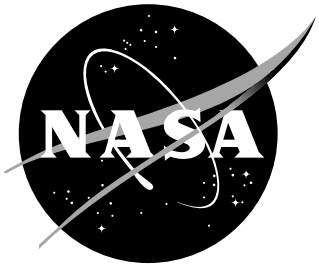
NASA/TM–20230014463



Implementation, Realization and an Effective Solver of Two-Equation Turbulence Models

S. Langer
DLR, Deutsches Zentrum für Luft- und Raumfahrt
Institut für Aerodynamik und Strömungstechnik,
Braunschweig, Germany

R. C. Swanson
NASA Langley Research Center, Hampton, Virginia

October 2023

The NASA STI Program Office ... in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program Office plays a key part in helping NASA maintain this important role.

The NASA STI Program Office is operated by Langley Research Center, the lead center for NASA's scientific and technical information. The NASA STI Program Office provides access to the NASA STI Database, the largest collection of aeronautical and space science STI in the world. The Program Office is also NASA's institutional mechanism for disseminating the results of its research and development activities. These results are published by NASA in the NASA STI Report Series, which includes the following report types:

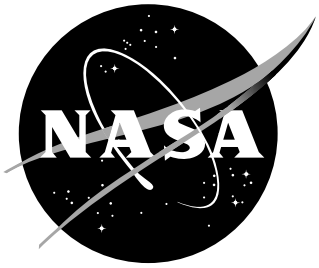
- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers, but having less stringent limitations on manuscript length and extent of graphic presentations.
- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.
- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.
- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.
- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services that complement the STI Program Office's diverse offerings include creating custom thesauri, building customized databases, organizing and publishing research results ... even providing videos.

For more information about the NASA STI Program Office, see the following:

- Access the NASA STI Program Home Page at ***http://www.sti.nasa.gov***
- E-mail your question via the Internet to help@sti.nasa.gov
- Fax your question to the NASA STI Help Desk at (301) 621–0134
- Phone the NASA STI Help Desk at (301) 621–0390
- Write to:
  NASA STI Help Desk
  NASA Center for AeroSpace Information
  7115 Standard Drive
  Hanover, MD 21076–1320

NASA/TM–20230014463



Implementation, Realization and an Effective Solver of Two-Equation Turbulence Models

S. Langer
DLR, Deutsches Zentrum für Luft- und Raumfahrt
Institut für Aerodynamik und Strömungstechnik,
Braunschweig, Germany

R. C. Swanson
NASA Langley Research Center, Hampton, Virginia

National Aeronautics and
Space Administration

October 2023

Available from:

NASA Center for AeroSpace Information (CASI)
7115 Standard Drive
Hanover, MD 21076–1320
(301) 621–0390

National Technical Information Service (NTIS)
5285 Port Royal Road
Springfield, VA 22161–2171
(703) 605–6000

Abstract

Currently, when the Reynolds-averaged Navier-Stokes (RANS) equations are solved using turbulence modeling, most often the one-equation model of Spalart and Allmaras is used. Then, it is only necessary to solve the RANS equations in conjunction with a single transport equation for modeling turbulence. For this model, considerable assessment and analysis has been performed, allowing the possibility of a reliable solution method for an eddy viscosity required to compute the Reynolds stresses in the RANS equations. Such evaluation along with analysis has not been achieved to realize similar performance with two-equation models of the k-ω type.

The primary objective of this paper is to present and discuss the components of an effective numerical algorithm for solving the RANS equations and the two transport equations of k-ω type turbulence models. All the important details of the turbulence model as actually implemented are given, which is sometimes not done in various papers considering such modeling. The viability and effectiveness of this solution algorithm are demonstrated by solving both two-dimensional and three-dimensional aerodynamic flows. In all applications, a linear rate of convergence without oscillations or other evidence of unstable behavior is observed. This behavior is also particularly true when the proposed algorithm is applied to systematically refined mesh sequences, which is generally not observed with algorithms solving more than one transport equation. Thus, numerical integration errors are systematically reduced, allowing for a significantly more reliable assessment of the effectiveness of the turbulence model. Additionally, in this paper, analysis of the solution algorithm, including linear stability, is also performed for a particular flow problem.

Contents

1 Introduction

In the field of Computational Fluid Dynamics (CFD) there has been a significant effort to develop numerical algorithms for solving the Reynolds-averaged Navier-Stokes (RANS) equations in conjunction with turbulence models. This effort has continued with the objective to extend the viability of CFD to perform a complete design of complex aircraft configurations. Thus, the convergence and reliability of a solver must be verified. Extensive data assessments of solution accuracy are clearly required. Prominent examples of these data assessments are the numerous Drag Prediction Workshops (e.g., Refs. [1–4]) and High Lift Prediction Workshops (see for example Refs. [5–7]), which are carried out regularly.

Assertions about uncertainties and errors from computed numerical data are clearly a requirement to use data obtained by numerical simulations in the aerodynamic certification processes. When designing new aircraft using numerical data, these data must be proven to be reliable. Using past AIAA workshops as a guide, different CFD computer codes typically yield a wide spread in results, even when ostensibly solving the same equations. Reasons for the wide spread are difficult to determine. Considering the fact that all participants assume that they solve the same equations with a consistent and convergent method, differences in the results must disappear as the degrees of freedom increase. For some codes and some test cases, this behavior can actually be observed. However, for simulations at the border of the flight envelope or if two-equation turbulence models are used, there are often significant deviations in the results shown at the workshops; even when statistical analyses are carried out to quantify the uncertainties [2–4]. Since convergence histories are rarely shown for these applications, this suggests that convergence may not be achieved; and hence, the results may be severely modified by numerical errors.

Significant improvements have been achieved in the last several years for the Spalart-Allmaras (SA) turbulence model [8, 9]. The improved solvability of the single transport equation of the model using the SA negative form is a major factor for the subtantial increase in using this turbulence model. Considering the results of the AIAA Drag Prediction Workshop and High Lift Prediction Workshop series of the last few years, there are more computer codes appearing using this model that can also converge several orders of magnitude. In some cases, even fully converged solutions (i.e., solutions in which the residuals are reduced to near machine zero) are achieved. Perhaps, this is at least part of the reason for the focus on introducing correction terms (e.g., streamline curvature and rotation effects) in the SA model [10, 11]. Such correction terms can provide improvement in representation of the flow physics.

During roughly the same time period as for the SA model, a number of improvements were also made for two-equation turbulence models of the k-ω type (i.e., Wilcox model and Menter SST model). Also, there was some effort to improve the solver for the RANS equations and the two-equation models. There remains a need for improvements in the convergence behavior and reliability. Thus, solving the RANS equations with such models is frequently avoided due to the high difficulty in achieving reliable convergence. Typically, to converge the RANS equations in combination with k-ω type models is a much more difficult task compared to a one-equation model.

Such two-equation turbulence models have been successfully applied to airfoil flows with small to moderate separation (e.g., see Menter [12], Swanson [13]). However, the difficulty of simulating a broad range of separated flows, including onset and extent of separation, still remains with this class of two-equation models, as well as other eddy viscosity models. Due to the demands on the data required for a certification process together with the observations from the workshops, it is clear that there is still a great need for the development of solution methods for the RANS equations, in particular for two-equation turbulence models.

From the perspective of the authors of this paper, to design such a numerical method

providing both sufficiently accurate data together with a quantifiable bound on the error, the following four points are a minimum standard one has to consider:

a) The full differential or integral formulation of the equations together with the turbulence modeling equations is given.

b) Its exact implementation is documented.

c) A solution algorithm that is able to compute for a given number of degrees of freedom a solution free of numerical integration errors.

d) The solution algorithm works for systematic mesh refinement, that is, one can obtain mesh converged results.

As soon as one of these criteria is not satisfied, certain doubts about the assertions made from approximate solutions computed with the considered numerical method emerge.

From the four criteria mentioned above, obviously the first two criteria are the simplest to satisfy. Straightforwardly, one simply has to write down the actual implementation of the equations. Nothing should be hidden such as cut-off values for certain variables. This fact is mentioned here explicitly, though often not mentioned, because a significant number of implementations use certain strategies to cut-off or to restrict several of the variables. As examples, we refer to the technical documentions [14, 15]. A final solution is generally not examined with respect to activity of such limiting processes. To assess the impact corresponding to such limitings, there must be transparency of the solutions. Since often authors do not discuss or do not even mention these intrusions into the equations, only conjectures about the reasons for discrepancies between the approximate solution and experimental data can be made. One conjecture of relevance is that c) mentioned above is not independent of b).

The reason for this, which is possibly the major issue in constructing a reliable algorithm to approximately solve the RANS equations and the two transport equations of k-ω type turbulence models, is to ensure positivity of the turbulence kinetic energy k and the energy dissipation rate ω. Perhaps, the most relevant publications dealing with this topic are Refs. [16–19]. They apply an implicit algorithm to solve the RANS equations and the two equations of a k-ω model in a loosely coupled manner. The advantage of this approach is that only the Jacobian matrix for the turbulence equations is manipulated such that it is an M-matrix for each time step. Hence, starting with a positive k and ω, all successive values of these two dependent variables remain positive throughout the iteration process. Another idea to ensure positivity of k and ω uses variable subtitution (logarithmic form), which was suggested by Ilinca et al. [20]. A number of other finite element methods (FEMs), not only Continuous Petrov-Galerkin methods but also Discontinuous Galerkin (DG) methods, also employ this approach (e.g., Refs. [21–23]). In some FEMs, realizability conditions are also imposed [21, 22]. As a consequence, it is not obvious if the transformed boundary-value problem is still equivalent to the original one. Also, as discussed in Ref. [24], there are possible effects due to smoothness of the solution as well as the modified boundary condition that may have an impact on the resulting boundary-value problem. In this article, we suggest a rather simple algorithm to deal with this positivity issue.

Given the wide range of applications for which solutions to the RANS equations are required, it is desirable that a computer code should meet the following requirements:

- It works for a large variety of parameters defining the boundary-value problem to solve, such as
  - a variety of geometries,

  - a large number of inflow conditions, which includes a range from very low Mach number to hypersonic flows,
  - a broad range of Reynolds numbers,
  - a large number of different boundary conditions;
- It works for a broad range of parameters determining the actual solution method, such as
  - variation in CFL number,
  - inner linear solution methods,
  - linear and nonlinear multigrid as well as cycling strategies;
- It always converges to a steady-state solution, if a steady-state solution exists;
- It does not show significant loss in convergence rates with systematic mesh refinement studies, for example, an increase in the number of degrees of freedom are considered.

From the point of view of the authors, at this time, the design of a solution method for the RANS equations satisfying all these conditions is an open problem. It can be assumed that various interventions (using cutoff values or restriction of variables) into several solution methods have been incorporated to be in a position to compute steady-state solutions, at least for a small number of problems with a specific choice of parameters.

Considering this background, the primary objective of this paper is to propose a numerical algorithm that efficiently solves the RANS equations in combination with two-equation models of the k-ω type on structured and unstructured meshes. Numerical examples demonstrate that the number of nonlinear iterations is comparable to state-of-the art methods for one-equation models. In this regard, we refer to Refs. [25–31], where a similar number of nonlinear iterations is required using similar methods for the one-equation model of Spalart and Allmaras. We demonstrate that a reliable behavior of the solution algorithm that is without oscillations or a deterioration in the final convergence rate is not only possible for basic two-dimensional (2-D) test cases but also for three-dimensional (3-D) transonic test cases at the design point and for a 3-D high lift configuration. Beyond the pure heuristics and test cases, an analysis is performed in particular for the two-equation turbulence models, which reveals some of the special characteristics of these models and what one needs to consider in order to implement them successfully.

In Sections 2 and 3 of the paper, the integral form of the RANS equations and the turbulence modeling equations, respectively, are defined. This includes the boundary conditions necessary for a well defined boundary-value problem. Section 4 considers the components of the numerical algorithm, which are the focus of the paper. Then, in Section 5, analysis of properties, including linear stability, of the solution algorithm is considered. Both 2-D and 3-D examples of applications of the numerical solution algorithm are shown and discussed in Section 6. Concluding remarks of the paper are made in Section 7.

2 Governing Equations of Fluid Flow

To describe flow behavior we consider for the domain $D \subset \mathbb{R}^m$, $m = 2, 3$, that is, an open and connected set, and an interval $[0, T) \subset \mathbb{R}$, $T > 0$. The conservative form of the RANS equations is expressed in integral form by

$$0 \quad = \quad \frac{d}{dt} V_D\left(W\right)(t) + R_{\partial D}\left(W\right)(t), \qquad t \in [0, T), \tag{1a}$$

where the integral operators V_D and $R_{\partial D}$ are given by

$$V_D(W)(t) := \int_D W(x,t)\,dx \tag{1b}$$

$$R_{c,\partial D}(W)(t) := \int_{\partial D} \langle f_c(W(y,t)), n(y)\rangle\, ds(y), \tag{1c}$$

$$R_{v,\partial D}(W)(t) := \int_{\partial D} \langle f_v(W(y,t)), n(y)\rangle\, ds(y), \tag{1d}$$

$$R_{\partial D} := R_{c,\partial D} - R_{v,\partial D}, \tag{1e}$$

and $W : D \times [0,T) \to \mathbb{R}^{m+2}$,

$$W(x,t) := (\rho(x,t), \rho(x,t)u(x,t), \rho(x,t)E(x,t))^T, \tag{2}$$

denotes the vector field of conserved variables and n is the unit outward normal on ∂D. The terms f_c and f_v describe the convective and viscous contributions

$$f_c(W) := \begin{pmatrix} \rho u \\ \rho u_1 u + p e_1 \\ \vdots \\ \rho u_m u + p e_m \\ \rho H u \end{pmatrix}, \qquad f_v(W) := \begin{pmatrix} 0 \\ \tau_1(W) \\ \vdots \\ \tau_m(W) \\ \theta(W) \end{pmatrix}, \qquad m = 2, 3.$$

Here e_i is the ith unit vector. The quantities ρ, $u = (u_1, \ldots, u_m)^T$, E and

$$H := E + p/\rho \tag{3}$$

are the density, the velocity, the specific total energy, and the enthalpy of the fluid. The equation of state

$$p = (\gamma - 1)\rho\left(E - \frac{\|u\|_2^2}{2}\right) \tag{4}$$

defines the pressure p, and γ is the gas dependent ratio of specific heats, which is 1.4 for air. Assuming that an effective viscosity

$$\mu_{\text{eff}} := \mu_{\text{eff}}(W) = \mu_{\text{eff}}(W(x,t))$$

is given and using Stokes hypothesis, the bulk viscosity vanishes and the second coefficient of viscosity $\lambda = -2/3\ \mu_{\text{eff}}$. Then, the viscous stress tensor $\tau = \tau(W) = \tau(W(x,t))$ is given by

$$\tau(W) := \mu_{\text{eff}}\mathcal{S} + \lambda\ \text{div}(u)\text{Id} = 2\mu_{\text{eff}}\left(\mathcal{S} - \frac{1}{3}\ \text{div(u)Id}\right) = 2\mu_{\text{eff}}\overline{\mathcal{S}}, \tag{5}$$

$$\overline{\mathcal{S}} := \mathcal{S} - \frac{1}{3}\ \text{div}(u)\text{Id}, \tag{6}$$

and $\mathcal{S}$ denotes the mean strain-rate tensor, which is given by the symmetric part of the total derivative of flow velocity vector u,

$$\mathcal{S} := \frac{1}{2}\left(\frac{du}{dx} + \left(\frac{du}{dx}\right)^T\right), \qquad \text{i.e.,} \quad \mathcal{S}_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right). \tag{7}$$

Throughout the paper Id(x) = x denotes the identity operator. The viscous flux term for the energy equation is given by

$$\theta(W) := \tau(W)\,u + q(W), \tag{8a}$$
$$q(W) := \kappa_{\text{eff}}\ \text{grad}\ T. \tag{8b}$$

The effective viscosity μ_{eff} and effective conductivity κ_{eff} are computed by

$$\mu_{\text{eff}} := \mu_l + \mu_t, \qquad \kappa_{\text{eff}} := \kappa_l + \kappa_t, \tag{9}$$

and the laminar viscosity is given by Sutherland's law

$$\mu_l(W) := \mu_{l,\infty}\left(\frac{T}{T_\infty}\right)^{3/2}\frac{T_\infty + \bar{T}}{T + \bar{T}}, \qquad \mu_{l,\infty} := \frac{\rho_\infty u_\infty L}{Re}, \tag{10}$$
$$\kappa_l(W) := \frac{c_p \mu_l(W)}{Pr_l} \quad \text{and} \quad c_p := \Re\frac{\gamma}{\gamma - 1}, \tag{11}$$

where $T_\infty > 0$, $\rho_\infty > 0$ and $u_\infty > 0$ denote some constant reference temperature, density and velocity, $L > 0$ is some constant reference length scale, $Re > 0$ is the corresponding Reynolds number, $\bar{T} := 110.4\text{K}$ is Sutherland's constant, $\Re$ is the universal gas constant, and the laminar Prandtl number is given by $Pr_l := 0.72$.

