$$F_{j+1/2} = \sum_{i=1}^{p} \alpha_i^p \left(F^*(U_j, U_{j+i}) + \ldots + F^*(U_{j-i+1}, U_{j+1}) \right),$$

where α_i^p are coefficients. It follows

$$F_{j+1/2} - F_{j-1/2} = \sum_{i=1}^{p} \alpha_i^p \left(F^*(U_j, U_{j+i}) - F^*(U_{j-i}, U_j) \right).$$

We are now ready to state the answer to our question [19].

Theorem 8.16. *Consider a hyperbolic conservation law, solved using a semi-discrete scheme on conservation form. Provided*

$$2\sum_{i=1}^{p} i\alpha_i^p = 1, \ \sum_{i=1}^{p} i^{2s-1}\alpha_i^p = 0, \ s \in [2, \ldots, p],$$

the numerical flux

$$F_{j+1/2} = \sum_{i=1}^{p} \alpha_i^p \left(F^*(U_j, U_{j+i}) + \ldots + F^*(U_{j-i+1}, U_{j+1}) \right),$$

is consistent and the resulting scheme is of order $\mathcal{O}(h^{2p})$. Furthermore, if the numerical entropy flux is defined as

$$\Psi_{j+1/2} = \sum_{i=1}^{p} \alpha_i^p \left(\Psi^*(U_j, U_{j+i}) + \ldots + \Psi^*(U_{j-i+1}, U_{j+1}) \right),$$

with $\Psi^*(U_j, U_{j+i}) = U_j F^*(U_j, U_{j+i})$, the scheme is entropy conservative.

Proof. Let us first discuss the accuracy. If we assume that $u = U_j$ is smooth, we have

$$F^*(U_j, U_{j+i}) - F^*(U_{j-i}, U_j) = 2\sum_{l=0}^{p} \frac{(ih)^{2l+1}}{(2l+1)!} f^{(2l+1)}(U_j) + \mathcal{O}(h^{2l+2}),$$

such that

$$\frac{F_{j+1/2} - F_{j-1/2}}{h} = \frac{2}{h}\sum_{i=1}^{p} \alpha_i^p \sum_{l=0}^{p} \frac{(ih)^{2l+1}}{(2l+1)!} f^{(2l+1)}(U_j) + \mathcal{O}(h^{2p+2}).$$

Consistency requires that

$$\frac{2}{h}\sum_{i=0}^{p} \alpha_i^p ihf^{(1)}(U_j) = \frac{\partial f}{\partial x}(U_j) \ \Rightarrow \ 2\sum_{i=1}^{p} i\alpha_i^p = 1.$$

For the remaining terms to vanish we must require

$$\sum_{i=1}^{p} i^{2s-1}\alpha_i^p = 0, \ s = 2, \ldots, p.$$

Consider

problem we can consider

$$\frac{d}{dt}\begin{bmatrix} U \\ t \end{bmatrix} = \begin{bmatrix} L(U(t)) \\ 1 \end{bmatrix}, \qquad \begin{bmatrix} U(0) \\ t \end{bmatrix} = \begin{bmatrix} U_0 \\ 0 \end{bmatrix}.$$

Hence, any system with an explicit dependence on t can be written as an augmented autonomous system.

The coefficients of the classic Runge–Kutta method, expressed in the form of the Butcher tableau, are

$$\begin{array}{c|c} c & \mathrm{A} \\ \hline & b^T \end{array}, \quad \mathrm{A}_{ij} = a_{ij}, \ c_i = c_i, \ b_i = b_i, \ i,j = 1\ldots,s,$$

and are found by enforcing the order conditions. The details are classic and can be found in, e.g., [3]. The Runge–Kutta method is explicit if A is strictly lower triangular, and diagonally implicit if A is lower triangular. Otherwise one recovers a fully implicit Runge–Kutta method. For the explicit case, it is worth recalling that an s-stage explicit method can be at most of order s, while the order of an s-stage diagonally implicit Runge–Kutta method cannot exceed $s+1$; see [3].

