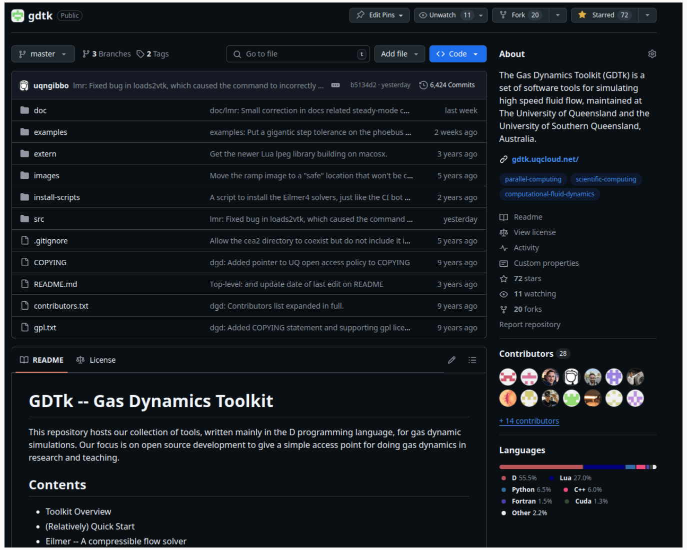


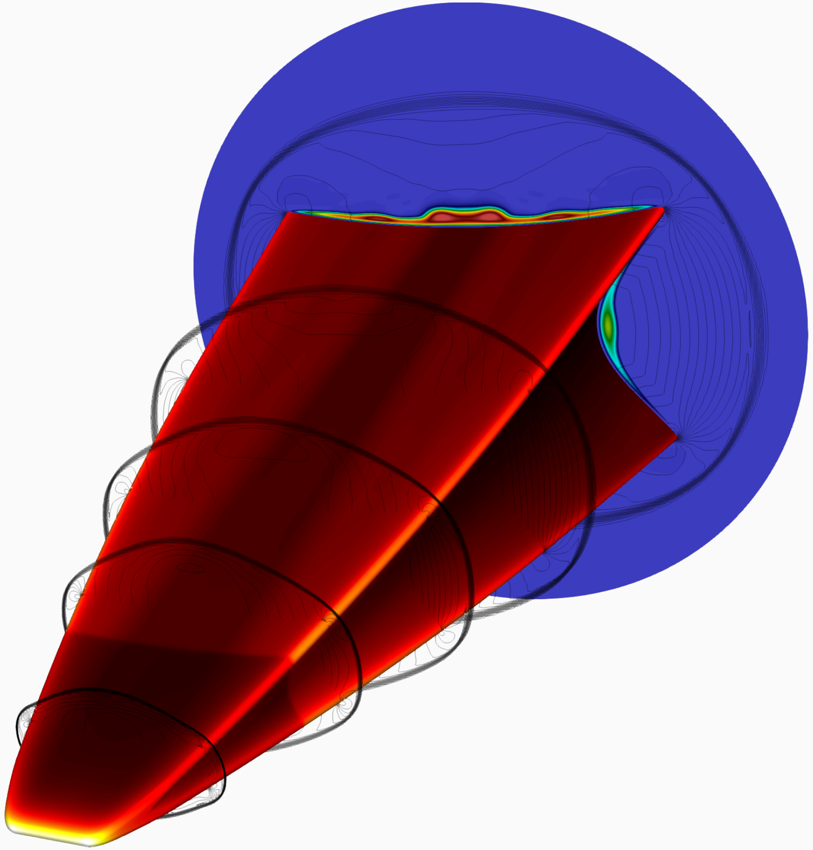
Fig. 4. Division of languages within Eilmer as of April 2022.



source: Gibbons et al. (2023) CPC, 282:108551 [7]

