Aerothermal Simulation and Model Order Reduction, Using the Open-Source Framework Feel++

JB Wahl¹  C. Prud’homme¹  Y. Hoarau²

¹Université de Strasbourg - IRMA
²Université de Strasbourg - ICube

September 12, 2017
ANR Project : CHORUS
Common Horizon of Open Research in Uncertainty for Simulation

- Methodological treatment of uncertainty management problems for multi disciplinary purposes
- Development of new mathematical models and algorithms to face scalability problems
- Accessibility of interoperable advanced algorithms linked to HPC capabilities for a large community in a recognized open source environment
Airbus Use-Case

Objective: Apply the Certified Reduced Basis Methods on an aerothermal simulation in an avionic bay

Model:
- Steady Navier-Stokes/Heat transfer
- Incompressible Newtonian Fluid
- Boussinesq Approximation
- Turbulent Flow

Governing Equations

\[
\begin{aligned}
\rho \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p - \nu \Delta \mathbf{u} &= \rho \beta (T - T_0) \mathbf{g}, & \text{in } \Omega \times [0, T_f], \\
\nabla \cdot \mathbf{u} &= 0, & \text{in } \Omega \times [0, T_f], \\
\mathbf{u} \cdot \nabla T - \kappa \Delta T &= 0, & \text{in } \Omega \times [0, T_f], \\
\text{Boundary Conditions.}
\end{aligned}
\]
Some Physics

Notations

- $\rho$: fluid density $(kg.m^{-3})$
- $\kappa$: thermal diffusivity $(m^2.s^{-1})$
- $\mu$: dynamic viscosity $(kg.m^{-1}.s^{-1})$
- $\nu$: kinematic viscosity $(m^2.s^{-1})$
- $\beta$: thermal expansion coefficient $(K^{-1})$

Fluid Physic

- $\mu \approx 10^{-5} m^2.s^{-1}$
- $\kappa \approx 10^{-5} m^2.s^{-1}$
- Length Scale $L \approx 1 m$
- $\Rightarrow Re \approx 10^5$, $Pr \approx 1$, $Pe \approx 10^5$
- $\Rightarrow$ Potentially Turbulent Flow: Fun stuff start here!
Numerical Notations

Spatial Discretization (FEM):
- $\Omega$ a bounded domain in $\mathbb{R}^d$, $d = 2, 3$ and $T_h$ an usual triangulation of $\Omega$,
- Aerothermal Finite Element Spaces $(u, p, T) \in X_h = [P_h^{p+1}]^d \times P_h^p \times P_h^{p+1}$, $p \geq 1$,
- $(\cdot, \cdot)$ the $L^2$ scalar product on $\Omega$,

Weak Formulation

$$\rho(u \cdot \nabla u, v) + \mu(\nabla u, \nabla v) - (p, \nabla \cdot v) - (\nabla \cdot u, q) - (T(u, p)n, v) = \rho \beta((T - T_0)g, v),$$

$$(u \cdot \nabla T, S) + \kappa(\nabla T, \nabla S) - \kappa(\nabla T \cdot n, S)_{\Gamma} = 0, \quad \forall (v, q, S) \in X_h$$

where $T(u, p) = \mu \nabla u - \mathbb{I} p$ is the usual stress tensor.
Numerical Methods Needed

Resolution Scheme for Non Linear Problem

Stabilization Methods

Slip Boundary Conditions
Pseudo Transient Continuation $\Psi_{tc}$

[Kelley and Keyes, 1998], [Coffey et al., 2003]

Consider Differential Algebraic Equation $F(w) = 0$, with $w = (u, p)$, discretization of the NS system

\[
\begin{aligned}
    u \cdot \nabla u + \nabla p - \mu \nabla^2 u &= f, \\
    \nabla \cdot u &= 0.
\end{aligned}
\]  

(3)

