
SU2
The Open-Source CFD Code

3rd Annual SU2 Developers Meeting

September 16th-18th, 2018

University of Strathclyde, Scottish Universities Insight Institute (SUII)

Glasgow, Scotland, UK

SU2

The Open-Source CFD Code

Meeting Agenda for Sunday September 16th

0900 – 0915: Welcome & Agenda

0915 – 1045: Introduction to developing in SU2: *Covering high level class design, how to modify the code, working with GitHub (branching, PRs, regressions), etc.*

1045 – 1615: Hack session: *Separate groups working on various problems (lunch and snacks/coffee offered in the room while working)*

1615 – 1700: Wrap-up Presentations: *Two-slide presentations on major progress for the day, including discussion*

1730 – open: Social at "The Counting House", *2 St Vincent Place, G1 2DH*

Meeting Agenda for Monday September 17th

0800 – 0830: Welcome & Year in review, *T. Economon (Bosch), J.J. Alonso (Stanford)*

0830 – 0900: SU2-NEMO - Thermochemistry and high-Mach aerothermodynamics, *M. Fossati (U. of Strathclyde), T. Magin, J.B. Scoggins, M. Pini, P. Colonna, R. Sanchez, T. Economon, D. Mayer, N. Beishuizen, C. Garbacz-Gomes, W.T. Meier, J.J. Alonso, T. van der Stelt*

0900 – 0930: Toward optimization for reactive flows in SU2, *N. Beishuizen (Bosch), D. Mayer, T. Economon*

0930 – 1000: Conjugate heat transfer problems and computing coupled discrete adjoints using AD, *O. Burghardt (TU Kaiserslautern), T. Albring, N. Gauger*

1000 – 1030: Coffee break

1030 – 1100: Physics-based RANS model-form UQ in SU2, *J. Mukhopadhaya (Stanford), A. Mishra, J.J. Alonso, G. Iaccarino*

1100 – 1130: Aeroacoustics prediction and optimization capabilities in SU2, *B. Zhou (NIA/NASA-Langley), T. Albring, N. Gauger, C. Ilario, T. Economon, J.J. Alonso, L. V. Lopes, H. Yao, S. Peng, L. Davidson*

1130 – 1200: Higher-order SU2: DG-FEM solver and WENO-FV solver with LES/ILES/WMLES applications, *E. van der Weide (U. of Twente), J.J. Alonso, D. Drikakis, K. Singh, P. Urbanczik, E. Molina, J.H. Choi*

1200 – 1300: Lunch

1300 – 1330: Unsteady optimization with SU2: application to turbomachinery design, *A. Rubino (TU Delft), M. Pini, N. Anand, P. Colonna*

1330 – 1400: Preliminary results on rotor-fuselage aerodynamics using SU2: status and challenges, *M. Morelli (Politecnico di Milano), G. Gori, A. Guardone*

1400 – 1430: Anisotropic mesh adaptation with the INRIA AMG library, *A. Loseille (INRIA), V. Menier, B. Munguia, J.J. Alonso*

1430 – 1500: Coffee break

1500 – 1530: Simulation and adjoint-based design for variable density incompressible flows with heat transfer, *T. Economon (Bosch)*

1530 – 1600: Implementation of pressure-based Navier-Stokes for wind energy applications, *A. Ravishankara (ECN part of TNO), H. Ozdemir, E. van der Weide*

1600 – 1630: SU2-IDS: International Developers Society, *T. Albring, R. Sanchez (TU Kaiserslautern), T. Economon, F. Palacios*

1630 – 1700: Wrap up, *J.J. Alonso (Stanford)*

In order to participate (in-person or virtually), please register for the meeting by following the link on the SU2 home page (https://su2code.github.io).

*Please note that all stated times are British Summer Time (BST). **The presenter author is underlined

Stanford · Technische Universität Kaiserslautern · TU Delft · Imperial College London · University of Twente. · BOSCH

Hackathon topics

1. Driver structure

2. Input/Output structure reformatting

3. Heterogeneous restart (e.g. RANS from an Euler and mesh mapping/interpolation)

4. Incompressible solver

5. Multiple config files and meshes multi-point optimization

6. General definitions for objective functions

7. SU2-NEMO thermochemistry structure and libraries interfaces

SU2
The Open-Source CFD Code

SU2-NEMO (Non-Equilibrium MOdels): Thermochemistry and high-Mach flows

M. Fossati, C. Garbacz-Gomez, J.J. Alonso, W.T. Maier, R. Sanchez, M. Pini, P. Colonna, T. van der Stelt, T. Magin, J.B. Scoggins, T. Economon, D. Mayer, N. Beishuizen

Stanford | ENGINEERING
Aeronautics & Astronautics

University of Strathclyde

Technische Universität Kaiserslautern

TU Delft

von KARMAN INSTITUTE FOR FLUID DYNAMICS

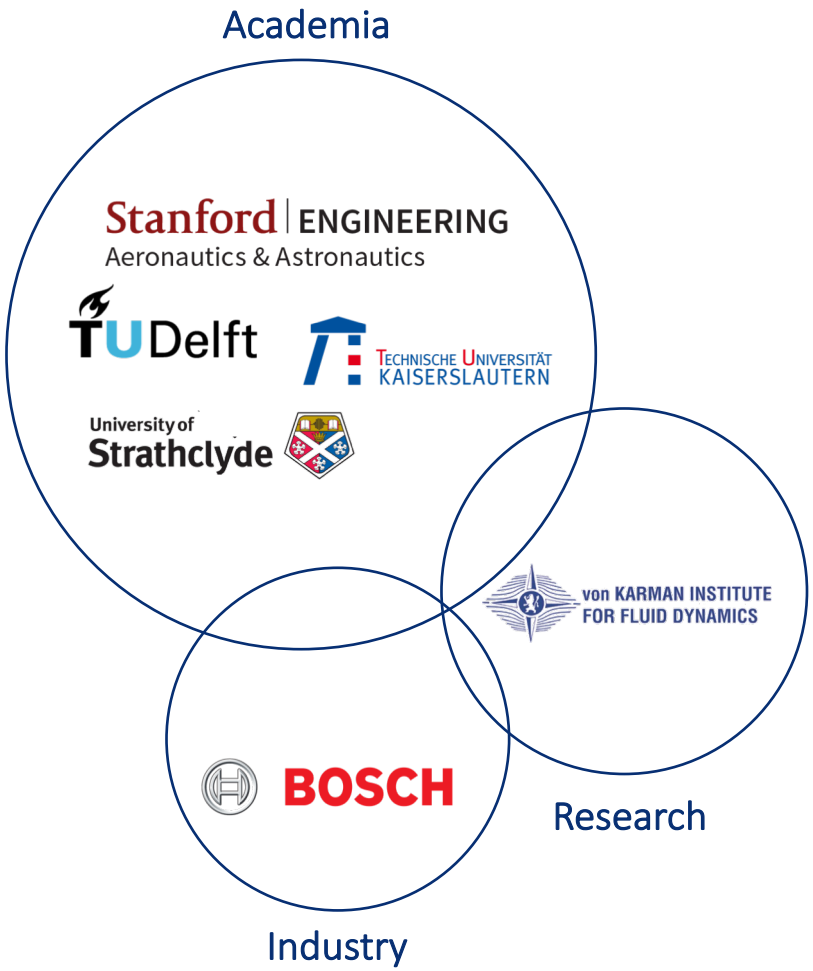
BOSCH

Ambition and objectives

"Enhance the multi-physics characteristics of SU2 and extend the spectrum of applications, with a focus on design"