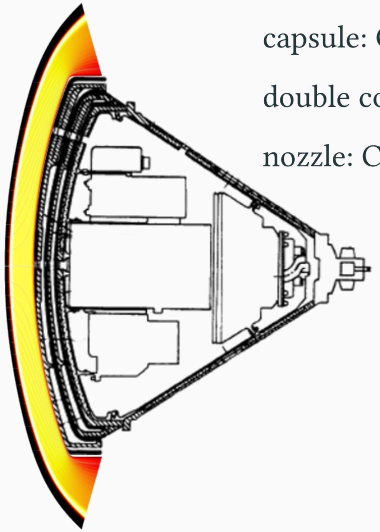
Eilmer: features and capabilities

- 2D/3D compressible flow simulation (on body-fitted grids)
- cell-centred finite volume
- 2nd-order with shock-capturing (routine); 4th-order for specialist work
- Gas models include: ideal, thermally perfect, multi-temperature and state-specific
- Kinetics: finite-rate chemistry and thermal nonequilibrium modelling
- Inviscid, laminar, turbulent flows
- Solid domains with conjugate heat transfer
- User-controlled moving grid capability
- Boundary shock-fitting method for blunt-body shock layers
- Transient, time-accurate updates with Runge-Kutta family integrators
- Steady-state accelerator using Newton-Krylov method
- User-defined customisations for boundary conditions, source terms, pre- and post-processing
- Parallel computations using shared memory or distributed memory (MPI)
- Native grid generation and 3rd-party import capability
- Unstructured mesh partitioning via Metis
- Adjoint solver for efficient sensitivities evaluation



Eilmer: what it's used for

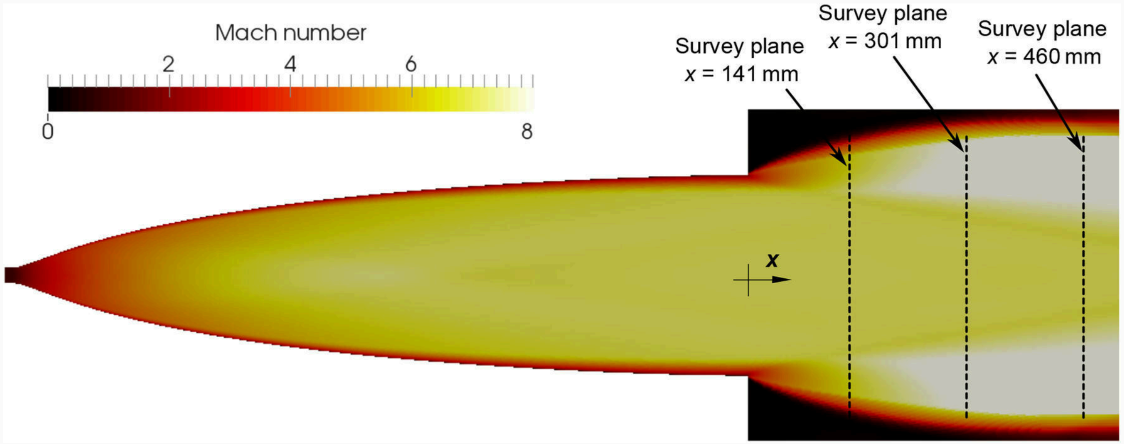
- Experiment support
- Flow physics investigation
- Flow path analysis, design and optimisation

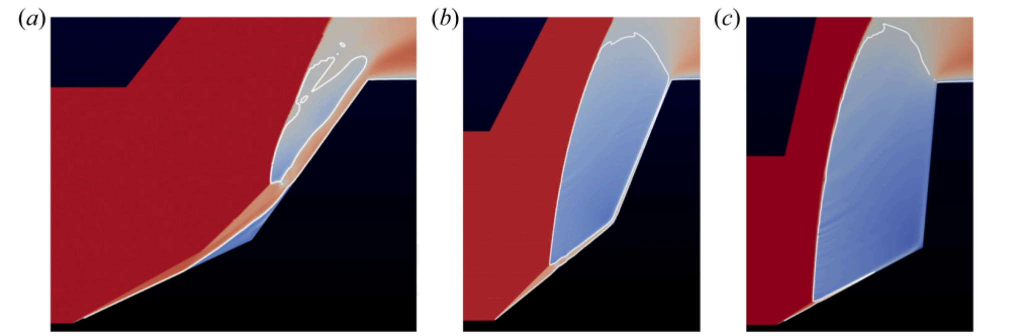


capsule: Gibbons et al. (2023) CPC, 282:108551 [7]

double cone: Hornung et al. (2021) JFM, 916:A5 [9]

nozzle: Chan et al. (2018) JPP, 34(6) [3]





