7. Time Integration Methods

In this chapter, we will discuss various approaches to achieving time accurate solutions to the equations of low-speed or incompressible flow. Of course, these methods will be useful for integrating other systems of equations, but our principle focus will be low-speed or incompressible flows. We will cover several of the commonly used approaches to integration in time. The principal approaches are

- Lax-Wendroff-type (LW).
- Runge-Kutta (R-K), including total variation diminishing (TVD)[1] R-K or Strongly Stability Preserving (SSP) R-K.
- Linear Multi-step Methods (LMM).
- Implicit Methods.

Some of the above methods will be discussed more specifically in Chap. 10 in relation to their implementation in conjunction with the artificial compressibility method. Each of these methods has specific advantages and drawbacks. We will briefly cover each of these and introduce some of the specific methods for each of the above types. First, we will cover some basic principles employed in the construction, implementation, analysis and evaluation of these methods. This discussion will also serve to introduce the basic issues involved when considering a method for use with integrating the low-speed or incompressible flow equations.

7.1 Time Integration of the Flow Equations

For the purposes of this chapter we will consider the spatial discretization as an abstract operator. The fluid flow is described by equations of motion subject to a constraint imposed by the divergence-free condition. As was

[1] The total variation is defined as follows: In a discretized domain, a variable U is a function of the mesh and its total variation at a time instant n, is given by $\mathrm{TV}(U^n) \equiv \mathrm{TV}(U(t)) = \sum_{j=-\infty}^{+\infty} |U_{j+1}^n - U_j^n|$, where U is assumed to be either 0 or constant as the index j approaches the infinity, in order to obtain finite total variation. The total variation, $\mathrm{TV}(U(t))$ is a decreasing function of time, i.e., $\mathrm{TV}(U^{n+1}) \leq \mathrm{TV}(U^n)$ (see Sect. 13.2 for further discussion).

introduced in Chap. 5, there is no explicit equation for the pressure, *this is of extreme importance*. Various approaches to overcome this problem are discussed in Chaps. 10, 11 and 12.

Issues related to time-integration can also be studied considering the differentiation index of partial differential-algebraic equations [19, 374]. The index of a system of equations defines the nature of explicit or hidden constraints and characterizes the difficulty of solving the system. Simply stated a direct algebraic constraint typically produces an index one system. If the constraint on variables is implicit this will produce a higher index problem. Incompressible flow is an index two system[2] implying that it is quite difficult to solve thus many methods exhibit a loss of accuracy. This is indeed observed with respect to the solution of these equations as accurate solution of the pressure is often not achieved. While there are other explanations for this behavior, it is consistent with the index of the equations and the expectations arising from their solution.

Another approach quite often taken for high index problems is the index reduction where a transformation is applied to the system of equations to derive more explicit equations for the unknowns [19]. An example of this practice are pressure Poisson equation-based approaches to the incompressible flow equations. Care must be taken to ensure that the index reduction does not introduce anomalous solutions that satisfy the lower index system, but fail to solve the original high index problem.

Now, we will embark upon our introduction to the various methods for integrating systems of equations. Our starting point will be Lax-Wendroff methods where time and space differencing is interchanged to provide some unique characteristics as compared with the more traditional methods introduced later in the chapter. A detailed discussion of Lax-Wendroff time differencing for incompressible flows is given in Chap. 14.

7.2 Lax-Wendroff-Type Methods

Lax-Wendroff methods [321] are inherited from the tradition of compressible fluid dynamics solvers. These methods are focused on second-order methods. Another manner that is useful to describe these methods is as combined time and space differencing [171]. In this class of methods time derivative are replaced by spatial derivatives as described by the partial differential equations.

It is useful to review the development of Lax-Wendroff and some of its more useful derivative forms. Take the general form of a conservation law

$$\frac{\partial U}{\partial t} + \frac{\partial E\,(U)}{\partial x} = 0\,, \tag{7.1}$$

[2] This is because the constraint is applied to velocity, but the variable effected is pressure.

with some assumptions about the smoothness of derivatives (i.e., the Cauchy-Kovalevskaya theorem[3]) we can derive equations for the derivatives of this equation. As we will be developing a second-order method we need the following:

$$\frac{\partial^2 U}{\partial x \partial t} + \frac{\partial^2 E(U)}{\partial x^2} = 0 \,, \tag{7.2}$$

$$\frac{\partial^2 U}{\partial t^2} + \frac{\partial^2 E(U)}{\partial t \partial x} = 0 \,. \tag{7.3}$$

Next, expand the spatial terms in the above equations in terms of $E'(U)$ and $E''(U)$ where $E'(U) = \partial E(U)/\partial U$ (the Jacobian). Its useful to remember that

$$\frac{\partial U}{\partial t} + \frac{\partial E(U)}{\partial x} = 0 \rightarrow \frac{\partial U}{\partial t} + E'(U)\frac{\partial U}{\partial x} = 0 \,,$$

for sufficiently smooth functions. We can now introduce the necessary terms to cancel the first-order error in Taylor series expansions in time and space,

$$\frac{\partial}{\partial t}\left(\frac{\partial U}{\partial x}\right) + E'(U)\frac{\partial^2 U}{\partial x^2} + E''(U)\left(\frac{\partial U}{\partial x}\right)^2 = 0 \,. \tag{7.4}$$

By approximating these derivatives with centered derivatives, the Lax-Wendroff method can be derived which is consistent with the Taylor series to second-order,

$$\begin{aligned} \frac{U_j^{n+1} - U_j^n}{\Delta t} + \frac{E\left(U_{j+1}^n\right) - E\left(U_{j-1}^n\right)}{\Delta x} \\ = \frac{\Delta t\left[E\left(U_{j+1}^n\right) - 2E\left(U_j^n\right) + E\left(U_{j-1}^n\right)\right]}{\Delta x^2} \,. \end{aligned} \tag{7.5}$$

Richtmyer [446] derived a useful variant in a predictor-corrector format. The predictor formula is

$$U_{j+1/2}^{n+1/2} = \frac{1}{2}\left(U_j^n + U_{j+1}^n\right) - \frac{\Delta t}{2\Delta x}\left[E\left(U_{j+1}^n\right) - E\left(U_j^n\right)\right] \,, \tag{7.6}$$

and the corrector is

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} + \frac{E\left(U_{j+1/2}^{n+1/2}\right) - E\left(U_{j-1/2}^{n+1/2}\right)}{\Delta x} = 0 \,. \tag{7.7}$$