Comparison with Usual Newton Algorithm

**Algorithm 1:** Newton Algorithm

Choose $\varepsilon$;  
Set $w = w_0$;  
repeat  
\[\text{Solve } F'(w)s = -F(w);\]  
\[\text{Set } w = w + s;\]  
\[\text{Evaluate } F(w);\]  
until $\|F(w)\| < \varepsilon$;

**Algorithm 2:** $\Psi_{tc}$ Algorithm

Choose $\varepsilon$;  
Set $w = w_0$ and $\delta = \delta_0$;  
repeat  
\[\text{Solve } (\frac{1}{\delta} D + F'(w))s = -F(w);\]  
\[\text{Set } w = w + s;\]  
\[\text{Evaluate } F(w);\]  
\[\text{Update } \delta;\]  
until $\|F(w)\| < \varepsilon$;
Pseudo Transient Continuation

Scaling Matrix

- For NS : \( D = \begin{bmatrix} V & 0 \end{bmatrix} \)
- Purpose : balance the local CFL number : \( C_K = \Delta t \sum_{i=1}^{d} \frac{u_i}{h_i} \)
- Scaling factor by element : \( \omega_K = \frac{\|u\|_2}{h_e} \)
- \( \Rightarrow \) Non zero initial state : initiate with Stokes solution

Update Pseudo Time Step

\[
\delta_{n+1} = \delta_n \frac{\|F(w_{n-1})\|}{\|F(w_n)\|} = \delta_0 \frac{\|F(w_0)\|}{\|F(w_n)\|} \tag{4}
\]
Ψtc Preliminary Results

Test Case: NACA0012 geometry

- Far field velocity $U = 1 \, m.s^{-1}$
- $\mu = 10^{-4} \, m^2.s^{-1}$
- Incidence angle $\theta = 0$
- $\rho = 1 \, kg.m^{-3}$

![Figure: Evolution of the time step and residual in Ψtc, value vs number of Iteration](image)

Figure: Evolution of the time step and residual in Ψtc, value vs number of Iteration
SUPG and Associated Stabilization Methods

Generic Problem

\[ \mathcal{L} u_h = f, \quad +BC \]  

(5)

where \( \mathcal{L} = -\nabla \cdot (\kappa \nabla) + a \nabla \) is the advection diffusion operator with \( a \) the advection field and \( \kappa \) the diffusion coefficient.

Standard Galerkin Method :

\[
B(u, v) = L(v), \\
B(u, v) = (\kappa \nabla u, \nabla v) + (a \nabla u, v) + (\kappa \nabla u \cdot n, v)_{\Gamma}, \\
L(v) = (f, v)
\]

(6)

SUPG Formulation :

\[
B_{SUPG}(u, v) = L_{SUPG}(v), \\
B_{SUPG}(u, v) = B(u, v) + (\tau_K \mathcal{L} u, a \nabla v), \\
L_{SUPG}(v) = L(v) + (\tau_K f, a \nabla v)
\]

(7)

GLS Formulation :

\[
B_{GLS}(u, v) = L_{GLS}(v), \\
B_{GLS}(u, v) = B(u, v) + (\tau_K \mathcal{L} u, \mathcal{L} v), \\
L_{GLS}(v) = L(v) + (\tau_K f, a \nabla v)
\]

(8)
Stabilization Parameter

Scaling Constant

Depends of:

- Shape of the cells,
- Polynomial Order,
- Satisfy the inverse estimate inequality

\[
C \sum_{K \in \mathcal{T}_h} h_e^2 ||\Delta v||^2_K \leq ||\nabla v||^2, \quad v \in \mathcal{X}_h
\]  \hspace{1cm} (9)

Computation:

- Resolution of local eigen value problem

\[
\lambda_K = \max_{v \in (P^2(K) \setminus \mathbb{R})} \frac{||\Delta v||^2_K}{||\nabla v||^2_K}
\]  \hspace{1cm} (10)

