For a nondeforming mesh, Equation (A-1) can be written as

\[
\frac{1}{f} \frac{\partial Q}{\partial t} = R(Q) \tag{B-1}
\]

where

\[
R = - \left[ \frac{\partial (\hat{F} - \hat{F}_v)}{\partial \xi} + \frac{\partial (\hat{G} - \hat{G}_v)}{\partial \eta} + \frac{\partial (\hat{H} - \hat{H}_v)}{\partial \zeta} \right] \tag{B-2}
\]

The time term can be discretized with backward differencing:

\[
\frac{(1 + \phi)(Q^{n+1} - Q^n) - \phi(Q^n - Q^{n-1})}{\Delta t} = R(Q^{n+1}) \tag{B-3}
\]

where the superscripts indicate time level. When \( \phi = 0 \) the method is first-order temporally accurate; when \( \phi = 1/2 \) the method is second-order accurate. This equation is implicit because the right-hand side is a function of the unknown flow variables at time level \( n + 1 \).

The CFL3D code is advanced in time with an implicit approximate-factorization method. The implicit derivatives are written as spatially first-order accurate, which results in block-tridiagonal inversions for each sweep. However, for solutions that utilize FDS the block-tridiagonal inversions are usually further simplified with a diagonal algorithm (with a spectral radius scaling of the viscous terms).

Because of the method which the left-hand side is treated for computational efficiency in steady-state simulations (approximate factorization, first-order accuracy), second-order temporal accuracy is forfeited for unsteady computations. One method for recovering the desired accuracy is through the use of sub-iterations. Two different sub-iteration strategies have been implemented in CFL3D. The first method is termed “pseudo time sub-iteration (\( \tau \)-TS)”. The method is also often referred to as the “dual time stepping” method. The other method, termed “physical time sub-iteration (\( t \)-TS),” follows Pulliam.\(^{28}\)

For the \( \tau \)-TS method, a pseudo time term is added to the time-accurate Navier-Stokes equations.
This equation is then discretized and iterated in $m$, where $m$ is the sub-iteration counter.

\[
\frac{1}{J} \frac{\partial Q}{\partial \tau} + \frac{(1 + \phi)(Q^{n+1} - Q^n) - \phi(Q^n - Q^{n-1})}{J \Delta t} = R(Q^{n+1}) \quad (B-4)
\]

In Equation (B-5), $\phi$ and $\phi'$ govern the order of accuracy of the physical and pseudo time terms, respectively. In practice, the pseudo time term is treated as first order (i.e., $\phi' = 0$), but the general form is shown here for completeness. As $m \to \infty$, the pseudo time term vanishes if the sub-iterations converge and $Q^{m+1} \to Q^n$. If $R$ is linearized with

\[
R(Q^{m+1}) \equiv R(Q^m) + \frac{\partial R}{\partial Q} \Delta Q^m \quad (B-6)
\]

and the quantity $-(1 + \phi)Q^m / (J \Delta t)$ is added to both sides of Equation (B-5), then Equation (B-5) becomes

\[
\left[ \frac{(1 + \phi')}{J \Delta \tau} + \frac{1 + \phi}{J \Delta t} \right] I + \delta_\xi A + \delta_\eta B + \delta_\zeta C \Delta Q^m = \phi' \Delta Q^{m-1} + \frac{\phi \Delta Q^{n-1}}{J \Delta t} - \frac{(1 + \phi)(Q^{m} - Q^{n})}{J \Delta t} + R(Q^m) \quad (B-7)
\]

where

\[
\Delta Q^m = Q^{m+1} - Q^m \quad (B-8)
\]

\[
A = \frac{\partial (\hat{F} - \hat{F}_v)}{\partial Q} \quad (B-9)
\]

\[
B = \frac{\partial (\hat{G} - \hat{G}_v)}{\partial Q} \quad (B-10)
\]

\[
C = \frac{\partial (\hat{H} - \hat{H}_v)}{\partial Q} \quad (B-11)
\]

Equation (B-7) is approximately factored and written in primitive variable form; it is solved as a series of sweeps in each coordinate direction as
\[
\begin{align*}
\left[ \left( \frac{(1 + \phi')M}{J\Delta \tau} + \frac{(1 + \phi)M}{J\Delta t} \right) + \delta_z A^* \right] \Delta q' = & \quad \frac{\phi'M\Delta q^{m-1}}{J\Delta \tau} + \frac{\phi M\Delta q^{n-1}}{J\Delta t} - \\
& \quad \frac{(1 + \phi)M(q^m - q^n)}{J\Delta t} + R(q^m)
\end{align*}
\] (B-12)