Yet another variant is the MacCormack scheme using alternating backward and forward differences [362],

$$\frac{U_j^{n+1,*} - U_j^n}{\Delta t} + \frac{E\left(U_j^n\right) - E\left(U_{j-1}^n\right)}{\Delta x} = 0 \,, \tag{7.8}$$

[3] The Cauchy-Kovalevskaya theorem basically states that an analytic solution of a partial differential equation exists through a Taylor expansion (and ample application of the chain rule).

and the forward differenced corrector,

$$\frac{U_j^{n+1} - U_j^{n+1,*}}{\Delta t} + \frac{E\left(U_{j+1}^{n+1,*}\right) - E\left(U_j^{n+1,*}\right)}{\Delta x} = 0 \,. \tag{7.9}$$

Still higher order methods can be found by continuing this process replacing third-order (and then fourth-order) time derivatives with spatial derivatives. The drawback of this method is that as the order increases the complexity of the algorithm increases geometrically for nonlinear equations. One of the best examples of this process can be found in [251]. More recently Toro et al. [545] have returned to this method. Additionally, Qiu and Shu [434] have applied Lax-Wendroff techniques for time accuracy for weighted essentially nonoscillatory methods (WENO) methods. For ENO-type differencing, the simplicity of method-of-lines approaches has supplanted the Lax-Wendroff style methods for time accuracy of greater than second-order.

These principles can be generally applied to incompressible flow. The key difference is the necessity of the divergence-free velocity field, at each (sub)step. While not absolutely necessary, the failure to enforce the divergence-free velocity at the mid-time-step has been shown to be prone to a weak nonlinear instability at large CFL numbers [266]. If the CFL number is kept at less than a half there is no evidence of this instability (the authors of the Book have verified this behavior). This experience acts as a general word of caution for integrating the incompressible flow equations with respect to pressure.

7.3 Other Approaches to Time-Centering

The chief mechanism to achieve time accuracy in this class of methods is to exchange spatial derivatives for time derivatives. As the PDE becomes more complicated, it becomes more important to include all of the terms in computing the differential balances. For incompressible flows the most important term is the pressure in its coupling to the divergence-free condition. The pressure gradient can be included explicitly, but (nonlinear) stability is enhanced by solving the pressure at the time-level appropriate for accuracy.

The chief decisions that one has to make is whether to include inter-cell coupling in the method. This can take several distinct forms: upwinding or Riemann solvers applied to the convective fluxes, the pressure solution and the treatment of viscous terms (all these issues are discussed in a much greater detail in the subsequent chapters). With upwinding the time-centered values one has the advantage of having a unique-single-valued velocity at the cell edges. This velocity field is useful in providing for a pressure solution thus providing for appropriate feedback to the velocity field.

Another alternative scheme for completing the time-centering is Hancock's scheme. Given the old time, cell-edge values of $U_{i+1/2,j}^n$ (i, j are grid

indices on a 2-D domain and n stands for the old time instant) the time-centering is computed without the use of characteristic extrapolation. Therefore, the time-derivative is completely taken from the underlying differential equation. For instance, if

$$\frac{\partial U}{\partial t} + \nabla \cdot E\left(U\right) = 0 \,,$$

then the time-centered value at $(i + 1/2, j)$ would be given by

$$U_{i+1/2,j}^{n+1/2} = U_{i+1/2,j}^{n} - \frac{\Delta t}{2} \nabla \cdot E\left(U^{n}\right) \,,$$

where the derivatives $\nabla \cdot E\left(U^{n}\right)$ are evaluated cell-by-cell using the spatial derivatives. Another alternative is to evaluate the derivatives by solving for the coupling between zones (a Riemann solution for compressible flow and advection plus pressure solve for incompressible flows). The convection is computed as it is in Chap. 14, but without characteristic extrapolation. This causes the stability condition for the algorithm to change to

$$\sum_{i=1}^{\text{dimensions}} \frac{\left|U_i\right| \Delta t}{\Delta x_i} \leq 1 \,.$$

Now we will discuss more classical methods for solving systems equations by considering them as systems of ODEs (method-of-lines approach).

7.4 Runge-Kutta Methods

Runge-Kutta methods are commonly used for integrating ODEs where the evaluation of the function $f\left(u, t\right)$ is thought of as being inexpensive. The presentation is done in terms of this general function because the time integration methods apply to general (ordinary and partial) differential equations. Accuracy is built up through solving the ODE in a series of steps (or stages). The incompressible flow equations are typically viewed as having expensive function evaluations because of the elliptic equation arising from the pressure solution. Thus, the number of stages in a Runge-Kutta method is equal to the number of pressure-equation solutions. This character is essential to the accuracy and stability of the overall method for incompressible flow. In other words, all intermediate velocity fields employed by the algorithm are made divergence-free. Another key advantage of Runge-Kutta methods is that time step changes are dealt with simplicity. This is because the methods are self-contained within a time step and do not require the storage across more than one time step. For LMMs the method becomes more complicated for uneven time step sizes because it is derived through a sequence of interpolation and integration over several time levels.