- \[ C = \frac{1}{\lambda_K h_e^2} \]
Stabilization Parameter

Option 1  [Leopoldo P. Franca, 1992a,b]

\[ \tau_1 = \frac{h_e}{2|a|_2} \xi(\alpha), \]

\[ \xi(\alpha) = \begin{cases} 
\alpha, & 0 \leq \alpha \leq 1, \\
1, & \alpha \geq 1,
\end{cases} \]

\[ m = \min\{\frac{1}{3}, 2C\} \]

\[ \alpha = \frac{m|a|_p h_e}{2\kappa}, \]

\[ |a|_p = \begin{cases} 
\left(\sum_{i=1}^d |a_i|^p\right)^{1/p}, & 1 \leq p < \infty, \\
\max_{i=1,\ldots,d} |a_i|, & p = \infty,
\end{cases} \]

Option 2  [Burda et al., 2005]

\[ \tau_2 = \frac{\xi(\alpha)}{\sqrt{\lambda_K} |a|_p}, \]

\[ \xi(\alpha) = \begin{cases} 
\alpha, & 0 \leq \alpha \leq 1, \\
1, & \alpha \geq 1,
\end{cases} \]

\[ \alpha = \frac{|a|_p}{2\kappa \sqrt{\lambda_K}}, \]
Advection Skew Mesh Test Case

[Brooks and Hughes, 1982]

- Constant advection field $\mathbf{a} = (\cos(\alpha), \sin(\alpha))$
- No reaction term
- No source term
- Square domain $\Omega = [0, 1] \times [0, 1]$
- Boundary Conditions
  - $u(x, y) = 1$ on $\Gamma_1$
  - $u(x, y) = 0$ on $\Gamma_0$

with $\Gamma_1 = \{(x, y), x = 0, 0 \leq y \leq 0.2\} \cup \{(x, y), y = 0\}$ and $\Gamma_0 = \partial \Omega \setminus \Gamma_1$.

**Figure**: Advection skew mesh setup
Advection Skew Mesh Results
Advection skew mesh, profiles for different stabilization methods, with $\alpha = \tan(0.5)$
Advection Skew Mesh Results

Profiles with SUPG Stabilization with parameters 1 and 2

(a) SUPG-1, $\alpha = \text{atan}(1)$

(b) SUPG-2, $\alpha = \text{atan}(1)$

(c) SUPG-1, $\alpha = \text{atan}(2)$

(d) SUPG-2, $\alpha = \text{atan}(2)$
Shock Capturing Methods
Isotropic Artificial Diffusion

Additional Term in the variational form:

\[
(R_h(u_h), \tau_K^{sc} z_h \cdot \nabla v_h),
\]

(11)

with the following definition for \( \tau_K^{sc} \)

Galeão and do Carmo (ID-GC) [Do Carmo and Galeão, 1991]

\[
\tau_K^{sc} = \tau_K \max \left( 0, \frac{|a|}{|z_h|} - 1 \right)
\]

(12)

Almeida et al. (ID-A) [Almeida and Silva, 1997]

\[
\tau_K^{sc} = \tau_K \max \left( 0, \frac{|a|}{|z_h|} - \zeta_K \right), \quad \text{with} \quad \zeta_K = \max \left( 1, \frac{a \cdot \nabla u_h}{R_h(u_h)} \right).
\]

(13)

where

- \( z_h = R_h(u_h) \frac{\nabla u_h}{|\nabla u_h|^2} \)
- \( R_h(u_h) \) is the discrete residual,
- \( \tau_K \) is the SUPG/GLS stabilization parameter
Shock Capturing Methods
Artificial Diffusion Added Orthogonally to Streamlines

Additional term:

\[
(\tau^{sc}_K \bar{D} \nabla u_h, \nabla v_h)
\]  (14)

with some proposition for \(\tau^{sc}_K\)

