3 HIGH-ORDER TIME-DISCRETIZATION SCHEMES

3.1 Introduction

The partial differential equation to be solved is of the general form

$$\partial_t u = H(u) \tag{3.1}$$

where H is a differential operator generally nonlinear. With a view to application to fluid dynamics, H is the sum of a nonlinear first-order term $N(u)$ and a linear second-order one $L(u)$, that is

$$H(u) = N(u) + L(u) \tag{3.2}$$

with

$$N(u) = -\partial_x F(u)\,, \quad L(u) = \nu \partial_{xx} u \quad (\nu \geq 0)\,. \tag{3.3}$$

When F is linear with constant coefficient, that is $F(u) = Au$, $A = const.$, the resulting equation is the advection-diffusion equation.

Three types of time-stepping may be used : (1) the fully explicit schemes, (2) the semi-implicit schemes where the linear part $L(u)$ is implicit and the nonlinear part $N(u)$ is explicit, and (3) the fully implicit schemes whose solution requires either an iterative procedure or a time-linearization technique like

$$H(u^{n+1}) \cong H(u^n) + \Delta t \partial_t H(u^n) \cong H(u^n) - \partial_x \left[F'(u^n)\left(u^{n+1} - u^n\right)\right] + \nu \partial_{xx}\left(u^{n+1} - u^n\right) \tag{3.4}$$

where u^n is the approximation to $u(x,t)$ at time $t^n = n\,\Delta t$, $n = 0, 1, \ldots$. In this lecture, we shall discuss only the fully explicit and the semi-implicit schemes.

High-order accuracy can be obtained by considering schemes involving several time levels. Thus, the discrete equation furnishing the solution u^{n+1} contains the values u^{n+1-j}, with $j = 0, \ldots, k$. Such a method is called a "multistep method". Its truncation error is generally $O(\Delta t^k)$. Multistep methods, especially based on the Adams-Bashforth technique (explicit schemes) or on a combination of Backward-Differentiation Formula (BDF) and Adams-Bashforth discretization (explicit or semi-implicit schemes), will be discussed in Section 3.2.

Another way to construct high-order methods is to consider only two levels of time (n and $n+1$), the accuracy being obtained by dividing the time interval in a sequence of intermediate stages, each of one furnishing an intermediate value. Accuracy of s-th order is reached with s intermediate stages. This type of method is called a "one-step method", the main example being the Runge-Kutta schemes, which will make the object of Section 3.3.

The stability of multistep and one-step methods will be studied in Section 3.4. In case of an explicit method, it will be seen that the allowable time-step generally decreases when the number of time-levels of the Adams-Bashforth scheme increases. On the other hand, the contrary occurs for the Runge-Kutta scheme : the allowable time-step increases with the number of intermediate stages. However, in the same time, the volume of calculations increases. Therefore, a comparison between multistep and one-step methods must take this point into account.

Another important question arising when comparing the methods is connected to the required storage. In general, the storage required by a multistep method is larger than this needed by Runge-Kutta methods in their "low-storage" version.

Therefore, the choice between one-step and multistep methods resumes, as usual, to a competition between computing time and storage, the decision being strongly computer dependent.

For a more comprehensive study of time-discretization methods, we refer to classical books on the numerical solution of differential equations, for example : Henrici [55], Gear [56], Lambert [57], Lapidus and Seinfeld [58], Hairer *et al.* [59] and Hairer and Wanner [60].

3.2 Multistep method

3.2.1 Explicit methods

The commonly used high-order explicit schemes are the well known Adams-Bashforth schemes. In these schemes, the time-derivative $\partial_t u$ is approximated with a two-level finite-difference formula and the spatial term H is approximated with a linear combination of H evaluated at k time-levels, so that the resulting finite-difference equation has a truncation error of order Δt^k. Therefore, the general Adams-Bashforth (AB) scheme is of the form :

$$\frac{u^{n+1}-u^n}{\Delta t} = \sum_{j=0}^{k-1} b_j H(u^{n-j}) \tag{3.5}$$

Another explicit multistep scheme consists of approximating $\partial_t u$ with a high-order finite-difference formula involving several time levels and H with an extrapolation similar to the above one. Such a discretization is particularly convenient when some part of $H(u)$, especially $L(u)$, is considered in an implicit way because it needs to be considered at level $n+1$ only. This discretization (called "Backward Differentiation" or "Backward-Euler") will be discussed below in detail. For the moment, only the explicit scheme (noted AB/BDE) is considered, it is of the general form

$$\frac{1}{\Delta t}\sum_{j=0}^{k} a_j u^{n+1-j} = \sum_{j=0}^{k-1} b_j H(u^{n-j}) \,. \tag{3.6}$$

Therefore, scheme (3.5) enters in the general form (3.6) with $a_0 = 1$, $a_1 = -1$ and $a_j = 0$ for $j > 1$.