7.4.1 Second-Order Runge-Kutta

In Chap. 6 we introduced the first-order explicit Runge-Kutta method: the forward Euler method. There are several second-order Runge-Kutta algorithms that are all essentially identical for linear equations. They are differentiated by their cost in a number of function evaluations as well as by the nonlinear error and stability properties.[4] Perhaps the simplest and best known of these is

$$\left.\begin{aligned} \frac{U^1 - U^n}{h} &= \tfrac{1}{2} f\left(U^n, t^n\right) \\ \frac{U^{n+1} - U^n}{h} &= f\left(U^1, t^{n+1/2}\right) \end{aligned}\right\}, \tag{7.10}$$

where $h = \Delta t$. Its stability is shown in Fig. 7.1. The truncation error for this method is

$$\frac{1}{6} h^2 U_{ttt}.$$

One of the second-order Runge-Kutta variants is also known as a TVD Runge-Kutta method[5] [493], more recently denoted as a strongly stability preserving (SSP) method [220]. It is also known as Heun's method [19, 403],

$$\left.\begin{aligned} \frac{U^1 - U^n}{h} &= f\left(U^n, t^n\right) \\ \frac{U^{n+1} - U^n}{h} &= \tfrac{1}{2}\left[f\left(U^n, t^n\right) + f\left(U^1, t^{n+1}\right)\right] \end{aligned}\right\}. \tag{7.11}$$

This class of methods permits the solution to retain certain favorable characteristics of the nonlinear spatial differencing. As Shu states [494] if one is concerned about the nonlinear stability of the integration procedure, then these methods deserve consideration. The conditions for these methods to have this property are fairly restrictive. At higher order the method becomes less attractive due to more restrictive stability conditions and non-standard spatial operators required. If these properties are not required, the linear stability of these methods is equivalent to the standard Runge-Kutta methods. This highlights the intrinsic difference between linear and nonlinear stability in a numerical method.

For incompressible flows computed using high-resolution methods, Runge-Kutta methods were first employed by Sussman et al. [522] and later by Shu and E [495] using ENO spatial discretizations (ENO schemes are presented in

[4] For simply hyperbolic PDEs the stability is defined by a CFL number, but we are considering more general properties of the differential equations.

[5] TVD Runge-Kutta methods were developed and presented by Shu and Osher [493] for hyperbolic PDEs (see Sect. 7.4.4).

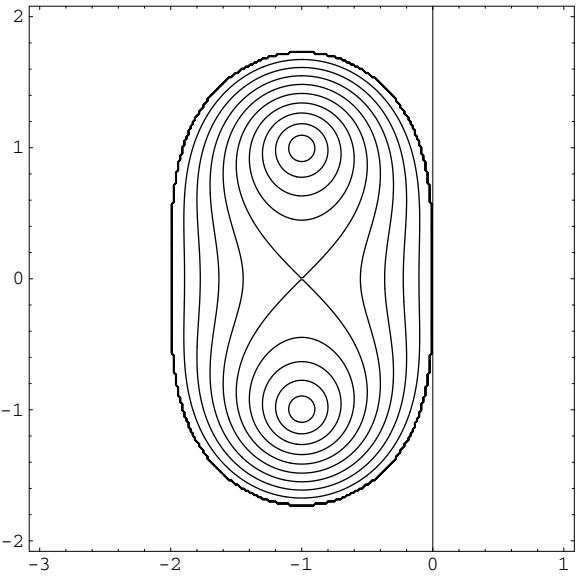

Fig. 7.1. The stability region, $|A| \leq 1$, for the second-order Runge-Kutta method. The horizontal axis describes the real part of the linear operator while the vertical axis describes the imaginary part of the generic operator.

Chap. 17). In both cases the projection operator (pressure solve) was applied once per stage.

As with the second-order method there is a variant of an SSP method producing larger CFL limits with the cost of more function evaluations. For example, a four-stage method is given by

$$\left.\begin{aligned} \frac{U^1 - U^n}{h} &= \tfrac{1}{2} f\left(U^n, t^n\right) \\ \frac{U^2 - U^1}{h} &= \tfrac{1}{2} f\left(U^1, t^{n+1/2}\right) \\ \frac{U^3 - \frac{1}{3}U^1 - \frac{2}{3}U^n}{h} &= \tfrac{1}{6} f\left(U^1, t^{n+1/2}\right) \\ \frac{U^{n+1} - U^2}{h} &= \tfrac{1}{2} f\left(U^2, t^{n+1/2}\right) \end{aligned}\right\}. \tag{7.12}$$

This carries a CFL limit of 2 rather than 1 as in (7.11).

For nonlinear problems there is a difference in (7.11) if the final step is evaluated using a midpoint rule

$$\frac{U^{n+1}-U^n}{h} = f\left(\frac{U^n+U^1}{2}, t^{n+1/2}\right) . \tag{7.13}$$

The midpoint rule, (7.13), produces smaller coefficients on the truncation for terms proportional to $\partial^2 f/\partial U^2$ and $\partial^3 f/\partial U^3$.

7.4.2 Third-Order Runge-Kutta

Heun's third-order method is [19, 403]

$$\left.\begin{aligned} \frac{U^1-U^n}{h} &= \tfrac{1}{3} f\left(U^n, t^n\right) \\ \frac{U^1-U^n}{h} &= \tfrac{2}{3} f\left(U^n, t^n\right) \\ \frac{U^{n+1}-U^n}{h} &= \tfrac{1}{4}\left[f\left(U^n, t^n\right) + 3f\left(U^2, t^{n+2/3}\right)\right] \end{aligned}\right\} . \tag{7.14}$$

The truncation error of this method is

$$\frac{1}{24} h^3 U_{tttt}.$$

The stability of the method is displayed in Fig. 7.2.

The TVD third-order method is also quite commonly used because it has the same stability conditions as the second-order methods. The form of the solution is

$$\left.\begin{aligned} \frac{U^1-U^n}{h} &= f\left(U^n, t^n\right) \\ \frac{U^2-U^n}{h} &= \tfrac{1}{4}\left[f\left(U^n, t^n\right) + f\left(U^1, t^{n+1}\right)\right] \\ \frac{U^{n+1}-U^n}{h} &= \\ \frac{1}{6}\Big[\, f\left(U^n, t^n\right) + 4f\left(U^2, t^{n+1/2}\right) + f\left(U^1, t^{n+1}\right) \,\Big] & \end{aligned}\right\} . \tag{7.15}$$

As in second-order Runge-Kutta, recent work [509] has shown a wider variety of these methods producing larger CFL limits with the cost of more function evaluations. For example, a four-stage method is given by