- Define the roadmap for a coordinated development of thermochemistry and nonequilibrium models
- Incorporate advanced models for finite-rate chemistry and thermal nonequilibrium
- Consolidate the implementation and use of advanced thermodynamic models
- Reboot under a new perspective the modelling of high-temperature effects

People and selected applications, to date



- High-Mach and high-enthalpy external aerothermodynamics
- Laminar to turbulent transition in highly-compressible regimes
- Non-ideal gas dynamics of complex fluids departing from ideal gas laws
- Laminar premixed combustion for domestic heating
- Combustion in industrial processes
- From atmospheric entry and meteor to biomass pyrolysis

Outline

1. Some background equations and models
2. Mutation++, Cantera and FluidProp
3. Internal gasdynamics: SU2 and FluidProp
4. External aerodynamics: SU2 and Mutation++
5. The road ahead

SU2
The Open-Source CFD Code

3rd Annual SU2 Developers Meeting

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0$$

$$\frac{\partial \rho \vec{u}}{\partial t} + \nabla \cdot \left(\rho \vec{u} \otimes \vec{u} + P\bar{I} - \bar{\tau}\right) = 0$$

$$\frac{\partial \rho e}{\partial t} + \nabla \cdot \left(\rho e \vec{u} + P\bar{I} \cdot \vec{u} - \bar{\tau} \cdot \vec{u} + \vec{q}\right) = 0$$

$$P = \rho \mathrm{R} T$$
$$e = c_v T$$

$$\tau_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3} \mu \delta_{ij} \nabla \cdot \vec{u}$$
$$\vec{q} = -k \nabla T$$

Some background equations and models

$$\frac{\partial \rho_s}{\partial t} + \nabla \cdot (\rho_s \vec{u} - \rho_s \vec{u}_{d,1}) = \dot{w}_s$$

$$\frac{\partial \rho \vec{u}}{\partial t} + \nabla \cdot \left(\rho \vec{u} \otimes \vec{u} + P\bar{I} - \bar{\tau}\right) = 0$$

$$\frac{\partial \rho e}{\partial t} + \nabla \cdot \left(\rho e \vec{u} + P\bar{I} \cdot \vec{u} - \bar{\tau} \cdot \vec{u} + \vec{q}\right) = \nabla \cdot \left(-\sum_s h_s \rho_s \vec{u}_{d,s}\right)$$

$$P = P(\rho_s, T)$$
$$e = e(X_s, T)$$

$$\tau_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - \frac{2}{3}\mu \delta_{ij} \nabla \cdot \vec{u}$$

$$\vec{q} = -k \nabla T$$

$$\mu = \sum_s \frac{\mu_s X_s}{\phi_s}, \quad k = \sum_s \frac{k_s X_s}{\phi_s}$$

Some background equations and models

$$\frac{\partial \rho_s}{\partial t} + \nabla \cdot (\rho_s \vec{u} - \rho_s \vec{u}_{d,1}) = \dot{w}_s$$

$$\frac{\partial \rho \vec{u}}{\partial t} + \nabla \cdot \left(\rho \vec{u} \otimes \vec{u} + P\bar{I} - \bar{\tau}\right) = 0$$

$$\frac{\partial \rho e}{\partial t} + \nabla \cdot \left(\rho e \vec{u} + P\bar{I} \cdot \vec{u} - \bar{\tau} \cdot \vec{u} + \vec{q}\right) = \nabla \cdot \left(-\sum_s h_s \rho_s \vec{u}_{d,s}\right) - \nabla \cdot \vec{q}_v$$

$$\frac{\partial \rho e_v}{\partial t} + \nabla \cdot (\rho e_v \vec{u} + \vec{q}_v + \sum_s e_{v,s}\, \rho_s \vec{u}_{d,s}) = \sum_s Q_s^v + \sum_s Q_S^{t-v}$$

$$P = P(\rho_s, T)$$

$$e = e(X_s, T)$$

$$\rho e_v = \sum_s \rho_s e_{v,s}$$

$$\vec{q}_v = -k_v \nabla T_v$$

$$e_{v,s} = \frac{R}{W_s} \frac{\theta_{v,s}}{\exp\left(\theta_{v,s}/T_v\right) - 1}$$

$$\tau_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - \frac{2}{3}\mu \delta_{ij} \nabla \cdot \vec{u}$$

$$\vec{q} = -k \nabla T$$

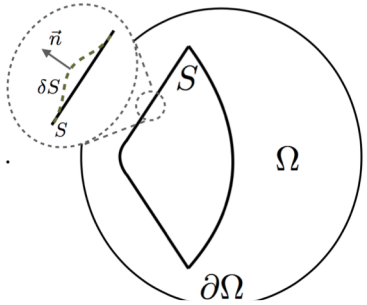
$$\mu = \sum_s \frac{\mu_s X_s}{\phi_s}, \quad k = \sum_s \frac{k_s X_s}{\phi_s}$$

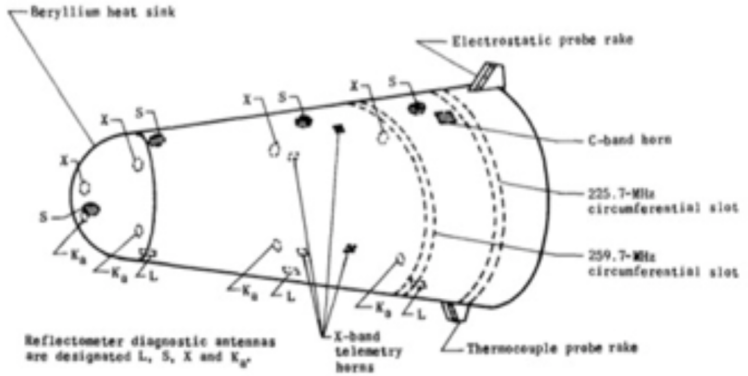
Some background equations and models: where were we?

Initial efforts by Sean Copeland (PhD, 2015, Stanford University):

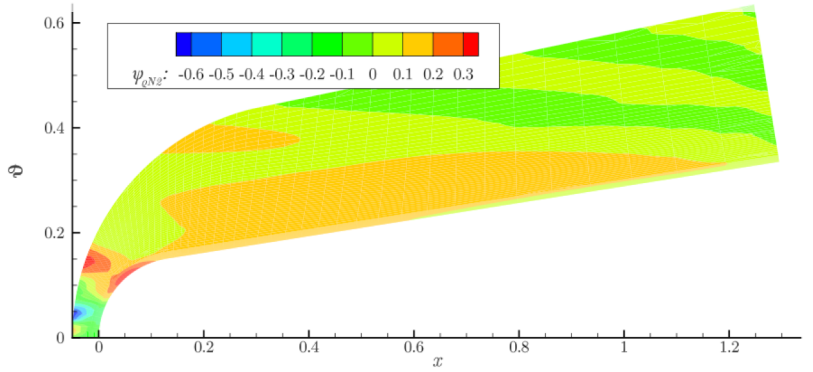
"A Continuous Adjoint Formulation for Hypersonic Flows in Thermochemical Nonequilibrium"

