VARIABLE-STEP UNCONDITIONALLY STABLE SECOND-ORDER APPROXIMATIONS

JOHN BURKARDT AND CATALIN TRENCHEA*

From (implicit) Backward-Euler to Variable-Step Second-Order Conservative Unconditionally-Stable, by adding two lines of code

1. One line of code to change a Backward Euler code into to a second-order, unconditionally stable, conservative method. For the numerical approximation of a general evolution equation:

$$y'(t) = f(t, y(t)), \tag{1.1}$$

on the mesh points $\{t_n\}_{n\geq 0}$, and with the timestep τ_n, such that:

$$t_{n+1} = t_n + \tau_n, \quad t_{n+1/2} = t_n + \tfrac{1}{2}\tau_n,$$

we recall the classical midpoint quadrature rule:

$$\frac{y_{n+1} - y_n}{\tau_n} = f(t_{n+1/2}, y_{n+1/2}), \tag{1.2}$$

where $y_n \approx y(t_n)$. The method (1.2) is ubiquitously presented and used [3, 7, 8, 13, 14, 16, 21, 22, 24–26] in the apparently different form:

$$\frac{y_{n+1} - y_n}{\tau_n} = f\Big(t_{n+1/2}, \frac{y_{n+1} + y_n}{2}\Big). \tag{1.3}$$

The reason for the wide use of (1.3) instead of (1.2) (see e.g. [22, page 133]) is due to the natural question: '*but which value should we take for $y_{n+1/2}$?*'. The method (1.3) is an implicit second-order A-stable time-stepping method, and is the preferred method for solving evolutive conservative systems of partial differential equations (PDEs), along with the second order backward differentiation formula (BDF2) for dissipative PDEs.

From an algorithmic viewpoint, increasing the numerical accuracy of a complex legacy code, based on the first-order backward Euler (BE) method, to a second-order A-stable method, can be a difficult task. One straightforward solution would be to apply non-intrusive minimal modifications to the algorithm, i.e., by adding a few lines of code and post-processing the original BE solution into a 'filtered' higher-order solution. This is currently done in geophysical fluid dynamics, to improve the quality of the solution to the leap-frog method, by filtering it with Robert-Asselin or Robert-Asselin-Williams filters [1, 20, 28, 31–33, 35–37]. Recently, the BE solution was filtered into the solution to a second-order linear multistep method (LMM), similar to a BDF2 solution (see e.g., [19]), with a reduced discrete curvature and numerical dissipation. Most LMMs [11, 22], when considered with variable steps, do not preserve the zero-stability or unconditional A-stability properties of the constant step versions. For example, the variable step version of the trapezoidal method (Crank-Nicolson) is unstable [22], [34, pp. 181-182]; similarly, BDF2 loses zero-stability and A-stability with variable stepsize.

An alternative non-intrusive modification to the BE method, with the goal of defining a family of second-order, variable step, unconditionally stable one-step methods, relies on the successful resolution of the above question regarding (1.2). This alternative is based on the fact that both the midpoint (1.2) and the trapezoidal methods can be viewed as a sequence of backward-Euler then forward-Euler methods, respectively a forward-Euler then a backward-Euler method, where the first computation is performed at the time $t_{n+1/2}$, see e.g., [21, page 223] and [12, page 57].

Consequently, consider implementing the midpoint rule (1.2) by solving a backward-Euler step at the half-integer time step $t_{n+1/2}$, followed by a forward-Euler step to t_{n+1}:

$$\frac{y_{n+1/2} - y_n}{\tau_n/2} = f(t_{n+1/2}, y_{n+1/2}), \tag{BE}$$

$$\frac{y_{n+1} - y_{n+1/2}}{\tau_n/2} = f(t_{n+1/2}, y_{n+1/2}). \tag{FE}$$

*Department of Mathematics, University of Pittsburgh, Pittsburgh, PA 15260, USA. Email: trenchea@pitt.edu.