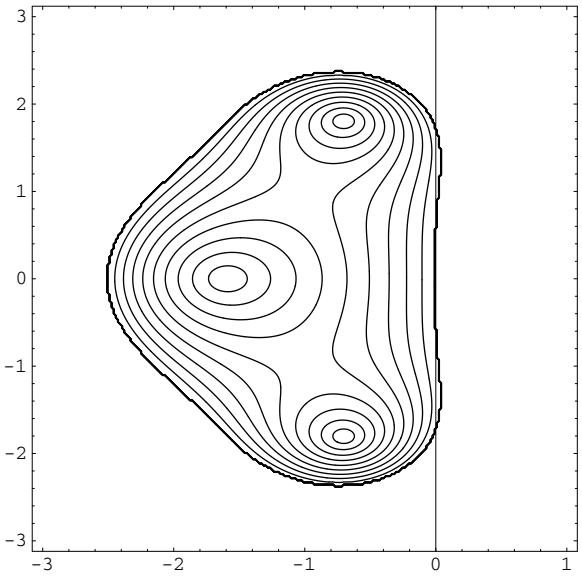

Fig. 7.2. The stability region, $|A| \leq 1$, for the third-order Heun's Runge-Kutta method.

$$\left.\begin{aligned} \frac{U^1 - U^n}{h} &= \tfrac{1}{2} f\left(U^n, t^n\right) \\ \frac{U^2 - U^1}{h} &= \tfrac{1}{2} f\left(U^1, t^{n+1/2}\right) \\ \frac{U^{n+1} - \frac{2}{3}U^2 - \frac{1}{3}U^n}{h} &= \tfrac{1}{3}\left[f\left(U^2, t^{n+1/2}\right) + f\left(U^1, t^{n+1}\right)\right] \end{aligned}\right\} . \tag{7.16}$$

This carries a CFL limit of 2 rather than 1 as in (7.11).

7.4.3 Fourth-Order Runge-Kutta

We will only give the classical fourth-order scheme. The method consists of the following four stages that build up the solution to design accuracy,

$$\left.\begin{aligned}
\frac{U^1 - U^n}{h} &= \tfrac{1}{2} f\left(U^n, t^n\right) \\
\frac{U^2 - U^n}{h} &= \tfrac{1}{2} f\left(U^1, t^{n+1/2}\right) \\
\frac{U^3 - U^n}{h} &= f\left(U^2, t^{n+1/2}\right) \\
\frac{U^{n+1} - U^n}{h} &= \tfrac{1}{6}\Big[\, f\left(U^n, t^n\right) \\
&\quad + 2f\left(U^1, t^{n+1/2}\right) + 2f\left(U^2, t^{n+1/2}\right) + f\left(U^3, t^{n+1}\right)\Big]
\end{aligned}\right\} \quad (7.17)$$

This method is one of the workhorse algorithms for the non-ODE-expert to integrate differential equations in time. The truncation error of this method is

$$\frac{1}{120} h^4 U_{ttttt}.$$

The stability of the method is displayed in Fig. 7.3. The reader would observe that Runge-Kutta methods gain larger stability regions with growing order.

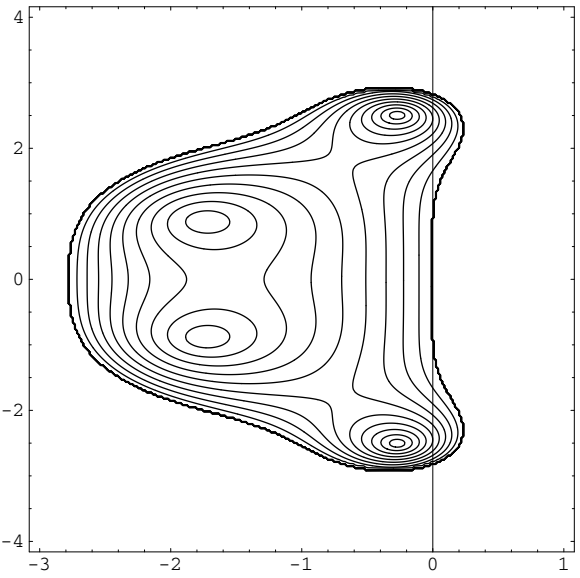

Fig. 7.3. The stability region, $|A| \leq 1$, for the fourth-order Runge-Kutta method.

Note that for fourth-order of accuracy, the TVD Runge-Kutta methods begin to incur a reduced stability (CFL) limit, and/or require non-standard spatial operators (i.e., anti-upwind schemes). As a consequence, they tend to be much less popular than the second- or third-order versions of this class of method. For initial value problems in fluid dynamics it has been found that the relative gain in accuracy and solution-quality reaches the point of diminishing returns at third-order.

7.4.4 TVD Runge-Kutta Methods Applied to Hyperbolic Conservation Laws

The family of explicit Runge-Kutta schemes of various orders of accuracy was introduced by Shu and Osher [493, 495] for hyperbolic conservation laws. Their goal was to develop r-th order approximations of a differential equation

$$U_t = L(U) \ , \tag{7.18}$$

where $L(U) = -\nabla \cdot E(U)$ is a spatial operator. We also define the operator $\mathcal{L}(U) = L(U) + \mathcal{O}(\Delta x^r)$ as an r-th order approximation to L. The general explicit Runge-Kutta method is given by

$$U^i = U^0 + \Delta t \sum_{k=0}^{i-1} c_{ik} \mathcal{L}\big(U^{(k)}\big), \quad i = 1, 2, ..., m \ , \tag{7.19}$$

$$U^{(0)} = U^n, \quad U^{(m)} = U^{n+1} \ , \tag{7.20}$$

where c_{ik} are coefficients the values of which depend on the order of accuracy of the scheme (see discussion below). If the differential equation contains source terms and/or the boundary conditions are time dependent, the operator $\mathcal{L}$ depends explicitly on time and in this case the Runge-Kutta method takes a more complicated form

$$U^i = U^0 + \Delta t \sum_{k=0}^{i-1} c_{ik} \mathcal{L}\big(U^{(k)}, t^{(0)} + d_k \Delta t\big) \ , \tag{7.21}$$

where

$$d_k = \sum_{i=0}^{k-1} c_{kl} \ . \tag{7.22}$$

To produce explicit schemes which satisfy the total variation diminishing (TVD) condition,[6] we can rewrite (7.19) as follows [495]

[6] The TVD condition states that the total variation of the solution (see Sect. 13.2 for its definition) with respect, for example, to spatial dimension x is uniformly bounded with respect to t; see Chap. 13 for further discussion on the total variation.