- continuum, steady, viscous, multi-component, gas mixture in thermochemical nonequilibrium
- Fast coupling assumptions
- Rigid-Rotator-Harmonic-Oscillator (RRHO) thermodynamics
- Transport properties
  - Diffusion — Fick's Law w/ closure terms (Sutton, 1998)
  - Viscosity — Newtonian fluid w/ Stokes' Hypothesis
  - Thermal Cond. — Fourier's Law
- Transport coefficients: Blottner/Eucken + Wilke's semi-empirical mixing rule
- Landau-Teller vibrational relaxation with Park's limiting cross section
- Finite-rate chemistry (Arrhenius-type)





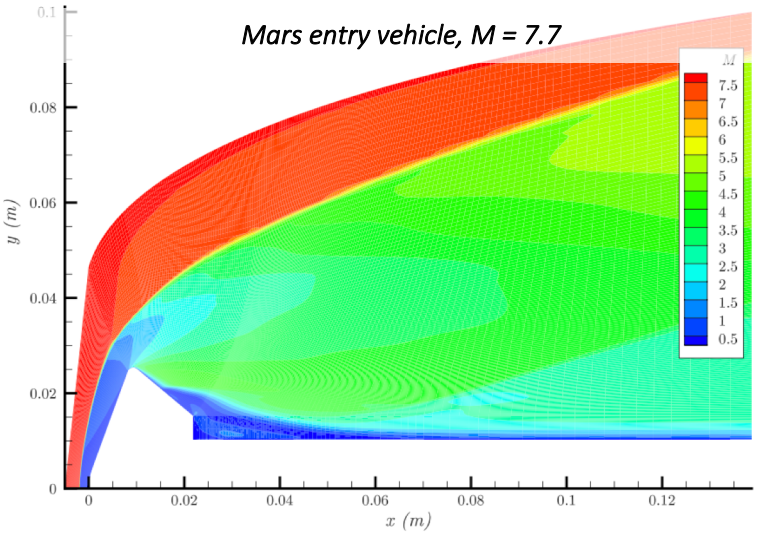
RAM-C test article (Jones, 1972)

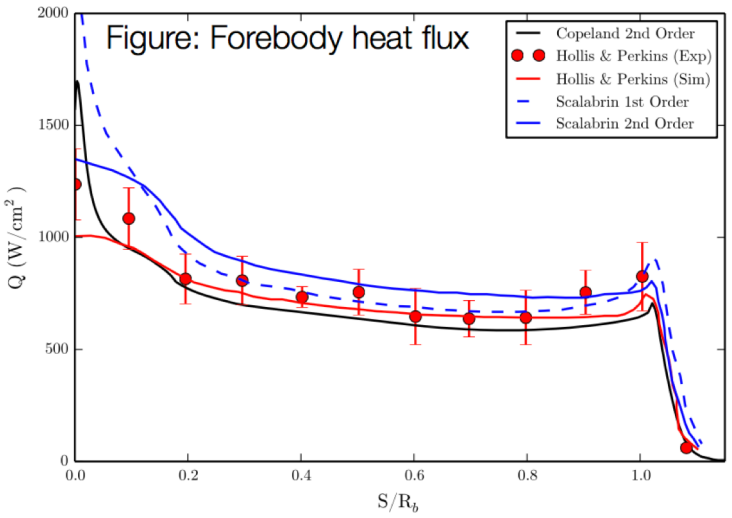


Some background equations and models: where were we?

- Derivation of continuous adjoint system, boundary conditions & surface sensitivities for steady, viscous, NE flow environments
- Implementation (in SU2) of flow and adjoint equations
- Adjoint equations formulated in a general way to support substitution of other thermochemical models
- All contributions in *feature_TNE2* and *feature_AdjTNE2*,

… but SU2 has continued to evolve…new and more general implementation needed!





Some background equations and models: where we are heading

$$\frac{\partial \rho_s}{\partial t} + \nabla \cdot (\rho_s \vec{u} - \rho_s \vec{u}_{d,1}) = \dot{w}_s$$

$$\frac{\partial \rho \vec{u}}{\partial t} + \nabla \cdot \left(\rho \vec{u} \otimes \vec{u} + P\bar{I} - \bar{\tau}\right) = 0$$

$$\frac{\partial \rho e}{\partial t} + \nabla \cdot \left(\rho e \vec{u} + P\bar{I} \cdot \vec{u} - \bar{\tau} \cdot \vec{u} + \vec{q}\right) = \nabla \cdot \left(-\sum_s h_s \rho_s \vec{u}_{d,s}\right) - \nabla \cdot \vec{q}_v$$

$$\frac{\partial \rho e_v}{\partial t} + \nabla \cdot \left(\rho e_v \vec{u} + \vec{q}_v + \sum_s e_{v,s}\, \rho_s \vec{u}_{d,s}\right) = \sum_s Q_s^v + \sum_s Q_S^{t-v}$$

$$P = P(\rho_s, T)$$

$$e = e(X_s, T)$$

$$\rho e_v = \sum_s \rho_s e_{v,s}$$

$$\vec{q}_v = -k_v \nabla T_v$$

$$e_{v,s} = \frac{R}{W_s} \frac{\theta_{v,s}}{\exp\left(\theta_{v,s}/T_v\right) - 1}$$

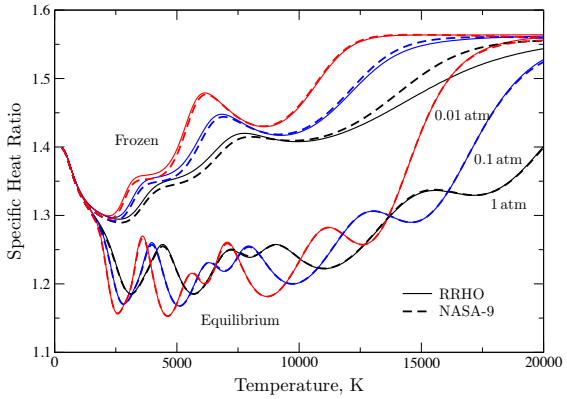
$$\tau_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - \frac{2}{3}\mu \delta_{ij} \nabla \cdot \vec{u}$$