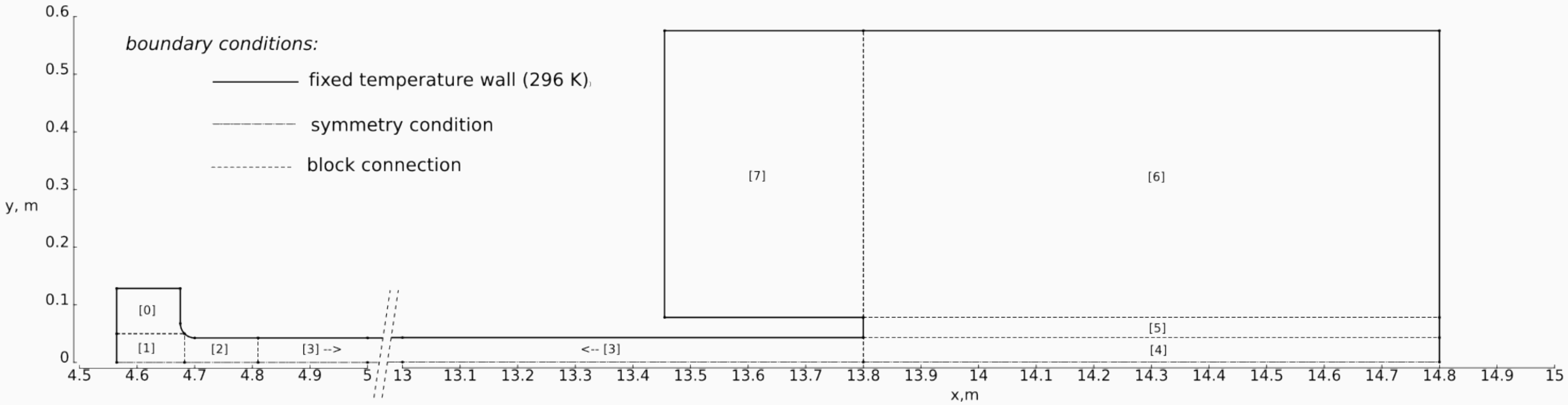
Eilmer for the simulation of impulse facilities

Tech specs:

- solves compressible Navier-Stokes equations in axially-symmetric geometries
- multi-block structured grids
- finite-volume; second-order in space with shock-capturing
- low-order Runge-Kutta family timesteppers for transient simulations
- Newton-Krylov accelerator for steady-state simulations

Features for impulse facility simulation:

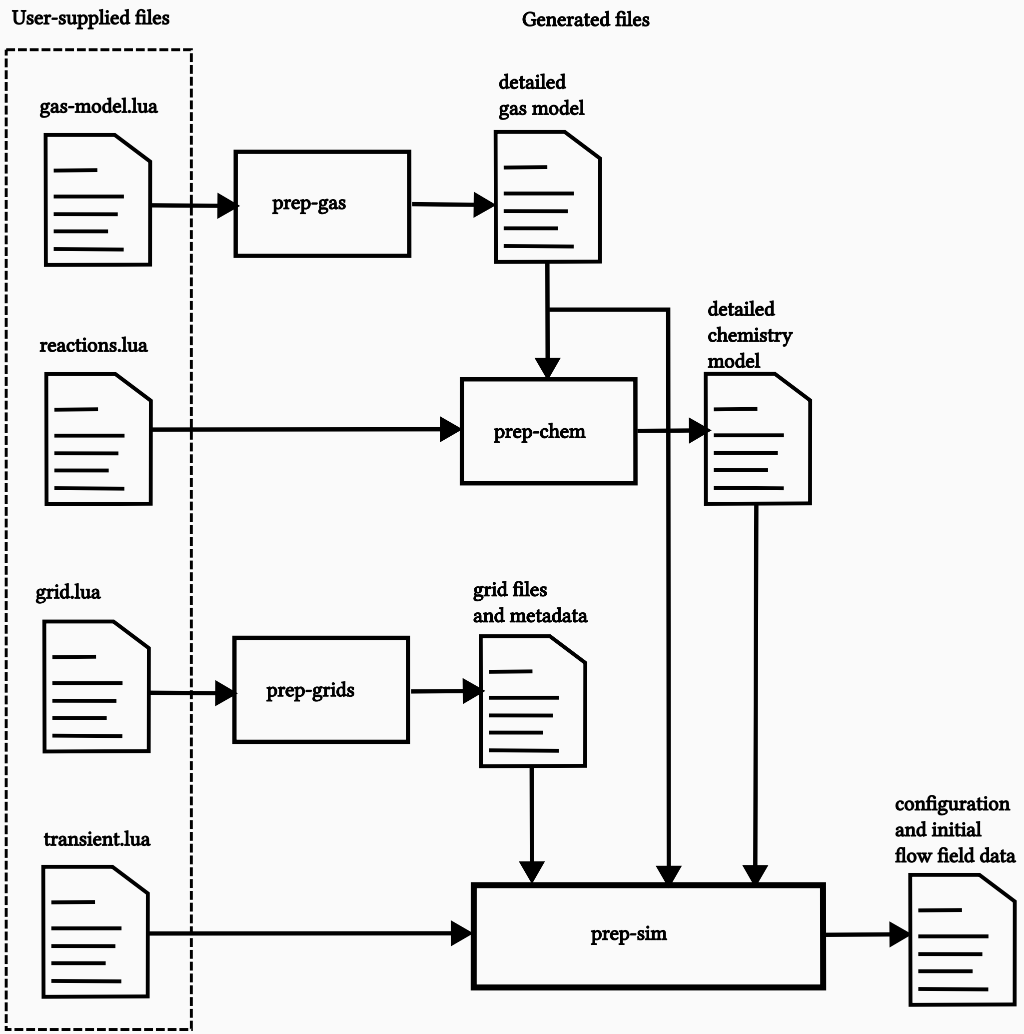
- adaptive flux calculators (for numerical stability near shocks)
- profile-type boundary conditions (for staging or hybrid L1d/eilmer sims)
- user-defined functions to add modelling behaviour, such as diaphragm rupture