Codina, modified by Knoblock (OD-CK) [Codina, 1993, John and Knobloch, 2006]

\[
\tau^{sc}_K = \frac{1}{2} \max \left(0, C - \frac{2\kappa}{Q_K(u_h) h_e} \right) h_e Q_K(u_h).
\]  (15)

Knobloch (OD-K) [John and Knobloch, 2006]

\[
\tau^{sc}_K = \tau_K(u_h) \frac{|a|^2 |R_h(u_h)|}{|a||\nabla u_h| + |R_h(u_h)|}
\]  (16)

with

- \(Q_K(u_h) = \frac{|R_h(u_h)|}{|\nabla u_h|}\)
- \(\bar{D} = I - a \otimes \frac{a}{|a|^2}\)
Shock Capturing Methods

Homemade Parameter

Additional term:

\[
(\tau_{sc}^K \bar{D} \nabla u_h, \nabla v_h)
\]  \hfill (17)

Shock Capturing Parameter

\[
\tau_{sc}^K = \tau_K(u_h) \max \left( 0, \frac{|a|^2 |R_h(u_h)| - Q_K^2(u_h) |R_h(u_h)|}{|a| |\nabla u_h| + |R_h(u_h)|} \right)
\]  \hfill (18)

with

\[
Q_K(u_h) = \frac{|R_h(u_h)|}{|\nabla u_h|}
\]

\[
\bar{D} = \mathbb{I} - a \otimes \frac{a}{|a|^2}
\]
Shock Capturing Methods

Convergence Study

- $\Omega = [0, 1] \times [0, 1]$
- $a = (1, 0)$
- $\kappa = 10^{-8}$
- $u(X) = \sin(\pi x)$
- $e_h = \frac{||u_h - u||_2}{||u||_2}$

Figure: Convergence Rate, error vs $h$. 

Figure: Convergence Rate, error vs $h$. 

$\mathcal{L}^2$ Norm

$\mathcal{L}^2$ Norm
Shock Capturing Methods

Advection Skew Mesh

Figure: Profiles for $\alpha = \tan(0.5)$, with different shock capturing methods

(a) ID-GC  (b) ID-A  (c) OD-CK
(d) OD-K  (e) HOME  (f) SUPG
Shock Capturing Methods

Advection Skew Mesh

Figure: Advection skew the mesh, cut along $x = 0.5$ for different shock capturing methods, $u vs y$
Shock Capturing Methods
Advection in a Rotating Flow Field

- $\Omega : (-0.5 \leq x, y \leq 0.5)$
- Advection Field $a = (-y, x)$
- BC : $u = 0$ on $\Gamma$

- $\kappa = 10^{-8}$

- BC : $u = \frac{1}{2}(\cos(4\pi y + \pi) + 1)$ on $[OA]$
- $[OA] : x = 0, -0.5 \leq y \leq 0.5$

![Diagram of a rotating flow field with boundary conditions and advection field]
Shock Capturing Methods
Advection in a Rotating Flow Field

(a) ID-GC  (b) ID-A  (c) OD-CK

(d) OD-K  (e) HOME  (f) SUPG
Slip Boundary Conditions

Problem Setting

System of equations

\[
\begin{cases}
    u \cdot \nabla u + \nabla p - \nabla \cdot (\mu \nabla u) = f, \\
    \nabla \cdot u = 0, \\
    u \cdot n = g_n, \quad \text{on } \Gamma \\
    (\mathbb{T}(u, p)n)_{\tau} = g_{\tau}, \quad \text{on } \Gamma
\end{cases}
\]  \tag{19}

where \( \mathbb{T}(u, p) = \mu \nabla u - \mathbb{I} p \) is the usual stress tensor.

Possible Solutions

- Boundaries Parallel to the Axes \( \Rightarrow \) Algebraic Elimination
- Nietsch penalization \cite{Urquiza_2014} : Extremely low convergence rate (Aborted)
- Penalization Using Lagrange Multiplier \cite{Urquiza_2014} : Validation On Going
Slip Boundary Conditions

Lagrange Multiplier Formulation

We introduce Lagrange Multiplier Space $\Lambda_h$, $\mathbb{P}^1$ discontinuous space on the boundary $\Gamma$.