A Runge–Kutta methods in its classic form is defined uniquely through the Butcher tableau unless the method is reducible. In the current context, a method is reducible if $a_{ij} = b_i = 0$ for some i and all j. In such a case, this stage can be eliminated without impacting the scheme. In what remains, we assume that all methods are irreducible.

Although the classic Runge–Kutta methods have proven themselves of substantial value and power for a broad range of problems, they present a challenge in the current context. Looking back at what we have discussed so far, all schemes are given as one-step schemes:

$$U^{n+1} = \mathrm{G}(U^n).$$

For this case, we have a solid theoretical understanding of stability and convergence. However, the classic Runge–Kutta methods are not in this form, suggesting that we need to revisit questions of stability and convergence.

The solution to this problem is to return to the basic approximation in (9.4) and the prediction of the intermediate stage values. The classic approach is powerful, yet it is clearly not unique and we can seek alternatives. In the current context, it is of particular interest to pursue the development of schemes of the form [11]

$$\begin{cases} U^{(0)} = U^n, \\ U^{(i)} = \sum_{j=1}^{s} \left(\alpha_{ij} U^{(j)} + k\beta_{ij} L(U^{(j)}) \right), \ i = 1\ldots s+1:, \\ U^{n+1} = U^{(s+1)}, \end{cases} \tag{9.5}$$

which is known as the Shu–Osher form of the Runge–Kutta method. In the following, we shall separate the discussion into explicit and implicit scheme.

9.1.1 ▪ Explicit strong stability preserving (SSP) Runge–Kutta schemes

For explicit schemes, the central result is as follows [11].

Proof. First of all we notice that

$$i = 1, \ldots, s+1 : \ \sum_{j=1}^{i} \alpha_{ij} = 1, \tag{9.6}$$

to ensure consistency. Next, consider

$$\begin{aligned} \|U^{(i)}\|_{TV} &= \left\| \sum_{j=1}^{i} \left(\alpha_{ij} U^{(j)} + k\beta_{ij} L(U^{(j)}) \right) \right\|_{TV} = \sum_{j=1}^{i} \left\| \alpha_{ij} \left(U^{(j)} + k\frac{\beta_{ij}}{\alpha_{ij}} L(U^{(j)}) \right) \right\|_{TV} \\ &\leq \sum_{j=1}^{i} \alpha_{ij} \left\| \left(U^{(j)} + k\frac{\beta_{ij}}{\alpha_{ij}} L(U^{(j)}) \right) \right\|_{TV} \end{aligned}$$

where we assume that α_{ij} is positive. Since

$$\left\| \left(U^{(j)} + k\frac{\beta_{ij}}{\alpha_{ij}} L(U^{(j)}) \right) \right\|_{TV} \leq \|U^{(j)}\|_{TV},$$

by TVD-stability of the forward Euler step provided $k\frac{\beta_{ij}}{\alpha_{ij}} \leq k_{FE}$, we immediately recover that $\|U^{(i)}\|_{TV} \leq \|U^n\|_{TV}$ by (9.6). This further implies

$$\|U^{n+1}\|_{TV} \leq \sum_{j=1}^{s+1} \alpha_{ij} \|U^{(j)}\|_{TV} \leq \sum_{j=1}^{s+1} \alpha_{ij} \|U^n\|_{TV} \leq \|U^n\|_{TV},$$

hence completing the proof. □

Hence, if we can identify methods in which α_{ij} and β_{ij} are semi-positive, i.e., the scheme is constructed as a convex combination of forward Euler schemes, and chosen to retain high-order accuracy of the scheme, we overcome the need for additional analysis as stability follows from the stability of the forward Euler scheme. Schemes with such a property are called Strong Stability Preserving (SSP) schemes. We refer to [6, 4] for comprehensive reviews of such methods beyond what we discuss here.