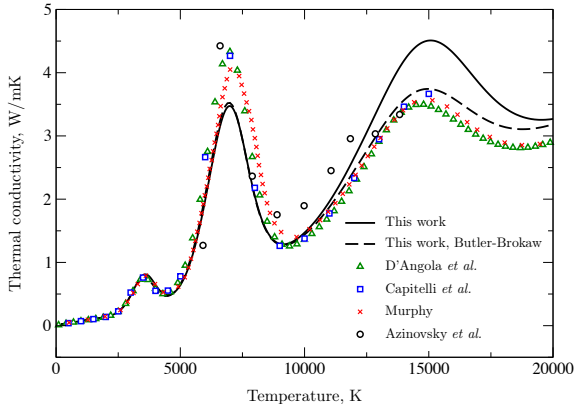
$$\vec{q} = -k \nabla T$$

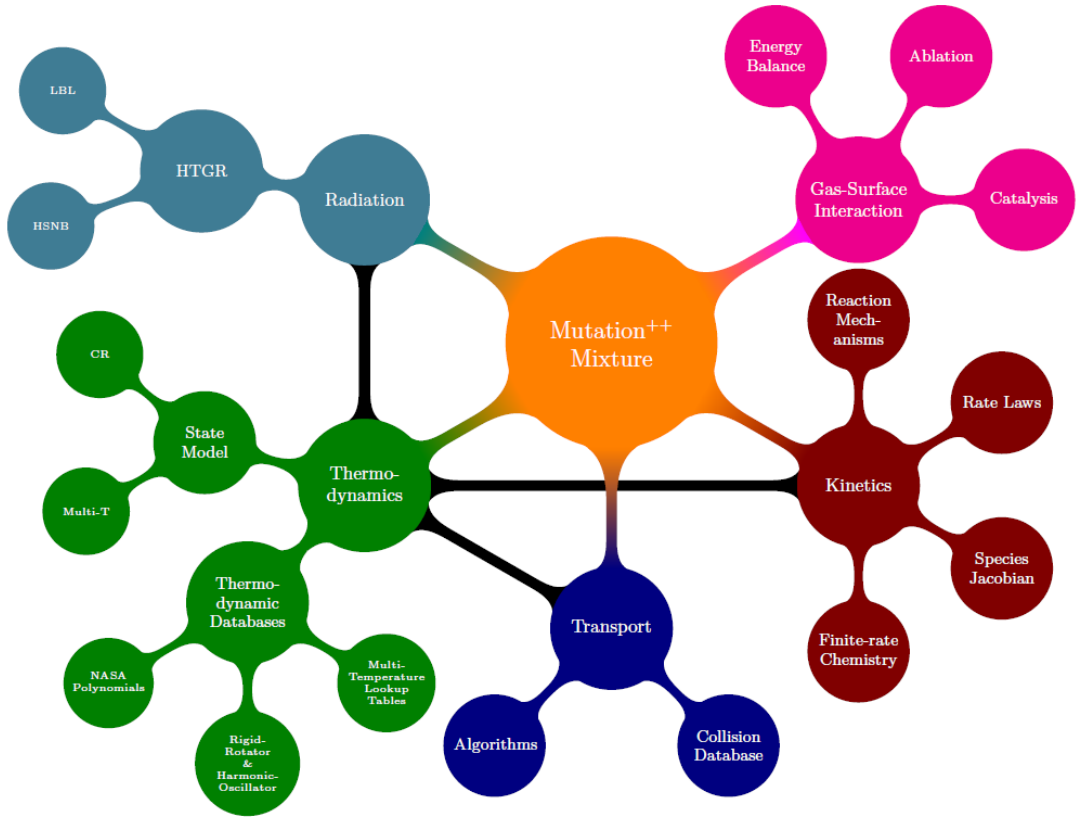
$$\mu = \sum_s \frac{\mu_s X_s}{\phi_s}, \quad k = \sum_s \frac{k_s X_s}{\phi_s}$$

Mutation++

- Thermodynamic properties
- Multicomponent transport properties
- Finite rate chemistry in thermal nonequilibrium
- A robust multiphase equilibrium solver







SU2 The Open-Source CFD Code

local state

physicochemical properties

Mutation++
- physicochemical models,
- algorithms, and
- data

https://sync.vki.ac.be/mpp/mutationpp

Cantera is an open-source suite of object-oriented software tools for problems involving chemical kinetics, thermodynamics, and/or transport processes. The software automates the chemical kinetic, thermodynamic, and transport calculations so that the users can efficiently incorporate detailed chemical thermo-kinetics and transport models into their calculations

- It is widely used in the combustion industry
- Low mach incompressible equilibrium chemistry

https://www.cantera.org/docs/sphinx/html/index.html

Some background equations and models: where we are heading

$$\frac{\partial \rho_s}{\partial t} + \nabla \cdot (\rho_s \vec{u} - \rho_s \vec{u}_{d,1}) = \dot{w}_s$$

$$\frac{\partial \rho \vec{u}}{\partial t} + \nabla \cdot \left(\rho \vec{u} \otimes \vec{u} + P\bar{I} - \bar{\tau}\right) = 0$$

$$\frac{\partial \rho e}{\partial t} + \nabla \cdot \left(\rho e \vec{u} + P\bar{I} \cdot \vec{u} - \bar{\tau} \cdot \vec{u} + \vec{q}\right) = \nabla \cdot \left(-\sum_s h_s \rho_s \vec{u}_{d,s}\right) - \nabla \cdot \vec{q}_v$$

$$\frac{\partial \rho e_v}{\partial t} + \nabla \cdot (\rho e_v \vec{u} + \vec{q}_v + \sum_s e_{v,s}\, \rho_s \vec{u}_{d,s}) = \sum_s Q_s^v + \sum_s Q_S^{t-v}$$

$$P = P(\rho_s, T)$$
$$e = e(X_s, T)$$

$$\rho e_v = \sum_s \rho_s e_{v,s}$$

$$\vec{q}_v = -k_v \nabla T_v$$

$$e_{v,s} = \frac{R}{W_s} \frac{\theta_{v,s}}{\exp\left(\theta_{v,s}/T_v\right) - 1}$$

$$\tau_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - \frac{2}{3}\mu \delta_{ij} \nabla \cdot \vec{u}$$

$$\vec{q} = -k \nabla T$$

$$\mu = \sum_s \frac{\mu_s X_s}{\phi_s}, \quad k = \sum_s \frac{k_s X_s}{\phi_s}$$

FluidProp

Thermophysical property calculator for arbitrary fluids:

- entire thermodynamic plane liquid and gas phase
- easily switching between all equations of state
- accurate properties also close to critical point
- multicomponent mixtures
- computes “real” critical points of mixtures
- advanced methods for phase equilibria
- *Look-up tables (under development)*

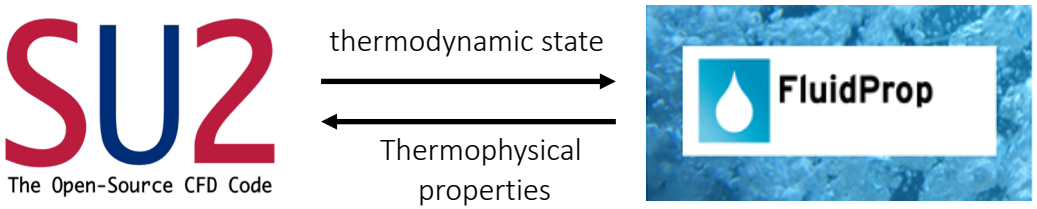
http://www.asimptote.nl/software/fluidprop

Fluids

Inorganic (CO, CO_2, H_2S, SO_2, O_2, N_2, …)
Hydrocarbons (linear, branched, cyclic, methane, toluene, …)
Alcohols, ketones (methanol, propanol, acetone, …)
Refrigerants (incl. recently developed, R12, R123, PP80, …)
Siloxanes (linear, cyclic, MM, MdM, MD6M, …)