For the determination of the coefficients a_j and b_j, Taylor's expansions around $(n+1)\Delta t$ are performed, let

$$\sum_{j=0}^{k} a_j\, u^{n+1-j} = \sum_{l=0}^{L} \Delta t^l\, A_l\, \partial_t^l u + O(\Delta t^{L+1}) \tag{3.7}$$

where ∂_t^l is the l-th time-derivative of u and

$$A_0 = \sum_{j=0}^{k} a_j\,, \quad A_l = \frac{(-1)^l}{l!} \sum_{j=0}^{k} j^l a_j\,, \quad l > 0$$

and

$$\sum_{j=0}^{k-1} b_j\, H(u^{n-j}) = \sum_{l=0}^{L-1} \Delta t^l\, B_l\, \partial_t^l H + O(\Delta t^L) \tag{3.8}$$

with

$$B_0 = \sum_{j=0}^{k-1} b_j\,, \quad B_l = \frac{(-1)^l}{l!} \sum_{j=0}^{k-1} (j+1)^l\, b_j\,, \quad l > 0.$$

Bringing these expansions into (3.6) we obtain

$$\frac{A_0}{\Delta t} u + (A_1\, \partial_t u - B_0 H) + \sum_{l=2}^{k} \Delta t^{l-1}\, \partial_t^{l-1}\, (A_l \partial_t u - B_{l-1} H) + O(\Delta t^k) = 0\,.$$

For the AB schemes (3.5), we have $A_0 = 0$, $A_1 = 1$ and $A_l = (-1)^{l+1}/l!$ for $l > 1$. The consistency condition is $B_0 = A_1$, that is

$$\sum_{j=0}^{k-1} b_j = 1$$

and the conditions for k-th order accuracy are

$$B_{l-1} = A_l\,, \quad l = 2, ..., k\,.$$

For the AB/BDE schemes (3.6), the time-derivative $\partial_t u$ and the spatial term $H(u)$ are required to be separately approximated to the order k. Therefore, we get the consistency conditions $A_0 = 0$, $A_1 = 1$ and $B_0 = 1$, that is

$$\sum_{j=0}^{k} a_j = 0\,, \quad \sum_{j=0}^{k} j\, a_j = 1\,, \quad \sum_{j=0}^{k-1} b_j = 1$$

Then, the conditions for k-th order accuracy are

$$A_l = 0\,, \quad B_{l-1} = 0\,, \quad l = 2, ..., k\,.$$

For both family of schemes, the truncation error is

$$\Delta t^k\, (A_{k+1} - B_k)\, \partial_t^{k+1} u + O(\Delta t^{k+1})\,. \tag{3.9}$$

Table 3.1 gives the values of the coefficients for schemes up to the fourth-order.

Scheme	order	a_0	a_1	a_2	a_3	a_4	b_0	b_1	b_2	b_3
AB2	2	1	-1				$\frac{3}{2}$	$-\frac{1}{2}$		
AB/BDE2	2	$\frac{3}{2}$	-2	$\frac{1}{2}$			2	-1		
AB3	3	1	-1				$\frac{23}{12}$	$-\frac{16}{12}$	$\frac{5}{12}$	
AB/BDE3	3	$\frac{11}{6}$	-3	$\frac{3}{2}$	$-\frac{1}{3}$		3	-3	1	
AB4	4	1	-1				$\frac{55}{24}$	$-\frac{59}{24}$	$\frac{37}{24}$	$-\frac{9}{24}$
AB/BDE4	4	$\frac{25}{12}$	-4	3	$-\frac{4}{3}$	$\frac{1}{4}$	4	-6	4	-1

Table 3.1. Coefficients of the Adams-Bashforth (AB) and Adams-Bashforth/Backward Differentiation (AB/BDE) schemes