In this article, we restrict ourselves to linear, two-equation turbulence models of the k-ω type, represented by the differential or integral equations stated in Section 3. The solutions of these equations reveal additional quantities in the considered fluid. These occurring variables extend the degrees of freedom given by the conservative variables W by a further unknown function

$$W_t : D \times [0,T) \to \mathbb{R}^2.$$

The additional variables are used to determine the eddy viscosity,

$$\mu_t = \mu_t(W_t(x,t), W(x,t)) \geq 0 \quad \text{for all} \quad (x,t) \in D \times [0,T),$$

required for (9). Given the eddy viscosity μ_t, the turbulent thermal conductivity is described by the algebraic relation

$$\kappa_t := c_p \frac{\mu_t}{Pr_t}, \qquad Pr_t := 0.9. \tag{12}$$

The dependent variables of the turbulence models are the turbulence kinetic energy per unit mass k and the dissipation rate ω of the turbulence kinetic energy. Before we state the two transport equations for k and ω, we define $t = (t_{ij})_{1\leq i,j\leq m}$ and the Reynolds stress tensor $\tau^{\text{RS}} = \left(\tau_{ij}^{\text{RS}}\right)_{1\leq i,j\leq m}$, using the strain-rate tensors $\mathcal{S}$ and $\overline{\mathcal{S}}$, given in (7) and (6), and the mean molecular stress tensor, $t = (t_{ij})_{1\leq i,j\leq m}$. Then,

$$t = 2\mu_{\text{eff}}\overline{\mathcal{S}}, \qquad \rho\tau^{\text{RS}} = 2\mu_t\overline{\mathcal{S}} - \frac{2}{3}\rho k\text{Id}. \tag{13}$$

Additionally, we define the mean rotation tensor Ω, which is the skew-symmetric part of the total derivative of flow velocity u,

$$\Omega := \frac{1}{2}\left(\frac{du}{dx} - \left(\frac{du}{dx}\right)^T\right), \qquad \text{i.e.} \quad \Omega_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i}\right). \tag{14}$$

3 Two-Equation Turbulence Models of k-ω Type

The principal elements of the k-ω type turbulence models being considered are described in this section. The emphasis is on the 1988 and 2006 Wilcox models and the 2003 Menter Shear Stress Transport (SST) model. Presentation and extensive discussion of these models is given in Refs. [32–34] and [12, 35–37]. An additional resource providing information on these and other turbulence models is NASA's **T**urbulence **M**odeling **R**esource (TMR) website [38]. Important details of these models, such as parameters and closure coefficients, can be found in the TMR or in the original publications. For convenience, these details are also given in the appendices.

3.1 Wilcox k-ω model

In this article, we consider k-ω models of the form

$$\frac{\partial(\rho k)}{\partial t} + \operatorname{div}(\rho k u) = \operatorname{div}((\mu_l + \sigma_k \mu_t)\operatorname{grad} k) + \rho Q_{k,(k,\omega)}, \tag{15a}$$

$$\frac{\partial(\rho \omega)}{\partial t} + \operatorname{div}(\rho \omega u) = \operatorname{div}((\mu_l + \sigma_\omega \mu_t)\operatorname{grad} \omega) + \rho Q_{\omega,(k,\omega)}. \tag{15b}$$

The eddy viscosity in these models is computed by

$$\mu_t = \mu_t(\mathbf{W}, k, \omega) = \rho k / z(\mathbf{W}, k, \omega) \tag{16}$$

with a given function z. For example, in the 1988 and 2006 models of Wilcox, we have

$$z(\mathbf{W}, k, \omega) = \omega, \quad \text{(1988 model)}, \tag{17}$$

$$z(\mathbf{W}, k, \omega) = \max\left\{\omega, C_{\lim}\sqrt{\frac{2\Omega : \Omega}{\beta^*}}\right\}, \quad \text{(2006 model)}. \tag{18}$$

The symbol : denotes a double dot tensor product. The source terms for the k and ω equations include production (Pr) and destruction (De) terms. A cross-diffusion term (Di) is also included in the source term ($\Psi = 1$) of the ω equation for the 2006 Wilcox model, but it is not included ($\Psi = 0$) in the 1988 model.

$$Q_{k,(k,\omega)} := \tilde{Pr}_{k,(k,\omega)} - De_{k,(k,\omega)}, \quad Q_{\omega,(k,\omega)} := Pr_{\omega,(k,\omega)} - De_{\omega,(k,\omega)} + \Psi Di_{\omega,(k,\omega)}, \tag{19}$$

where

$$Pr_{k,(k,\omega)} := \tau^{\mathrm{RS}} : \frac{du}{dx}, \qquad De_{k,(k,\omega)} := \beta^* k\omega \tag{20a}$$

$$Pr_{\omega,(k,\omega)} := \alpha\frac{\omega}{k}\tau^{\mathrm{RS}} : \frac{du}{dx}, \qquad De_{\omega,(k,\omega)} := \beta\omega^2. \tag{20b}$$

$$Di_{\omega,(k,\omega)} := \sigma_d \frac{1}{\omega}\langle \operatorname{grad} k, \operatorname{grad} \omega\rangle. \tag{20c}$$

The symbols σ_k and σ_ω represent fixed constants defined in Appendix A. In the 1988 Wilcox model, the production term of the k equation is generally limited according to

$$\tilde{Pr}_{k,(k,\omega)} := \min\left\{Pr_{k,(k,\omega)}, 20De_{k,(k,\omega)}\right\}, \tag{21}$$

so as to prevent very large Pr_k during the initial phases of a calculation. An example to demonstrate the impact and influence of Eq.(21) on possible solutions is presented in Section 7 of Ref. [24]. This change is not included in the 2006 Wilcox model due to the introduction of a stress limiter. This stress limiter can have a significant effect on a shock location. The effect of varying the stress limiter on the shock location for the RAE 2822 airfoil (Case 10) is shown in Swanson and Rossow [39].

3.2 Menter k-ω model (SST model)

In the 2003 Menter SST model, the constants σ_k and σ_ω in the two transport equations (15a) and (15b) are replaced with blending functions Φ_k and Φ_ω. These blending functions that depend on the σ_k and σ_ω in the k-ω and k-ε models. The source terms are also changed due to blending functions for the k-ω and k-ε models.

The eddy viscosity is determined from Eq. (16) with $z(\mathbf{W}, k, \omega)$ given by

$$z(\mathbf{W}, k, \omega) = \frac{1}{\min\left\{\frac{1}{\omega}, \frac{a_1}{F_2\sqrt{2S:S}}\right\}} = \max\left\{\omega, \frac{F_2\sqrt{2S:S}}{a_1}\right\}. \tag{22}$$

Here, F_2 is a blending function defined by

$$F_2 := \tanh\left(\left(\max\left\{\Gamma_1, 2\Gamma_3\right\}\right)^2\right) \tag{23}$$

where

$$\Gamma_1 := \frac{C_{\Gamma_1}\nu_l}{d^2\omega}, \qquad \Gamma_3 := \frac{\sqrt{k}}{\beta^*\omega d}, \quad \beta^* := 0.09, \tag{24}$$

d is the distance to the closest no-slip wall, and ν_l is the kinematic viscosity. According to Ref. [12], the constants are $a_1 = 0.31$ and $C_{\Gamma_1} = 500$. The source terms, having the same components as in Eq. (19), are given by

$$Q_{k,\mathrm{SST}} = Pr_{k,\mathrm{SST}} - De_{k,\mathrm{SST}}, \qquad Q_{\omega,\mathrm{SST}} = Pr_{\omega,\mathrm{SST}} - De_{\omega,\mathrm{SST}} + Di_{\omega,\mathrm{SST}}, \tag{25}$$

where

$$Pr_{k,\mathrm{SST}} := \tau^{\mathrm{RS}} : \frac{du}{dx}, \qquad De_{k,\mathrm{SST}} := \beta^* k\omega, \tag{26a}$$

$$Pr_{\omega,\mathrm{SST}} := \Phi_\gamma \frac{1}{\nu_t}\tau^{\mathrm{RS}} : \frac{du}{dx}, \qquad De_{\omega,\mathrm{SST}} := \Phi_\beta \omega^2, \tag{26b}$$

$$Di_{\omega,\mathrm{SST}} := 2\left(1 - F_1\right)\sigma_{\omega_2}\frac{1}{\omega}\left\langle \mathrm{grad}\ k, \mathrm{grad}\ \omega\right\rangle. \tag{26c}$$

The blending of the SST model is controlled by a function $\Phi = \Phi(x; \varepsilon_1, \varepsilon_2)$. This function is designed to detect the edge of the boundary layer, such that the SST model behaves inside the boundary layer like a k-ω model and outside like a k-ε model, exploiting the convex combination $\Phi : [0,1] \to [\varepsilon_1, \varepsilon_2]$,

$$\Phi\left(F_1; \varepsilon_1, \varepsilon_2\right) := F_1\varepsilon_1 + \left(1 - F_1\right)\varepsilon_2. \tag{27}$$

Details of the blending, blending functions, and coefficients of the SST model are given in Appendix B.

3.3 Simplifications of k-ω Models

Although these models are formulated in their compressible form (15), their actual implementation and usage is often based on their incompressible version. In fact, this occurs even when they are used with respect to compressible flow. Then, one assumes

$$\mathrm{div}\left(u\right) = 0, \tag{28}$$

yielding for the equation of mass conservation $0 = \frac{\partial\rho}{\partial t} + \langle \mathrm{grad}\ \rho, u\rangle$, and hence

$$\frac{\partial\left(\rho k\right)}{\partial t} + \mathrm{div}\left(\rho k u\right) = \rho\left(\frac{\partial k}{\partial t} + \mathrm{div}\left(k u\right)\right), \tag{29}$$

$$\frac{\partial\left(\rho\omega\right)}{\partial t} + \mathrm{div}\left(\rho\omega u\right) = \rho\left(\frac{\partial\omega}{\partial t} + \mathrm{div}\left(\omega u\right)\right). \tag{30}$$

Inserting Eq. (29) and Eq. (30) into Eq. (15) gives

$$\frac{\partial k}{\partial t} + \operatorname{div}(ku) = \frac{1}{\rho}\operatorname{div}\left(\left(\mu_l + \sigma_k \mu_t\right)\operatorname{grad} k\right) + Q_{k,(k,\omega)}, \tag{31a}$$

$$\frac{\partial \omega}{\partial t} + \operatorname{div}(\omega u) = \frac{1}{\rho}\operatorname{div}\left(\left(\mu_l + \sigma_\omega \mu_t\right)\operatorname{grad} \omega\right) + Q_{\omega,(k,\omega)}, \tag{31b}$$

and the independent variables $W_t = (\rho k, \rho\omega)$ are replaced by $W_t = (k, \omega)$. To be consistent using assumption given in Eq. (28), we have

$$\rho\tau = 2\mu_t \overline{\mathcal{S}}. \tag{32}$$

In a second step, assumption (28) can be integrated into (32). This yields $\overline{\mathcal{S}} = \mathcal{S}$ and finally

$$\rho\tau = 2\mu_t \mathcal{S}. \tag{33}$$

Remark *Often, it is not indicated if the assumption Eq. (28) is included into the formulation of the turbulence model, that is, if Eq. (32) or Eq. (33) is used for the formulation of the production terms. For the implementation considered here, we chose Eq. (32).*

3.4 Integral Form of k-ω Models

When integrating the diffusive terms in Eq. (31) over a control volume, they cannot be rewritten as a surface integral because of division with density ρ. Hence, a further approximation is introduced,

$$\frac{1}{\rho}\operatorname{div}\left(\left(\mu_l + \sigma_k \mu_t\right)\operatorname{grad} k\right) \approx \operatorname{div}\left(\left(\nu_l + \sigma_k \nu_t\right)\operatorname{grad} k\right), \tag{34a}$$

$$\frac{1}{\rho}\operatorname{div}\left(\left(\mu_l + \sigma_\omega \mu_t\right)\operatorname{grad} \omega\right) \approx \operatorname{div}\left(\left(\nu_l + \sigma_\omega \nu_t\right)\operatorname{grad} \omega\right). \tag{34b}$$

Integration of Eq. (31) using the approximation of Eq. (34) gives the integral equation

$$V_D\left(Q_{(k,\omega)}\left(W_t, W\right)\right)(t) = \frac{d}{dt} V_D\left(W_t\right)(t) + R_{\partial D,(k,\omega)}\left(W_t, W\right)(t), \tag{35}$$

where the integral operators are $R_{\partial D,(k,\omega)} := R_{c,\partial D,(k,\omega)} - R_{v,\partial D,(k,\omega)}$, and

$$R_{c,\partial D,(k,\omega)}\left(W_t, W\right)(t) := \int_{\partial D} \left\langle f_{c,(k,\omega)}\left(W_t\left(y,t\right), W\left(y,t\right)\right), n(y)\right\rangle ds(y),$$

$$R_{v,\partial D,(k,\omega)}\left(W_t, W\right)(t) := \int_{\partial D} \left\langle f_{v,(k,\omega)}\left(W_t\left(y,t\right), W\left(y,t\right)\right), n(y)\right\rangle ds(y).$$

Here, the convective $f_{c,(k,\omega)}$ and viscous $f_{v,(k,\omega)}$ contributions as well as the source terms $Q_{(k,\omega)}$ are summarized by

$$f_{c,(k,\omega)}\left(W_t, W\right) := \begin{pmatrix} ku \\ \omega u \end{pmatrix}, \tag{36a}$$

$$f_{v,(k,\omega)}\left(W_t, W\right) := \begin{pmatrix} \left(\nu_l + \sigma_k \nu_t\right)\operatorname{grad} k \\ \left(\nu_l + \sigma_\omega \nu_t\right)\operatorname{grad} w \end{pmatrix}, \tag{36b}$$

$$Q_{(k,\omega)}\left(W_t, W\right) := \begin{pmatrix} Pr_{k,(k,\omega)} - De_{k,(k,\omega)} \\ Pr_{\omega,(k,\omega)} - De_{\omega,(k,\omega)} + Di_{\omega,(k,\omega)} \end{pmatrix}. \tag{36c}$$

3.5 Nondimensionalization

For a numerical implementation, a suited scaling and nondimensionalization is often crucial. Due to the no-slip wall boundary condition (see Ref. [24] and Section 2) for ω,

$$\lim_{h\to 0+} \omega\left(x - hn(x)\right) h^2 = \frac{6\nu_l}{\beta}, \qquad x \in \partial D_{\mathrm{no-slip}}, \tag{37}$$

typically, the values for ω related to k and the other conservative variables Eq. (2) may differ in orders of magnitude. For this reason, a scaling of ω is introduced.