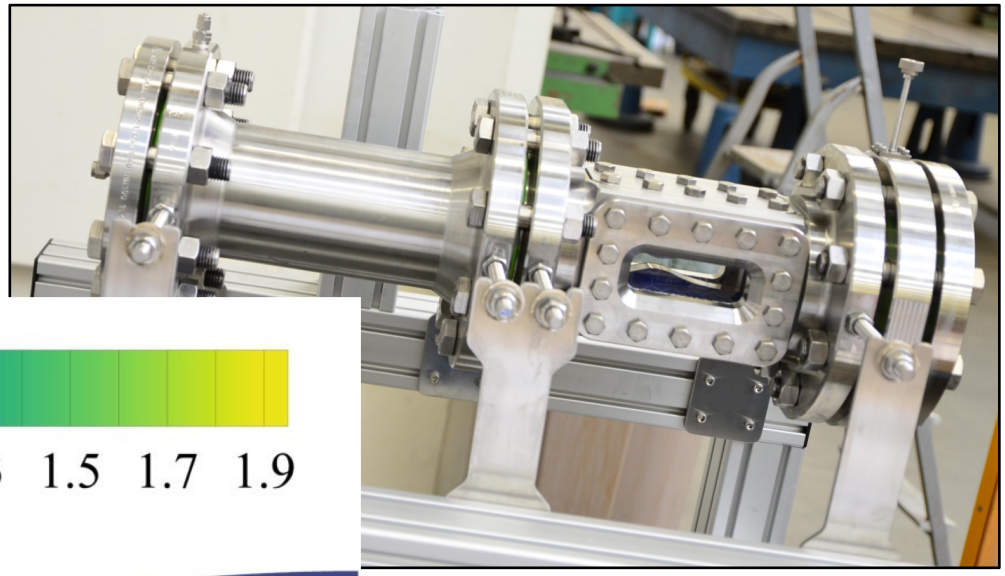
Equations of state

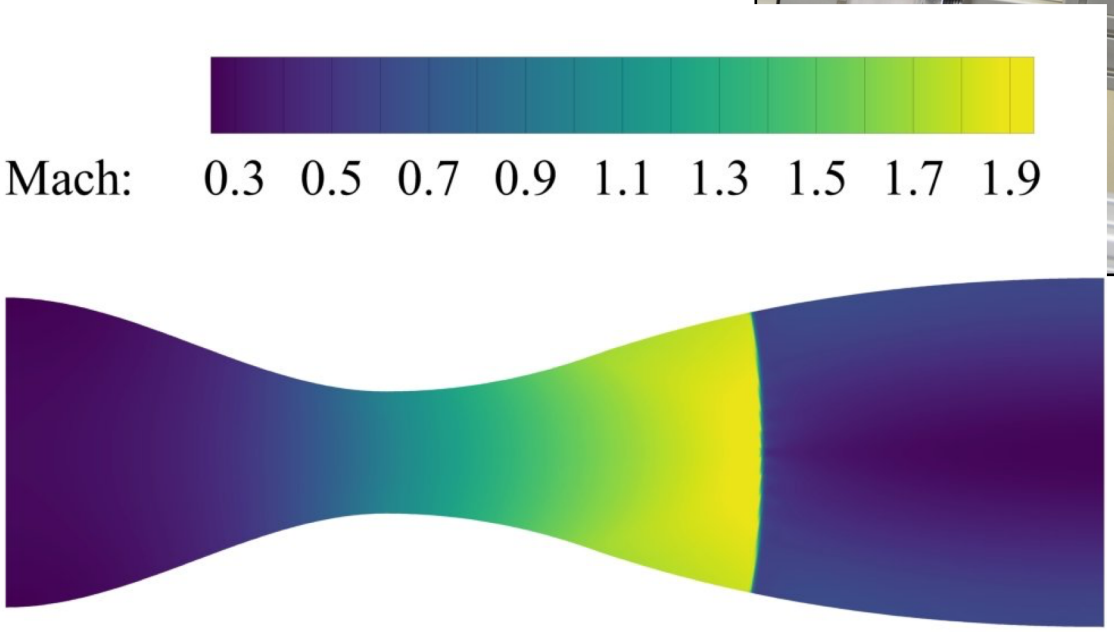
Fluid specific	water/steam IAPWS-IF97, absorption cooling LiBr/H2O	TPSI, ***NIST REFPROP 10***	multi-para-meter
Stat. mech	SAFT based EoSs, like sPC-SAFT, PC-SAFT, PCP-SAFT	Peng-Robinson variants PRSV and iPRSV	Cubic



Shock-induced flow with complex fluids

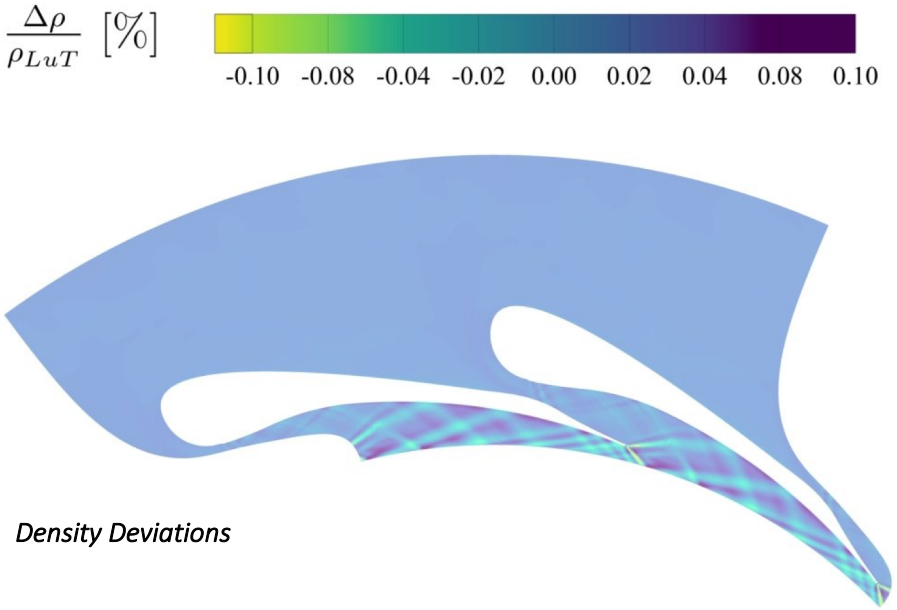
- Fluid: siloxane MM
- Fully turbulent, Roe scheme
- FluidProp-SW