3.2.2 Semi-implicit methods

The constraint on the size of the time-step Δt due to stability requirements associated with a fully explicit scheme (especially restrictive for second-order derivatives) leads naturally to increase the degree of implicitness of a scheme. The semi-implicit methods apply generally to nonlinear equations like (3.1)-(3.2) where the coefficients of the linear operator are constant. This is the case, for example, of the Navier-Stokes equations for incompressible fluids with constant viscosity. This linear term is considered implicitly and the nonlinear term is explicit, so that the resulting discrete operator is time-independent and can be inverted or diagonalized in a preprocessing stage performed before to start the time-integration. Such a time discretization is of common use associated with Fourier and Chebyshev spatial approximations.

The most used second-order implicit schemes are the Adams-Bashforth/Crank-Nicolson (AB/CN) scheme

$$\frac{u^{n+1} - u^n}{\Delta t} = \frac{1}{2}\left[3N(u^n) - N(u^{n-1})\right] + \frac{1}{2}\left[L(u^{n+1}) + L(u^n)\right] \tag{3.10}$$

and the Adams-Bashforth/Backward-Differentiation (AB/BDI2) scheme

$$\frac{3u^{n+1} - 4u^n + u^{n-1}}{2\Delta t} = 2N(u^n) - N(u^{n-1}) + L(u^{n+1})\,, \tag{3.11}$$

also denoted AB2/2BE (Ouazzani *et al.* [61]).

Concerning the stability, the scheme (3.11) damps the high frequencies better than (3.10). For large viscosity the scheme (3.11) is more stable than (3.10) ; in particular, for $\nu/|A|$ larger than a critical value depending on the spatial approximation, the scheme

(3.11) is (linearly) unconditionally stable (see Ouazzani *et al.* [61] for the Chebyshev approximation). On the other hand, when the viscosity is small the scheme (3.10) is slightly more stable than (3.11). However, for zero viscosity, both schemes are (weakly) unstable (see Section 3.4). Concerning the accuracy, it can be seen that the truncation error of (3.11) applied to the diffusion equation is larger than the error of (3.10). However, when applied to the incompressible Navier-Stokes equations associated with the tau or collocation Chebyshev method, the scheme (3.11) has shown to be slightly more accurate than (3.10), as experienced by Vanel *et al.* [62] and by Ehrenstein and Peyret [63].

An advantage of the AB/BDI2 scheme is that it does not necessitate to calculate $L(u^n)$. Extensions of AB/CN scheme (3.10) to higher order necessitates to consider $L(u)$ at several time-levels, therefore it is more expensive in terms of computing time and storage.

For all these reasons, it is recommended to use high-order semi-implicit schemes of type (3.11), that is schemes similar to (3.6) with addition of the implicit evaluation of $L(u)$. These schemes, (belonging to the family introduced by Crouzeix [64]), noted here AB/BDI, are

$$\frac{1}{\Delta t}\sum_{j=0}^{k} a_j u^{n+1-j} = \sum_{j=0}^{k-1} b_j N(u^{n-j}) + L(u^{n+1}) \tag{3.12}$$

with coefficients a_j and b_j given in Table 3.1. The truncation error is again given by the expression (3.9). The third-order (AB/BDI3) scheme of the family (3.12) has been applied to the solution of the Navier-Stokes equations associated with various spatial approximations : finite-element methods (Baker *et al.* [65]), Chebyshev tau method (Le Quéré [66]), Chebyshev collocation method (Botella [67], Botella and Peyret [68]) or spectral element method (Karniadakis *et al.* [69]).

3.2.3 Starting schemes

With a multistep scheme, there exists a difficulty for starting the solution since the only known value is $u^0 = u(t = 0)$ while the equation (3.6) needs the knowledge of $u^1,...,u^k$ for the calculation of u^{k+1}. The usual way to remove this difficulty is to use, at the first time-cycles, an one-step scheme like the Runge-Kutta scheme described in next Section. From the practical point of view, it is interesting to note that the starting scheme may be of one order accuracy less than the general scheme. This comes from classical error estimates (Gear [56]) showing that the approximation error $e(t) = O(\Delta t^k)$ becomes $O(\Delta t^{k+1})$ when t is close to zero, that is $t = m\Delta t$, with m sufficiently small.