$$U^{(i)} = \sum_{k=0}^{i-1} \left[a_{ik} U^{(k)} + b_{ik} \Delta t \mathcal{L}(U^{(k)}) \right] , \tag{7.23}$$

where

$$\sum_{k=0}^{i-1} a_{ik} = 1, \quad \text{and} \quad b_{ik} = c_{ik} - \sum_{l=k+1}^{i-1} c_{lk} a_{il} .$$

On the basis of (7.23), different orders of accuracy Runge-Kutta methods can be obtained. This is done by using Taylor series expansions and choose coefficients a_{ik} and b_{ik} that optimize the CFL restriction. We list below the various versions of the Runge-Kutta schemes as proposed by Shu and Osher [495]:

- *Second-order of accuracy*:
  The coefficients are given by

$$\left.\begin{aligned} a_{20} &= 1 - a_{21} \\ b_{20} &= 1 - \frac{1}{2b_{10}} - a_{21} b_{10} \\ b_{21} &= \frac{1}{2b_{10}} \end{aligned}\right\} , \tag{7.24}$$

  where b_{10} and a_{21} are free parameters. The optimal scheme is given by

$$\left.\begin{aligned} U^{(1)} &= U^{(0)} + \Delta t \mathcal{L}\big(U^{(0)}\big) \\ U^{(2)} &= U^{(0)} + \tfrac{1}{2} \Delta t \mathcal{L}\big(U^{(0)}\big) + \tfrac{1}{2} \Delta t \mathcal{L}\big(U^{(1)}\big) \end{aligned}\right\} , \tag{7.25}$$

  with the CFL coefficient being equal to 1. Equation (7.25) is the modified Euler method [207].

- *Third-order of accuracy*:
  The coefficients are given by

$$\left.\begin{aligned} a_{32} &= 1 - a_{31} - a_{30} , \\ b_{32} &= \frac{3b_{10} - 2}{6P(b_{10} - P)} , \\ b_{21} &= \frac{1}{6 b_{10} b_{32}} , \end{aligned}\right\}$$

$$\left.\begin{aligned} b_{31} &= \frac{1/2 - a_{32}b_{10}b_{21} - Pb_{32}}{b_{10}} \,, \\ b_{30} &= 1 - a_{31}b_{10}a_{32}P - b_{31} - b_{32} \,, \\ b_{20} &= P - a_{21}b_{10} - b_{21} \,, \end{aligned}\right\} \tag{7.26}$$

where $P = b_{20} + a_{21}b_{10} + b_{21}$, a_{21}, a_{30}, a_{31} and b_{10} are free parameters. The following (optimum) version of the scheme has been proposed [495]

$$\left.\begin{aligned} U^{(1)} &= U^{(0)} + \Delta t\mathcal{L}\big(U^{(0)}\big) \\ U^{(2)} &= \tfrac{3}{4}U^{(0)} + \tfrac{1}{4}U^{(1)} + \tfrac{1}{4}\Delta t\mathcal{L}\big(U^{(1)}\big) \\ U^{(3)} &= \tfrac{1}{3}U^{(0)} + \tfrac{2}{3}U^{(2)} + \tfrac{2}{3}\Delta t\mathcal{L}\big(U^{(2)}\big) \end{aligned}\right\}, \tag{7.27}$$

with the CFL coefficient being equal to 1.

- *Fourth-order of accuracy*:
  The fourth-order version is written as [207]

$$\left.\begin{aligned} U^{(1)} &= U^{(0)} + \tfrac{1}{2}\Delta t\mathcal{L}\big(U^{(0)}\big) \\ U^{(2)} &= \tfrac{1}{2}U^{(0)} - \tfrac{1}{4}\Delta t\widetilde{\mathcal{L}}\big(U^{(0)}\big) + \tfrac{1}{2}U^{(1)} + \tfrac{1}{2}\Delta t\mathcal{L}\big(U^{(1)}\big) \\ U^{(3)} &= \tfrac{1}{9}U^{(0)} - \tfrac{1}{9}\Delta t\widetilde{\mathcal{L}}\big(U^{(0)}\big) + \tfrac{2}{9}U^{(1)} - \tfrac{1}{3}\Delta t\widetilde{\mathcal{L}}\big(U^{(1)}\big) \\ &\quad + \tfrac{2}{3}U^{(2)} + \Delta t\mathcal{L}\big(U^{(2)}\big) \\ U^{(4)} &= \tfrac{1}{3}U^{(0)} - \tfrac{1}{6}\Delta t\mathcal{L}\big(U^{(1)}\big) + \tfrac{1}{3}U^{(2)} + \tfrac{1}{3}\big(U^{(3)}\big) \\ &\quad + \tfrac{1}{6}\Delta t\mathcal{L}\big(U^{(3)}\big) \end{aligned}\right\}, \tag{7.28}$$

  with the CFL coefficient being equal to 2/3.
  The operator $\widetilde{\mathcal{L}}$ is also a discrete approximation to the spatial operator L, which satisfies the TVD condition under the same CFL restriction.

- *Fifth-order of accuracy*:
  The formula for the fifth-order method is given by [314, 495]

