Trajectory-Based Conjugate Heat Transfer Simulation of the BoLT-II Flight Experiment

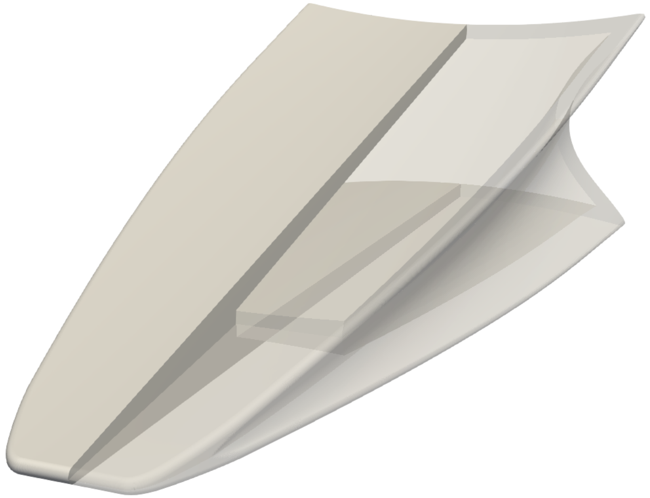
Centre for Hypersonics

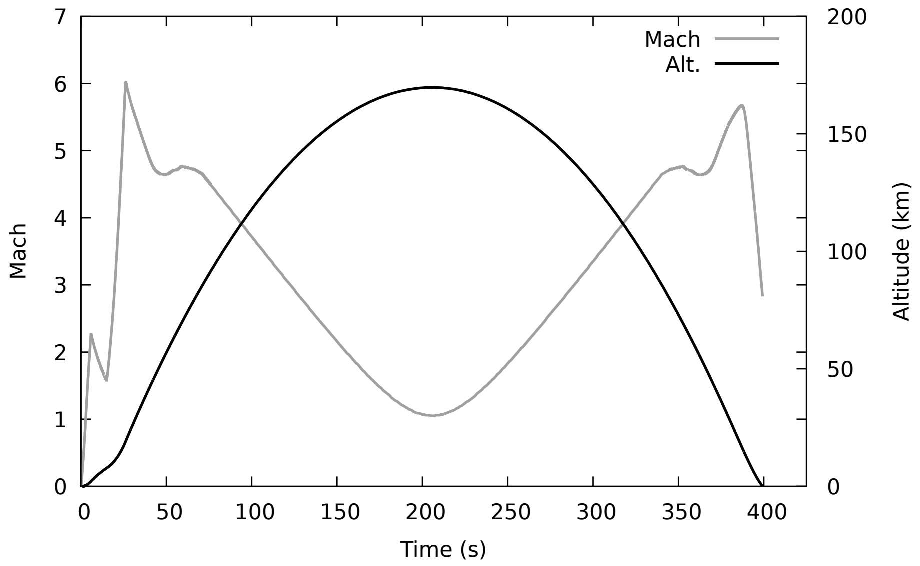
Dr Kyle Damm, Dr Damian Curran, Dr Rowan Gollan, A/Prof. Anand Veeraragavan

Outline

- Motivation: **BoLT-II** thermal analysis
- **Numerical methods for CHT**:
  - Fluid-Solid coupling
  - Fluid-Solid solver
- Accelerated transient simulation of solid domain via **Super-Time-Stepping:**
  - Overview
  - Performance investigation on a hypersonic flow problem
- Demonstration: **BoLT-II trajectory-based** simulation