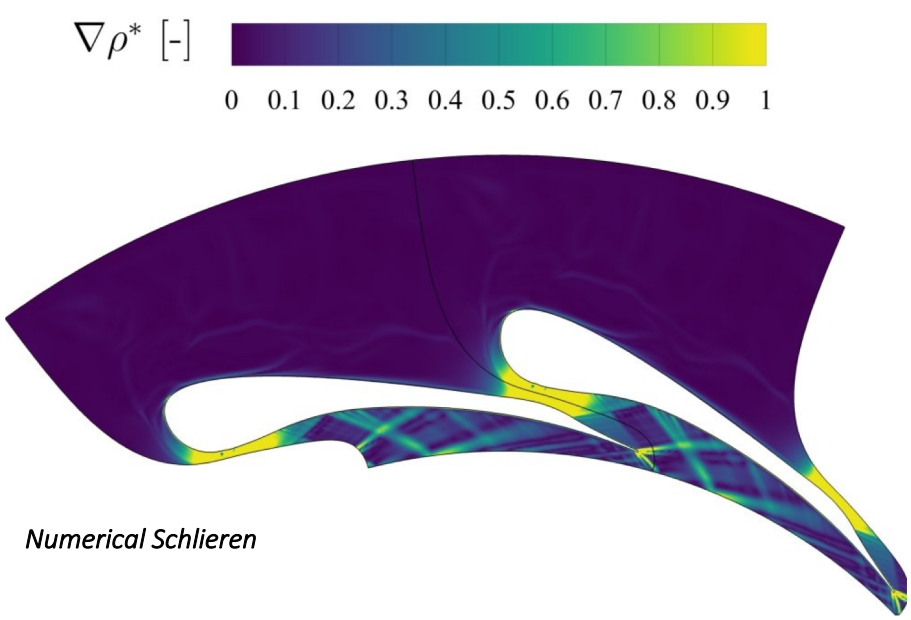


Flow Past Supersonic ORC Turbine Cascade

- Fluid: siloxane MM
- Fully turbulent, Roe scheme
- Comparison FluidProp-SW vs FluidProp-LuT