Weak Formulation

Find $(u_h, p_h, \xi_h) \in V_h \times Q_h \times \Lambda_h$ such that

$$\mathcal{B}_{\gamma,h}((u_h, p_h, \xi_h), (v, q, \lambda)) = \mathcal{L}_{\gamma,h}(v, q, \lambda), \quad \forall (v, q, \lambda) \in V_h \times Q_h \times \Lambda_h, \quad (20)$$

where

$$\mathcal{B}_{\gamma,h}((u_h, p_h, \xi_h), (v, q, \lambda_h)) = \int_{\Omega} (u_h \cdot \nabla) u_h \cdot v \, d\Omega + \int_{\Omega} \mu \nabla u_h : \nabla v \, d\Omega$$

$$- \int_{\Omega} p_h \nabla \cdot v \, d\Omega - \alpha \int_{\Omega} \nabla \cdot u_h q_h \, d\Omega$$

$$+ \int_{\Gamma} \xi_h v \cdot n \, d\Gamma + \int_{\Gamma} u_h \cdot n \lambda_h \, d\Gamma$$

$$- \gamma \int_{\Gamma} h_f (\xi_h + n \nabla (u_h, p_h)) (\lambda_h + \delta n \nabla (v_h, q_h)) \, d\Gamma, \quad (21)$$

and

$$\mathcal{L}_{\gamma,h}(v, q, \lambda) = \int_{\Omega} f \cdot v \, d\Omega + \int_{\Gamma} g_n \lambda_h \, d\Gamma + \int_{\Gamma} g_\tau \cdot (v_h)_\tau \, d\Gamma \quad (22)$$
Slip Boundary Conditions

Implementation Details

Use of dynamic block structure for matrices and vectors:

```cpp
initMatrix( sparse_matrix_ptrt& M )
{
    if ( M_slip_bc )
    {
        BlocksBaseGraphCSR myblockGraph(2,2);
        myblockGraph(0,0) = stencil(_test=Xh,_trial=Xh,...)->graph();
        myblockGraph(0,1) = stencil(_test=Xh,_trial=Lh,...)->graph();
        myblockGraph(1,0) = stencil(_test=Lh,_trial=Xh,...)->graph();
        myblockGraph(1,1) = stencil(_test=Lh,_trial=Lh,...)->graph();
        M = backend()->newBlockMatrix(_block=myblockGraph);
    }
    else
    {
        M = backend()->newMatrix(Xh,Xh);
    }
}
```
Slip Boundary Conditions
Implementation Details

Use of dynamic block structure for matrices and vectors:

```cpp
initVector( vector_ptrt & V )
{
    if ( M_slip_bc )
    {
        BlocksBaseVector< double > myblockVec(2);
        myblockVec(0,0) = backend() -> newVector( Xh );
        myblockVec(1,0) = backend() -> newVector( Lh );
        V = backend() -> newBlockVector( _block=myblockVec,... );
    }
    else
    V = backend() -> newVector( Xh );
}
```
Slip Boundary Conditions

Implementation Details

Use block structure with bilinear forms

```cpp
for( auto const& condition : M_slip_bc ) {
    form2( _test=Xh, _trial=Xh, _matrix=M_diff_m )
        += integrate (markedfaces(mesh,marker(condition)),
                        - gamma*hFace() * Sigma * Sigmat);

    form2( _test=Xh, _trial=Lh, _matrix=M_diff_m,
            _rowstart=0, _colstart=2 )
        += integrate( markedfaces(mesh,marker(condition)),
                        idt(xi)*inner(id(u),N())
                        - gamma*hFace() * idt(xi) * Sigma);

    form2( _test=Lh, _trial=Xh, _matrix=M_diff_m,
            _rowstart=2, _colstart=0 )
        += integrate( markedfaces(mesh,marker(condition)),
                        id(xi)*inner(idt(u),N())
                        - gamma*hFace() * id(xi) * Sigmat);

    form2( _test=Lh, _trial=Lh, _matrix=M_diff_m,
            _rowstart=2, _colstart=2 )
        += integrate( markedfaces(mesh,marker(condition)),
                        - gamma*hFace() * id(xi) * idt(xi) );
}
```
Slip Boundary Conditions