Searching for α_{ij} and β_{ij} we recall the requirement that

$$k\frac{\beta_{ij}}{\alpha_{ij}} \leq k_{FE} \ \Rightarrow k \leq \frac{\alpha_{ij}}{\beta_{ij}} k_{FE} = C(\alpha, \beta) k_{FE}.$$

Clearly, we should seek coefficients that maximize the SSP coefficient $C(\alpha, \beta)$ defined as

$$C(\alpha, \beta) = \min_{ij} \frac{\alpha_{ij}}{\beta_{ij}}.$$

Some of the complications associated with this optimization are illustrated in the following example [4].

Example 9.4. Consider the standard, second order accurate, explicit 2-stage Runge-Kutta scheme

$$U^{(1)} = U^n + kL(U^n), \ \ U^{n+1} = U^n + \frac{k}{2}\left(L(U^n) + L(U^{(1)}) \right),$$

$$\frac{d^n u(0)}{dt^n} = \begin{cases} 0 & n < l \\ (l-1)! & n \geq l \end{cases},$$

such that

$$u(k) = \sum_{n=0}^{\infty} \frac{(l-1)!}{(l+n)!} k^{l+n}.$$

Matching the coefficients yields the order conditions

$$b^T \mathrm{A}^n \mathrm{C}^{l-1} e = \frac{(l-1)!}{(l+n)!}, \quad 1 \leq l+n \leq p,$$

for a scheme with a local truncation error of k^{p+1}.

The conditions derived in this way are both necessary and sufficient for $p \leq 3$ but only necessary for $p > 3$. To derive the complete set of order conditions for Runge-Kutta methods of arbitrary order, one must use the theory of rooted trees [3].

Let us also consider the stage order of the scheme, given as the order of accuracy of the solution at the individual stage levels at which the local solution is an approximation to $u(c_i k)$. Utilizing Taylor expansions and matching terms as above, we recover the condition for stage order $\tilde{p}$ as

$$\mathrm{A}\mathrm{C}^{l-1} e = \frac{1}{l} \mathrm{C}^l e, \quad l = 1, \ldots, \tilde{p}, \tag{9.10}$$

Note that for $l = 1$, we recover the basis consistency requirement

$$\mathrm{A}e = c, \tag{9.11}$$

It is worth observing that since the second stage of an explicit Runge-Kutta method is a forward Euler step, the stage order can not exceed one for any explicit scheme. Similarly, for a diagonally implicit Runge-Kutta method, the first stage is a backward Euler step, again restricting the stage order to one. Only by considering fully implicit Runge-Kutta methods or specially designed explicit schemes, avoiding this second stage, can this restriction be overcome. ∎

As we have already seen in Ex. 9.4, the development of SSP schemes is not unique, which make a simple connection to classic Runge-Kutta methods complicated. Let us begin by expressing the classic Runge-Kutta method (9.4) for the scalar problem in the form

$$U = eU^n + kQL, \tag{9.12}$$

where $U = [U^{(1)}, \ldots, U^{(s)}, U^{n+1}]^T, L = [L(U^{(1)}), \ldots, L(U^{(s)}), L(U^{n+1})]^T$, e is an $s+1$ one-vector and

$$\mathrm{Q} = \begin{bmatrix} \mathrm{A} & 0 \\ b^T & 0 \end{bmatrix},$$

contains the Runge-Kutta coefficients on Butcher form.

To relate these to the SSP coefficients, let us consider the modified Shu-Osher form

$$\begin{cases} U^{(i)} = v_i U^n + \sum_{j=1}^{s} \left(\alpha_{ij} U^{(j)} + k\beta_{ij} L(U^{(j)}) \right), & i = 1 \ldots s \\ U^{n+1} = U^{(s+1)} \end{cases}.$$

Consistency requires

$$i = 1 \ldots s+1 : v_i + \sum_{j=1}^{s} \alpha_{ij} = 1 \;\Rightarrow\; v + \alpha e = e, \tag{9.13}$$

where α is $(s+1) \times s$ matrix of coefficients. Expressing the modified scheme in compact form yields

$$U = v U^n + \alpha U + k\beta L, \tag{9.14}$$

where the $(s+1) \times (s+1)$ matrices α and β are given as

$$\alpha = \begin{bmatrix} \alpha & 0 \end{bmatrix}, \beta = \begin{bmatrix} \beta & 0 \end{bmatrix},$$

and α and β are $(s+1) \times s$ matrices of SSP coefficients.