$$
\left.
\begin{aligned}
U^{(1)} &= U^{(0)} + \tfrac{1}{2}\Delta t\mathcal{L}\big(U^{(0)}\big) \\
U^{(2)} &= \tfrac{3}{4}U^{(0)} + \tfrac{1}{4}U^{(1)} + \tfrac{1}{8}\Delta t\mathcal{L}\big(U^{(1)}\big) \\
U^{(3)} &= \tfrac{3}{8}U^{(0)} - \tfrac{1}{8}\Delta t\widetilde{\mathcal{L}}\big(U^{(0)}\big) + \tfrac{1}{8}U^{(1)} - \tfrac{1}{16}\Delta t\widetilde{\mathcal{L}}\big(U^{(1)}\big) \\
&\quad + \tfrac{1}{2}U^{(2)} + \tfrac{1}{2}\Delta t\mathcal{L}\big(U^{(2)}\big) \\
U^{(4)} &= \tfrac{1}{4}U^{(0)} - \tfrac{5}{64}\Delta t\widetilde{\mathcal{L}}\big(U^{(0)}\big) + \tfrac{1}{8}U^{(1)} - \tfrac{13}{64}\Delta t\widetilde{\mathcal{L}}\big(U^{(1)}\big) \\
&\quad + \tfrac{1}{8}U^{(2)} + \tfrac{1}{8}\Delta t\mathcal{L}\big(U^{(2)}\big) + \tfrac{1}{2}U^{(3)} + \tfrac{9}{16}\Delta t\mathcal{L}\big(U^{(1)}\big) \\
U^{(5)} &= \tfrac{89537}{2880000}U^{(0)} + \tfrac{2276219}{40320000}\Delta t\mathcal{L}\big(U^{(0)}\big) + \tfrac{407023}{2880000}U^{(1)} \\
&\quad + \tfrac{407023}{672000}\Delta t\mathcal{L}\big(U^{(1)}\big) + \tfrac{1511}{12000}U^{(2)} + \tfrac{1511}{2800}\Delta t\mathcal{L}\big(U^{(2)}\big) \\
&\quad + \tfrac{87}{200}U^{(3)} - \tfrac{261}{140}\widetilde{\mathcal{L}}\big(U^{(3)}\big) + \tfrac{4}{15}U^{(4)} + \tfrac{8}{7}\mathcal{L}\big(U^{(4)}\big) \\
U^{(6)} &= \tfrac{4}{9}U^{(0)} + \tfrac{1}{15}U^{(1)} - \tfrac{8}{45}\widetilde{\mathcal{L}}\big(U^{(1)}\big) + \tfrac{8}{45}U^{(3)} \\
&\quad + \tfrac{2}{3}\Delta t\mathcal{L}\big(U^{(3)}\big) + \tfrac{14}{45}U^{(5)} + \tfrac{7}{90}\Delta t\mathcal{L}\big(U^{(5)}\big)
\end{aligned}
\right\}, \tag{7.29}
$$

with the CFL coefficient being equal to 7/30.

Another class of methods introduced recently are the natural continuous extension (NCE) methods [57]. There are also a large class of implicit Runge-Kutta methods based on various quadratures. We will not cover these methods explicitly and the details about these methods can be found in a number of textbooks that exclusively cover numerical methods for ODEs (e.g., Ascher and Petzold [19]).

7.5 Linear Multi-step Methods

Linear multi-step methods are another general class of ODE methods. These are less commonly associated with the solution of hyperbolic PDEs [19]. They have the advantage of only requiring one function evaluation per time step. Thus, they are favored for expensive function evaluations such as the pressure Poisson equation. On the other hand, these methods require the storage of the function for one or more preceding time steps. Moreover, the necessity of using the function evaluations from previous time steps make the start up for these methods problematic. This is usually handled with the use of lower order methods or Runge-Kutta methods until sufficient memory has been built up.

7.5.1 Adams-Bashforth Method

The second-order version of this method is quite commonly used and takes a simple form,

$$\frac{U^{n+1} - U^n}{h} = \frac{1}{2}\left[3f\left(U^n, t^n\right) - f\left(U^{n-1}, t^{n-1}\right)\right] \ . \tag{7.30}$$

One of the first things to notice is that the truncation error takes a different form

$$-\frac{5h^2}{12} f\left(u, t\right) \ .$$

On the negative side, the stability is more limiting for this method than second-order Runge-Kutta methods, being roughly half their value. This is shown in Fig. 7.4. Note that multi-step methods have k (k is the step number) amplification factors and one must look at the worst of these to determine stability.

As noted earlier, variable time step sizes also cause some difficulty in relation to that formulas change between different time steps. For variable time step sizes one can use Lagrange interpolation over the time interval t^n to t^{n+1} and then integrate the interpolated function. This yields the following method,

$$\frac{U^{n+1} - U^n}{h^n} = f\left(U^n, t^n\right) + \frac{h^n}{2h^{n-1}}\left[f\left(U^n, t^n\right) + f\left(U^{n-1}, t^{n-1}\right)\right] \ , \tag{7.31}$$

where $h^n = t^{n+1} - t^n$ and $h^{n-1} = t^n - t^{n-1}$. Another approach stated in a manner similar to Runge-Kutta methods, is given by

$$\left.\begin{aligned} \frac{U^1 - U^n}{h^n} &= \frac{1}{2}\frac{U^n - U^{n-1}}{h^{n-1}} \\ \frac{U^{n+1} - U^n}{h} &= f\left(U^1, t^{n+1/2}\right) \end{aligned}\right\} \ . \tag{7.32}$$

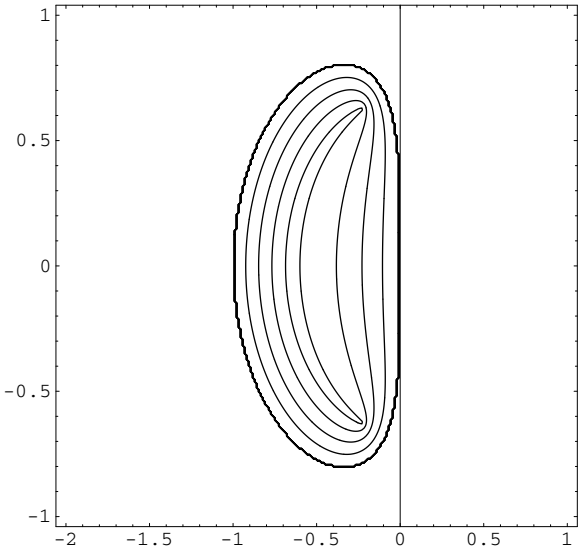

Fig. 7.4. The stability region, $\max_k |A_k| \leq 1$, for the second-order Adams-Bashforth method. For multi-step methods k is the step number.

A back-substitution of the first stage into the second confirms that this method is linearly equivalent to the classical second-order Adams-Bashforth method (7.30). This approach requires the storage of the time derivative from the previous time step. The more classical approach to this problem is to conduct an interpolation over the previous two time steps and integrate thus producing the following,

$$\frac{U^{n+1} - U^n}{h^n} = \left(1 - \frac{h^n}{h^{n-1}}\right) f\left(U^n, t^n\right) - \frac{h^n}{h^{n-1}} f\left(U^{n-1}, t^{n-1}\right) . \quad (7.33)$$

The third-order Adams-Bashforth method is

$$\frac{U^{n+1} - U^n}{h} = \frac{1}{12}\Big[23 f\left(U^n, t^n\right) - 16 f\left(U^{n-1}, t^{n-1}\right) + 5 f\left(U^{n-2}, t^{n-2}\right)\Big] . \quad (7.34)$$

The Adams-Bashforth methods are contrasted with Runge-Kutta methods in that the stability region decreases with increasing order. This is easily seen in comparing Figs. 7.4 and 7.5.