We point out that solving the equations (BE)-(FE) is equivalent to, and reduces to only solving (BE), and then applying a time-filter, as the (FE) step is equivalent to a linear extrapolation. Hence we evaluate $y_{n+1} = 2y_{n+1/2} - y_n$, and the equation (BE)-(FE) can be thought of as a single process designated as (BEFE):

$$\begin{cases} \dfrac{y_{n+1/2} - y_n}{\tau_n/2} = f(t_{n+1/2}, y_{n+1/2}), \\ y_{n+1} = 2y_{n+1/2} - y_n. \end{cases} \tag{BEFE}$$

Notice that the second step can also be written as:

$$y_{n+1/2} = \frac{y_{n+1} + y_n}{2},$$

and therefore both (1.2) *and* (1.3) *yield exactly the same numerical approximations*, i.e., (1.2) is a second-order accurate, unconditionally A-stable method. An important characteristic of (1.2) is the fact that it is a one-leg two-step method, which, as in the case of the trapezoidal rule, makes it easy to view it as a variable-step method, without losing the stability property. There are several options as to how to adapt the time-step τ_n (see e.g., [17, 18]), namely how to estimate the local truncation error.

This implementation (BEFE) of the midpoint method is consequential from the viewpoint of its potential applications for time-stepping methods of complex partial differential equations. The first advantage is the ease of non-intrusive implementation: it takes one line of code to transform a first-order dissipative method to a second-order conservative stable method. (We recall that Dahlquist's barrier limits the accuracy of A-stable linear multistep methods to second-order.) A second advantage is that the constant in the local truncation error of (1.3), when seen in the implementation (BEFE) is $\frac{1}{24}$, instead of the usual $-\frac{1}{12}$. Thirdly, since midpoint is a one step method, time-adaptivity can be implemented with non-intrusive minimal algorithmic changes.

For coupled complex systems, like magnetohydrodynamics, ocean-atmosphere, groundwater-surface water, or fluid-structure interactions, the current trend is to employ partitioning methods of implicit-explicit type, which solve each equation separately by a legacy code, and transfer information between the subdomains and algorithms. This breaking of the monolithic approach ubiquitously comes at the cost of stability. Most existing partitioned stable methods are only first-order accurate in time. The (BEFE) implementation opens the path of extending the current partitioning first-order stable methods to second-order accurate variable-step unconditionally stable methods, by manipulating the computed solution at $t_{n+1/2}$ in a stable manner. Recently, this approach has been applied to problems in fluid-structure interaction [2], magnetohydrodynamics and ocean-atmosphere modeling. Note also that the computed solution at $t_{n+1/2}$ allows further manipulation, such as modular spatial filtering, in order to improve the qualitative properties of the numerical simulations [29, 30].

2. Generalization to a θ-like method. We remark also that (BEFE) is a particular instance of the one-leg 'θ-like' method:

$$\frac{y_{n+1} - y_n}{\tau_n} = f(t_{n+\theta}, y_{n+\theta_n}), \tag{2.1}$$

implemented as:

$$\begin{cases} \dfrac{y_{n+\theta_n} - y_n}{\theta_n \tau_n} = f(t_{n+\theta_n}, y_{n+\theta_n}), \\ \dfrac{y_{n+1} - y_{n+\theta_n}}{(1-\theta_n)\tau_n} = f(t_{n+\theta_n}, y_{n+\theta_n}). \end{cases} \tag{2.2}$$

which can be rewritten as:

$$\begin{cases} \dfrac{y_{n+\theta_n} - y_n}{\theta_n \tau_n} = f(t_{n+\theta_n}, y_{n+\theta_n}), \\ y_{n+1} = \frac{1}{\theta_n} y_{n+\theta} - \left(\frac{1}{\theta_n} - 1\right) y_n. \end{cases} \tag{2.3}$$

Notice that (2.1) is not the classical linear multistep θ method [15, page 182], but Cauchy's one-leg version (see e.g. [4, pp. 40], also [6, 9, 10]):

$$\frac{y_{n+1}-y_n}{\tau_n} = f\big(t_{n+\theta}, \theta y_{n+1} + (1-\theta) y_n\big),$$

since, as above, we have from the second part of (2.3) that $y_{n+\theta} = \theta y_{n+1} + (1-\theta) y_n$.