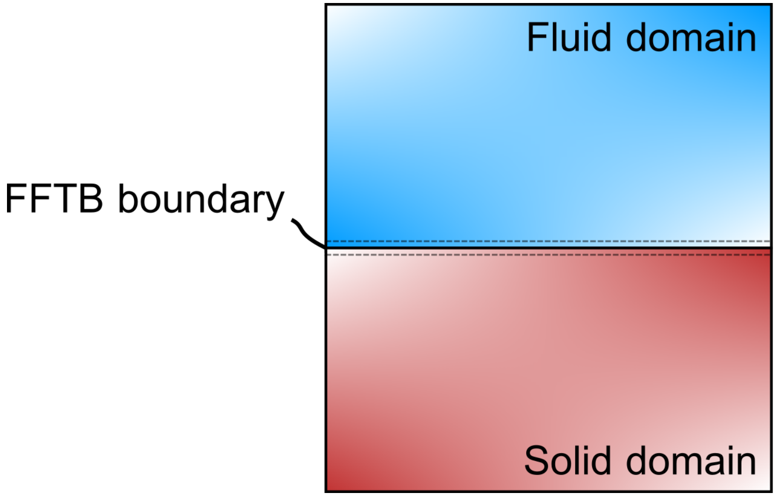
Motivation: BoLT-II Thermal Loading





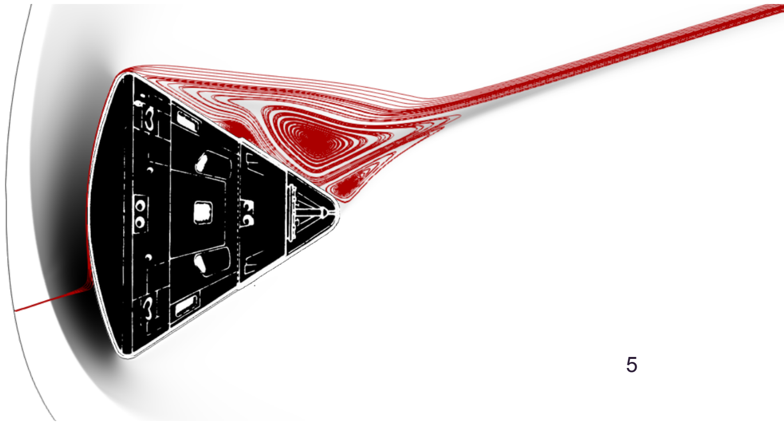
Fluid-Solid Coupling

- Partitioning approach - separate fluid solver and solid solver
- For hypersonic flows the fluid time scale << solid time scale
    A. Steady-state flow fields at each trajectory point
    B. Transient evolution of heat soak into solid
- Coupling strategy (**Flux Forward Temperature Back**):
    1. Choose Δt_{couple}
    2. Perform a fluid domain steady-state update (fixed temperature boundary condition)
    3. Transfer heat flux from fluid domain to solid domain
    4. Perform solid domain transient update (fixed heat flux boundary condition)
    5. Transfer temperature from solid domain to fluid domain
    6. Repeat steps 2-5 for N stages where $T_{flight} = N\Delta t_{couple}$



Fluid-Solid Solver

- **Eilmer** - open-source multi-physics solver (http://gdtk.uqcloud.net/)
- 2D/3D cell-centered finite volume compressible flow solver
  - Solves the Euler, Navier-Stokes, **RANS** equations
  - Operates on **structured**/unstructured grids (native, GridPro, SU2)
  - Several gas models: **ideal**, thermally perfect, multi-temperature, state-specific
- 2D/3D cell-centered finite volume **solid solver** (energy conservation equation)
- Time-accurate updates using Runge-Kutta family of integrators
- **Jacobian-Free Newton-Krylov** (JFNK) method for accelerated steady-state convergence



Super-Time-Stepping

- Investigated **Super-Time-Stepping (STS)** for accelerated transient time integration
- STS is a type of **Stabilised Explicit Runge-Kutta** method
- **Runge-Kutta-Legendre** variant implemented (s-stages):

$$\begin{aligned} y_0 &= y(t_0) \\ y_1 &= y_0 + \tilde{\mu}_1 \Delta t y'_0 \\ y_j &= \mu_j y_{j-1} + \nu_j y_{j-2} + \tilde{\mu}_j \Delta t y'_{j-1} \\ y(t+\Delta t) &= y_s \end{aligned}$$

$$\begin{aligned} \mu_j &= (2j-1)/j \\ \nu_j &= (1-j)/j \\ \tilde{\mu}_j &= 2\mu_j/(s^2+2) \end{aligned}$$