Throughout this subsection, we denote dimensional variables using the sign $\hat{}$, for example dimensional density is denote by $\hat{\rho}$. To nondimensionalize $\hat{k}$ and $\hat{\omega}$, we choose as reference values

$$k_{\mathrm{ref}} = u_{\mathrm{ref}}^2 \qquad \text{and} \qquad \omega_{\mathrm{ref}} = \frac{u_{\mathrm{ref}}}{L_{\mathrm{ref}}}. \tag{38}$$

Then the nondimensional variables are given by

$$k = \frac{\hat{k}}{u_{\mathrm{ref}}^2} \qquad \text{and} \qquad \omega = \frac{\hat{\omega}}{\omega_{sc}\omega_{\mathrm{ref}}}, \qquad \omega_{sc} = \frac{Re}{\sqrt{\gamma} M_\infty L}, \tag{39}$$

where ω_{sc} denotes the additional scaling for ω. Such scaling may be motivated by the following argumentation. Boundary condition Eq. (37) is realized by

$$\omega_{\mathrm{no-slip}}\left(p_{i,\mathrm{bdry}}\right) = \frac{60\nu_l\left(W(p_{i,\mathrm{bdry}})\right)}{\beta \left\| p_{i,\mathrm{bdry}} - p_{i,n} \right\|_2^2}, \qquad p_{i,bdry} \in \partial D_{\mathrm{no-slip}}, \tag{40}$$

where $p_{i,bdry}$ denotes the point on the no-slip wall and $p_{i,n}$ the closest, next discrete point in direction $-n\left(p_{i,bdry}\right)$. Obviously, with respect to possible mesh refinements the distance to the closest wall satisfies

$$d_{h,i} \approx \left\| p_{i,\mathrm{bdry}} - p_{h,i,n} \right\|_2 \to 0, \qquad h \to 0,$$

and hence possibly numerical instabilities emerge due to a significant increase in the boundary values $\omega_{\mathrm{no-slip}}$. To resolve the flow inside the boundary layer, meshes are in general generated such that this distance scales with the Reynolds number, $d_i = d_i(Re) \sim C/Re$. As a consequence, the choice of ω_{sc} is some normalization

$$\omega_{\mathrm{no-slip}} \sim \tilde{C} \frac{1}{\omega_{sc}} \frac{\frac{\sqrt{\gamma} M_\infty L}{Re}}{\left(\frac{C}{Re}\right)^2} \sim \overline{C},$$

and the largest values for ω are in a range of the other variables.

This scaling parameter needs to be correctly incorporated in all terms and equations. For a detailed study, we refer to Refs. [40, 41], here we just report the results. For example, to obtain the scaled, nondimensional version of the eddy viscosity Eq. (16) together with Eq. (22) for the SST model, we have

$$\mu_t = \frac{1}{\omega_{sc}} \rho k \min\left\{ \frac{1}{\omega}, \frac{\omega_{sc} a_1}{F_2 \sqrt{2\Omega : \Omega}} \right\}.$$

Then, introducing Eq. (38) into Eq. (36a), Eq. (36b) and Eq. (36c), we compute

$$f_{c,(k,\omega)}\left(\hat{W}_t, \hat{W}\right) = u_{\mathrm{ref}} \begin{pmatrix} k_{\mathrm{ref}} & 0 \\ 0 & \omega_{\mathrm{ref}}\omega_{sc} \end{pmatrix} f_{c,(k,\omega)}\left(W_t, W\right), \tag{41}$$

$$f_{v,(k,\omega)}\left(\hat{W}_t, \hat{W}\right) = u_{\mathrm{ref}} \begin{pmatrix} \frac{k_{\mathrm{ref}}}{\omega_{sc}} & 0 \\ 0 & \omega_{\mathrm{ref}} \end{pmatrix} \tilde{f}_{v,(k,\omega)}\left(W_t, W\right), \tag{42}$$

where

$$\tilde{f}_{v,(k,\omega)}\left(W_t, W\right) := \begin{pmatrix} \left(\Gamma(T) + \sigma_k \nu_t\right) \operatorname{grad} k \\ \left(\Gamma(T) + \sigma_\omega \nu_t\right) \operatorname{grad} \omega \end{pmatrix}, \qquad \Gamma(T) = T^{3/2}\left(\frac{1 + C_{\text{suth}}}{T + C_{\text{suth}}}\right),$$

and C_{suth} denotes Sutherland's constant. For the source terms of Eq. (36c), we obtain

$$\begin{aligned}
Pr_{k,(k,\omega)}\left(\hat{W}_t, \hat{W}\right) &= \frac{u_{\text{ref}}^3}{L_{\text{ref}}\omega_{sc}} Pr_{k,(k,\omega)}\left(W_t, W\right), && \text{(43a)}\\
De_{k,(k,\omega)}\left(\hat{W}_t, \hat{W}\right) &= \frac{u_{\text{ref}}^3 \omega_{sc}}{L_{\text{ref}}} De_{k,(k,\omega)}\left(W_t, W\right), && \text{(43b)}\\
Pr_{\omega,(k,\omega)}\left(\hat{W}_t, \hat{W}\right) &= = \frac{u_{\text{ref}}^2}{L_{\text{ref}}^2} Pr_{\omega,(k,\omega)}\left(W_t, W\right), && \text{(43c)}\\
De_{\omega,(k,\omega)}\left(\hat{W}_t, \hat{W}\right) &= \frac{u_{\text{ref}}^2 \omega_{sc}^2}{L_{\text{ref}}^2} De_{\omega,(k,\omega)}\left(W_t, W\right), && \text{(43d)}\\
Di_{\omega,(k,\omega)}\left(\hat{W}_t, \hat{W}\right) &= \frac{u_{\text{ref}}^2}{L_{\text{ref}}^2} Di_{\omega,(k,\omega)}\left(W_t, W\right). && \text{(43e)}
\end{aligned}$$

Introducing the mapping $g : D \to \hat{D}$, $x \mapsto L_{\text{ref}} x$, which maps the computational domain D to its physical domain $\hat{D}$, and application of substitution formulae

$$\begin{aligned}
\int_{\hat{D}} v(x)dx &= L_{\text{ref}}^m \int_D v(g(y))dy, && \text{(44a)}\\
\int_{\partial\hat{D}} \langle v(y), n(y)\rangle \, ds(y) &= L_{\text{ref}}^{m-1} \int_{\partial D} \langle v(g(y)), n(y)\rangle \, ds(y). && \text{(44b)}
\end{aligned}$$

Using Eqs. (41), (42) and Eqs. (43a)–(43e), the integral Eq. (35) can be converted to:

$$\begin{aligned}
&\begin{pmatrix} \frac{L_{\text{ref}}^2 u_{\text{ref}}^3}{\omega_{sc}} \int_D Pr_{k,(k,\omega)}dy - L_{\text{ref}}^2 u_{\text{ref}}^3 \omega_{sc} \int_D De_{k,(k,\omega)}dy \\ L_{\text{ref}} u_{\text{ref}}^2 \int_D Pr_{\omega,(k,\omega)}dy - L_{\text{ref}} u_{\text{ref}}^2 \omega_{sc}^2 \int_D De_{\omega,(k,\omega)}dy + L_{\text{ref}} u_{\text{ref}}^2 \int_D Di_{\omega,(k,\omega)}dy \end{pmatrix}\\
= &\begin{pmatrix} L_{\text{ref}}^2 u_{\text{ref}}^3 & 0 \\ 0 & \omega_{sc} L_{\text{ref}} u_{\text{ref}}^2 \end{pmatrix} \left\{ \frac{d}{dt} V_D(W_t)(t) + R_{c,\partial D,(k,\omega)}\left(W_t, W\right)(t) \right\}\\
- &\begin{pmatrix} \frac{1}{\omega_{sc}} L_{\text{ref}}^2 u_{\text{ref}}^3 & 0 \\ 0 & L_{\text{ref}} u_{\text{ref}}^2 \end{pmatrix} \int_{\partial D} \left\langle \tilde{f}_{v,(k,\omega)}\left(W_t, W\right), n \right\rangle ds(y).
\end{aligned}$$

Multiplication of the whole system with the diagonal matrix $\operatorname{diag}\left(\frac{1}{L_{\text{ref}}^2 u_{\text{ref}}^3}, \frac{1}{\omega_{sc}} \frac{1}{L_{\text{ref}} u_{\text{ref}}^2}\right)$ gives the mathematically equivalent system of equations

$$\begin{aligned}
&\begin{pmatrix} \omega_{sc}^{-1} \int_D Pr_{k,(k,\omega)}dy - \omega_{sc} \int_D De_{k,(k,\omega)}dy \\ \omega_{sc}^{-1} \int_D Pr_{\omega,(k,\omega)}dy - \omega_{sc} \int_D De_{\omega,(k,\omega)}dy + \omega_{sc}^{-1} \int_D Di_{\omega,(k,\omega)}dy \end{pmatrix}\\
= &\frac{d}{dt} V_D(W_t)(t) + R_{c,\partial D,(k,\omega)}\left(W_t, W\right)(t)\\
- &\begin{pmatrix} \omega_{sc}^{-1} & 0 \\ 0 & \omega_{sc}^{-1} \end{pmatrix} \int_{\partial D} \left\langle \tilde{f}_{v,(k,\omega)}\left(W_t, W\right), n \right\rangle ds(y). \qquad \text{(45)}
\end{aligned}$$

The system of equations Eq. (45) is the actual system of equations which is implemented and solved. Naturally, the choice $\omega_{sc} = 1$ or any other reasonable choice is also possible.

3.6 Boundary-Value Problem

Following the presentation in Ref. [24], a suitable formulation of the complete boundary-value problem is required. Defining the hyperplane

$$E := \{(x_1, x_2, x_3) : a_1 x_1 + a_2 x_2 + a_3 x_3 + C_\omega = 0\} \subset \mathbb{R}^3,$$

and according to E

$$\begin{aligned} V^- &:= \{(x_1, x_2, x_3) : a_1 x_1 + a_2 x_2 + a_3 x_3 + C_\omega < 0\} \subset \mathbb{R}^3, \\ V^+ &:= \{(x_1, x_2, x_3) : a_1 x_1 + a_2 x_2 + a_3 x_3 + C_\omega > 0\} \subset \mathbb{R}^3, \end{aligned}$$

we formulate the following boundary-value problem:

Exterior turbulent flow problem, formulation:
Find a function $W^\dagger$ that satisfies the steady RANS equations in $V^+ \setminus \overline{D}$, that is

$$\frac{d}{dt} W^\dagger(x,t) = 0 \qquad \text{for all} \quad x \in V^+ \setminus \overline{D}, \quad t \geq T^\dagger > 0,$$

and satisfies the (adiabatic) no-slip wall boundary conditions and

$$\begin{aligned} \lim_{h\to\infty} W(x + hy, t) &= W_\infty, \qquad x \in E, \quad \langle a, y\rangle > -C_\omega, \\ \lim_{h\to 0} W(x + hy, t) &= W_\infty, \qquad x \in E, \quad \langle a, y\rangle > -C_\omega. \end{aligned}$$

Additionally, find a function W_t that satisfies the k-ω turbulence model in $V^+ \setminus \overline{D}$, and satisfies the boundary conditions

$$(k, \omega) = (0, \infty) \qquad \text{on} \qquad \partial D$$

and

$$\begin{aligned} \lim_{h\to\infty} (k(x + hy), \omega(x + hy)) &= (0, 0), \qquad x \in E, \quad \langle a, y\rangle > 0, \\ \lim_{h\to 0} (k(x + hy), \omega(x + hy)) &= (\infty, \infty), \qquad x \in E, \quad \langle a, y\rangle > 0. \end{aligned}$$

The boundary conditions are realized following the presentation in [24].

4 Solution Algorithm

4.1 Multigrid and Implicit Smoother

The discretization strategy followed is based on a finite-volume formulation. The inviscid terms are discretized using a central difference scheme with an added matrix-valued artificial viscosity [29]. A first-order upwind scheme is applied to the convective part of the turbulent flow equations. Gradients required for the viscous terms and for the source terms are computed using a Green-Gauss formulation (see [42]). For a detailed description of the discretization strategy and boundary conditions applied, we refer to [29, 30, 40].

The discretization of the mean flow Eqs. (1) together with the turbulent flow Eqs. (35) yields the system of ordinary differential equations

$$\frac{d}{dt} \begin{pmatrix} \mathbf{W}(t) \\ \mathbf{W}_t(t) \end{pmatrix} = \begin{pmatrix} -\mathbf{M}_{\text{mean}}^{-1} \mathbf{R}_{\text{mean}} (\mathbf{W}(t), \mathbf{W}_t(t)) \\ -\mathbf{M}_{\text{turb}}^{-1} \mathbf{R}_{\text{turb}} (\mathbf{W}(t), \mathbf{W}_t(t)) \end{pmatrix}, \tag{46}$$

where

$$\begin{aligned}\mathbf{M}_{\text{mean}} &:= \text{diag}\left(\text{diag}\left(\text{vol}\left(\Omega_i\right)\right)\right) \in \mathbb{R}^{5N_{\text{elem}}\times 5N_{\text{elem}}} \\ \mathbf{M}_{\text{turb}} &:= \text{diag}\left(\text{diag}\left(\text{vol}(\Omega_i)\right)\right) \in \mathbb{R}^{2\cdot N_{\text{elem}}\times 2\cdot N_{\text{elem}}}\end{aligned}$$

denote the mass matrix for mean and turbulent flow equations. To approximately solve Eq. (46), we assume that the mean flow equations depend only on $\mathbf{W}$, and $\mathbf{W}_t$ acts only as a parameter here, whereas the turbulent flow equations depend only on $\mathbf{W}_t$, and $\mathbf{W}$ acts as a parameter. Hence, we rewrite system (46) as

$$\frac{d}{dt}\mathbf{W}(t) = -\mathbf{M}_{\text{mean}}^{-1}\mathbf{R}_{\text{mean}}\left(\mathbf{W}(t);\mathbf{W}_t(t)\right) \tag{47a}$$

$$\frac{d}{dt}\mathbf{W}_t(t) = -\mathbf{M}_{\text{turb}}^{-1}\mathbf{R}_{\text{turb}}\left(\mathbf{W}_t(t);\mathbf{W}(t)\right). \tag{47b}$$

Equations (47a) and (47b) are then solved sequentially in a loosely coupled manner (see loosely and fully coupled methods Ref. [43]).

To approximate a steady-state solution, we apply a nonlinear multigrid method (see Ref. [44]), which is called the Full Approximation Scheme (FAS), to the mean flow equation (47a). The turbulent flow equation (47b) is solved in a single grid mode, only. A multistage diagonally implicit Runge-Kutta method is used as a smoother (see Ref. [40])

$$\begin{aligned}\mathbf{W}^{(0)} &:= \mathbf{W}^n \\ \mathbf{W}^{(j)} &= \mathbf{W}^{(0)} - \alpha_{j+1,j}\mathbf{P}_j\left(\mathbf{W}^{(j-1)}\right)^{-1}\mathbf{R}\left(\mathbf{W}^{(j-1)}\right),\ j=1,\ldots,s \qquad (48)\\ \mathbf{W}^{n+1} &= \mathbf{W}^{(s)}, \\ \mathbf{P}_j\left(\mathbf{W}^{(j-1)}\right) &:= (\Delta T)^{-1}\mathbf{M} + \frac{d\mathbf{R}}{d\mathbf{W}}\left[\mathbf{W}^{(j-1)}\right], \quad \Delta T := \text{diag}\left(\text{diag}\left(\Delta t_i\right)\right), \qquad (49)\end{aligned}$$

where we have negelected the subindices mean and turb, and Δt_i denotes the local time step. To apply Eq. (48), the linear equation

$$\mathbf{Prec}_j^{-1}\mathbf{P}_j\mathbf{x} = \alpha_{j+1,j}\mathbf{Prec}_j^{-1}\mathbf{R}(\mathbf{W}^{(j-1)}). \tag{50}$$

needs to be solved. Here, $\mathbf{Prec}_j$ denotes some preconditioner, which is the major ingredient of the algorithm. To find an approximation of Eq. (50), we apply a (left) preconditioned GMRES method (see for example Ref. [45]) with initial guess $\mathbf{x}^{(0)} = 0$:

- Solve (approximately) $\mathbf{Prec}_j\mathbf{r}_0 = \alpha_{j+1,j}\mathbf{R}\left(\mathbf{W}^{(j-1)}\right)$
- Compute $\beta := \|\mathbf{r}_0\|_2, \mathbf{v}_1 := \frac{1}{\beta}\mathbf{r}_0$
- for $k = 1,\ldots,m$
  - Solve (approximately) $\mathbf{Prec}_j\mathbf{w} = \mathbf{P}_j\mathbf{v}_k$
  - for $i = 1,\ldots k$
    - $h_{i,k} := \langle\mathbf{w},\mathbf{v}_i\rangle$
    - $\mathbf{w} := \mathbf{w} - h_{i,k}\mathbf{v}_i$
  - $h_{k+1,k} = \|\mathbf{w}\|_2, \mathbf{v}_{k+1} = \frac{1}{h_{k+1,k}}\mathbf{w}$
- $\mathbf{V} = (\mathbf{v}_1,\ldots,\mathbf{v}_m), H_m = (h_{i,k})_{1\le i\le k+1, 1\le k\le m}$
- Solve $\mathbf{y}_m := \text{argmin}_{\mathbf{y}}\|\beta e_1 - H_m\mathbf{y}\|_2$ by Given's-rotations

- $\mathbf{x}^{(m)} := \mathbf{x}^{(0)} + \mathbf{V}_m \mathbf{y}_m$

In the GMRES method, the matrix-vector multiplication of the operator given in (50) applied to a vector is approximated by a finite-difference of the residual operator,

$$\frac{d\mathbf{R}}{d\mathbf{W}}(\mathbf{W})\,\mathbf{h} \approx \frac{1}{2\epsilon}\left(\mathbf{R}(\mathbf{W}+\epsilon\mathbf{h}) - \mathbf{R}(\mathbf{W}-\epsilon\mathbf{h})\right).$$

The choice of a suitable $\epsilon > 0$ is not trivial, as cancellation and approximation errors need to be balanced. A possible method can be found in [42, Chapter 6.2.5].