We recall that in the following we mean stability in the sense of G-stability (Dahlquist 1975, see e.g., [8] or [23, p.308]), which is equivalent to A-stability for constant step linear multistep methods.

PROPOSITION 2.1. *The midpoint method* (BE)-(FE)*, and the θ-method* (2.2) *for $\theta_n \geq \frac{1}{2}$, are unconditionally-stable, and the following equality holds:*

$$\frac{1}{2}\|y_{n+1}\|^2 - \frac{1}{2}\|y_n\|^2 + \frac{2\theta_n - 1}{2}\|y_{n+1} - y_n\|^2 = \tau_n \langle f(t_{n+\theta_n}, y_{n+\theta_n}), y_{n+\theta_n}\rangle.$$

Proof. We prove the result only for (2.2), since the midpoint method is obtained by taking $\theta_n = 1/2$. Multiplying both equations in (2.2) by $\theta_n \tau_n y_{n+\theta_n}$ and $(1-\theta_n)\tau_n y_{n+\theta_n}$ respectively, and applying the polarization identity we obtain:

$$\begin{aligned}
&\frac{1}{2}\|y_{n+\theta_n}\|^2 - \frac{1}{2}\|y_n\|^2 + \frac{1}{2}\|y_{n+\theta_n} - y_n\|^2 = \theta_n \tau_n f(t_{n+\theta_n}, y_{n+\theta_n}) y_{n+\theta_n},\\
&\frac{1}{2}\|y_{n+1}\|^2 - \frac{1}{2}\|y_{n+\theta_n}\|^2 - \frac{1}{2}\|y_{n+1} - y_{n+\theta_n}\|^2 = (1-\theta_n)\tau_n f(t_{n+\theta_n}, y_{n+\theta_n}) y_{n+\theta_n}.
\end{aligned}$$

Summation and the use of (2.2) completes the argument. □

3. Time-step adaptivity. We begin this section by a small observation: the local truncation error of the midpoint method (BEFE) is:

$$T_{n+1} = \frac{1}{24}\tau_n^3 y'''(t_{n+1/2}) + \mathcal{O}(\tau_n^5). \tag{3.1}$$

The same formula holds for the '$\theta-$like' method (2.1), provided $\theta_n = \frac{1}{2} + \frac{1}{2}\tau_n^2$.

Therefore, we can adaptively adjust the time step τ_n by enforcing an estimate of the local truncation error (1.2), denoted $\widehat{T}_{n+1}$, to equal a tolerance, i.e., such that the $\|\widehat{T}_{n+1}\| \approx \texttt{tol}$ (see e.g. [18]). The time-step τ_n^{new} which imposes that $\hat{T}_{n+1}$ is sufficiently small is given by:

$$\tau_n^{new} = \tau_n \left| \frac{\texttt{tol}}{\|\widehat{T}_{n+1}\|} \right|^{\frac{1}{3}}. \tag{3.2}$$

There are numerous ways in which the time-step adaptivity can be implemented (see e.g. [18]), out of which we present three methods. The first choice is based on the estimation of the LTE using Taylor expansions. The other two options estimate the local truncation error by the difference between the numerical midpoint solution and a second-order, and respectively a third-order approximation, given by formulae similar to the explicit Adams-Bashforth 2 (AB2) and Adams-Bashforth 3 (AB3) methods. These two methods are related to the classical AB2 and AB3 (see e.g., [22, p. 398]), the difference being that they use the function values evaluated at half-times

$f_{n+1/2}, f_{n-1/2}, f_{n-3/2}, f_{n-5/2}$.