Convergence Study with Stokes Equations  [Urquiza et al., 2014]

Square Case

- $\Omega : [-1, 1] \times [-1, 1]$
- $u_e = (2y(1 - x^2), -2x(1 - y^2))$
- $(\nabla(u_e, p_e) n)_{\tau} = u_e = (2y(1 - x^2), -2x(1 - y^2))$
- $g_n = 0, \ g_{\tau} = (2y(1 - x^2), -2x(1 - y^2))$

Ring Case

- $\Omega : 1 \leq x^2 + y^2 \leq 4$
- $u_e = (-y \sqrt{x^2 + y^2}, x \sqrt{x^2 + y^2})$
- $g_n = 0, \ g_{\tau} = (\nabla(u_e, p_e) n)_{\tau} = (-2y, 2x)$
- Strong Dirichlet on the inside boundary ($x^2 + y^2 = 1$)

Both Cases

- $\mu = 1$
- $h = 0.2, 0.1, 0.05, 0.025$
- $p_e = 0$
Square Case Convergence Study

Error on $u$, $\mathcal{L}^2$-norm

Figure: Convergence study on square test case

(a) $\alpha = 1, \delta = 1$  (b) $\alpha = 1, \delta = -1$  (c) $\alpha = -1, \delta = 1$  (d) $\alpha = -1, \delta = -1$
Square Case Convergence Study
Error on $u$, $H^1$-norm

Figure: Convergence study on square test case

(a) $\alpha = 1, \delta = 1$  (b) $\alpha = 1, \delta = -1$  (c) $\alpha = -1, \delta = 1$  (d) $\alpha = -1, \delta = -1$
Square Case Convergence Study
Error on $p$, $L^2$-norm

$\|p_h - p_e\|_2$

(a) $\alpha = 1, \delta = 1$   (b) $\alpha = 1, \delta = -1$   (c) $\alpha = -1, \delta = 1$   (d) $\alpha = -1, \delta = -1$

Figure: Convergence study on square test case
Ring Case Convergence Study

Error on $u$, $L^2$-norm

\[\|u_h - u_e\|_2\]

\[\begin{array}{c}
\text{Dirichlet} & \gamma = 10^{-4} & \gamma = 10^{-2} \\
\gamma = 1 & \gamma = 10^2 & \gamma = 10^4
\end{array}\]

(a) $\alpha = 1, \delta = 1$

(b) $\alpha = -1, \delta = -1$

Figure: Convergence study on ring test case
Ring Case Convergence Study

Error on $u$, $H^1$-norm

![Graph showing convergence study](image)

(a) $\alpha = 1, \delta = 1$

(b) $\alpha = -1, \delta = -1$

Figure: Convergence study on ring test case
Ring Case Convergence Study

Error on $p$, $L^2$-norm

<table>
<thead>
<tr>
<th>Condition</th>
<th>$\gamma$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dirichlet</td>
<td>$10^{-4}$</td>
</tr>
<tr>
<td>$\gamma = 1$</td>
<td>$10^{-2}$</td>
</tr>
<tr>
<td>$\gamma = 10^2$</td>
<td>$10^4$</td>
</tr>
</tbody>
</table>