One can be easily convinced of this fact by considering the simple problem

$$d_t u = \lambda u \quad (\lambda = \text{negative constant})$$
$$u(0) = u_0$$

whose solution is $u(t) = u_0 e^{\lambda t}$. The equation is approximated with the first-order Euler scheme

$$\frac{u^{n+1} - u^n}{\Delta t} = \lambda u^{n+1}, \quad u^0 = u_0$$

whose solution is

$$u^n = u_0 (1 - \lambda \Delta t)^{-n} .$$

Now, if $t = n\Delta t$ is fixed ($t \leq T$), and $\Delta t \to 0$, we easily find

$$e(t) = u^n - u(t) = \frac{u_0}{2} \lambda \Delta t e^{\lambda t} t + O(e^{\lambda t} t^2 \Delta t^2) .$$

Therefore, the error is of first-order but it becomes of second-order if $t = m\Delta t$, with m sufficiently small.

An example of starting scheme used in association with a third-order multistep scheme is the second-order Runge-Kutta/Crank-Nicolson family described in Section 3.3. In particular, the scheme

$$\begin{aligned} \frac{u_1 - u^n}{\Delta t} &= N(u^n) + L(u^n) \\ \frac{2u^{n+1} - u_1 - u^n}{2\Delta t} &= \frac{1}{2} \left[N(u_1) + L(u^{n+1}) \right] \end{aligned} \tag{3.13}$$

has been used by Botella [26] for the incompressible Navier-Stokes equations discretized in time with the scheme AB/BDI3 and in space with a Chebyshev collocation approximation.

3.3 One-step methods : Runge-Kutta schemes

3.3.1 General explicit Runge-Kutta schemes

In the one-step method of Runge-Kutta type, the calculation of the solution at level $n+1$ involves only the solution at level n. The accuracy is obtained through the calculation of intermediate values at s stages. In general, the order accuracy k of the scheme is equal to the number of stages s. But, for various reasons (e.g. low-storage, stability), it may be useful to consider one more stage than necessary to get the accuracy.

Considering the general equation (3.1) the general s-stage Runge-Kutta scheme is

$$\begin{aligned} K_1 &= H(u^n) \\ K_i &= H\left(u^n + \Delta t \sum_{j=1}^{i-1} a_{i,j} K_j \right), \quad i = 2, ..., s \\ u^{n+1} &= u^n + \Delta t \sum_{j=1}^{s} b_j K_j . \end{aligned} \tag{3.14}$$

The sets of coefficients $a_{i,j}$ and b_j characterize the scheme. When H depends explicitly on time, that is $H = H(t, u)$, the quantity K_i is defined by evaluating H at time $t^n + c_i \Delta t$. These coefficients c_i which characterize the intermediate time-levels satisfy the consistency conditions

$$c_i = \sum_{j=1}^{i-1} a_{i,j} \,, \quad i = 2, ..., s \tag{3.15}$$

with $c_1 = 0$. It is convenient to preserve the coefficients c_i even if H does not explicitly depend on t.

Another way to write the general Runge-Kutta scheme (3.14) is

$$\begin{aligned} u_0 &= u^n \\ u_i &= u_0 + \Delta t \sum_{j=0}^{i-1} a_{i+1,j+1} H(u_j) \,, \quad i = 1, ..., s-1 \\ u^{n+1} &= u_0 + \Delta t \sum_{j=0}^{s-1} b_{j+1} H(u_j) \,. \end{aligned} \tag{3.16}$$

Finally, a third form of Runge-Kutta schemes may be found in the literature, it is

$$\begin{aligned} u_0 &= u^n \\ u_i &= u_{i-1} + \Delta t \sum_{j=0}^{i-1} \alpha_{i,j} H(u_j) \,, \quad i = 1, ..., s-1 \\ u^{n+1} &= u_{s-1} + \Delta t \sum_{j=0}^{s-1} \beta_j H(u_j) \end{aligned} \tag{3.17}$$

with

$$\alpha_{i,j} = a_{i+1,j+1} - a_{i,j+1} \,, \quad i = 2, ..., s-1; \;\; j = 0, ..., i-2 \,,$$