Result: *Adaptive midpoint rule*

initialization: set `tol`, compute y_1 and τ_0 with a one step second-order accurate method, such that τ_0 is in the convergence range (see e.g., [5, page 367]);

compute y_2 and τ_1 with a second order accurate method, $t_2 = t_1 + \tau_1$;

$t^{\text{new}} = t_2$, $\tau^{\text{new}} = \tau_1$;

for $n \geq 2$ (i.e., t^{new}, τ^{new}, y_n, y_{n-1}, y_{n-2} are given);

while $t^{\text{new}} \leq T$ **do**

  $\tau_n \leftarrow \tau^{\text{new}}$;

  evaluate y_{n+1} with the midpoint rule (1.2);

  evaluate $\widehat{T}_{n+1}$ with (LTE-Taylor), (LTE-AB2) or (LTE-AB3);

  $\tau^{\text{new}} \leftarrow \tau_n \left| \mathtt{tol}/\|\widehat{T}_{n+1}\| \right|^{\frac{1}{3}}$;

  if $\|\widehat{T}_{n+1}\| \leq \mathtt{tol}$ **then**

    $t_{n+1} \leftarrow t_n + \tau^{\text{new}}$, $t^{\text{new}} \leftarrow t_{n+1}$, $n+1 \leftarrow n$

  end

end

3.1. Estimation of the local truncation error using Taylor expansions. In order to estimate the numerical value of $\widehat{T}_{n+1}$, we need to evaluate $y'''(t_n)$. We proceed by using Taylor expansions:

$$
\begin{aligned}
y'(t_{n+1/2}) &= y'(t_n) + \frac{\tau_n}{2} y''(t_n) + \frac{\tau_n^2}{8} y'''(t_n) + \mathscr{O}(\tau_n^3),\\
y'(t_{n-1/2}) &= y'(t_n) - \frac{\tau_{n-1}}{2} y''(t_n) + \frac{\tau_{n-1}^2}{8} y'''(t_n) + \mathscr{O}(\tau_{n-1}^3),\\
y'(t_{n-3/2}) &= y'(t_n) - \frac{2\tau_{n-1}+\tau_{n-2}}{2} y''(t_n) + \frac{(2\tau_{n-1}+\tau_{n-2})^2}{8} y'''(t_n) + \mathscr{O}(\tau_{n-1}^3 + \tau_{n-2}^3),
\end{aligned}
$$

which, eliminating $y'(t_n)$ and $y''(t_n)$, gives:

$$
\frac{y'(t_{n+1/2}) - y'(t_{n-1/2})}{\tau_n + \tau_{n-1}} - \frac{y'(t_{n-1/2}) - y'(t_{n-3/2})}{\tau_{n-1} + \tau_{n-2}} = \frac{1}{8}(\tau_n + 2\tau_{n-1} + \tau_{n-2}) y'''(t_n) + \mathscr{O}(\tau_n^2 + \tau_{n-1}^2 + \tau_{n-2}^2).
$$

Using the numerical method (1.2), the LTE (3.1) can finally be estimated in terms of the computed solutions:

$$
\begin{aligned}
\widehat{T}_{n+1} &= \frac{\tau_n^3}{3(\tau_n + 2\tau_{n-1} + \tau_{n-2})} \Big(\frac{f_{n+1/2} - f_{n-1/2}}{\tau_n + \tau_{n-1}} - \frac{f_{n-1/2} - f_{n-3/2}}{\tau_{n-1} + \tau_{n-2}} \Big) \qquad \text{(LTE-Taylor)}\\
&= \frac{\tau_n^3}{3(\tau_n + 2\tau_{n-1} + \tau_{n-2})} \Big(\frac{\frac{y_{n+1}-y_n}{\tau_n} - \frac{y_n - y_{n-1}}{\tau_{n-1}}}{\tau_n + \tau_{n-1}} - \frac{\frac{y_n - y_{n-1}}{\tau_{n-1}} - \frac{y_{n-1}-y_{n-2}}{\tau_{n-2}}}{\tau_{n-1} + \tau_{n-2}} \Big)\\
&= \frac{\tau_n^3}{3(\tau_n + 2\tau_{n-1} + \tau_{n-2})} \Big(y_{n+1} \frac{1}{\tau_n(\tau_n + \tau_{n-1})} - y_n \frac{\tau_n + \tau_{n-1} + \tau_{n-2}}{\tau_n \tau_{n-1}(\tau_{n-1} + \tau_{n-2})}\\
&\qquad + y_{n-1} \frac{\tau_n + \tau_{n-1} + \tau_{n-2}}{\tau_{n-1}\tau_{n-2}(\tau_n + \tau_{n-1})} - y_{n-2} \frac{1}{\tau_{n-2}(\tau_{n-1} + \tau_{n-2})} \Big).
\end{aligned}
$$

3.2. Estimation of the local truncation error using a variable step AB-2 solution. Here we estimate the local truncation error at t_{n+1} by evaluating the difference between the $\mathscr{O}(\Delta t^2)$ midpoint-solution y_{n+1} and another second-order approximation, $y_{n+1}^{\widetilde{AB2}}$, obtained by a variable-step Adams-Bashforth 2-like method. Let $\Pi_1(t)$ be the polynomial interpolating $f(y(t))$ at nodes $\{t_{n-1/2}, t_{n-3/2}\}$ and values $\{f_{n-1/2}, f_{n-3/2}\}$, which by (BEFE) denote:

$$
f_{n-1/2} = \frac{y_n - y_{n-1}}{\tau_{n-1}}, \qquad f_{n-3/2} = \frac{y_{n-1} - y_{n-2}}{\tau_{n-2}}.
$$

Then the solution to the AB2-like with variable step is:

$$
y_{n+1}^{\widetilde{AB2}} = y_n + \int_{t_n}^{t_{n+1}} \Pi_1(t)\, dt = y_n + f_{n-1/2} \frac{\tau_n(\tau_n + 2\tau_{n-1} + \tau_{n-2})}{\tau_{n-1} + \tau_{n-2}} - f_{n-3/2} \frac{\tau_n(\tau_n + \tau_{n-1})}{\tau_{n-1} + \tau_{n-2}}
$$

$$= y_n \frac{(\tau_n+\tau_{n-1})(\tau_n+\tau_{n-1}+\tau_{n-2})}{\tau_{n-1}(\tau_{n-1}+\tau_{n-2})} - y_{n-1}\frac{\tau_n(\tau_n+\tau_{n-1}+\tau_{n-2})}{\tau_{n-1}\tau_{n-2}} + y_{n-2}\frac{\tau_n(\tau_n+\tau_{n-1})}{\tau_{n-2}(\tau_{n-1}+\tau_{n-2})}, \tag{AB2-like}$$

and its local truncation error (under the 'localization assumption', i.e. back values are exact, see e.g. [18, p.70], [27, p.56]) can be written:

$$\widetilde{T_{n+1}^{AB2}} = \tau_n^3 y'''(t_{n+1/2})\left(\frac{1}{24}+\frac{1}{8}\left(1+\frac{\tau_{n-1}}{\tau_n}\right)\left(1+2\frac{\tau_{n-1}}{\tau_n}+\frac{\tau_{n-2}}{\tau_n}\right)\right).$$

For brevity, we denote the error coefficient in the right hand side, which depends on timestep ratios, by:

$$\mathscr{R}_n = \frac{1}{24}+\frac{1}{8}\left(1+\frac{\tau_{n-1}}{\tau_n}\right)\left(1+2\frac{\tau_{n-1}}{\tau_n}+\frac{\tau_{n-2}}{\tau_n}\right).$$