Solving for U in (9.14) and using (9.13) yields

$$U = e U^n + k(\mathrm{I} - \alpha)^{-1} \beta L,$$

from which, by comparing with (9.12), we recover the connection between the two schemes as

$$(\mathrm{I} - \alpha)^{-1} \beta = \mathrm{Q}, \tag{9.15}$$

provided that $(\mathrm{I} - \alpha)$ is invertible. However, by considering the simple problem of $u' = 0$ in (9.14) it is easily realized that this is a natural condition for all practical methods. Hence, any SSP method can be expressed in terms of its Butcher tableaux through (9.15).

When trying to establish the reverse connection we should expect to encounter problems based on Ex. 9.4. Let us therefore restrict attention to SSP schemes with a given SSP coefficient r, and assume that all coefficients have the same SSP coefficient, i.e., $\alpha = r\beta$. In this case (9.15) yields a canonical SSP scheme with

$$\beta = \mathrm{Q}(\mathrm{I} + r\mathrm{Q})^{-1}, \;\; \alpha = r\beta, \;\; v = (\mathrm{I} - \alpha)e.$$

Clearly, we must choose r such that α_{ij} and β_{ij} are all semi-positive. This has a number of important consequences. First of all, by choosing $r = 0$ we immediately see that for a Runge-Kutta method to be SSP, the Butcher coefficients A_{ij} and b_i must all be non-negative, since $\mathrm{Q} \geq 0$ in an elementwise manner.

If we now assume that $0 < r \ll 1$, we can express

$$\beta = \sum_{k=0}^{\infty} (-1)^k r^k \mathrm{Q}^{k+1} = \mathrm{Q} - r\mathrm{Q}^2 + r^2\mathrm{Q}^3 + \ldots \geq 0.$$

If $b_j = 0$, then $\mathrm{Q}_{s+1,j} = 0$ which, for small r, requires that $(\mathrm{Q}^2)_{s+1,j} = 0$ to ensure positivity. However,

$$(\mathrm{Q}^2)_{s+1,j} = \sum_{i=1}^{s+1} b_i a_{ij} = 0.$$

Since a_{ij} and b_i are all semi-positive, this requires that either b_i are all zero or a_{ij} are all zero. In either case, the Runge-Kutta method is reducible. Hence, we have established the result [8]

Theorem 9.7. *Any irreducible Runge-Kutta method with a positive SSP coefficient has* $A \geq 0$ *and* $\boldsymbol{b} > 0$.

This result has a number of significant implications [8].

Theorem 9.8. *Any irreducible Runge-Kutta method with* $A \geq 0$ *must have stage-order* $\tilde{p} \leq 2$.

Proof. Recalling the stage order conditions (9.10) and (9.11), we have

$$i = 1 \ldots s \text{+}\cancel{1}\text{:} \quad \sum_{j=1}^{s} a_{ij} c_j = \frac{c_i^2}{2}, \quad \sum_{j=1}^{s} a_{ij} = c_i,$$

respectively. Let us assume that c_i are ordered such that $0 \leq c_1 \leq \ldots \leq c_s$. Since A is semi-positive, c_i are all semi-positive. Multiply the second order condition with c_i and subtract the first to recover

$$\sum_{j=1}^{s} a_{ij}(c_i - c_j) = \frac{c_i^2}{2}.$$

If we consider $i = 1$, it is clear that the only solution is $a_{1j} = c_1 = 0$, as in an explicit method. In this case, the local stage error is 2.