Eilmer workflow

1. pre-processing
2. running a simulation
3. post-processing

```
$ lmr prep-gas
$ lmr prep-grid
$ lmr prep-sim
$ lmr snapshot2vtk --all
$ lmr help
```



Supersonic turbulent flow over a flat plate

`gdtk/examples/lmr/2D/turbulent-flat-plate`

Nick N. Gibbons 2024-02-26

This example is a turbulent, supersonic flow over a flat plate at Mach 6.5, from Ye and Morgan (1994). In T4, the boundary layer at this condition is actually transitional, though here it is treated as fully turbulent from the leading edge. Both structured and unstructured variants are available, and the one equation "Edwards" variant of the Spalart-Allmaras turbulence model is used.

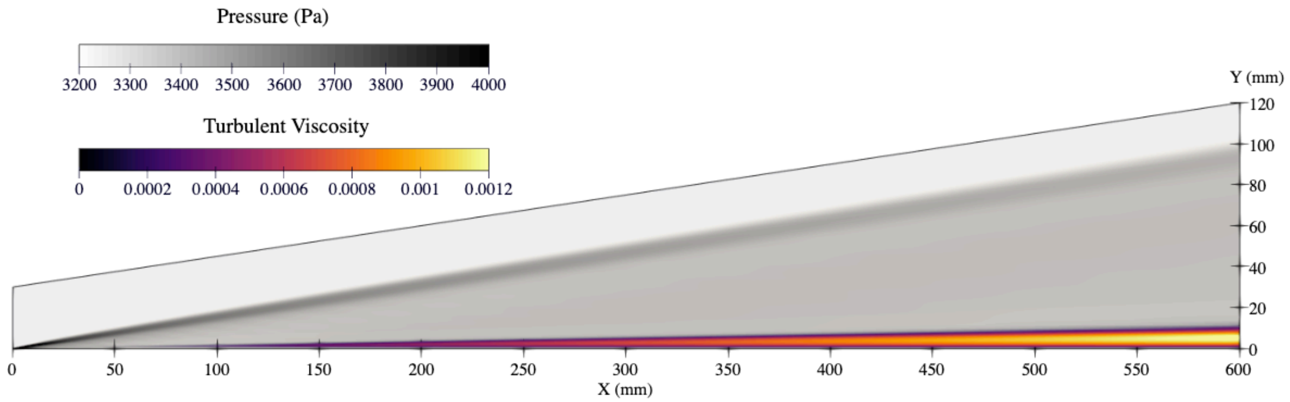
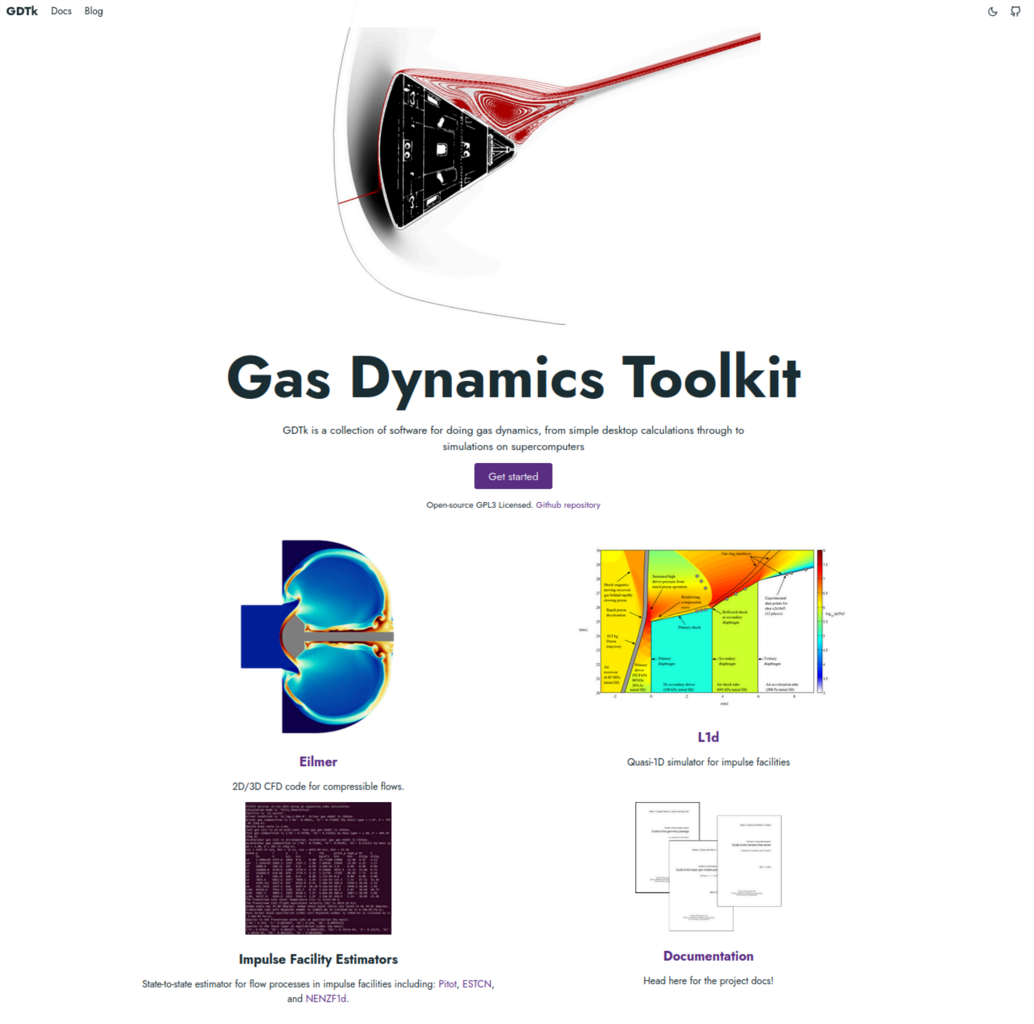