Then, from (3.1) and the expression above, we obtain the following approximation of the local truncation error of the midpoint rule (BEFE):

$$\widehat{T}_{n+1} = (y_{n+1}^{\text{midpoint}} - y_{n+1}^{\widetilde{AB2}})\frac{1}{1-1/(24\mathscr{R}_n)}, \tag{LTE-AB2}$$

where $y_{n+1}^{\text{midpoint}}$ denotes the midpoint solution from (BEFE), and $y_{n+1}^{\widetilde{AB2}}$ is given in (AB2-like).

3.3. Estimation of the local truncation error using a variable step AB-3 solution. We choose to estimate the local truncation error at t_{n+1} by evaluating the difference between the $\mathscr{O}(\Delta t^2)$ midpoint-solution y_{n+1} and a third-order approximation, u_{n+1}, obtained by the variable-step Adams-Bashforth 3 method (see e.g., [22, p. 398]). We denote:

$$f_{n-1/2} = \frac{y_n - y_{n-1}}{\tau_{n-1}}, \qquad f_{n-3/2} = \frac{y_{n-1}-y_{n-2}}{\tau_{n-2}}, \qquad f_{n-5/2} = \frac{y_{n-2}-y_{n-3}}{\tau_{n-3}},$$

and let $\Pi_2(t)$ be the polynomial interpolating $f(y(t))$ at nodes $\{t_{n-1/2}, t_{n-3/2}, t_{n-5/2}\}$ and values $\{f_{n-1/2}, f_{n-3/2}, f_{n-5/2}\}$:

$$\Pi_2(t) = f_{n-1/2} + \frac{f_{n-1/2}-f_{n-3/2}}{t_{n-1/2}-t_{n-3/2}}(t-t_{n-1/2}) + \frac{\frac{f_{n-1/2}-f_{n-3/2}}{t_{n-1/2}-t_{n-3/2}} - \frac{f_{n-3/2}-f_{n-5/2}}{t_{n-3/2}-t_{n-5/2}}}{t_{n-1/2}-t_{n-5/2}}(t-t_{n-1/2})(t-t_{n-3/2}).$$

Hence:

$$\begin{aligned} u_{n+1} \approx y_n + \int_{t_n}^{t_{n+1}} \Pi_2(t)\,dt = y_n + \tau_n\Bigg[& f_{n-1/2} + \frac{f_{n-1/2}-f_{n-3/2}}{t_{n-1/2}-t_{n-3/2}}\frac{\tau_n+\tau_{n-1}}{2} \\ & + \frac{\frac{f_{n-1/2}-f_{n-3/2}}{t_{n-1/2}-t_{n-3/2}} - \frac{f_{n-3/2}-f_{n-5/2}}{t_{n-3/2}-t_{n-5/2}}}{t_{n-1/2}-t_{n-5/2}} \cdot \left(\frac{1}{3}\tau_n^2+\frac{1}{2}\tau_{n-1}^2+\frac{3}{4}\tau_n\tau_{n-1}+\frac{1}{4}\tau_n\tau_{n-2}+\frac{1}{4}\tau_{n-1}\tau_{n-2}\right)\Bigg], \end{aligned}$$

and therefore the local truncation error can be approximated by:

$$\begin{aligned} \widehat{T}_{n+1} = \tau_n\Bigg[& f_{n-1/2} + \frac{f_{n-1/2}-f_{n-3/2}}{t_{n-1/2}-t_{n-3/2}}\frac{\tau_n+\tau_{n-1}}{2} \\ & + \frac{\frac{f_{n-1/2}-f_{n-3/2}}{t_{n-1/2}-t_{n-3/2}} - \frac{f_{n-3/2}-f_{n-5/2}}{t_{n-3/2}-t_{n-5/2}}}{t_{n-1/2}-t_{n-5/2}} \cdot \left(\frac{1}{3}\tau_n^2+\frac{1}{2}\tau_{n-1}^2+\frac{3}{4}\tau_n\tau_{n-1}+\frac{1}{4}\tau_n\tau_{n-2}+\frac{1}{4}\tau_{n-1}\tau_{n-2}\right)\Bigg]. \end{aligned} \tag{LTE-AB3}$$