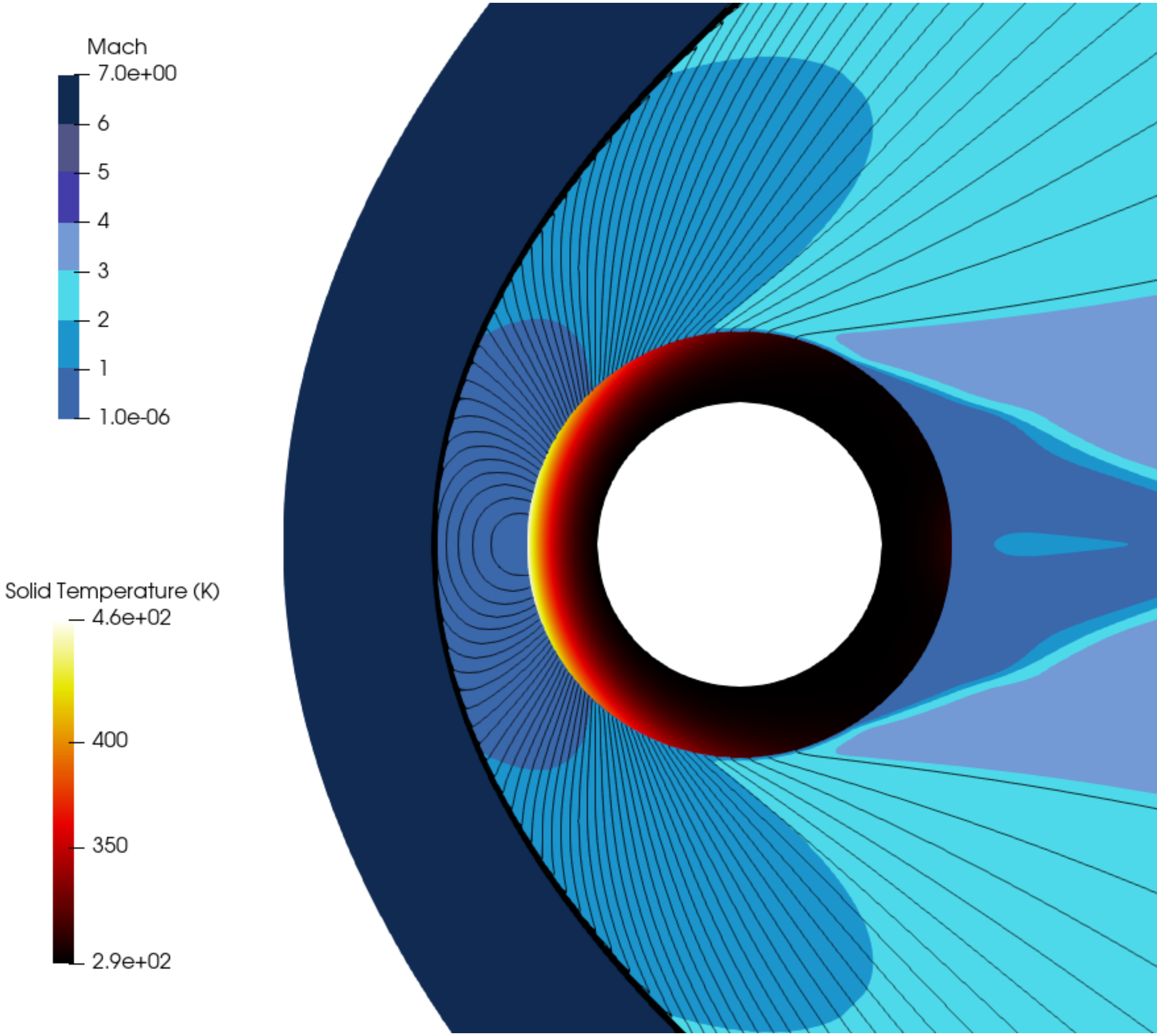
- Inner stages are **RK-like**
- Enables an **increase in the stable time-step**:

$$\Delta t = \Delta t_{explicit} \left(\frac{s^2 + s}{2} \right)$$

- Theoretical speed-up of **O(s)** relative to an **Euler update scheme**
- Algorithm adaptively chooses **appropriate s value** for each step

Super-Time-Stepping

- CHT simulation of **hypersonic** flow over a hollow cylinder
  - Based on experiments by Wieting (1987)
  - Validation results presented in paper
- Simulated **5 seconds** of wind tunnel experiment
- Using $\Delta t_{\text{couple}} = 1.0\ s$ (i.e. 5 flow/solid solutions)
- Compare wall-clock time for:
  - Traditional Euler scheme
  - STS scheme employing s_{max} = 5, 10, 50, 100, 200



Super-Time-Stepping

method	Δt (s)	Wall-clock (s)	Speedup factor
Euler	6.947×10^{-5}	305.46	1.0
STS ($s = 5$)	1.042×10^{-3}	144.66	2.1
STS ($s = 10$)	3.821×10^{-3}	58.54	5.2
STS ($s = 50$)	8.858×10^{-2}	9.28	32.9
STS ($s = 100$)	3.508×10^{-1}	4.58	66.7
STS ($s = 169$)	9.980×10^{-1}	2.76	110.7