Figure 1. Pressure and turbulent viscosity color maps for the Mach 6.5 flat plate.

ON THIS PAGE

https://gdtk.uqcloud.net/docs/eilmer/examples-catalogue/

Where to find out more



https://gdtk.uqcloud.net

Thanks!

The GDTk Team

Peter Jacobs, Kyle Damm, Reece Otto and the team of PhD students in the CFD group

Peter Jacobs

a large number of the tools presented today were initiated, developed, documented and maintained under his custodianship for over 35+ years (and continues today)

Bibliography

[1] R. J. Goozée, P. A. Jacobs, and D. R. Buttsworth, "Simulation of a complete reflected shock tunnel showing a vortex mechanism for flow contamination," *Shock waves*, vol. 15, no. 3, pp. 165–176, 2006.

[2] R. J. Gollan, "Computataional modelling of high-temperature gas effects with application to hypersonics flows," 2008.

[3] W. Y. K. Chan, P. A. Jacobs, M. K. Smart, S. Grieve, C. S. Craddock, and L. J. Doherty, "Aerodynamic Design of Nozzles with Uniform Outflow for Hypervelocity Ground-Test Facilities," *Journal of Propulsion and Power*, vol. 34, no. 6, pp. 1467–1478, 2018, doi: 10.2514/1.B36938.

[4] P. Jacobs, R. Gollan, K. Damm, and N. Gibbons, "GDTk: The Gas Dynamics Toolkit." 2024.

[5] N. N. Gibbons, "equilibrium-c: A Lightweight Modern Equilibrium Chemistry Calculator for Hypersonic Flow Applications." 2024.

[6] P. Jacobs, "Calculation of Test Flow Conditions for Reflected-Shock Tunnels." [Online]. Available: https://gdtk.uqcloud.net/pdfs/eilmer-talk-pj-2020-dec.pdf

[7] N. N. Gibbons, K. A. Damm, P. A. Jacobs, and R. J. Gollan, "Eilmer: An open-source multi-physics hypersonic flow solver," *Computer Physics Communications*, vol. 282, no. 108551, 2023, doi: doi.org/10.1016/j.cpc.2022.108551.

[8] C. James, "An Introduction to Pitot3." [Online]. Available: https://gdtk.uqcloud.net/pdfs/cmj-cfh-talk-07-2021.pdf

[9] H. Hornung, R. Gollan, and P. Jacobs, "Unsteadiness boundaries in supersonic flow over double cones," *Journal of Fluid Mechanics*, vol. 916, p. A5, 2021.