4.2 Construction of Preconditioner

The preconditioner has two contributions:

a) the linear operator $\mathbf{Prec}_j$ itself,

b) an iterative solution method for approximately solving the linear systems

$$\mathbf{Prec}_j \mathbf{w} = \mathbf{P}_j \mathbf{v}_k. \tag{51}$$

For the construction of the linear operators we follow the approach presented in [29, 40]:

$$\mathbf{Prec}_{j,\text{mean}} := (\Delta T)^{-1}\,\mathbf{M}_{\text{mean}} + \varepsilon\alpha_{jj}\frac{d\tilde{\mathbf{R}}^{\text{comp}}_{\text{mean}}}{d\mathbf{W}}, \tag{52}$$

$$\mathbf{Prec}_{j,\text{turb}} := (\Delta T)^{-1}\,\mathbf{M}_{\text{turb}} + \varepsilon\alpha_{jj}\frac{d\tilde{\mathbf{R}}^{\text{comp},(k,\omega)}_{\text{turb}}}{d\mathbf{W}_t}. \tag{53}$$

Here, $d\tilde{\mathbf{R}}^{\text{comp}}_{\text{prec}}/d\mathbf{W}$ is a linearization of a residual based on a compact discretization scheme. The parameter ε is introduced to allow for over and under relaxation.

The focus in this article is the presentation of the preconditioning technique for the k-ω equations. To realize this preconditioner, the derivatives of the source terms to construct $\mathbf{Prec}_{j,\text{turb}}$ do not include the destruction terms

$$\frac{\partial De_{k,(k,\omega)}}{\partial k_i} \qquad \text{and} \qquad \frac{\partial De_{\omega,(k,\omega)}}{\partial \omega_i}. \tag{54}$$

The necessity for this modification is discussed in Section 5.1.

4.3 Solving Linear Systems and Truncation Criteria

To realize the preconditioned GMRES method approximately solving (50), we need to define appropriate truncation criteria. To this end, we distinguish two principal implementations of Eq. (48):

M1) When the number of GMRES steps is 0 and only the preconditioning step is evaluated,

M2) When the number of GMRES steps is $\neq 0$.

To approximately solve (50) for the mean flow equations we follow both strategies M1 and M2:

- When choosing M1, we apply either at most 250 symmetric Gauss-Seidel sweeps or iterate until
$$\frac{\|\mathbf{Prec}_j \mathbf{r}_0 - \alpha_{j+1,j}\mathbf{R}\left(\mathbf{W}^{(j-1)}\right)\|_2}{\|\alpha_{j+1,j}\mathbf{R}\left(\mathbf{W}^{(j-1)}\right)\|_2} < 10^{-2}.$$

- When choosing M2, we truncate the GMRES iteration either after at most 20 steps or if
$$\frac{\|\mathbf{P}_j\mathbf{x}^{(m)} - \alpha_{j+1,j}\mathbf{R}(\mathbf{W}^{(j-1)})\|_2}{\|\alpha_{j+1,j}\mathbf{R}(\mathbf{W}^{(j-1)})\|_2} < 10^{-2}.$$
For approximately solving the linear systems (51), we apply five symmetric Gauss-Seidel sweeps.

4.4 Choice of Time Step

To compute the local time steps Δt_i in (49) for both the mean flow equations and for the turbulence modeling equations, we use an approximation to the spectral radius of the diagonal blocks of $\frac{d\mathbf{R}}{d\mathbf{W}}$, that is,

$$\begin{aligned}\Delta t_i \quad := \quad & \mathrm{CFL}\cdot \mathrm{vol}\,(D_i)\left[\sum_{j\in\mathcal{N}(i)} \mathrm{svol}(e_{ij})\left(\rho\left(\frac{\partial \mathcal{H}^{1st,\mathrm{Roe}}}{\partial W_i}\right)\right.\right. \\ & \left.\left. +C_v\rho\left(\frac{\partial\left\langle f_v\left(W_i,W_j\right),n_{e_{ij}}\right\rangle}{\partial W_i}\right)^{\mathrm{TSL},\mu=\mathrm{const}}\right)\right]^{-1}, \qquad C_v := 8.\end{aligned}$$

The CFL number is chosen according to

$$\mathrm{CFL} \quad = \quad \min\left\{\mathrm{CFL}_{\mathrm{init}}\cdot f(n), \mathrm{CFL}_{\max}\right\}, \tag{55a}$$

$$f(n) \quad = \quad \begin{cases} 1, & n<10, \\ \alpha^{n-10}, & n\geq 10, \end{cases} \qquad \alpha>1. \tag{55b}$$

Typical values are $\mathrm{CFL}_{\mathrm{init}} = 3$ and $\mathrm{CFL}_{\max} = 1000$. For more details, we refer to Refs. [29, 40, 46].

4.5 Positivity of k and ω

To deal with the problem of positivity of k and ω, we simply introduced a damping of the updates. For example, Algorithm Eq. (48) gives for the variables k_i and ω_i, $i = 1,\ldots,N_{elem}$, the updates

$$k_i^{(j)} \quad = \quad k_i^{(0)} - \Delta k_i, \tag{56a}$$

$$\omega_i^{(j)} \quad = \quad \omega_i^{(0)} - \Delta\omega_i, \tag{56b}$$

where $(\Delta k_i, \Delta\omega_i)$ denotes the symbol for ith entry of the solution vector that one obtains by evaluating

$$\alpha_{j+1,j}\mathbf{P}_j^{-1}\mathbf{R}\left(\mathbf{W}^{(j-1)}\right).$$

Direct application of Eq. (56) often yields negative values, in particular for k. Most often this is observed for high-lift test cases, but for almost all test cases, negative values show up for k and ω at least during the starting phase of the iteration. Therefore, we replaced the update of Eq. (56) by an application of Algorithm 1.

Algorithm 1 represents a kind of damped Newton method introducing a further effect of regularization. Expressed in formulae, Algorithm 1 realizes the following condition:

$$s_{n,i}^{(k)} \quad = \quad \min_{n\in\mathbb{N}_0}\left\{\frac{1}{2^n}\right\} \quad \text{such that} \quad k_i^{(j)} > 0,$$

$$s_{n,i}^{(\omega)} \quad = \quad \min_{n\in\mathbb{N}_0}\left\{\frac{1}{2^n}\right\} \quad \text{such that} \quad \omega_i^{(j)} > 0.$$

Algorithm 1 Update for k-ω model

procedure LOOP OVER ALL MESH POINTS TO UPDATE k AND ω
  for $i = 1, \ldots, N_{elem}$ **do**
    $s_n = 1$
    for $n = 1, 2, \ldots$ **do**
      $k_i^{\text{new}} = k_i^{(0)} - s_n \Delta k_i$
      if $k_i^{\text{new}} > 0$ **then**
        $k_i^{(j)} = k_i^{\text{new}}$
        break
      else
        $s_{n+1} = \frac{s_n}{2}$
  for $i = 1, \ldots, N_{elem}$ **do**
    $s_n = 1$
    for $n = 1, 2, \ldots$ **do**
      $\omega_i^{\text{new}} = \omega_i^{(0)} - s_n \Delta \omega_i$
      if $\omega_i^{\text{new}} > 0$ **then**
        $\omega_i^{(j)} = \omega_i^{\text{new}}$
        break
      else
        $s_{n+1} = \frac{s_n}{2}$

A side effect of this approach is the fact that the updates may become arbitrarily small, yielding an overall convergence corruption. However, so far, none of the considered cases have been observed to have a stalled convergence. Compared with many other methods tried to ensure positivity of k and ω, Algorithm 1 was found to be superior with the present implementation. The simplicity of Algorithm 1 is another argument for its use. Nevertheless, Algorithm 1 cannot guarantee convergence. Hence, future work may require focus on other mechanisms to ensure positivity of k and ω without reformulating the k-ω model itself. On the other hand, the damping of updates

$$\begin{aligned} k_i^{\text{new}} &= k_i^{(0)} - s_n \Delta k_i \\ \omega_i^{\text{new}} &= \omega_i^{(0)} - s_n \Delta \omega_i \end{aligned}$$

is not a severe restriction and is justified in the following sense. Using in general (48) to compute the updates, we have

$$(\Delta k_i, \Delta \omega_i) = \alpha_{j+1,j} \left[\left((\Delta t)^{-1} \mathbf{M} + \alpha_{jj} \frac{d\mathbf{R}}{d\mathbf{W}} \left(\mathbf{W}^{(j-1)} \right) \right)^{-1,\text{app}} \mathbf{R}(\mathbf{W}^{(j-1)}) \right]_i .$$

A necessary criterion for convergence is $\|\mathbf{R}(\mathbf{W}^{(j-1)}\| \to 0$, that is in particular

$$\left(\mathbf{R}(\mathbf{W}^{(j-1)} \right)_i \to 0.$$

If the solution of the turbulence modeling equations converges, then at some iterate the updates Δk_i and $\Delta \omega_i$ are so small that additional damping is not necessary, and at the same time, positivity of k and ω is ensured. In this sense, Algorithm 1 is not a severe restriction. In the case that this algorithm is active all over the iteration, the solution of the turbulence modeling equations does not converge. Then, one either needs to question the application to the considered test case or the numerical method being applied. In particular,

if this is the case, it can be assumed that with the implemented solution method no positive function k and/or ω can be computed.

To illustrate the mode of operation of Algorithm 1, a plot of the number of k limitations and ω limitations is given in Figs. 1 and 2 for the Wilcox model of 1988 and for the SST model, respectively. To approximate a solution, we performed for each multigrid cycle on the mean flow equations 20 subiterations on the turbulence modeling equations. The number of multigrid cycles is plotted on the upper x-axis, and the total number of subiterations on the lower x-axis. From Fig. 1 (left), we can observe that not only the total number of limitations for k goes to zero, but also within each subiteration the number of limitations is significantly reduced. For the Wilcox model of 1988, no limitation of the ω variable is required, which is clearly seen in Fig. 1 (right).

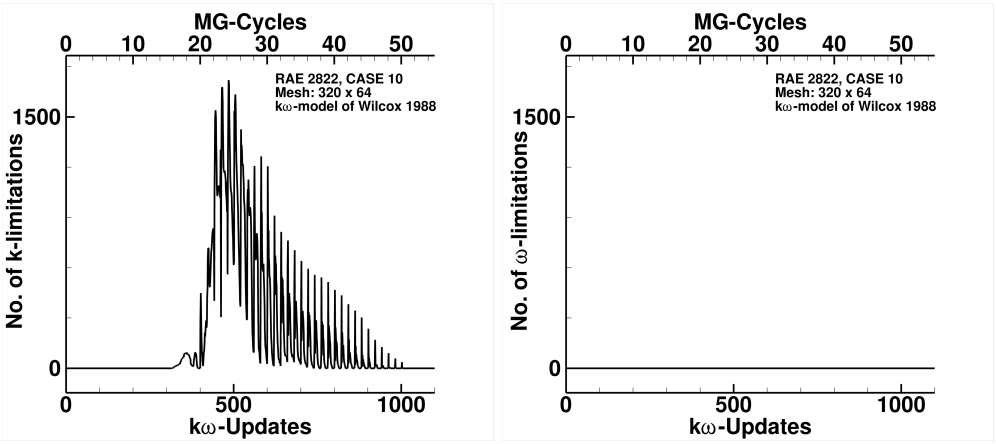


Figure 1: Number of limitations for the 1988 Wilcox k-ω model.

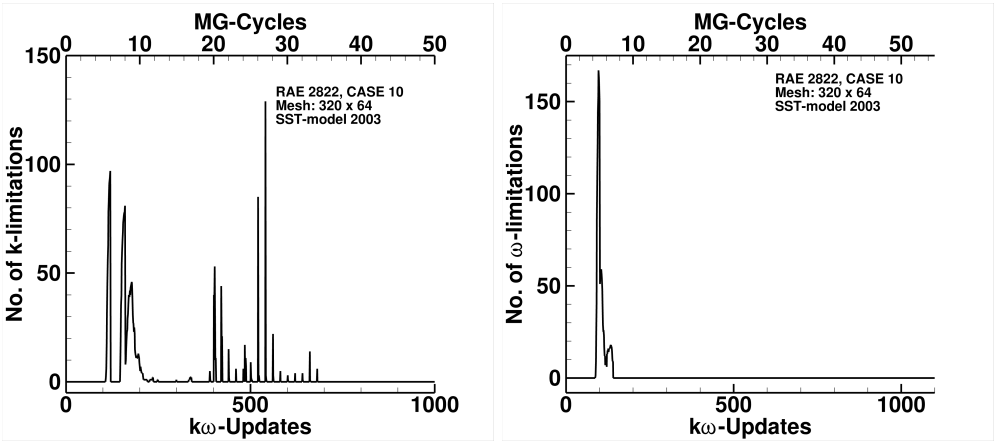


Figure 2: Number of limitations for the Menter SST model.

For the considered example, the number of limitations required for the SST-model is significantly smaller. Moreover, Fig. 2 (right) shows that for the SST-model, limitations for ω are also performed. As expected from considerations above, at some level in convergence,

the number of limitations for both the k and ω variables is 0. This means, that discrete, positive solutions for k and ω are obtained. This confirms numerically that the suggested limitation given by Algorithm 1 yields discrete solutions satisfying positivity for k and ω if convergence of the equations is observed. However, as long as one of the functions for k or ω is locally negative, the algorithm cannot converge.

5 Considerations for the k-ω Model

In this section, at least for some of the heuristics and arguments previously given, we are going to present both an analysis for the linear and nonlinear parts of the solution algorithm.

5.1 Analysis of Linear Part

In Section 4.1, it is stated that the derivatives of the the destruction terms of Eq. (54) are neglected for the construction of the preconditioner. To illustrate this necessity, we use the fact that the splitting method

$$x_{m+1} = Mx_m + Nb$$

converges if and only if $\rho(M) < 1$ (e.g., see Ref. [47]). For the Jacobi and Gauss-Seidel methods, the iteration matrix M is given by

$$M_{\text{Jac}} = D^{-1}(D - A) \qquad \text{and} \qquad M_{\text{GS}} = -(D + L)^{-1} R.$$

Using Arnoldi's method, as in the inner part of the GMRES method Refs. [29, 31], we can aproximate the spectrum of these operators for the preconditioner Eq. (53), including and neglecting Eq. (54). Figures 3 and 4 show the approximated eigenvalue distributions at the beginning and end of a nonlinear iteration, respectively. For each figure, Eq. (54) is included on the left and excluded on the right. In particular, at the beginning of the iteration, an inclusion of Eq. (54) changes the spectrum of the iteration matrices such that convergence of both the Gauss-Seidel and the Jacobi methods is not possible, whereas exclusion of Eq. (54) gives a converging method. For the final state, the situation improves, but as one can observe in Fig. 4 (left), there are still a few approximate eigenvalues outside the range of stability. Hence, for the overall nonlinear iteration process, it is necessary and beneficial to neglect Eq. (54). Moreover, when comparing the spectra of the Jacobi and the Gauss-Seidel methods, we observe a more equally distributed one for the latter one.