Figure: Convergence study on ring test case

(a) $\alpha = 1, \delta = 1$

(b) $\alpha = -1, \delta = -1$
Turbulence Modelization

**DNS** : Direct Numerical Simulation
- Too expensive

**RANS Model** : Reynolds Average Navier Stokes
- Temporal averaging of the turbulence
- Adapted to stationary problems
- Issue : Boundary layer treatment

**LES** : Large Eddy Simulation
- Spatial averaging of the turbulence
- Adapted to transient problems
RANS Formulation

Reynolds Averaged Navier Stokes

- Time averaged Equations
- Approximation of the Reynolds Stress Tensor:

\[-\tau_{ij} = \nu_t \frac{\partial u_i}{\partial x_j}\]  \(\text{(23)}\)

- Use of turbulent Prandt:

\[\kappa_t = \frac{\nu_t}{Pr_t}\]  \(\text{(24)}\)

Equations

\[\begin{cases}
\rho \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p - \nabla \cdot ((\mu + \mu_t)\nabla \mathbf{u}) = \rho \beta (T - T_0) \mathbf{g}, & \text{in } \Omega \times [0, T_f], \\
\nabla \cdot \mathbf{u} = 0, & \text{in } \Omega \times [0, T_f], \\
\mathbf{u} \cdot \nabla T - \nabla \cdot ((\kappa + \kappa_t)\nabla T) = 0, & \text{in } \Omega \times [0, T_f], \\
+ \text{Boundary Conditions,} \\
+ \text{Closure Equations.}
\end{cases}\]  \(\text{(25)}\)
Some RANS Models

- Spalart-Allmaras: one equation
- $k - \epsilon$ (and associated): two equations (most common model)
- $k - \omega$ (and associated): two equations
- SST (Menter’s Shear Stress Transport): two equations
- Reynolds Stress equation Model (RSM): seven equations
Spalart Allmaras Equations

- Eddy Viscosity:
  \[
  \nu_t = \bar{\nu} f_{v1}
  \]  \hspace{1cm} (26)

- Equation

\[
\frac{\partial \bar{\nu}}{\partial t} + \mathbf{u} \cdot \nabla \bar{\nu} = \frac{1}{\sigma} [\nabla \cdot ((\nu + \bar{\nu}) \nabla \nu) + c_{b2} (\nabla \bar{\nu} \cdot \nabla \bar{\nu})] - c_{b1} (1 - f_{t2}) \bar{S} \bar{\nu} \\
\hspace{1cm} + [c_{w1} f_w - \frac{c_{b1}}{\kappa^2} f_{t2}] \left( \frac{\bar{\nu}}{d} \right)^2 = 0, \hspace{1cm} (27)
\]

with

- \( \chi = \frac{\bar{\nu}}{\nu} \)
- \( f_{v1} = \frac{\chi^3}{\chi^3 + c_{v1}^3} \)
- \( \bar{S} = \Omega + \frac{\bar{\nu}}{\kappa^2 d^2} f_{v2} \)
- \( \Omega = \sqrt{\left( \nabla \times \mathbf{u} \right) \cdot \left( \nabla \times \mathbf{u} \right)} \)
- \( f_{v2} = 1 - \frac{\chi}{1 + \chi f_{v1}} \)
- \( f_{t2} = c_{t3} \exp(-c_{t4} \chi^2) \)
- \( f_w = g \left( \frac{1 + c_{w3}^6}{g^6 + c_{w3}^6} \right)^{1/6} \)
- \( g = r + c_{w2} (r^6 - r) \)
- \( r = \min \left( \frac{\bar{\nu}}{S \kappa^2 d^2}, 10 \right) \)

- \( c_{b1}, c_{b2}, c_{w1}, c_{w2}, c_{w3}, c_{v1}, c_{t3}, c_{t4}, \kappa, \) and \( \sigma \) are empirical constants
$k - \omega$ SST Model