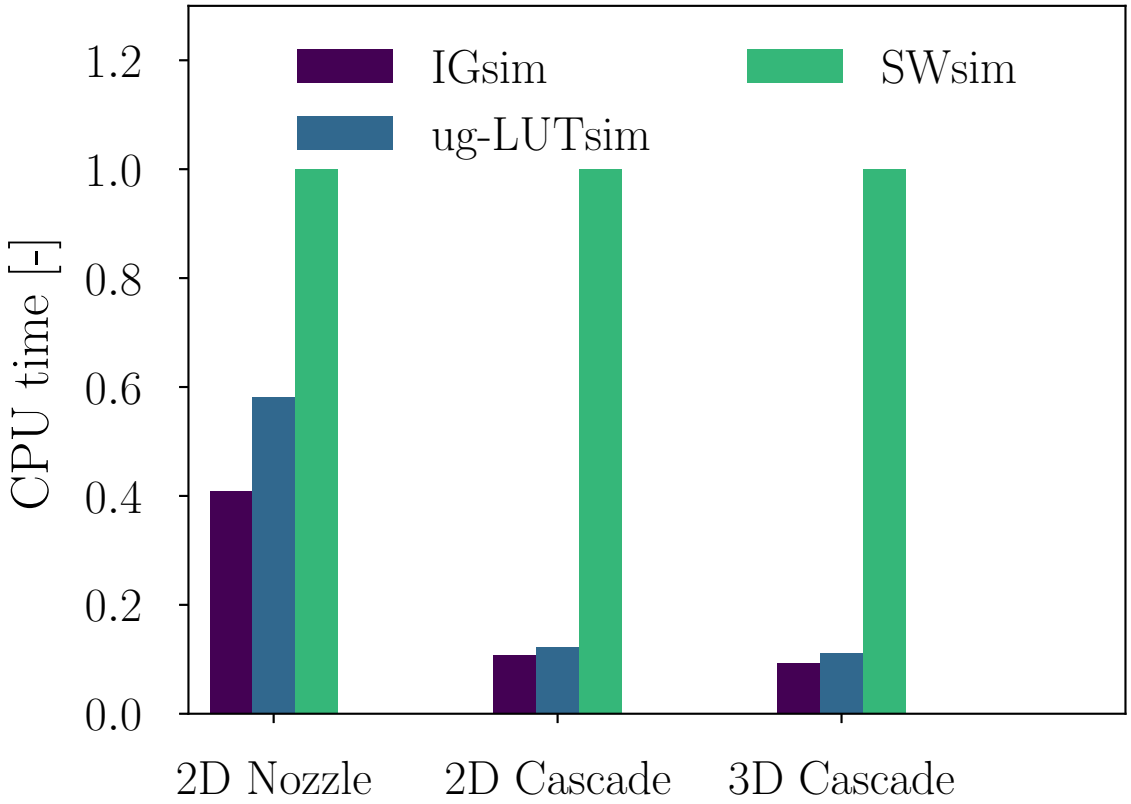


Density Deviations



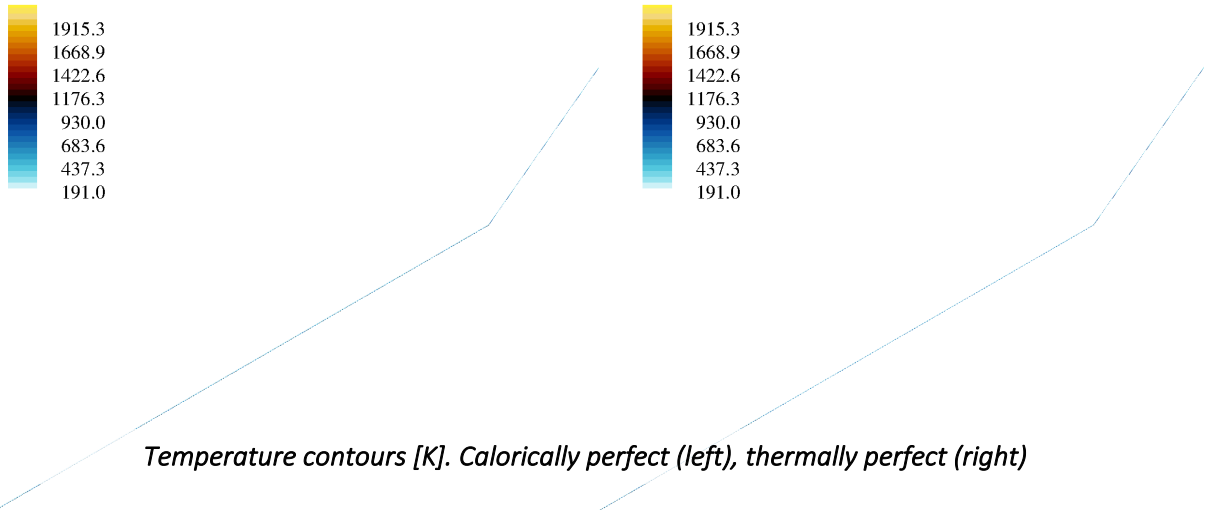
Numerical Schlieren

Thermodynamic Models: Computational Cost

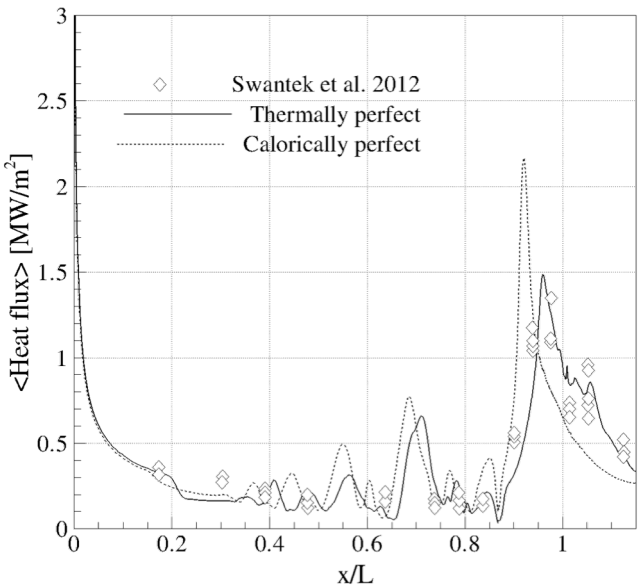


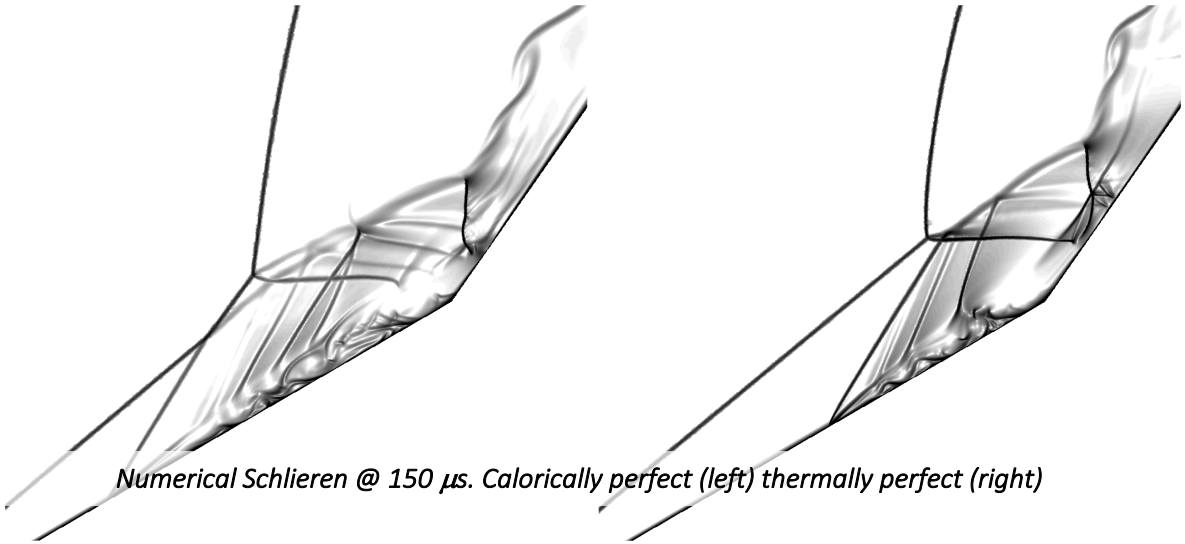
External aerodynamics: Double wedge at M7.11

$Mach_{FS}$	7.11
Unit Reynolds [m^{-1}]	55,880
$Temperature_{FS}$ [K]	191
$Pressure_{FS}$ [Pa]	391.735
$Temperature_{W}$ [K]	300
Fluid	N_2O_2
θ_1 / θ_2	30° / 55°



Temperature contours [K]. Calorically perfect (left), thermally perfect (right)

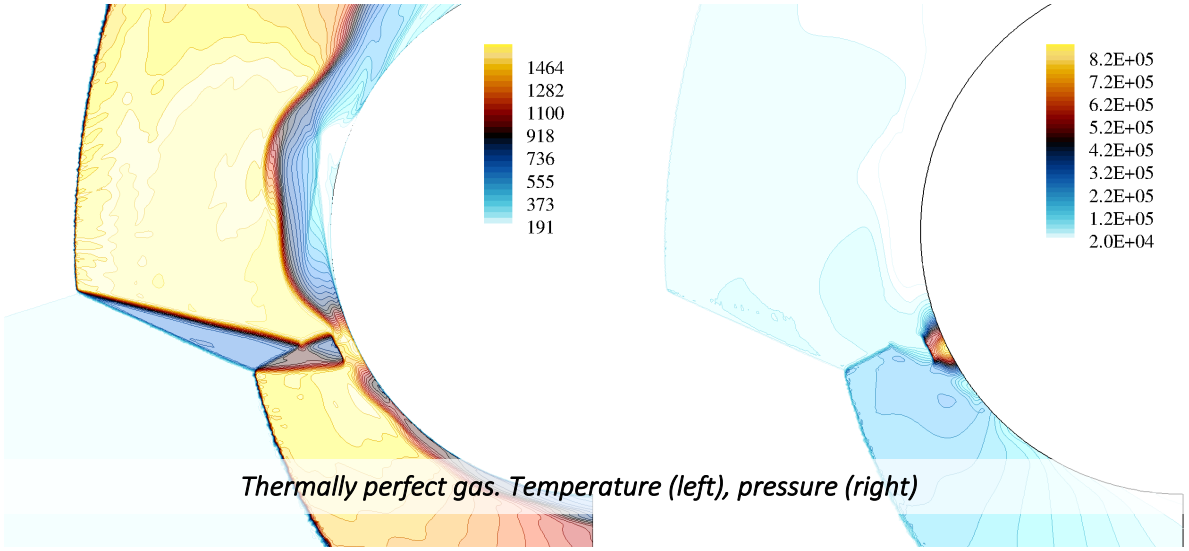




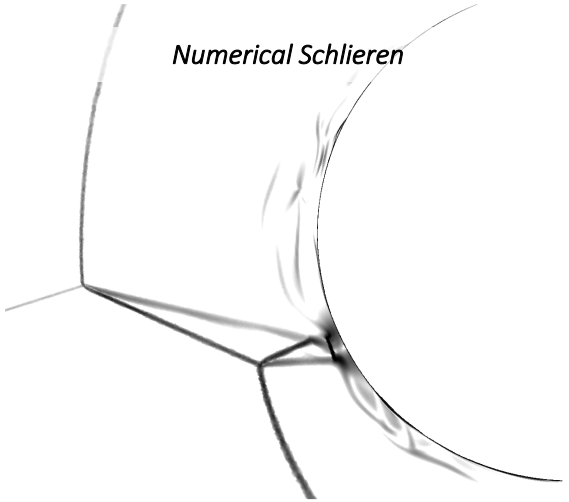
Numerical Schlieren @ 150 μs. Calorically perfect (left) thermally perfect (right)

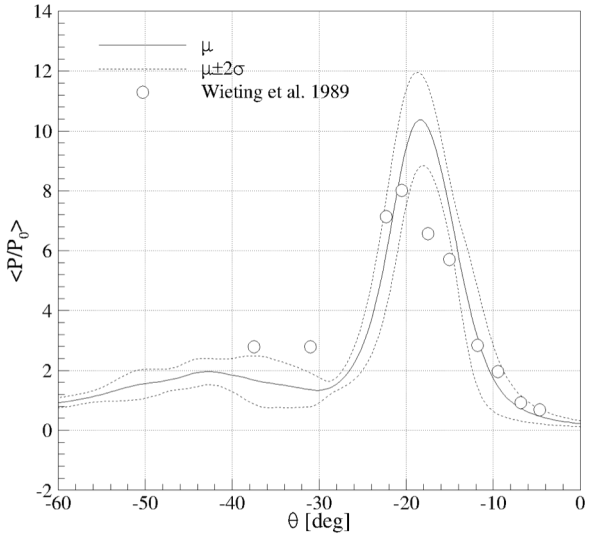
External aerodynamics: Edney IV interaction