5.2 Analysis of Nonlinear Part

To understand if we can expect convergence of the solution for the turbulence modeling equations for the outer nonlinear iteration, at least in a small neighborhood of a solution $\left(\mathbf{W}^\dagger, \mathbf{W}_t^\dagger\right)$, for example,

$$\mathbf{R}_{\text{turb}}\left(\mathbf{W}_t^\dagger, \mathbf{W}^\dagger\right) = 0.$$

We assume a small perturbation, that is, $\|\mathbf{W}_t\| < \epsilon$ of $\mathbf{W}_t^\dagger$. Using a Taylor series expansion and neglecting terms of higher order, we approximate

$$\begin{aligned}
\frac{d\mathbf{W}(t)}{dt} &= \frac{d\left(\mathbf{W}^\dagger + \mathbf{W}(t)\right)}{dt} = -\mathbf{M}^{-1}\mathbf{R}\left(\mathbf{W}^\dagger + \mathbf{W}\right) \\
&\approx -\mathbf{M}^{-1}\left(\mathbf{R}\left(\mathbf{W}^\dagger\right) + \frac{d\mathbf{R}}{d\mathbf{W}}\left(\mathbf{W}^\dagger\right)\mathbf{W}\right) \\
&= -\mathbf{M}^{-1}\mathbf{A}\mathbf{W}(t), \qquad \mathbf{A} := \frac{d\mathbf{R}}{d\mathbf{W}}\left(\mathbf{W}^\dagger\right).
\end{aligned}$$

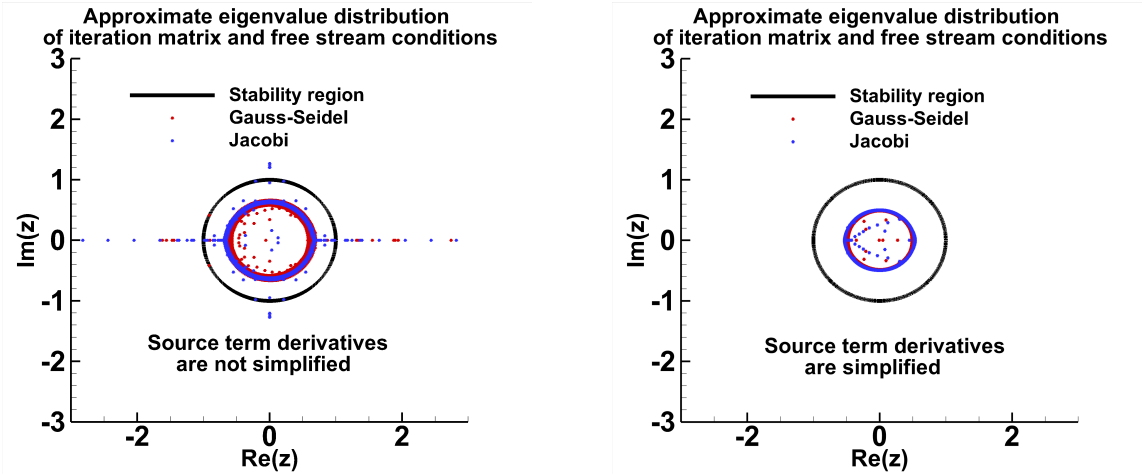


Figure 3: Eigenvalue distribution of Jacobi and Gauss-Seidel method in the beginning of nonlinear iteration.

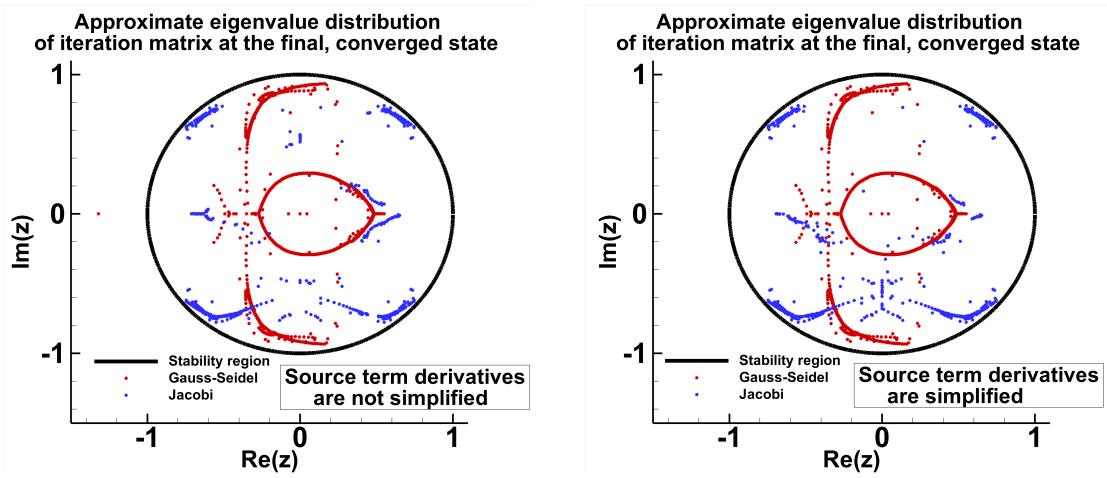


Figure 4: Eigenvalue distribution of Jacobi and Gauss-Seidel method for the final nonlinear iteration.

Then, the update of the multistage Runge-Kutta scheme Eq. (48) can be expressed by a polynomial expression,

$$\begin{aligned} \mathbf{W}^{n+1} &= q_s\left(\mathbf{Prec}^{-1,\mathrm{app}}\mathbf{A}\right)\mathbf{W}^n, \\ q_s(z) &= 1+\sum_{j=1}^{s}(-1)^j z^j \Pi_{i=s-j+1}^{s}\alpha_{i+1,i}. \end{aligned} \tag{57}$$

Thus, the multistage Runge-Kutta scheme (48) is stable, that is, it defines a contracting operator in a linear sense for all $z \in \mathbb{C}$, if $|q_s(z)| < 1$.

To compute approximations to the eigenvalues of $\mathbf{P}^{-1,\mathrm{app}}\mathbf{A}$, we again exploit the GMRES method and its connection to the Arnoldi process (se Refs. [29, 31]). As a test case, we consider Case 9 from the experiments of Cook, McDonald and Firmin [48] and a 320×64 C-

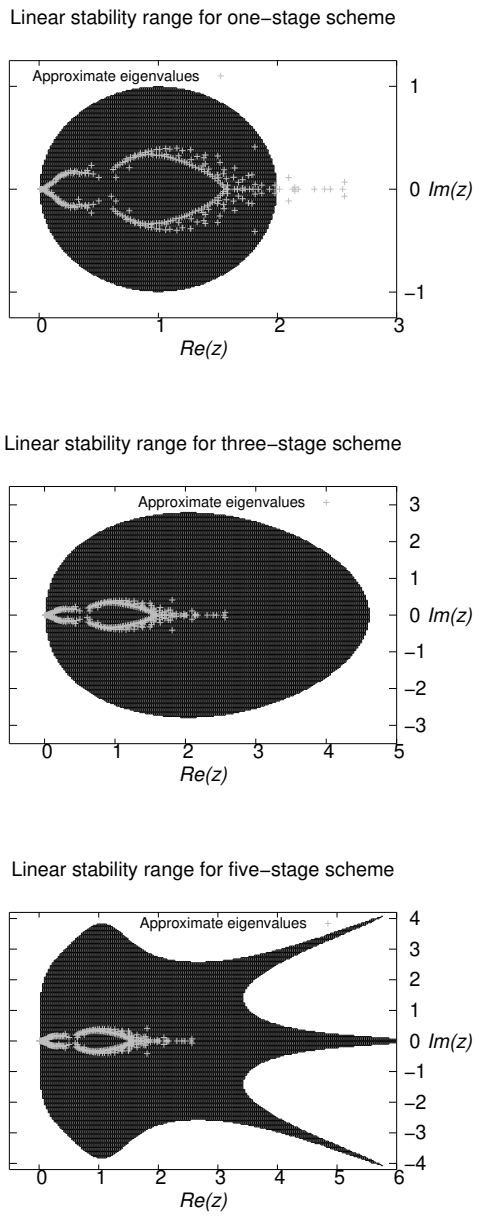


Figure 5: Range of stability for multistage schemes and approximate spectrum for two equation k-ω model of CASE 9.

type mesh. This mesh is a member of a family of meshes used later in Section 6 on Numerical Examples. To transform the spectrum, we restrict ourselves to the one-stage scheme with coefficients $\alpha_{21} = 1.0$, the three-stage scheme with coefficients $\alpha_{21} = 0.15$, $\alpha_{32} = 0.4$, $\alpha_{43} = 1$ and the five-stage scheme with coefficients $\alpha_{21} = 0.0695$, $\alpha_{32} = 0.1602$, $\alpha_{43} = 0.2898$, $\alpha_{54} = 0.5060$ and $\alpha_{65} = 1.0$ (see Ref. [25]). Figure 5 shows the range of stability of these multistage schemes together with the approximate spectrum. For this test case and for both the three-stage and the five-stage schemes, we can expect a stable method, whereas the one stage scheme is expected to diverge.

6 Numerical Examples

To demonstrate the applicability of the suggested algorithms, we consider some standard 2-D and challenging 3-D test cases, which have been computed frequently in the literature. The 2-D cases are transonic flows over an airfoil. The 3-D cases include relatively low-speed flow over a wing-body geometry and transonic flow over a wing-body configuration at a high angle of attack. For the 2-D cases, we consider both the Wilcox and SST k-ω models. The SST model is used in the 3-D computations. The density residual is evaluated by

$$\text{density residual}(n) \quad := \quad \sqrt{\sum_{j=1}^{N_{elem}} \frac{\left(\mathbf{R}_{j,\text{mean},\rho}\left(\mathbf{W}^{\mathrm{T_n}}\right)\right)^2}{(\text{vol}(\Omega_j))^2}} \Big/ \sqrt{\sum_{j=1}^{N_{elem}} \frac{\left(\mathbf{R}_{j,\text{mean},\rho}\left(\mathbf{W}_\infty\right)\right)^2}{(\text{vol}(\Omega_j))^2}},$$

for k-ω model we evaluate both corresponding residuals

$$k-\text{residual}(n) \quad := \quad \sqrt{\sum_{j=1}^{N_{elem}} \frac{\left(\mathbf{R}_{j,\text{turb},k}\left(k^{\mathrm{T_n}},\omega^{\mathrm{T_n}}\right)\right)^2}{(\text{vol}(\Omega_j))^2}} \Big/ \sqrt{\sum_{j=1}^{N_{elem}} \frac{\left(\mathbf{R}_{j,\text{turb},k}\left(k_\infty,\omega_\infty\right)\right)^2}{(\text{vol}(\Omega_j))^2}},$$

$$\omega-\text{residual}(n) \quad := \quad \sqrt{\sum_{j=1}^{N_{elem}} \frac{\left(\mathbf{R}_{j,\text{turb},\omega}\left(k^{\mathrm{T_n}},\omega^{\mathrm{T_n}}\right)\right)^2}{(\text{vol}(\Omega_j))^2}} \Big/ \sqrt{\sum_{j=1}^{N_{elem}} \frac{\left(\mathbf{R}_{j,\text{turb},\omega}\left(k_\infty,\omega_\infty\right)\right)^2}{(\text{vol}(\Omega_j))^2}}.$$

To satisfy near machine zero, a computation is truncated, and the result is accepted as soon as density residual$(n) < 10^{-14}$. The computations on unstructured grids were performed in parallel using MPI and either the $\mathrm{C^2A^2S^2E^2}$ or CARA cluster of DLR.

6.1 RAE 2822 Airfoil

The first examples considered correspond to the RAE 2822 airfoil. These examples have been chosen because they are frequently considered when attempting to validate turbulence models. There are two cases, Case 9 and Case 10, for transonic flow over the RAE 2822 airfoil [48]. The flow conditions for these cases are given in Table 1. In Case 9, there is a fairly strong shock wave occurring on the upper surface of the airfoil; where as in Case 10, there is a sufficiently strong shock on the upper surface to cause significant separation of the flow behind the shock.

We perform the computations on a sequence of C-type structured meshes described in Table 2. The meshes have a C-type topology. The finest mesh consists of 1280 cells around the airfoil (1024 cells on the airfoil) and 256 cells in normal direction. The normal mesh spacing at the surface of the finest mesh is approximately 3×10^{-6}, and the maximum surface cell aspect ratio is about 560.

Table 1: Flow Conditions for RAE 2822 airfoil

Cases	M_∞	AoA	Re
Case 9	0.73	2.79°	$6.5 \cdot 10^6$
Case 10	0.75	2.81°	$6.2 \cdot 10^6$

Table 2: Mesh data for RAE 2822 airfoil

	Coarse	Medium	Fine
Mesh size	320×64	640×128	1280×256
No. of quadrilaterals	20480	81920	327680
No. of cells on the airfoil	256	512	1024

Table 3: Case 9: Computed lift and drag coefficients with 1988 k-ω model

Model	Grid	C_L	C_D	$(C_D)_p$	$(C_D)_v$
$k\omega$-1988	320×64	0.792254	0.0195732	0.0120721	0.0075010
$k\omega$-1988	640×128	0.804623	0.0191719	0.0122047	0.0069673
$k\omega$-1988	1280×256	0.806221	0.0189818	0.0121783	0.0068035

Table 4: Case 9: Computed lift and drag coefficients with SST model

Model	Grid	C_L	C_D	$(C_D)_p$	$(C_D)_v$
SST	320×64	0.742041	0.0163938	0.0103164	0.0060774
SST	640×128	0.761013	0.0164595	0.0106297	0.0058298
SST	1280×256	0.766296	0.0164743	0.0107338	0.0057406

Table 5: Case 10: Computed lift and drag coefficients with 1988 k-ω model

Model	Grid	C_L	C_D	$(C_D)_p$	$(C_D)_v$
$k\omega$-1988	320×64	0.800997	0.0309963	0.0236907	0.0073056
$k\omega$-1988	640×128	0.814742	0.0300047	0.0232394	0.0068074
$k\omega$-1988	1280×256	0.815570	0.0298718	0.0232158	0.0066560

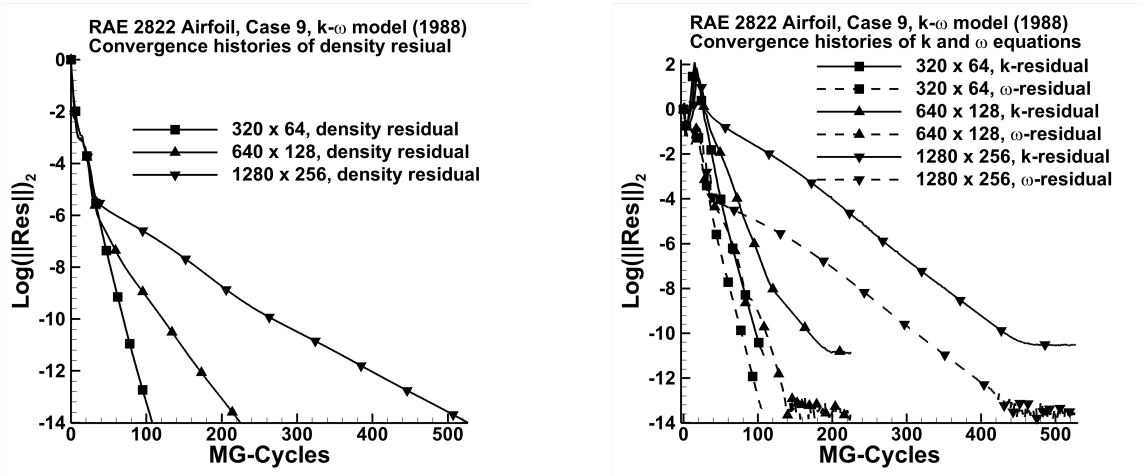
Table 6: Case 10: Computed lift and drag coefficients with SST model

Model	Grid	C_L	C_D	$(C_D)_p$	$(C_D)_v$
SST	320×64	0.722263	0.0241748	0.0187140	0.0054608
SST	640×128	0.742187	0.0246749	0.0194276	0.0052473
SST	1280×256	0.743644	0.0248062	0.0194821	0.0053241

The initial numerical calculations for the two RAE 2822 cases were performed with the complete solution algorithm of this paper, which includes agglomerative multigrid and GMRES, in the framework of an unstructured grid computer code. Tables 3–6 include the predicted lift and total drag coefficients, including the pressure and skin-friction contributions. In Figs. 6–9, convergence histories using the 1988 version of the Wilcox k-ω and the 2003 Menter SST turbulence models are presented. The final plots, which are included in Fig. 14, show the computed surface pressure distributions for the two RAE cases compared with the experimental data of Cook, McDonald and Firman [48].