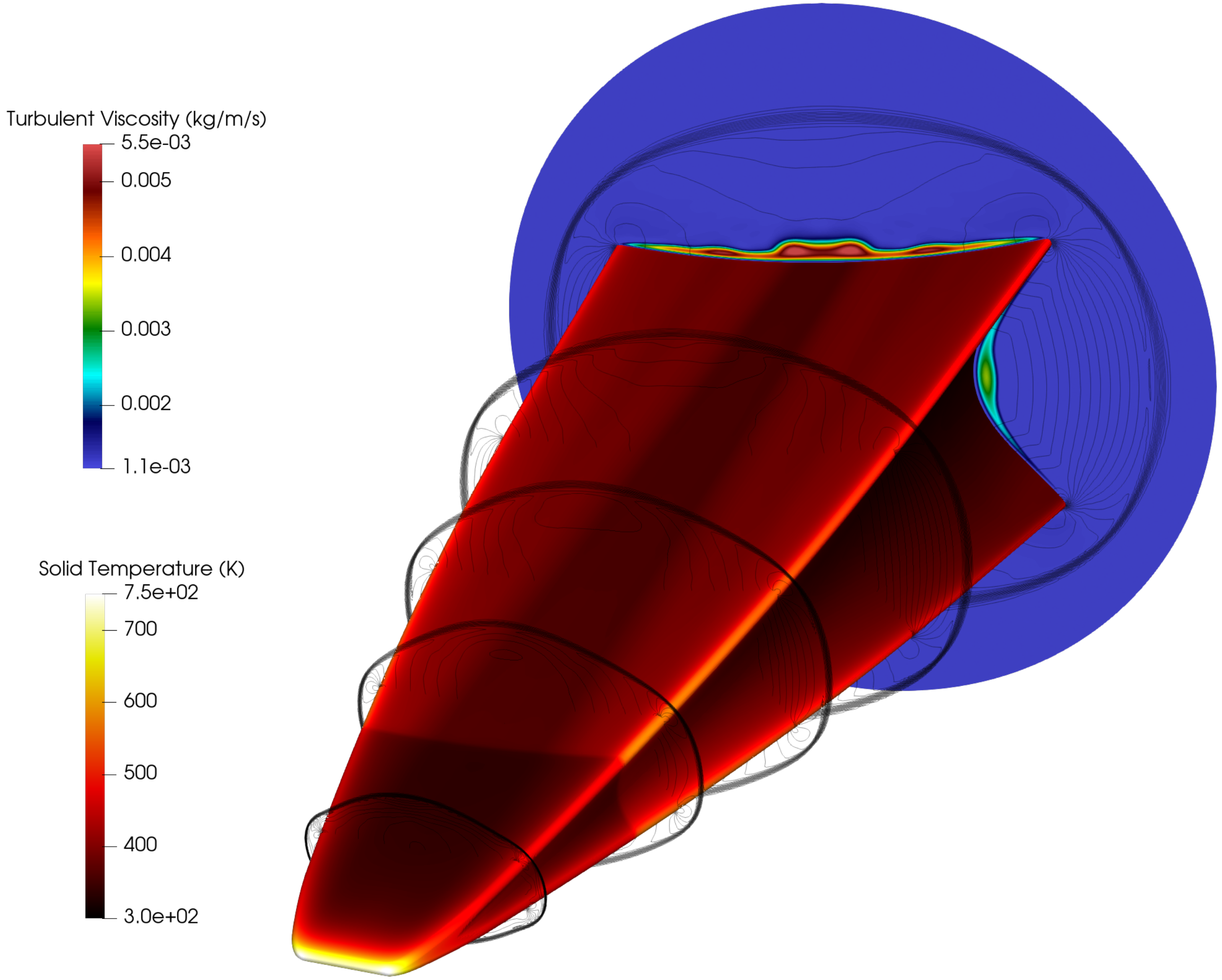
- Results are for the largest time-step taken during a simulation (corresponding to **$s \leq s_{max}$**)
- STS allows for a stable time-step **orders of magnitude** larger than the stable Euler time-step
- The speedup factor is approximately **O(s)**
- The time-step for s = 200 is larger than 1.0 s, hence the adaptive algorithm is invoked on the first step

BoLT-II Trajectory Simulation

- Simulated **portion** of the **descent trajectory**
- From $\mathbf{t_{start} = 396\ s}$ to $\mathbf{t_{final} = 406\ s}$
- $\mathbf{\Delta t_{couple} = 2\ s}$ (i.e. 5 flow/solid solutions)
- Uniform initial **solid temperature** at 300 K
- Each flow simulation is initialized by the **freestream condition**
- Spalart-Allmaras **turbulence** model
- Approximately 8.5 million fluid cells and 3.5 million solid cells

BoLT-II Trajectory Simulation

- JFNK solver converges fluid domain in **500 steps**
- Solid domain **STS breakdown**:
  - $\mathbf{s_{max} = 1000}$
  - Maximum time-step $\mathbf{\Delta t_{max} = 1.757\ s}$
  - **6 solid time-steps** to simulate 2.0 seconds
- Equivalent Euler time-stepping:
  - $\mathbf{\Delta t_{explicit} = 3.5 \times 10^{-6}\ s}$
  - Would require over **500,000** Euler time-steps!
  - Theoretical **speedup factor of approx. 500**



Conclusions

Super-Time-Stepping…

- Is a **simple to implement explicit** update scheme for solving parabolic PDEs
- Allows large stable time-steps - orders of magnitude larger than an Euler time-step
- Achieves a speedup factor of **O(s)**
- Has been an enabling technology for large scale **BoLT-II trajectory-based CHT** simulations

The University of Queensland
Australia

Contact

Dr Kyle Damm

Postdoctoral Research Fellow

kyle.damm@uqconnect.edu.au
+61 437308617

CRICOS 00025B