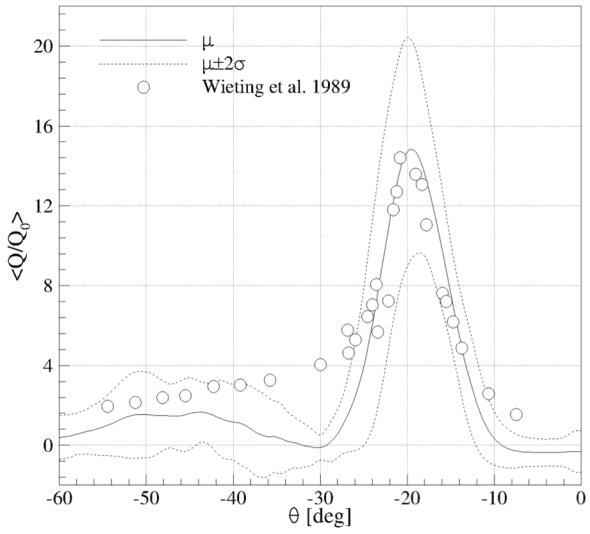
Mach	8.03 / 5.25
Unit Reynolds [m^{-1}]	6.75 10^6
Temperature [K]	111.56 / 238.04
Pressure [Pa]	985.01 / 6,996
$Temperature_W$ [K]	294.44
Fluid	N_2O_2
Impinging shock [deg]	18.1114



Thermally perfect gas. Temperature (left), pressure (right)

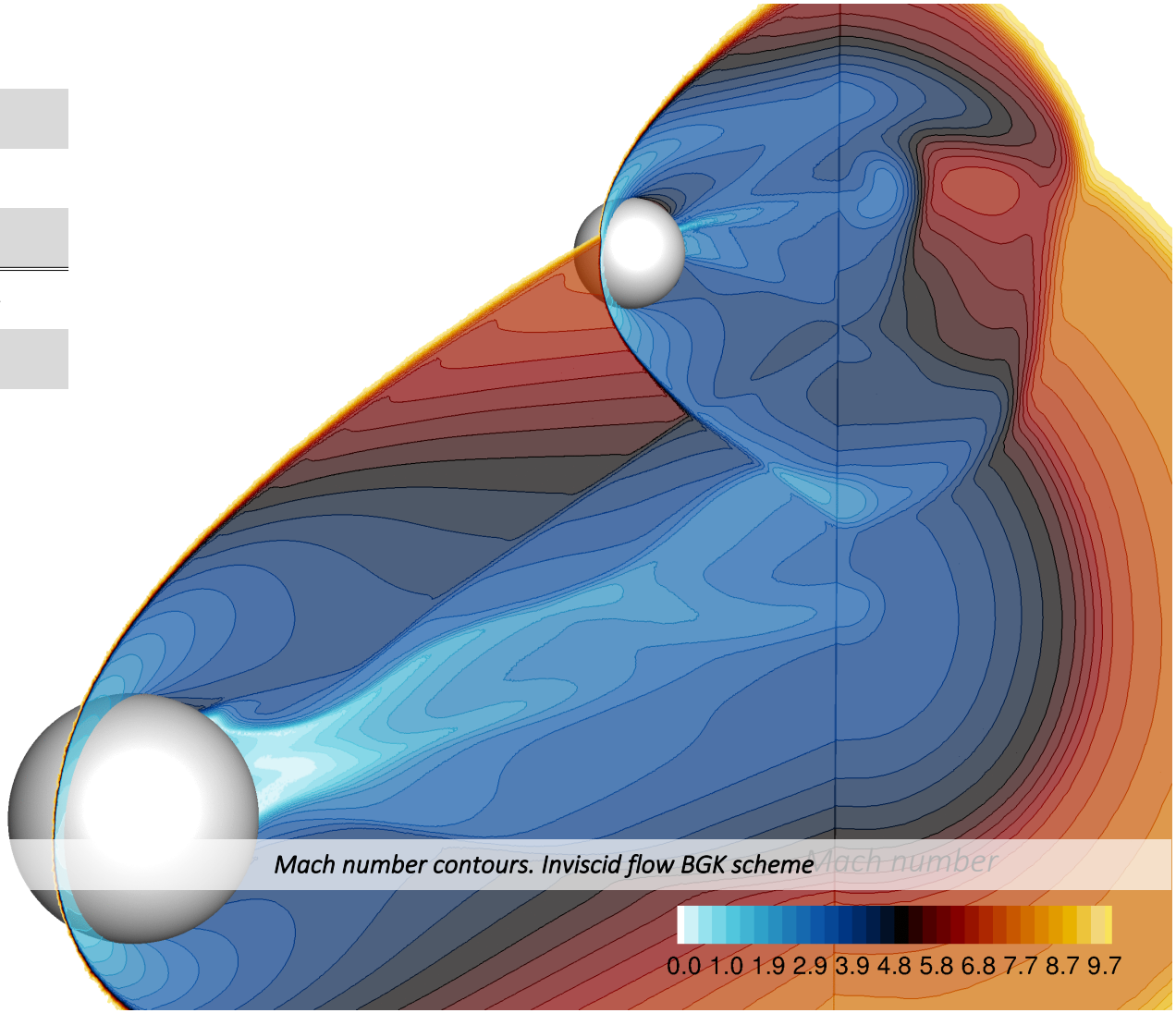






External aerodynamics: Proximal bodies

Mach	10
Temperature [K]	111.56
Pressure [Pa]	985.01
Fluid	N_2O_2
D_P/D_S	18.1114
y/D_P	1.088
x/D_P	4



Mach number contours. Inviscid flow BGK scheme

The road ahead …

- Define and implement a smart thermochemistry interface
- Augment the library of schemes for improved robustness with high Mach (including MUSCL)
- Extend the BC formulations to account for radiative equilibrium and potentially slip flow (for high-Mach)
- Formulate and implement models for the finite-rate energy exchange (i.e. multiple temperatures and energy modes)
- Ensure consistency with the algorithm differentiation for adjoint formulation
- Treatment of the Jacobian matrix of source terms
- Consolidate the multispecies and finite rate chemistry models with attention to the stiffness of the problem
- Introduce compressibility effects for turbulence modeling
- Transition modeling in highly compressible flows
- Coupling with conjugate heat transfer approach
- Coupling with Maxwell to account for MHD

… and counting

Thank you,
Happy to take any questions

Hackathon topic(s)

Q: What would be a “smart” interface for the thermodynamics and chemistry libraries (e.g. look-up tables, more integrated connection)?

Q: What is smart and what level of flexibility (i.e. plug-and-play) do we want to ensure/consider?

Q: What would be the impact on the config. file wrt to the thermodynamic models?

FLUID_MODEL

- STANDARD_AIR — Perfect ideal gas model for air, i.e. R =287.058 J/kgK, γ = 1.4 etc.
- USER_DEFINED† — Mainly anything else for which more details are required

```
if (FLUID_MODEL == USER_DEFINED){ more options might be needed
THERMAL_EOS          (Ideal_gas, Van_der_Waals, Peng_robinson, etc.)
CALORIC_EOS          (Perfect_gas, NASA_fit, etc.)
COMPOSITION          (Mass/Molar fractions of species)
}
```

CHEMISTRY (Frozen, Equilibrium, Finite-rate)

```
if (CHEMISTRY == Equilibrium){ even more options might be needed}
if (CHEMISTRY == Finite_rate){ a lot of options might be needed}
```