REFERENCES

[1] R. ASSELIN, *Frequency filter for time integrations*, Mon. Wea. Rev., 100 (1972), pp. 487–490.
[2] M. BUKAC AND C. TRENCHEA, *Boundary update via resolvent for fluid-structure interaction*, tech. rep., University of Pittsburgh, 2018.
[3] J. C. BUTCHER, *Numerical methods for ordinary differential equations*, John Wiley & Sons, Ltd., Chichester, third ed., 2016. With a foreword by J. M. Sanz-Serna.

[4] A.-L. CAUCHY, *Équations différentielles ordinaires*, Éditions Études Vivantes, Ltée., Ville Saint-Laurent, QC; Johnson Reprint Corp., New York, 1981. Cours inédit. Fragment. [Unpublished course. Fragment], With a preface by Jean Dieudonné, With an introduction by Christian Gilain.
[5] S. D. CONTE AND C. DE BOOR, *Elementary numerical analysis*, vol. 78 of Classics in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2018.
[6] G. DAHLQUIST, *On one-leg multistep methods*, SIAM J. Numer. Anal., 20 (1983), pp. 1130–1138.
[7] G. DAHLQUIST AND A. K. BJÖRCK, *Numerical methods*, Dover Publications, Inc., Mineola, NY, 2003. Translated from the Swedish by Ned Anderson, Reprint of the 1974 English translation.
[8] G. G. DAHLQUIST, *On stability and error analysis for stiff non-linear problems, Part I*, Dept. of Comp. Sci. Roy. Inst. of Technology, Report TRITA-NA-7508 (1975).
[9] ———, *Error analysis for a class of methods for stiff non-linear initial value problems*, in Numerical Analysis, G. Watson, ed., vol. 506 of Lecture Notes in Mathematics, Springer Berlin Heidelberg, 1976, pp. 60–72.
[10] ———, *G-stability is equivalent to A-stability*, BIT, 18 (1978), pp. 384–401.
[11] G. G. DAHLQUIST, W. LINIGER, AND O. NEVANLINNA, *Stability of two-step methods for variable integration steps*, SIAM J. Numer. Anal., 20 (1983), pp. 1071–1085.
[12] D. R. DURRAN, *Numerical methods for fluid dynamics*, vol. 32 of Texts in Applied Mathematics, Springer, New York, second ed., 2010. With applications to geophysics.
[13] W. E, *Principles of multiscale modeling*, Cambridge University Press, Cambridge, 2011.
[14] C. W. GEAR, *Numerical initial value problems in ordinary differential equations*, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1971.
[15] V. GIRAULT AND P.-A. RAVIART, *Finite element approximation of the Navier-Stokes equations*, vol. 749 of Lecture Notes in Mathematics, Springer-Verlag, Berlin, 1979.
[16] E. GODLEWSKI AND P.-A. RAVIART, *Numerical approximation of hyperbolic systems of conservation laws*, vol. 118 of Applied Mathematical Sciences, Springer-Verlag, New York, 1996.
[17] P. GRESHO, R. SANI, AND M. ENGELMAN, *Incompressible flow and the finite element method: advection-diffusion and isothermal laminar flow*, Incompressible Flow & the Finite Element Method, Wiley, 1998.
[18] D. F. GRIFFITHS AND D. J. HIGHAM, *Numerical methods for ordinary differential equations*, Springer Undergraduate Mathematics Series, Springer-Verlag London, Ltd., London, 2010. Initial value problems.
[19] A. GUZEL AND W. LAYTON, *Time filters increase accuracy of the fully implicit method*, BIT, 58 (2018), pp. 301–315.
[20] A. GUZEL AND C. TRENCHEA, *The Williams step increases the stability and accuracy of the hoRA time filter*, Appl. Numer. Math., 131 (2018), pp. 158–173.