$$\alpha_{i,j} = a_{i+1,j+1} \,, \quad i = 1, ..., s-1 \,, \quad j = i-1 \,,$$

$$\beta_j = b_{j+1} - a_{s,j+1} \,, \quad j = 0, ..., s-2 \,,$$

$$\beta_{s-1} = b_s \,.$$

The truncation error is determined, as usual, from Taylor's expansion. The consistency (1st-order accuracy) requires

$$\sum_{i=1}^{s} b_i = 1 \,. \tag{3.18}$$

The Runge-Kutta scheme is second-order accurate if

$$\sum_{i=1}^{s} b_i c_i = \frac{1}{2}\,. \tag{3.19}$$

For third-order accuracy the following conditions have to be added :

$$\frac{1}{2}\sum_{i=1}^{s} b_i c_i^2 = \frac{1}{6}\,, \quad \sum_{i,j=1}^{s} b_i a_{i,j} c_j = \frac{1}{6}\,. \tag{3.20}$$

In case of a 3-stage scheme ($s = 3$), the conditions (3.18)-(3.20) give a system of 4 equations for 6 coefficients. Therefore, two of them can be chosen according to various criteria : accuracy, low-storage (Williamson [70]) or TVD (Shu and Osher [71], Shu [38], Gottlieb and Shu [72]). For example, the optimal third-order TVD scheme, often used with ENO method, is obtained for

$$a_{2,1} = 1\,, \quad a_{3,1} = a_{3,2} = \frac{1}{4}\,, \quad b_1 = b_2 = \frac{1}{6}\,, \quad b_3 = \frac{2}{3}\,.$$

Note that this scheme may be written under the form

$$\begin{aligned} u_1 &= u^n + \Delta t H(u^n) \\ u_2 &= \frac{3}{4}u^n + \frac{1}{4}u_1 + \frac{1}{4}\Delta t H(u_1) \\ u^{n+1} &= \frac{1}{3}u^n + \frac{2}{3}u_2 + \frac{2}{3}\Delta t H(u_2) \end{aligned} \tag{3.21}$$

which presents the advantage to necessitate only one evaluation of H at each stage. Such a scheme, however, is not "low-storage" in the sense which will be discussed in next Section.

Fourth-order accuracy requires, besides (3.18)-(3.20), the following equations to be satisfied :

$$\begin{aligned} &\frac{1}{6}\sum_{i=1}^{s} b_i\, c_i^3 = \frac{1}{24}\,, \quad \sum_{i,j=1}^{s} b_i\, c_i\, a_{i,j}\, c_j = \frac{1}{8}\,, \\ &\frac{1}{2}\sum_{i,j=1}^{s} b_i\, a_{i,j}\, c_j^2 = \frac{1}{24}\,, \quad \sum_{i,j,k=1}^{s} b_i\, a_{i,j}\, a_{j,k}\, c_k = \frac{1}{24}\,. \end{aligned} \tag{3.22}$$

The classic four-stage fourth-order Runge-Kutta scheme is defined by the following parameters :

$$\begin{aligned} &a_{1,1} = 0\,, \quad a_{2,1} = \frac{1}{2}\,, \quad a_{3,1} = 0\,, \quad a_{3,2} = \frac{1}{2}\,, \quad a_{4,1} = a_{4,2} = 0\,, \quad a_{4,3} = 1 \\ &b_1 = \frac{1}{6}\,, \quad b_2 = b_3 = \frac{1}{3}\,, \quad b_4 = \frac{1}{6} \end{aligned} \tag{3.23}$$

The conditions (3.22) as well as those ensuring fifth-order accuracy are given by Carpenter and Kennedy [73].

For further purpose it is interesting to write down the Taylor expansion to the fifth-order for the case where H is linear (with constant coefficients) :

$$u^{n+1} = u^n + \Delta t \left(\sum_{i=1}^{s} b_i\right) H(u^n) \ +\Delta t^2 \left(\sum_{i=1}^{s} b_i c_i\right) H^2(u^n) + \Delta t^3 \left(\sum_{i,j=1}^{s} b_i a_{i,j} c_j\right) H^3(u^n)$$
$$+\Delta t^4 \left(\sum_{i,j,k=1}^{s} b_i a_{i,j} a_{j,k} c_k\right) H^4(u^n) + O(\Delta t^5) \tag{3.24}$$

which is useful for the analysis of stability.