There are also linear multi-step methods of a SSP type [220, 494, 509]. These methods are the generalization of TVD time discretizations sharing the same qualities and should be employed if strong nonlinear stability is desired [494]. The second-order method can be written,

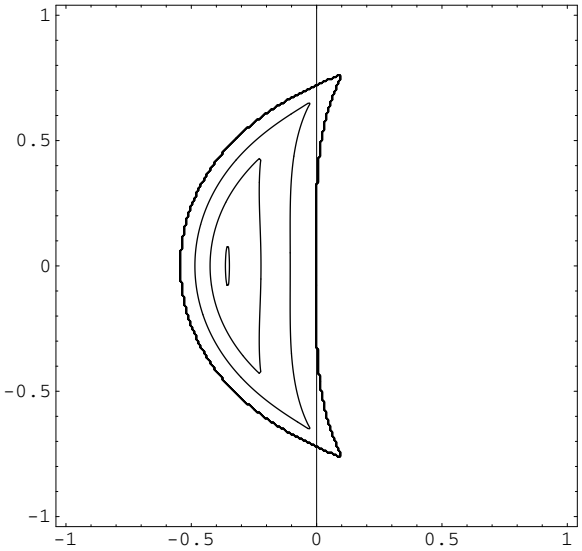

Fig. 7.5. The stability region, $\max_k |A_k| \leq 1$, for the third-order Adams-Bashforth method.

$$\frac{4U^{n+1} - 3U^n - U^{n-1}}{6h} = f\left(U^n, t^n\right) \ . \tag{7.35}$$

This method is stable for a relatively larger region than it is SSP, i.e., a time step limit of 1/2.

There is also a third-order SSP-LMM method, which has a reasonable form,

$$\frac{27U^{n+1} - 16U^n - 11U^{n-3}}{60h} = \frac{4}{5} f\left(U^n, t^n\right) + \frac{1}{5} f\left(U^{n-2}, t^{n-2}\right) \ . \tag{7.36}$$

This method has a stability limit of 1/2 for retaining the properties of SSP methods.

General Remark:

Each of the various methods for integrating the incompressible flow equations has its distinct advantages and disadvantages. In the case of pressure-Poisson methods, the main issue is the number of pressure Poisson solutions that are required per unit time step. Thus, one must factor in the CFL limit (or the CFL number that is practical for a given accuracy) into this.

Perhaps one of the practical trade offs can be seen by comparing the CFL limit and number of steps between the third-order Runge-Kutta and Adams-Bashforth methods. The three steps of the Runge-Kutta method require a CFL number of less than 1.5 with one pressure solution per step. The Adams-

Bashforth method has a CFL limit of 0.73 with one pressure solution (overall). Thus, by efficiency alone the Adams-Bashforth method is preferable.

7.5.2 Adams-Moulton Method

Implicit methods will entail more complex solution algorithms as the velocity and pressure solution are coupled. In general, this will also require the solution of coupled nonlinear equations. Linearizations can be performed, and indeed are required for effective preconditioners, but must be used with extreme caution without achieving some degree of nonlinear convergence. This can be found through the use of an efficient modern Newton's method like Newton-Krylov [86]. Using explicit methods only the pressure solution necessarily entails numerical linear algebra. As before, stability of the solution depends on having a (approximate or exact) divergence-free velocity field. With the implicit solution this property comes automatically.

Adams-Moulton methods are implicit in that the update formula contains the current time level [19]. These methods are derived like the explicit Adams-Bashforth methods, but the interpolant includes the current time level (and one level back in time for a given order of accuracy). The basic method for the time derivative is the the backward Euler method,

$$\frac{U^{n+1} - U^n}{h} = f\left(U^{n+1}, t^{n+1}\right) \ . \tag{7.37}$$

Like the forward Euler method a first-order accuracy is obtained. Unlike the explicit Euler scheme, the method is extremely stable and in fact is unstable for a small region only. Fig. 7.6 shows the stability properties of the method.

The next method in a sequence that raises the accuracy of time integration is the Crank-Nicholson method. This is simply a trapezoidal integration rule, but unlike the explicit case, one must implicitly compute the new time values. The update formula is quite simple,

$$\frac{U^{n+1} - U^n}{h} = \frac{1}{2}\left[f\left(U^n, t^n\right) + f\left(U^{n+1}, t^{n+1}\right)\right] \ . \tag{7.38}$$

One obvious variation of this method is the implicit midpoint rule,

$$\frac{U^{n+1} - U^n}{h} = \frac{1}{2} f\left(\frac{U^n + U^{n+1}}{2}, \frac{t^n + t^{n+1}}{2}\right) \ . \tag{7.39}$$

Both of the above variants have desirable stability properties. The stability region is shown in Fig. 7.7. The characteristic of being stable for all dissipative operators is known as "A-stability".[7] Crank-Nicolson and implicit midpoint methods are all A-stable methods.

[7] A-stable methods are those which have regions of stability containing the whole of the left-hand half plane in the stability diagram, i.e., Re$G < 0$, where G is defined in Sect. 6.2.

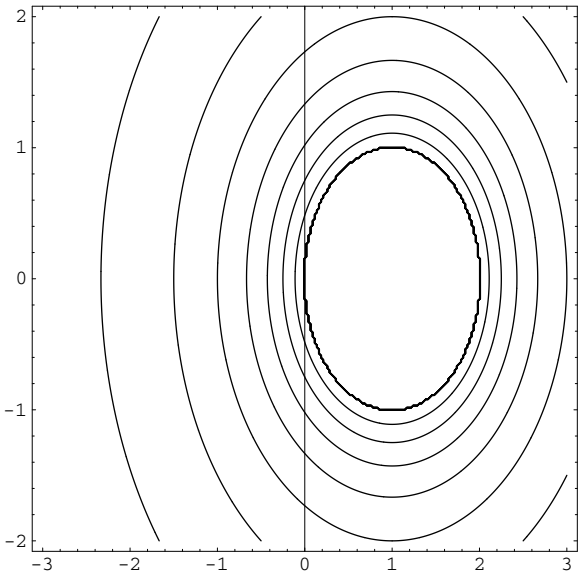

Fig. 7.6. The stability region, $\max_k |A_k| \leq 1$, for the first-order backward Euler method.