\[
\left[ \left( \frac{(1 + \phi')M}{J\Delta \tau} + \frac{(1 + \phi)M}{J\Delta t} \right) + \delta_\eta B^* \right] \Delta q'' = \left( \frac{(1 + \phi')M}{J\Delta \tau} + \frac{(1 + \phi)M}{J\Delta t} \right) \Delta q'
\] (B-13)

\[
\left[ \left( \frac{(1 + \phi')M}{J\Delta \tau} + \frac{(1 + \phi)M}{J\Delta t} \right) + \delta_z C^* \right] \Delta q^m = \left( \frac{(1 + \phi')M}{J\Delta \tau} + \frac{(1 + \phi)M}{J\Delta t} \right) \Delta q''
\] (B-14)

\[
q^{m+1} = q^m + \Delta q^m
\] (B-15)

where the primitive variables are

\[
q = \begin{bmatrix} \rho \\ u \\ v \\ w \\ p \end{bmatrix}
\] (B-16)

\[
M = \frac{\partial Q}{\partial q}
\] (B-17)

\[
A^* = \frac{\partial (\hat{F} - \hat{F}_v)}{\partial q}
\] (B-18)

\[
B^* = \frac{\partial (\hat{G} - \hat{G}_v)}{\partial q}
\] (B-19)

\[
C^* = \frac{\partial (\hat{H} - \hat{H}_v)}{\partial q}
\] (B-20)

The quantity \(\Delta \tau\) is based on a constant CFL number set by the input parameter \texttt{cfl\_tau}
(See “LT5 - Time Step Parameters” on page 21). Multigrid is used to drive \(\Delta q^m\) to zero in
a reasonable number of sub-iterations.

In the t-TS method, Equation (B-3) is merely iterated in \(m\), where \(m\) is the sub-iteration counter:
The quantity \(- (1 + \phi) \frac{Q^n}{(J \Delta t)}\) is added to both sides, the residual is linearized, and the equation is approximately factored and written in primitive variable form as

\[
\frac{(1 + \phi)(Q^{m+1} - Q^n) - \phi(Q^n - Q^{n-1})}{J \Delta t} = R(Q^{m+1}) \tag{B-21}
\]

As \(m \to \infty\), \(q^{m+1} \to q^{n+1}\). When only one series of sweeps is performed, \(q^m = q^n\) and the standard time-accurate CFL3D scheme is recovered (i.e., no sub-iterations). Unlike the \(\tau\)-TS method, this sub-iteration procedure (Equation (B-22) through Equation (B-25)) utilizes only one time step: the physical time step \(\Delta t\) (constant).

Prior to the execution of Equation (B-25) in the code, the corrections are constrained in order to maintain the positivity of the thermodynamic variables \(\rho\) and \(p\). For example, the update to pressure is taken as

\[
p^{n+1} = p^n + \Delta p\left[1 + \phi \left(\alpha_c + \frac{\Delta p}{p^n}\right)\right]^{-1} \tag{B-26}
\]

whenever \(\Delta p/p^n \leq \alpha_c\). Currently, \(\alpha_c = -0.2\) and \(\phi_c = 2.0\).

In the limit of \(\Delta p/p^n \to -\infty\), \(p^{n+1} \to p^n/2\). This modification improves the robustness of the method by allowing it to proceed through local transients encountered during the convergence process which would otherwise terminate the calculation.

When running steady-state computations (\(dt < 0\)), the time step advanced locally in each cell is related to the input CFL number by

\[
\Delta t = \frac{\text{CFL}}{|\nabla \xi| t_1 + |\nabla \eta| t_2 + |\nabla \zeta| t_3} \tag{B-27}
\]
where

\begin{align*}
t_1 &= |\overline{U}| + a + 2|\nabla \xi| (\mu + \mu_T) \max \left( 4 \frac{\gamma}{3 \Pr} \frac{M}{Re_{LR}} \right) \frac{1}{\rho} \\
t_2 &= |\overline{V}| + a + 2|\nabla \eta| (\mu + \mu_T) \max \left( 4 \frac{\gamma}{3 \Pr} \frac{M}{Re_{LR}} \right) \frac{1}{\rho} \\
t_3 &= |\overline{W}| + a + 2|\nabla \zeta| (\mu + \mu_T) \max \left( 4 \frac{\gamma}{3 \Pr} \frac{M}{Re_{LR}} \right) \frac{1}{\rho}
\end{align*}

(B-28)

where \( \overline{U} = U/|\nabla \xi| \), \( \overline{V} = V/|\nabla \eta| \), \( \overline{W} = W/|\nabla \zeta| \) and \( U \), \( V \), and \( W \) are defined in Equation (A-7) in Appendix A. The viscous scaling terms (the last term in each equation of Equation (B-28)) are only used when the solution includes viscous terms. They arise from a spectral radius scaling (see Coakley\textsuperscript{14}).