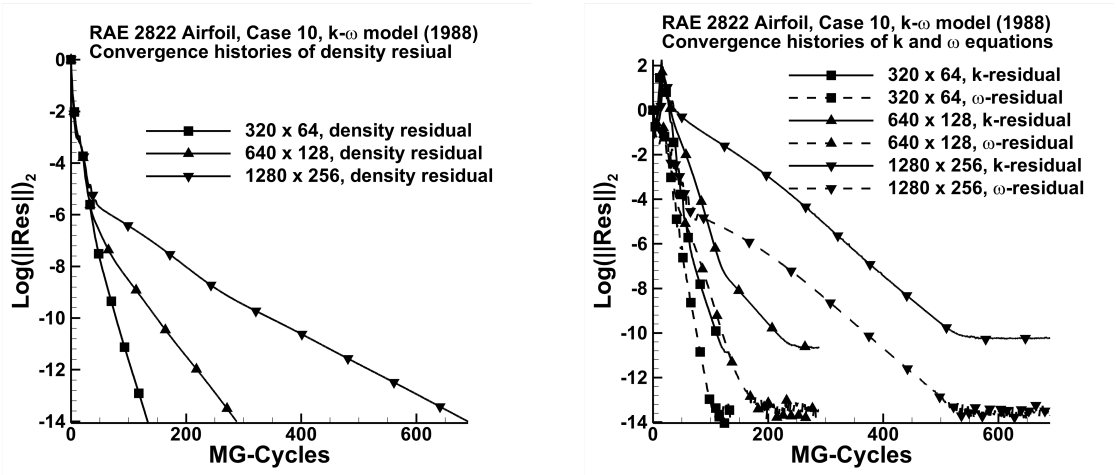
In the second set of results for Cases 9 and 10, the numerical computations were performed using the *core* of the solution algorithm described previously in the framework of a structured grid computer code. Results are for both the 2006 version of the Wilcox k-ω

and the 2003 Menter SST turbulence models. The convergence history results from these calculations are displayed in Figs. 10 to 13. As shown in these figures, the residuals for both the RANS equations and the transport equations of the turbulence models indicate a consistent, reliable linear convergence rate, and solutions are attained in only 125 to 150 multigrid cycles. Additional discussion of applications of the structured grid code are given in Ref. [13].



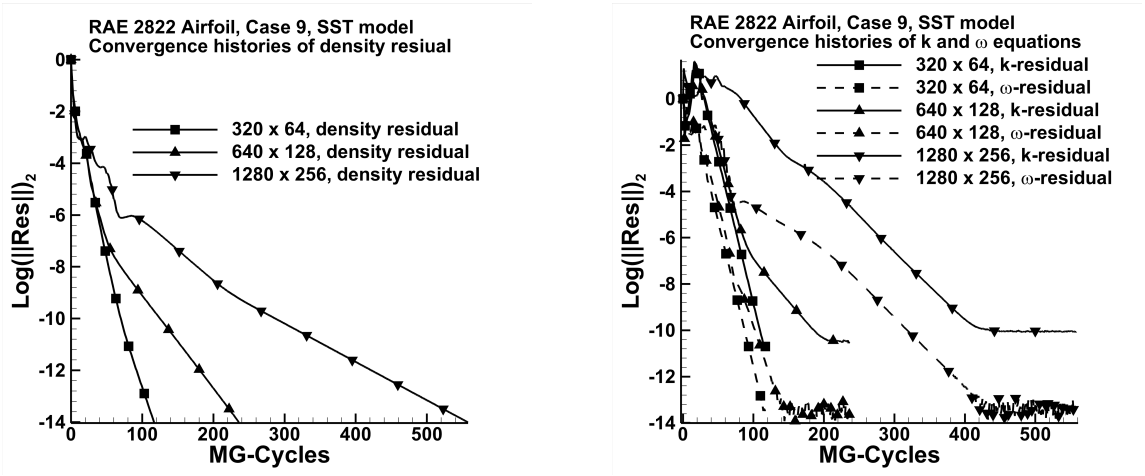
(a) (b)

Figure 6: Convergence histories for Case 9 and the k-ω model: (a) mean flow equations, (b) k-ω equations.



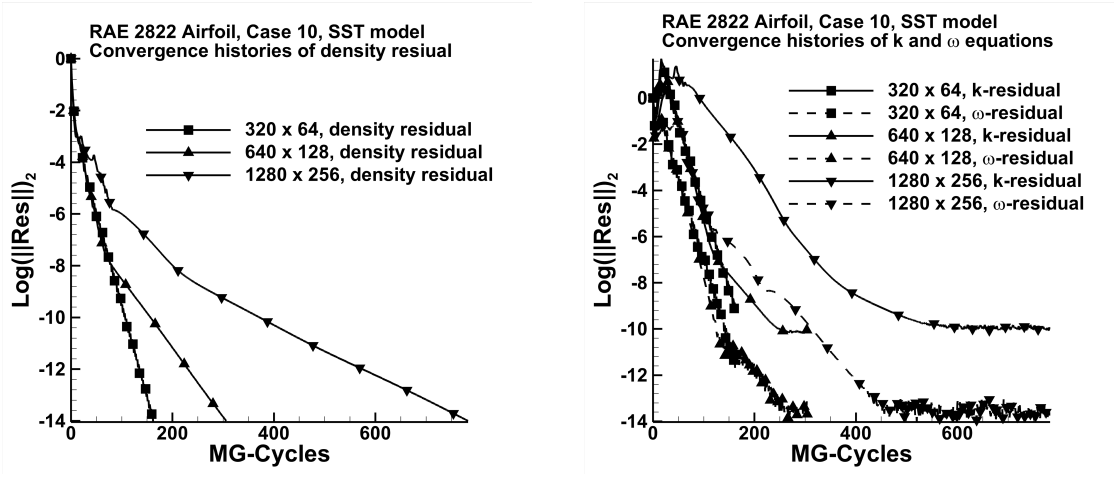
(a) (b)

Figure 7: Convergence histories for Case 10 and the k-ω model: (a) mean flow equations, (b) k-ω equations.



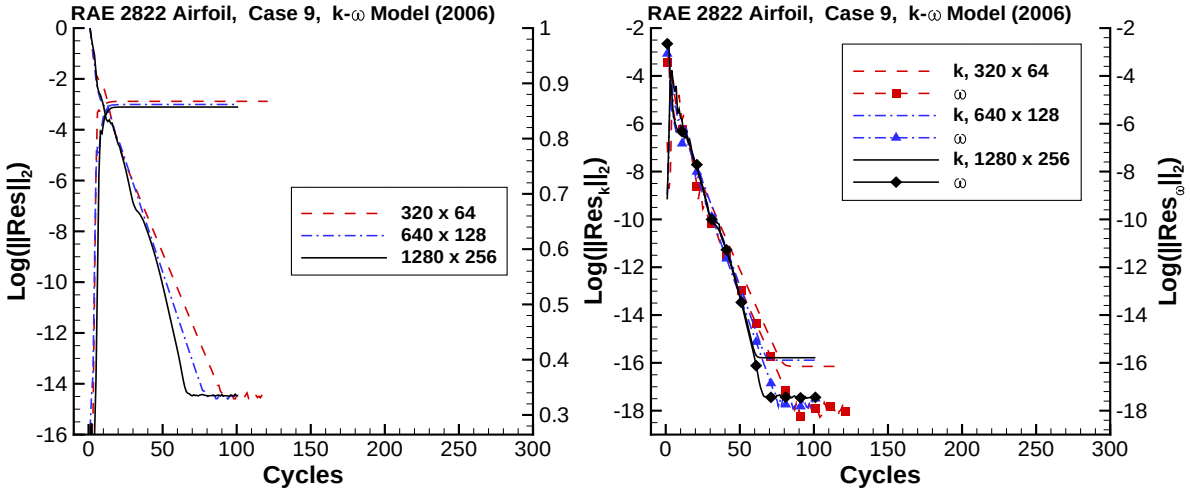
(a) (b)

Figure 8: Convergence histories for Case 9 and the SST model: (a) mean flow equations, (b) k-ω equations.



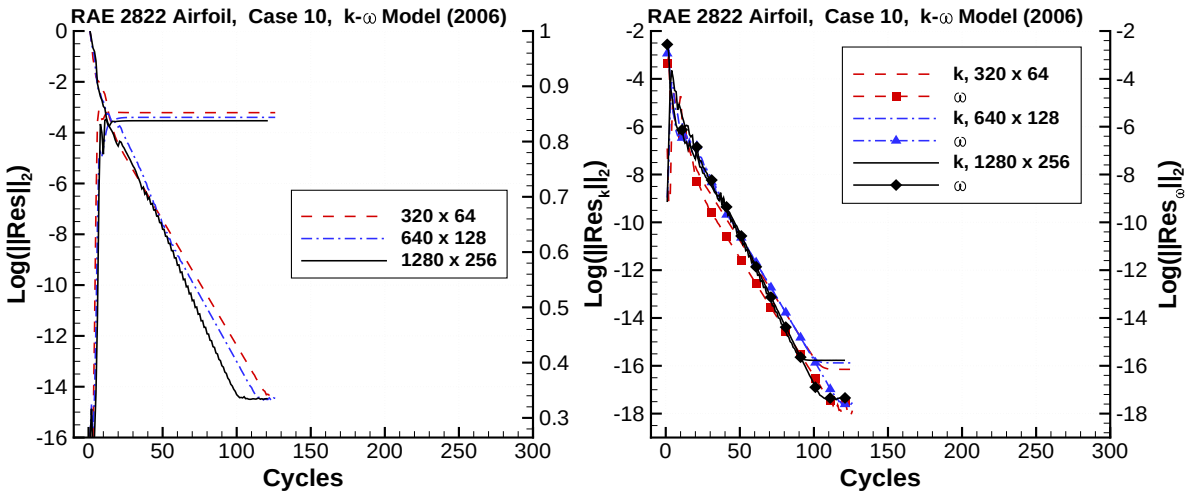
(a) (b)

Figure 9: Convergence histories for Case 10 and the SST model: (a) mean flow equations, (b) k-ω equations.



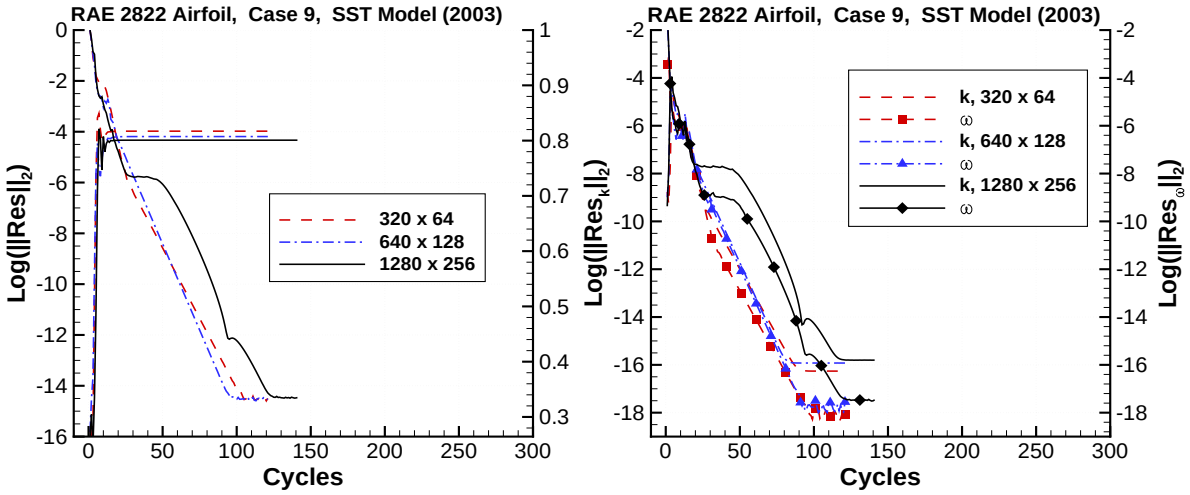
(a) (b)

Figure 10: Convergence histories of the solution computed for turbulent flow over RAE 2822 airfoil (Case 9). The effects of turbulence are represented with the Wilcox k-ω model (2006). A family of grids is considered: 320×64, 640×128, and 1280×256 cells; (a) mean flow equations, (b) k-ω equations.



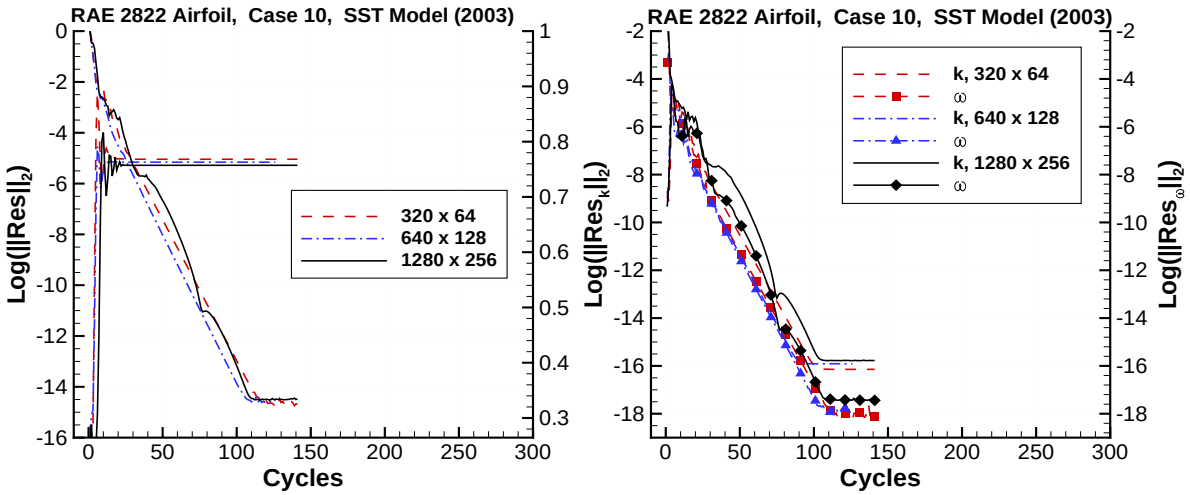
(a) (b)

Figure 11: Convergence histories of the solution computed for turbulent flow over RAE 2822 airfoil (Case 10). The effects of turbulence are represented with the Wilcox k-ω model (2006). A family of grids is considered: 320×64, 640×128, and 1280×256 cells; (a) mean flow equations, (b) k-ω equations.