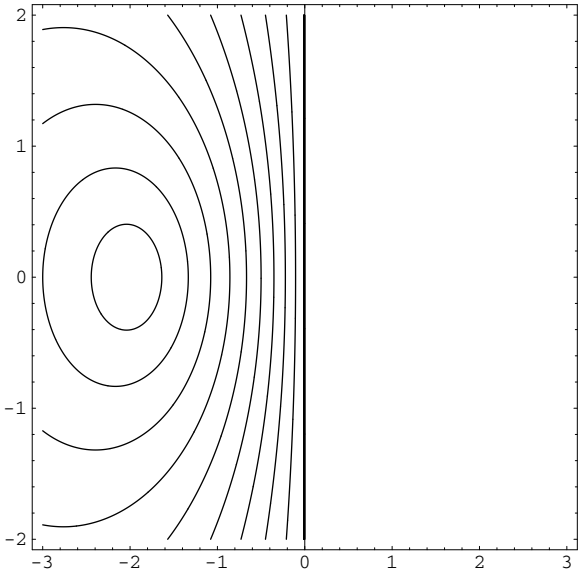

Fig. 7.7. The stability region, $\max_k |A_k| \leq 1$, for the second-order Crank-Nicholson method.

In the Adams-Moulton family of methods, the next method in the sequence is a third-order method,

$$\frac{U^{n+1} - U^n}{h} = \frac{1}{12}\Big[5f\left(U^{n+1}, t^{n+1}\right) + 8f\left(U^n, t^n\right) - f\left(U^{n-1}, t^{n-1}\right)\Big]. \tag{7.40}$$

In this case the price paid for this extra order of accuracy is quite high in terms of stability. Rather than unconditional stability for negative real functions, there is now an uncomfortably small stability region. This is shown in Fig. 7.8. As the order of Adams-Moulton methods increases, the stability region continues to shrink.

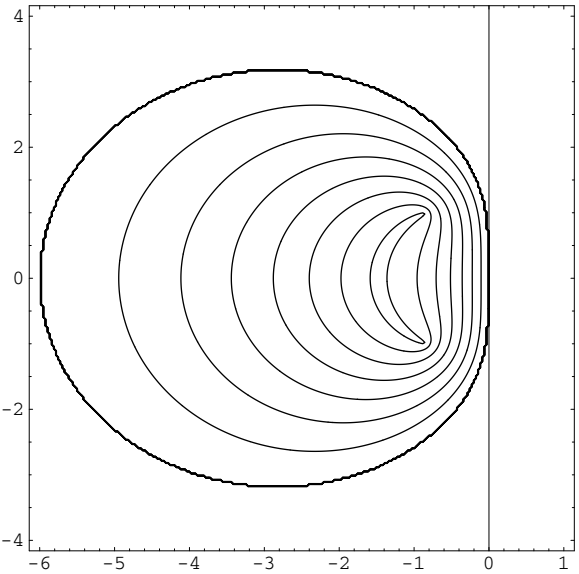

Fig. 7.8. The stability region, $\max_k |A_k| \leq 1$, for the third-order Adams-Moulton method.

An alternative to treating these methods as implicit methods are predictor-corrector methods where an explicit method is used to "predict" the time advanced solution which is substituted into the implicit difference formula in the "corrector" step. Typically, this results in an enhanced stability region over the purely explicit method. As a simple example consider the coupling of forward Euler with the trapezoidal rule, producing Heun's method.

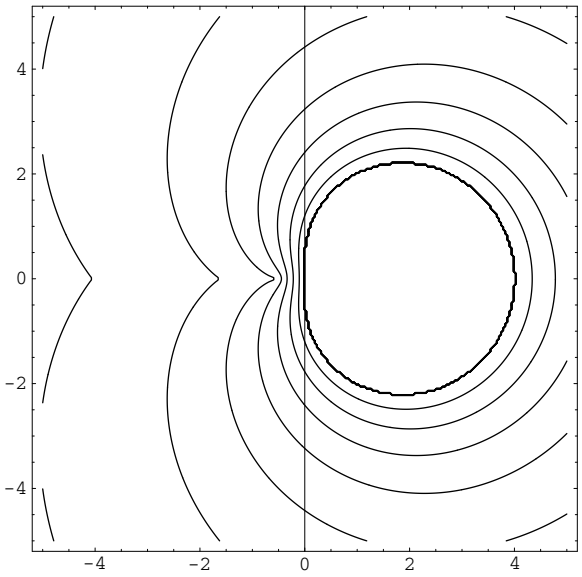

Fig. 7.9. The stability region, $\max_k |A_k| \leq 1$, for the second-order backward differentiation formula.

7.5.3 Backward Differentiation Formulas

The last category of methods we will introduce are backwards differentiation formulas (BDFs) [19]. These methods are quite popular for solving stiff systems of equations. This is because of their simple form and large stability regions. For the same order of accuracy these methods have much larger stability regions than the Adams-Moulton methods. BDF methods are chiefly characterized by only evaluating the function at the advance time, $n + 1$. Accuracy is achieved through high-order approximations to U_t.

BDFs and Adams-Moulton methods share their first-order incarnation, the backward Euler method. The second-order BDF update equation is different however, i.e.,

$$\frac{3U^{n+1} - 4U^n + U^{n-1}}{2h} = f\left(U^{n+1}, t^{n+1}\right). \tag{7.41}$$

The large stability region is shown in Fig. 7.9.

The third-order method includes one more time level in the approximation of U_t,

$$\frac{11U^{n+1} - 18U^n + 9U^{n-1} - 2U^{n-2}}{6h} = f\left(U^{n+1}, t^{n+1}\right). \tag{7.42}$$

Its stability region is slightly smaller than the second-order method, but nevertheless is large and significantly larger than the Adams-Moulton method.