The turbulent eddy viscosity is computed by

$$\mu_t = \frac{\rho a_1 k}{\max (a_1 \omega, SF_2)}$$

with turbulent kinetic energy $k$ and specific dissipation $\omega$ are solutions of

$$\begin{cases}
\frac{\partial k}{\partial \omega} + u \nabla k + \beta^* k \omega - \nabla \cdot \left[ \frac{\mu + \sigma_k \mu_t}{\rho} \nabla k \right] = P_k, \\
\frac{\partial \omega}{\partial t} + u \nabla \omega + \beta \omega^2 - \nabla \cdot \left[ \frac{\mu + \sigma_w \mu_t}{\rho} \nabla \omega \right] = \frac{\gamma}{\mu_t} P_k + 2(1 - F1) \frac{\sigma_{w2}}{\omega} \nabla k \cdot \nabla \omega,
\end{cases}$$

with the following closure relations

$$F_1 = \tanh (\text{arg}_1^4), \quad \text{arg}_1 = \min \left[ \max \left( \frac{\sqrt{k}}{\beta^* \omega y}, \frac{500 \mu}{\rho y^2 \omega} \right), \frac{4 \sigma_{w2} k}{CD_{k\omega} y^2} \right]$$

$$F_2 = \tanh (\text{arg}_2^2), \quad \text{arg}_2 = \max \left( \frac{2 \sqrt{k}}{\beta^* \omega y}, \frac{500 \mu}{\rho y^2 \omega} \right)$$

$$P_k = \min (\bar{T} : \nabla u, 10 \beta^* k \omega), \quad CD_{k\omega} = \max \left( 2 \rho \frac{\sigma_{w2}}{\omega} \nabla k \nabla \omega, 10^{-10} \right)$$

$$S = \sqrt{2(\bar{S} : \bar{S})}.$$
**$k - \omega$ SST Model**

### Closure Details

Each closure constant is a blend of an inner and an outer constant, via

$$\phi = F_1 \phi_1 + (1 - F_1) \phi_2$$

(34)

where $\phi_1$ represent constant 1 and $\phi_2$ represent constant 2. The closure constant are

$$\begin{align*}
\sigma_k &= 0.85, & \sigma_\omega &= 0.5, & \beta_1 &= 0.075, & \gamma_1 &= 5/9 \\
\sigma_k &= 1.0, & \sigma_\omega &= 0.856, & \beta_2 &= 0.0828, & \gamma_2 &= 0.44
\end{align*}$$

(35)

$$\beta^* = 0.09, \quad \kappa = 0.41, \quad a_1 = 0.31.$$  

We also remind some usual notations. $\bar{\tau}$ is the Reynolds stress tensor, defined, for incompressible flow, by

$$\bar{\tau}_{ij} = 2\mu_t \bar{S}_{ij} - \frac{2}{3} \rho k \delta_{ij}$$

(36)

where $\delta_{ij}$ is the Kronecker delta and $\bar{S}$ is given by stress tensor,

$$\bar{S}_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right).$$

(37)

We can also rewrite this tensors as

$$\bar{\tau} = 2\mu_t \bar{S} - \frac{2}{3} \rho k I,$$

$$\bar{S} = \frac{1}{2} (\nabla u + (\nabla u)^T).$$

(38)
Near-Wall Treatments

**Option 1 : Full Resolution of the Boundary Layer**
- Expensive
- Complex Meshing
- Accuracy
- Stabilization Issue

**Option 2 : Modelization of the Boundary Layer**
- Can be inappropriate (accuracy issue)
- Need of specific numerical tools (slip conditions)
- Less expensive
Example of Boundary Layer Mesh
Example of Boundary Layer Mesh (zoom)
Turbulence Modelization Current Work

Spalart Allmaras
- Model implemented
- No wall functions (Full Resolution)
- Stable up to $Re = 60000$ on Naca0012 test case
- Instability issue in the boundary layer

$k - \omega$ SST model
- Model implemented
- With wall functions
- Waiting for Slip Boundary Conditions
Naca 12 use-case
Bibliography I