If we now seek an additional order of accuracy, we have the additional constraint

$$\sum_{j=1}^{s} a_{ij} c_j^2 = \frac{c_i^3}{3}.$$

Proceeding as above, we select the smallest $c_i > 0$ and recover

$$\sum_{j=1}^{s} a_{ij}(c_i^2 - c_j^2) = \frac{2c_i^3}{3}.$$

Since $0 < c_i \leq c_j$ this is not possible with $a_{ij} \geq 0$. Hence, the stage satisfying this condition requires that $a_{ij} = 0$, in which case the method is reducible. This implies that the stage-order of the Runge-Kutta method can not exceed 2. □

This has a further consequence [8].

Theorem 9.9. *For any Runge-Kutta method of order* p *with* $\boldsymbol{b} > 0$, *it holds that*

$$\tilde{p} \geq \lfloor \frac{p-1}{2} \rfloor,$$

where $\tilde{p}$ *is the stage order.*

Proof. If we define the stage error as

Table 11.1. *Coefficients to recover maximal linear accuracy of $\mathcal{O}(h^{2m-1})$ by a convex combination of m stencils of size m*

m	r_0	$d_{-1}^{r_0}$	$d_0^{r_0}$	$d_1^{r_0}$	$d_2^{r_0}$	$d_3^{r_0}$
1	-1	1.0000				
	0		1.0000			
2	-1	0.3333	0.6667			
	0		0.6667	0.3333		
3	-1	0.1000	0.6000	0.3000		
	0		0.3000	0.6000	0.1000	
4	-1	0.0286	0.3429	0.5143	0.1143	
	0		0.1143	0.5143	0.3429	0.0286

```
% Purpose: Compute linear weights for maximum accuracy 2m-1,
% using stencil shifted $r_0=-1,0$ points upwind.
A = zeros(m,m); b = zeros(m,1);

% Setup linear system for coefficients
for i=1:m
  col = ReconstructWeights(m,i-1+r0);
  A(1:(m+1-i),i) = col(i:m)';
end

% Setup righthand side for maximum accuracy and solve
crhs = ReconstructWeights(2*m-1,m-1+r0);
b = crhs(m:(2*m-1))'; d = A\b;
return
```

For consistency (11.4) we expect that

$$\sum_{r=r_0}^{m-1+r_0} d_r^{r_0} = 1.$$

Table 11.1 shows the computed coefficients for a few values of m. We note that all values of $d_r^{r_0}$ are positive, i.e., $v_{j+1/2}$ is computed as a convex combination of $v_{j+1/2}^{(r)}$. Furthermore, we observe that

$$d_r^{-1} = d_{m-2-r}^0$$

as a consequence of the symmetry of the stencils.

Finally, by symmetry we recover that

$$v(x_{j-1/2}) \simeq v_{j-1/2} = \sum_{r=r_0}^{m-1+r_0} d_{m-1-r}^{r_0} v_{j-1/2}^{(r)},$$

as an $\mathcal{O}(h^{2m-1})$ approximation to $v(x_{j-1/2})$.

since

$$\sum_{r=r_0}^{m-1+r_0} (\omega_r^{r_0} - d_r^{r_0}) v_{j+1/2}^{(r)} = \sum_{r=r_0}^{m-1+r_0} (\omega_r^{r_0} - d_r^{r_0}) \left[v_{j+1/2}^{(r)} - v(x_{j+1/2}) \right] = \mathcal{O}(h^{2m-1}).$$

Here we have used the consistency relations

$$\sum_{r=r_0}^{m-1+r_0} \omega_r^{r_0} = \sum_{r=r_0}^{m-1+r_0} d_r^{r_0} = 1.$$

Clearly, the path to success lies in a careful definition of $\omega_r^{r_0}$.