To close this Section, we mention the loss of accuracy in time for first-order hyperbolic equations when the boundary conditions are time-dependent, for example $u(0,t) = g(t)$. It is shown by Carpenter *et al.* [74] that the conventional method of imposing boundary conditions, that is $u_i(0,t) = g(t + c_i\Delta t)$ at stage i in Eq.(3.16), reduces the truncation error near the boundary to first-order, leading to a global accuracy of second-order only, whatever the order of the considered Runge-Kutta scheme. Procedures for reducing this loss of accuracy is given by the quoted authors.

3.3.2 Low-storage explicit Runge-Kutta schemes

One property that can be required to a time-discretization scheme is to necessitate the less storage possible. This requirement may be primordial especially for turbulent flow calculations which generally need a very large number of degrees of freedom. Therefore, a significant effort is continuing to be devoted to reduce the storage needed by the Runge-Kutta schemes.

Assuming the spatial discretization of $H(u)$ to involve N degrees of freedom (modes or grid values). A Runge-Kutta scheme requiring 2N memory locations is called a 2N-storage scheme, 3N-stage scheme if it requires 3N locations ... Note that 2N is the storage required by the usual first-order Euler scheme.

Williamson [70] has shown that all second-order schemes can be cast in 2N-storage form. On the other hand, only some three-stage third-order schemes can be cast into a 2N-storage version. For four-stage fourth-order schemes it has been shown (Fyfe [75]) that all of them can be 3N-storage but only very special schemes (not commonly used) can be cast in the 2N-storage form.

In the present Section, after a presentation of the general 2N-storage scheme, we shall present the three-stage third-order scheme proposed by Williamson [70] and which is broadly used in high-order CFD. Then, we shall discuss the four-stage, third-order, 2N-storage scheme proposed recently by Carpenter and Kennedy [73], in which the additional stage is used to improve the stability properties. Finally, fourth-order 3N- and 2N-storage schemes will be described.

a) General 2N-storage scheme

The algorithm is of the form

$$
\begin{aligned}
u_0 &= u^n \\
Q_j &= A_j\, Q_{j-1} + \Delta t H(u_{j-1}) \\
u_j &= u_{j-1} + B_j\, Q_j\,, \quad j = 1, \ldots, s \\
u^{n+1} &= u_s\,,
\end{aligned}
\tag{3.25}
$$

with $A_1 = 0$. Williamson [70] gives the expression of A_j and B_j in terms of the coefficients of the general Runge-Kutta scheme (3.14). This can be done only if these coefficients have certain ratios to each other. This is the reason why all the Runge-Kutta schemes cannot be put in the 2N-storage form. The expressions are :

$$
\begin{aligned}
B_j &= a_{j+1,j}\,, \quad j \neq s \\
B_s &= b_s \\
A_j &= \frac{b_{j-1} - B_{j-1}}{b_j}\,, \quad j \neq 1\,,\; b_j \neq 0 \\
A_j &= \frac{a_{j+1,j-1} - c_j}{B_j}\,, \quad j \neq 1\,,\; b_j = 0\,.
\end{aligned}
\tag{3.26}
$$

b) Third-order schemes

The three-stage third-order scheme proposed by Williamson [70] is defined by the values

$$
\begin{aligned}
&A_1 = 0\,, \quad A_2 = -\frac{5}{9}\,, \quad A_3 = -\frac{153}{128} \\
&B_1 = \frac{1}{3}\,, \quad B_2 = \frac{15}{16}\,, \quad B_3 = \frac{8}{15}
\end{aligned}
\tag{3.27}
$$

and can be read as :

$$
\begin{aligned}
u_0 &= u^n \\
Q_1 &= \Delta t H(u_0) \\
u_1 &= u_0 + \frac{1}{3} Q_1 \\
Q_2 &= -\frac{5}{9} Q_1 + \Delta t H(u_1) \\
u_2 &= u_1 + \frac{15}{16} Q_2 \\
Q_3 &= -\frac{153}{128} Q_2 + \Delta t H(u_2) \\
u_3 &= u_2 + \frac{8}{15} Q_3 \\
u^{n+1} &= u_3\,.
\end{aligned}
\tag{3.28}
$$

This scheme is often used in association with spectral approximations for solving the compressible Navier-Stokes equations (see e.g. Guillard *et al.* [76]).