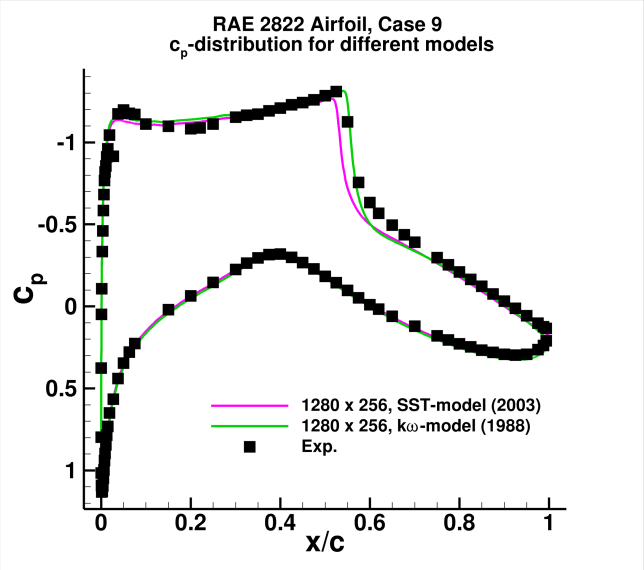
(a) (b)

Figure 12: Convergence histories of the solution computed for turbulent flow over RAE 2822 airfoil (Case 9). The effects of turbulence are represented with the Menter SST model (2003). A family of grids is considered: 320×64, 640×128, and 1280×256 cells; (a) mean flow equations, (b) k-ω equations.



(a) (t)

Figure 13: Convergence histories of the solution computed for turbulent flow over RAE 2822 airfoil (Case 10). The effects of turbulence are represented with the Menter SST model (2003). A family of grids is considered: 320×64, 640×128, and 1280×256 cells; (a) mean flow equations, (b) k-ω equations.



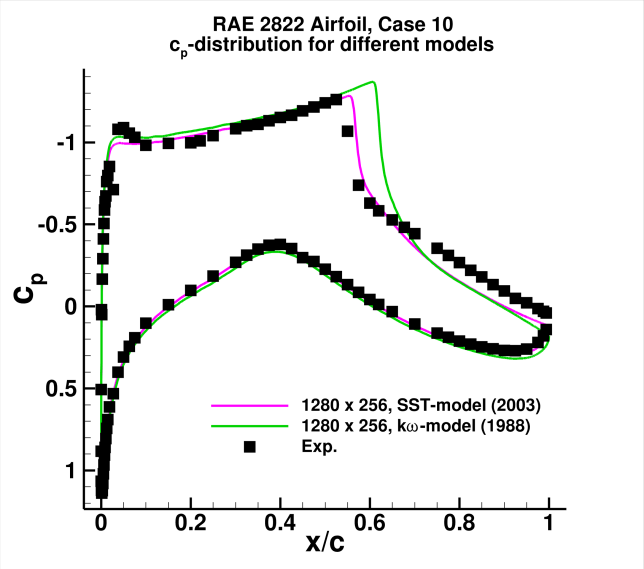


Figure 14: Surface pressure coefficient (C_p) distributions for Case 9 (top) and Case 10 (bottom) with comparisons to experimental data.

6.2 Transonic Turbulent Flow over a Common Research Model

The next test case is the first of the two 3-D flow computations in this paper. This first case is turbulent flow over the Common Research Model (CRM). It was considered at the fifth AIAA Drag Prediction Workshop. The meshes provided by the workshop are block-structured. A sequence of hybrid meshes was generated from the pure hexahedral meshes. Table 7 provides information about the grids used in the computations. The relevant physical conditions for this case are also given. The computed lift and drag coefficients for both mesh sequences are given in Table 8. There is a detailed description of the meshes given in Ref. [49].

Table 7: Mesh data for DPW5 CRM.

	Hybrid meshes		Hexahedral meshes	
Level	No. of Tetrahedra	No. of Prisms	No. of Hexahedrons	No. of points
L1	2555904	425984	638976	660177
L2	8626176	1437696	2156544	2204089
L3	20766720	3301376	5111808	5196193
L4	69728256	11261952	17252352	17441905

Table 8: lift and drag coefficient for CRM and SST model.

	Hexahedral meshes		Hybrid meshes	
	C_L	C_D	C_L	C_D
L1	0.48636	0.02783	0.50094	0.03012
L2	0.46915	0.02505	0.47431	0.02589
L3	0.46985	0.02470	0.47168	0.02494
L4	0.47426	0.02467	0.47633	0.02488

- Geometry: Wing-body configuration, fifth AIAA Drag Prediction Workshop
- Reynolds number: $Re = 5.0 \cdot 10^6$
- Inflow Mach number: $M_\infty = 0.85$
- Angle of attack 2.15°

It should be pointed out that it is difficult to find reports in the literature in which systematic mesh refinement studies together with two-equation turbulence models (e.g., the SST-model) are shown for 3-D flows, and the solution algorithm shows a consistent and reliable convergence behavior. In this paper, the objective is not to perform extensive mesh refinement studies and attempt to evaluate accuracy, but rather to demonstrate that the algorithm presented is effective in reducing residuals well below the truncation error of the numerical algorithm. Again, a necessary requirement to make a determination of accuracy with any turbulence model is to remove the integration error for a succession of meshes. Note, detailed information about accuracy for the 3-D flow examples considered herein can be found in Ref. [29] and [31].

Figure 15 shows the convergence histories for the sequence of meshes for the considered test case. Application of the algorithm presented in Section 4 made it possible to reach similar consistent convergence behavior for both the hexahedral and hybrid sequence of

meshes, and the stopping criterion was realized in only a few hundred multigrid cycles. In Fig. 16, the surface C_p distribution for the Common Research Model when using hybrid and hexahedral meshes is displayed.

6.3 NASA TRAP Wing

As a final example, to investigate the behavior of the proposed solution algorithm towards the incompressible limit, we examine the NASA Trap Wing considered at the first AIAA High-Lift prediction workshop [5]. For this test case, the importance of an adequate low-speed preconditioner for the governing flow equations must be emphasized, as there are low-speed regions as well as regions in which the flow is significantly accelerated. The low-speed preconditioner applied in this algorithm has a numerical dissipation that is a function of the local Mach number (see Ref. [50] for details).

For the numerical computations, we used two meshes. The meshes were generated using VGrid and are marked as UH6 in Table 2 of Ref. [5]. Characteristics of these meshes are given in Table 9. Figure 17 provides a sense of the gridpoint distribution on the wing-body surface and symmetry plane for Grid 2. In addition, this figure shows the eddy viscosity contours corresponding to this grid on the symmetry plane.

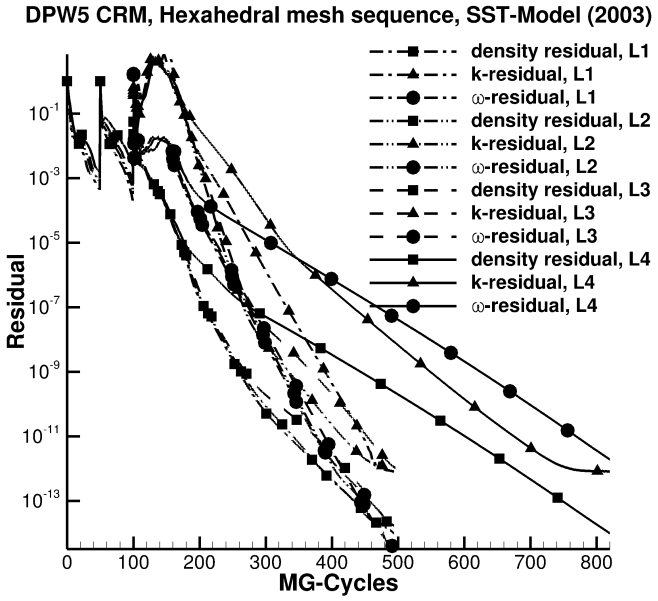
Table 9: NASA TRAP Wing: Mesh data.

Mesh	No. of points	No. of elements
Grid 1	3727008	10169092
Grid 2	11047965	38017477

The flow conditions for this 3-D example are given by

- Geometry: NASA TRAP Wing
- Reynolds number: $Re = 4.3 \cdot 10^6$
- Inflow Mach number: $M_\infty = 0.2$
- Angle of attack 28°

The high angle of attack (α) is an especially important flow condition, due to the numerical challenge that it represents in achieving a fully converged solution, as evident in the literature. In addition, high α aerodynamics can introduce numerous complexities into the flow field, including recirculation, vortical flow regions and various strong viscous types of interactions. Convergence histories for these computations with the SST model are displayed in Figs. 18. As in the results for the previous examples, these histories also exhibit near machine zero convergence.



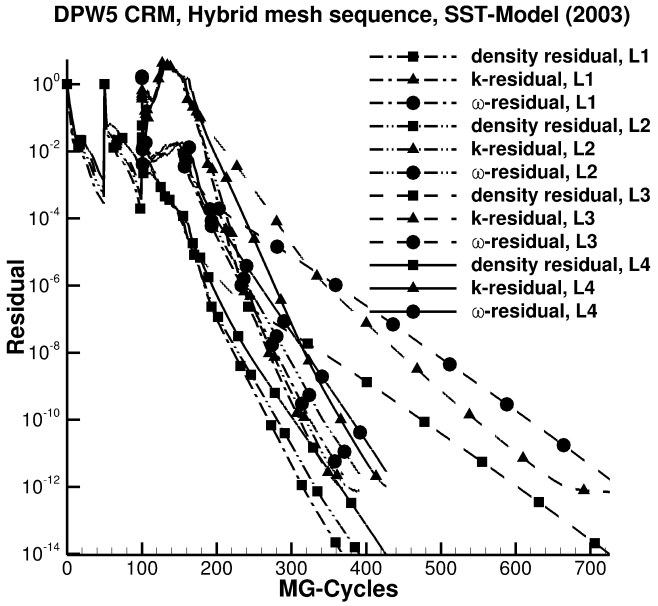


Figure 15: Convergence histories for DPW5 CRM using SST model: hex mesh (top) and hybrid mesh (bottom).

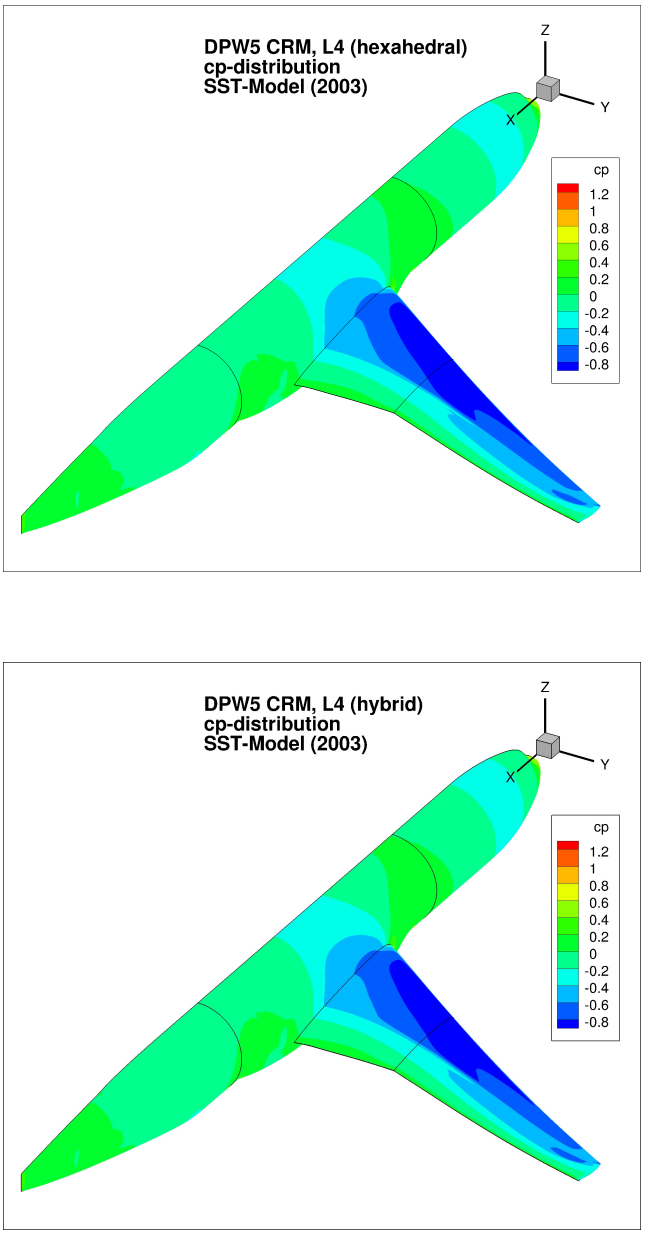
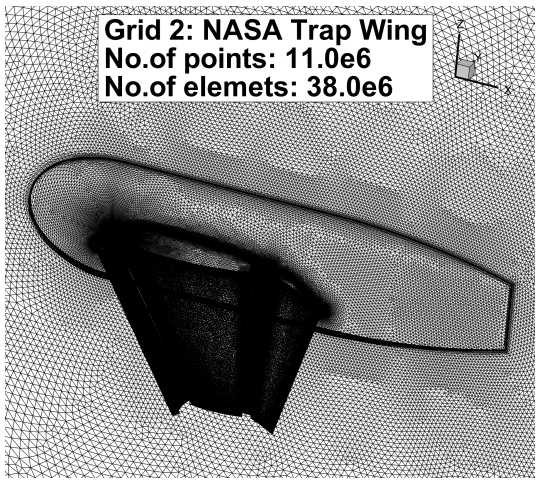
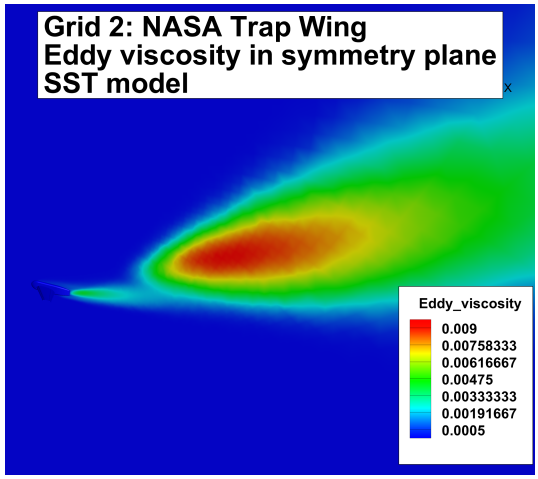


Figure 16: Surface C_p distribution for DPW5 CRM using SST model (computation on hexahedral mesh (top) and hybrid mesh (bottom)).



(a)



(b)

Figure 17: (a) Grid of NASA Trap Wing; (b) Computed eddy viscosity in the symmetry plane with the SST model.

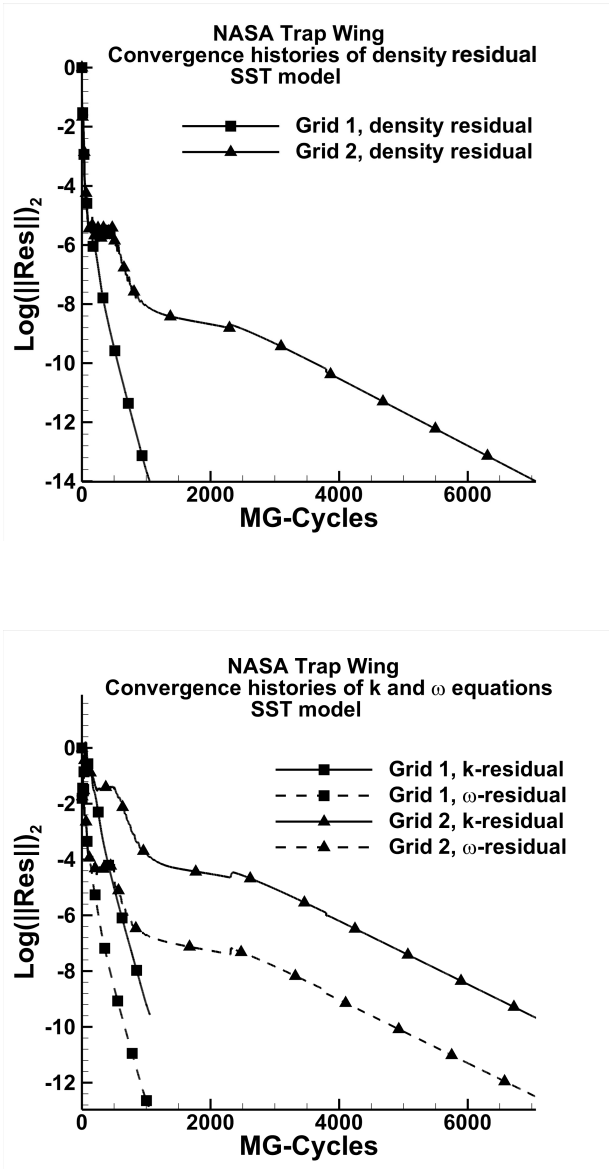


Figure 18: Convergence histories of mean flow and turbulent flow equations for the NASA Trap Wing using the SST model.