[21] E. HAIRER, C. LUBICH, AND G. WANNER, *Geometric numerical integration*, vol. 31 of Springer Series in Computational Mathematics, Springer, Heidelberg, 2010. Structure-preserving algorithms for ordinary differential equations, Reprint of the second (2006) edition.
[22] E. HAIRER, S. P. NØRSETT, AND G. WANNER, *Solving ordinary differential equations. I*, vol. 8 of Springer Series in Computational Mathematics, Springer-Verlag, Berlin, second ed., 1993. Nonstiff problems.
[23] E. HAIRER AND G. WANNER, *Solving ordinary differential equations. II*, vol. 14 of Springer Series in Computational Mathematics, Springer-Verlag, Berlin, 2010. Stiff and differential-algebraic problems, Second revised edition.
[24] W. HUNDSDORFER AND J. VERWER, *Numerical solution of time-dependent advection-diffusion-reaction equations*, vol. 33 of Springer Series in Computational Mathematics, Springer-Verlag, Berlin, 2003.
[25] A. ISERLES, *A first course in the numerical analysis of differential equations*, Cambridge Texts in Applied Mathematics, Cambridge University Press, Cambridge, 1996.
[26] J. D. LAMBERT, *Computational methods in ordinary differential equations*, John Wiley & Sons, London-New York-Sydney, 1973. Introductory Mathematics for Scientists and Engineers.
[27] ———, *Numerical methods for ordinary differential systems*, John Wiley & Sons, Ltd., Chichester, 1991. The initial value problem.
[28] W. LAYTON, Y. LI, AND C. TRENCHEA, *Recent developments in IMEX methods with time filters for systems of evolution equations*, J. Comput. Appl. Math., 299 (2016), pp. 50–67.
[29] W. LAYTON, N. MAYS, M. NEDA, AND C. TRENCHEA, *Numerical analysis of modular regularization methods for the BDF2 time discretization of the Navier-Stokes equations*, ESAIM Math. Model. Numer. Anal., 48 (2014), pp. 765–793.
[30] W. LAYTON, L. G. REBHOLZ, AND C. TRENCHEA, *Modular nonlinear filter stabilization of methods for higher Reynolds numbers flow*, J. Math. Fluid Mech., 14 (2012), pp. 325–354.
[31] Y. LI AND C. TRENCHEA, *A higher-order Robert–Asselin type time filter*, J. Comput. Phys., 259 (2014), pp. 23–32.
[32] ———, *Analysis of time filters used with the leapfrog scheme*, in Proceedings of the VI Conference on Computational Methods for Coupled Problems in Science and Engineering, Venice, Italy, May 2015, pp. 1261–1272.
[33] A. J. ROBERT, *The integration of a spectral model of the atmosphere by the implicit method*, Proc. WMO-IUGG Symp. on NWP, Tokyo, Japan Meteorological Agency, (1969), pp. 19–24.
[34] H. J. STETTER, *Analysis of discretization methods for ordinary differential equations*, Springer-Verlag, New York-Heidelberg, 1973. Springer Tracts in Natural Philosophy, Vol. 23.
[35] P. D. WILLIAMS, *A proposed modification to the Robert–Asselin time filter*, Mon. Wea. Rev., 137 (2009), pp. 2538–2546.
[36] ———, *The RAW filter: An improvement to the Robert–Asselin filter in semi-implicit integrations*, Mon. Wea. Rev., 139 (2011), pp. 1996–2007.
[37] ———, *Achieving seventh-order amplitude accuracy in leapfrog integrations*, Mon. Wea. Rev., 141 (2013), pp. 3037–3051.