It will be shown in Section 3.4 that the allowable time-step required for (linear) stability increases with the order of accuracy. Taking advantage of this property, Carpenter and Kennedy [73] have devised a 2N-storage, four-stage, third-order accurate scheme possessing the same stability constraint that the four-stage, fourth-order scheme. More precisely, thanks to supplementary coefficients introduced by the additional stage, it is possible to have an amplification factor of fourth-order type, that is the amplification factor associated with (3.24). Incidentally, it is interesting to observe that the above property makes the considered third-order scheme of fourth-order accuracy in the linear case.

The requirement on the amplification factor is not sufficient to determine all the coefficients. Therefore, the remaining freedom is exploited by imposing the coefficients to be rational and to minimize the main part of the truncation error.

The four-stage third-order scheme constructed on the above described requirements is defined by the set :

$$\begin{aligned} A_1 &= 0\,, & A_2 &= -\frac{205}{243}\,, & A_3 &= -\frac{243}{38}\,, & A_4 &= -\frac{2}{9}\,, \\ B_1 &= \frac{19}{36}\,, & B_2 &= \frac{27}{19}\,, & B_3 &= \frac{2}{9}\,, & B_4 &= \frac{1}{4}\,. \end{aligned} \tag{3.29}$$

c) Fourth-order scheme

The classic four-stage scheme defined by the coefficients (3.23) can be written (Blum [77]) in the following 3N-storage form

$$\begin{aligned} &u_0 = u^n \\ &P_0 = 0 \\ &Q_0 = u^n \\ &P_j = A_j\, P_{j-1} + B_j\, Q_{j-1} \\ &Q_j = C_j\, Q_{j-1} + H(u_{j-1}) \\ &u_j = u_{j-1} + \Delta t\,(D_j\, P_j + E_j\, Q_j) \qquad j = 1, ..., s = 4 \\ &u^{n+1} = u_s \end{aligned} \tag{3.30}$$

with coefficients given in Table 3.2

Like for the third-order scheme, Carpenter and Kennedy [73] have considered fourth-order schemes with five-stage. By increasing the number of stages, that is the number of arbitrary coefficients, they constructed a scheme according to the following properties: (1) 2N-storage, (2) large domain of stability, (3) small truncation error.

j	A_j	B_j	C_j	D_j	E_j
1	0	1	0	0	$\frac{1}{2}$
2	0	1	0	$-\frac{1}{2}$	$\frac{1}{2}$
3	$\frac{1}{6}$	0	$-\frac{1}{2}$	0	1
4	1	-1	2	1	$\frac{1}{6}$

Table 3.2. Coefficients of the classic four-stage fourth-order 3N-storage Runge-Kutta scheme

The values of the coefficients A_j and B_j of equations (3.25)-(3.26) are

$$\begin{array}{ll} A_1 = 0 & B_1 = 0.1028639988105 \\ A_2 = -0.4801594388478 & B_2 = 0.7408540575767 \\ A_3 = -1.4042471952 & B_3 = 0.7426530946684 \\ A_4 = -2.016477077503 & B_4 = 0.4694937902358 \\ A_5 = -1.056444269767 & B_5 = 0.1881733382888 \end{array} \tag{3.31}$$

A rational form of these coefficients is given by Carpenter and Kennedy [73].

The above scheme has been used for the study of the interaction shock-vortex by means of the solution of the compressible Euler equations with ENO method (Erlebacher *et al.* [78]). These authors report that this scheme is more efficient than Williamson's third-order 2N-storage scheme (3.28) because of the larger allowable time-step (higher by a factor of 1.9). The 2N-storage third-order and fourth-order schemes respectively defined by (3.29) and (3.31) were recently applied (Wilson *et al.* [79]) to the incompressible Navier-Stokes equations in association with fourth-order and sixth-order Hermitian finite-difference approximations.

3.3.3 Semi-implicit Runge-Kutta schemes

As already explained, the constraint on the time-step due to the explicit nature of a scheme can be reduced by considering a semi-implicit time-discretization. The viscous terms $L(u)$ (generally linear) which lead to a very restrictive time-step (for moderate viscosity) are treated in an implicit way. The convective (generally nonlinear) terms $N(u)$ are evaluated explicitly.

In this Section, we present a family of two-stage second-order schemes and, then, a 2N-storage three-stage scheme whose accuracy is third-order for the explicit term and second-order for the implicit one.