7 Concluding Remarks

In this paper an effective numerical algorithm for solving the RANS equations and the two transport equations of k-ω type turbulence models has been presented. The integral form of the RANS and k-ω equations along with the corresponding boundary conditions for a well defined boundary-value problem have been presented and described. Details of the k-ω type two-equation models, which include all elements exactly as implemented, have been given.

It is important to emphasize that the two-equation models considered in the paper have been implemented according to the original papers of Wilcox [32, 34] and Menter [37]. Consequently, there are no additional limiters on the dependent variables, which are often used in the application of the models and not discussed. Furthermore, there are no realizability conditions being imposed, as seen in some finite-element methods (e.g., [21, 22]), which use a logarithmic variable substitution. While one can certainly argue that changing the turbulence model for the purpose of what one can call *numerical compatibility* is appropriate, from the viewpoint of this paper, this should only be acceptable if the integrity of the model is not compromised.

To demonstrate the effectiveness and capability of the numerical solution algorithm, several examples of aerodynamic applications have been presented. Both 2-D and 3-D flows have been considered with varying degrees of difficulties due to the complexity of the turbulent flows being considered. In all cases, the convergence histories have been shown. The computed lift and drag coefficients have also been given.

An analysis of the numerical algorithm is addressed, which reflects some of the properties and stability of the solution algorithm. An approximate eigenspectrum of the preconditioner for the inner part of the GMRES method has been determined using Arnoldi's method. The impact of the destruction terms, which are part of the source terms of a two-equation turbulence model, using Jacobi and Gauss-Seidel methods has been presented. Neglecting these terms in the Jacobian of the implicit preconditioner is necessary, in general, and highly beneficial. The distribution of the eigenvalues of the amplification matrix has been obtained for two-stage, three-stage and five-stage Runge-Kutta schemes with the implicit preconditioner. The advantage of the present technique with respect to local mode analysis [51] is that the stability is determined for a specific problem being solved.

As a final remark, in Sections 3.3 and 3.4 the incompressible version of the considered models was introduced and finally implemeneted. When comparing with results from other codes using the original compressible version of the models for subsonic and transonic flow cases, no major differences between the results were observed. However, at the outset there seem to be no obvious obstacles to apply the methods proposed in this article also directly to the compressible version of the models. Future work will address how the methods proposed here can be directly applied to these compressible versions.

References

1. D. W. Levy, T. Zickuhr, J. Vassberg, S. Agrawal, R. A. Wahls, S. Pirzadeh, and M. J. Hemsch. Data Summary from the First AIAA Computational Fluid Dynamics Drag Prediction Workshop. *J. Aircr.*, 40(5):875–882, 2003.

2. D. W. Levy, K. R. Laflin, E. N. Tinoco, J. C. Vassburg, M. Mani, B. Rider, C. L. Rumsey, R. A. Wahls, J. H. Morrison, O. P. Brodersen, S. Crippa, D. J. Mavriplis, and M. Murayama. Summary Data from the Fifth Computational Fluid Dynamics Drag Prediction Workshop. *J. Aircr.*, 51(4):1194–1213, 2014.

3. E. N. Tinoco, P. Brodersen, S. Keye, K. R. Laflin, E. Feltrop, J. C. Vassberg, M. Mani, B. Rider, R. A. Wahls, J. H. Morrison, D. Hue, C. J. Roy, D. J. Mavriplis, and M. Murayama. Summary Data from the Sixth AIAA CFD Drag Prediction Workshop: CRM Cases. *J. Aircr.*, 55(4):1352–1379, 2018.

4. J. M. Derlaga and J. H. Morrison. Statistical Analysis of Solutions from the Sixth AIAA Computational Fluid Dynamics Drag Prediction Workshop. *J. Aircr.*, 55(4):1388–1400, July 2019.

5. C. L. Rumsey, J. P. Slotnick, M. Long, A. R. Stuever, and T. R. Wayman. Summary of the First AIAA CFD High-Lift Prediction Workshop. *J. Aircr.*, 48(6):2068–2079, 2011.

6. C. L. Rumsey and J. P. Slotnick. Overview and Summary of the Second AIAA High Lift Prediction Workshop. *AIAA Paper 2014-0747*, 2014.

7. C. L. Rumsey, J. P. Slotnick, and A. J. Sclafani. Overview and Summary of the Third AIAA High Lift Prediction Workshop. *J. Aircr.*, 56(2):621–644, 2019.

8. P. R. Spalart and S. R. Allmaras. A One-Equation Turbulence Model for Aerodynamic Flows. *AIAA Paper 92-0439*, January 1992.

9. S. R. Allmaras, F. T. Johnson, and P. R. Spalart. Modifications and Clarifications for the Implementation of the Spalart-Allmaras Turbulence Model. In *International Conference on Computational Fluid Dynamics 7, Hawaii*, number ICCFD7-1902 in Conference Proceeding Series, 2012.

10. M. L. Shur, M. K. Strelets, A. K. Travin, and P. Spalart. Turbulence Modeling in Rotating and Curved Channels: Assessing the Spalart-Shur Correction. *AIAA J.*, 38(5):784–792, May 2000.

11. P. R. Spalart. Strategies for turbulence modelling and simulations. *Int. J. Heat Fluid Flow*, 21(3):252–263, 2000.

12. F. R. Menter. Two-Equation Eddy-Viscosity Turbulence Models for Engineering Applications. *AIAA J.*, 32(8):1598–1605, August 1994.

13. R. C. Swanson. Solving Two-Equation Turbulence Models: *With a Perspective on Solving Transport Equations*. Technical Memorandum, NASA/TM–20210016636, NASA Langley Research Center, Hampton, VA, October 2021.

14. S. L. Krist, R. T. Biedron, and C. L. Rumsey. CFL3D User's Manual (version 5.0). Technical Memorandum, NASA/TM–1998-208444, NASA Langley Research Center, Hampton, Va, 1998.

15. T. Gerhold, J. Evans, and M. Galle. Technical Documentation of the DLR τ-Code. Technical report, 1997. LIDO-Berichtsjahr=1999.

16. Y. Mor-Yossef and Y. Levy. Unconditionally positive implicit procedure for two-equation turbulence models: Application to k-ω turbulence models. *J. Comput. Phys.*, 220(1):88–108, 2006.

17. Y. Mor-Yossef and Y. Levy. The unconditionally positive-convergent implicit time integration scheme for two-equation turbulence models: Revisited. *Comput. Fluids*, 38(10):1984–1994, 2009.

18. M. Wasserman, Y. Mor-Yossef, I. Yavneh, and J.B. Greenberg. A robust implicit multigrid method for rans equations with two-equation turbulence models. *J. Comput. Phys.*, 229(16):5820–5842, 2010.

19. M. Wasserman, Y. Mor-Yossef, and J.B. Greenberg. A positivity-preserving, implicit defect-correction multigrid method for turbulent combustion. *J. Comput. Phys.*, 316:303–337, 2016.

20. F. Ilinca, J.-F. Hétu, and D. Pelletier. A unified finite element algorithm for two-equation models of turbulence. *Comput. Fluids*, 27(3):291–310, 1998.

21. F. Bassi, A. Crivellini, S. Rebay, and M. Savini. Discontinuous Galerkin solution of the Reynolds-averaged Navier–Stokes and k-ω turbulence model equations. *Comput. Fluids*, 34(4–5):507–540, 2005.

22. R. Hartmann, J. Held, and T. Leicht. Adjoint-based error estimation and adaptive mesh refinement for the RANS and k-ω turbulence model equations. *J. Comput. Phys.*, 230(11):4268–4284, 2011.

23. V. Venkatakrishnan, S. R. Allmaras, D. S. Kamenetskii, and F. T. Johnson. Higher Order Schemes for the Comprssible Navier-Stokes Equations. *AIAA Paper 2003-3987*, 2003.

24. S. Langer and R. C. Swanson. On Boundary-Value Problems for RANS Equations and Two-Equation Turbulence Models. *J. Sci. Comput.*, 85(20), October 2020.

25. R. C. Swanson, E. Turkel, and C.-C. Rossow. Convergence acceleration of Runge-Kutta schemes for solving the Navier-Stokes equations. *J. Comput. Phys.*, 224(1):365–388, May 2007.

26. R. C. Swanson and C.-C. Rossow. An efficient solver for the RANS equation and a one-equation turbulence model. *Comput. Fluids*, 42:13–25, 2011.

27. H. Nishikawa, B. Diskin, J. L. Thomas, and D. H. Hammond. Recent advances in agglomerated multigrid. In *51st AIAA Aerospace Sciences Meeting*, AIAA Paper 2013-863, Texas, 2013.

28. Hiroaki Nishikawa, Boris Diskin, James L. Thomas, and D. H. Hammond. Recent advances in agglomerated multigrid. In *51st AIAA Aerospace Sciences Meeting*, AIAA Paper 2013-863, Texas, 2013.

29. Stefan Langer. Agglomeration multigrid methods with implicit Runge-Kutta smoothers applied to aerodynamic simulations on unstructured grids. *J. Comput. Phys.*, 277(0):72–100, 2014.

30. S. Langer, A. Schwöppe, and N. Kroll. The DLR Flow Solver TAU - Status and Recent Algorithmic Developments. In *Proc. 52nd Aerospace Sciences Meeting, January 2014*, number 2014-0080 in Conference Proceeding Series. AIAA, 2014.

31. S. Langer, A. Schwöppe, and N. Kroll. Investigation and Comparison of Implicit Smoothers Applied in Agglomeration Multigrid. *AIAA J.*, 53(8):2080–2096, 2015.

32. D. C. Wilcox. Reassessment of the Scale-Determining Equation for Advanced Turbulence Models. *AIAA J.*, 26(11):1299–1310, November 1988.

33. D. C. Wilcox. *Turbulence Modeling for CFD.* DCW Industries, Inc., La Cañada, CA, third edition, 2006.

34. David C. Wilcox. Formulation of the k-ω turbulence model revisited. *AIAA J.*, 46(11):2823–2838, November 2008.

35. F. R. Menter. Improved Two-Equation k-ω Turbulence Models for Aerodynamic Flows. Technical report, NASA, October 1992.

36. F. R. Menter. Zonal Two Equation k-ω Turbulence Models for Aerodynamic Flows. Technical report, NASA, 1993.

37. F. R. Menter, M. Kuntz, and R. Langtry. Ten Years of Industrial Experience with the SST Turbulence Model. In *Turbulence, Heat and Mass Transfer 4, pp. 625-632, eds: K. Hanjalic and Y. Nagano, M. J. Tummers, Begel House, Inc.*, 2003.

38. http://turbmodels.larc.nasa.gov. Turbulence modeling resource website, 2013.

39. R. C. Swanson and C.-C. Rossow. An Initial Investigation of the Effects of Turbulence Models on the Convergence of the RK/Implicit Scheme. Technical Memorandum, NASA/TM–2008-215342, NASA Langley Research Center, Hampton, VA, August 2008.

40. S. Langer. Preconditioned Newton methods to approximate solutions of the Reynolds averaged Navier-Stokes equations. Technical report, Institut für Aerodynamik und Strömungstechnik, June 2018.

41. S. Langer. An initial investigation of solving RANS equations in combination with two-equation turbulence models. Technical report, Institut für Aerodynamik und Strömungstechnik, September 2019.

42. J. Blazek. *Computational Fluid Dynamics: Principles and Applications.* Elsevier Science Ltd., Oxford, UK, 2005.

43. S. Langer and G. S. Martinez. Loosely coupled and coupled solution methods for the RANS equations and a one-equation turbulence model. *Comput. Fluids*, 232:1–17, October 2021.

44. U. Trottenberg, C. W. Oosterlee, and A. Schuller. *Multigrid.* Academic Press, Inc., Orlando, FL, USA, 2001.

45. Y. Saad. *Iterative Methods for Sparse Linear Systems.* International Thomson Publishing, Boston, 1996.

46. S. Langer. Hierarchy of Preconditioning Techniques for the solution of the Navier-Stokes equations discretized by 2nd order unstructured finite volume methods. In *ECCOMAS 2012 CONFERENCE*, Conference Proceeding Series, September 2012.

47. G. H. Golub and C. F. van Loan. *Matrix Computations.* The John Hopkins University Press, Baltimore, second edition, 1983.

48. P. H. Cook, M. A. McDonald, and M. C. P. Firmin. Aerofoil RAE 2822 pressure distributions and boundary layer and wake measurements. *AGARD-AR*, 138, 1979.

49. John C. Vassberg. A Unified Baseline Grid about the Common Research Model Wing-Body for the Fifth AIAA CFD Drag Prediction Workshop. In *Proc. 29th AIAA Applied Aerodynamics Conference, Honolulu, HI, June 2011*, number 2011-3508 in Conference Proceeding Series. AIAA, 2011.

50. S. Langer. Investigations of a compressible second order finite volume code towards the incompressible limit. *Comput. Fluids*, 149:119–137, 2017.

51. R. C. Swanson, E. Turkel, and S. Yaniv. Analysis of a RK/Implicit Smoother for Multigrid. In *Computational Fluid Dynamics 2010: Proceedings of the Sixth International Conference on Computational Fluid Dynamics, ICCFD6, St Petersburg, Russia, on July 12-16, 2010*, pages 409–417. Springer-Verlag Berlin Heidelberg, 2011.

Appendix A

Wilcox k-ω Models

A.1 1988 Wilcox model

The constants of the 1988 model are

$$\sigma_k = \frac{1}{2}, \qquad \sigma_\omega = \frac{1}{2}, \qquad \alpha = \frac{5}{9}, \qquad \beta = \frac{3}{40}, \qquad \beta^* = \frac{9}{100}. \tag{A1}$$

A.2 2006 Wilcox model

The closure coefficients for the 2006 k-ω model are as follows:

$$\alpha = \frac{13}{25}, \quad \beta = \beta_o f_\beta, \quad \beta^* = \frac{9}{100}, \quad \sigma_k = \frac{3}{5}, \quad \sigma_\omega = \frac{1}{2}, \quad \sigma_{do} = \frac{1}{8}, \quad \beta_o = 0.0708, \tag{A2}$$

$$Pr_t = \frac{8}{9}, \quad f_\beta = \frac{1+85\chi_\omega}{1+100\chi_\omega}, \quad \chi_\omega \equiv \left|\frac{\Omega_{ij}\Omega_{jk}\hat{S}_{ki}}{(\beta^*\omega)^3}\right|, \quad \hat{S}_{ki} = S_{ki} - \frac{1}{2}\frac{\partial \tilde{u}_m}{\partial x_m}\delta_{ki},$$

$$\sigma_d = \begin{cases} 0, & \frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j} \le 0 \\ \sigma_{do}, & \frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j} > 0 \end{cases}$$

where Pr_t is the turbulent Prandtl number. As indicated in the NASA Turbulence Modeling Resource [38],

$$\alpha = \frac{\beta_o}{\beta^*} - \frac{\sigma_\omega \kappa^2}{\sqrt{\beta^*}}, \tag{A3}$$

where κ is the von Kàrmàn constant (0.4).

Appendix B

Menter SST Turbulence Model

To realize smooth blending, the function $F_1 : [0, \infty) \rightarrow [0, 1]$ is modeled using the hyperbolic tangent,

$$F_1 \quad = \quad F_1(\Gamma_{F_1}) = \tanh\left(\Gamma_{F_1}^4\right), \tag{B1}$$