Following [17], let us consider the following choice

$$\omega_r^{r_0} = \frac{\alpha_r}{\sum_{s=0}^{m-1} \alpha_s}, \quad \alpha_r = \frac{d_r^{r_0}}{(\varepsilon + \beta_r)^{2p}}, \tag{11.8}$$

where $\varepsilon \ll 1$; typically $\varepsilon = 10^{-6}$. The appropriate value of ε is closely related to the magnitude of β_r and, for very accurate solutions, a smaller value may be needed. For a discussion of this matter we refer to [16, 1]. The value of p can be changed to modify the impact of the smoothness indicator, although a value of $p = 1$ is often used.

We first consider the smooth case. In this case, (11.7) is satisfied provided

$$\beta_r = C(h)\left(1 + \mathcal{O}(h^{m-1})\right).$$

On the other hand, if $\beta_r = \mathcal{O}(1)$, we recover $\omega_r^{r_0} = \mathcal{O}(h^{2p})$, which severely reduces the impact of this stencil. This is precisely the behavior we are looking for.

The final step in the design of the algorithm is the definition of the smoothness indicator β_r. Intuitively, we seek a cheap and smooth indicator, related to the total variation of the solution. To avoid the complexity of evaluating the L^1-norm, [17] proposes

$$\beta_r = \sum_{l=1}^{m-1} \int_{x_{j-1/2}}^{x_{j+1/2}} h^{2l-1} \left(\frac{d^l \pi_r}{dx^l} \right)^2 dx. \tag{11.9}$$

We recall that π_r is the reconstruction polynomial for the specific stencil, while the factor of h^{2l-1} ensures grid-scale independence. Since we assume an equidistant grid, the specific reference to r_0 is omitted.

The smoothness indicators can be expressed in explicit form as

$$\beta_0 = (\bar{v}_{j+1} - \bar{v}_j)^2, \quad \beta_1 = (\bar{v}_j - \bar{v}_{j-1})^2,$$

for $m = 2$, and

$$\beta_0 = \frac{13}{12}\left(\bar{v}_j - 2\bar{v}_{j+1} + \bar{v}_{j+2}\right)^2 + \frac{1}{4}\left(3\bar{v}_j - 4\bar{v}_{j+1} + \bar{v}_{j+2}\right)^2, \tag{11.10}$$

$$\beta_1 = \frac{13}{12}\left(\bar{v}_{j-1} - 2\bar{v}_j + \bar{v}_{j+1}\right)^2 + \frac{1}{4}\left(\bar{v}_{j-1} - \bar{v}_{j+1}\right)^2,$$

$$\beta_2 = \frac{13}{12}\left(\bar{v}_{j-2} - 2\bar{v}_{j-1} + \bar{v}_j\right)^2 + \frac{1}{4}\left(\bar{v}_{j-2} - 4\bar{v}_{j-1} + 3\bar{v}_j\right)^2,$$

To exemplify the evaluation of the smoothness indicator, let us first consider the case of $m=2$ and $r=0,1$, characterized by the grid $[x_{-1/2-r}, x_{1/2-r}, x_{3/2-r}]$. The resulting operator is

$$\mathrm{Q}^0=\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix},\ \mathrm{Q}^1=\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix},$$

and the smoothness indicators for $m=2$ are given as

$$\beta_0=(\bar{v}_{j+1}-\bar{v}_j)^2,\ \beta_1=(\bar{v}_j-\bar{v}_{j-1})^2.$$

For $m=3$ and $r=0,1,2$, we obtain

$$\mathrm{Q}^0=\frac{1}{6}\begin{bmatrix} 20 & -31 & 11 \\ -31 & 50 & -19 \\ 11 & -19 & 8 \end{bmatrix},\ \mathrm{Q}^1=\frac{1}{6}\begin{bmatrix} 8 & -13 & 5 \\ -13 & 26 & -13 \\ 5 & -13 & 8 \end{bmatrix},\ \mathrm{Q}^2=\frac{1}{6}\begin{bmatrix} 8 & -19 & 11 \\ -19 & 50 & -31 \\ 11 & -31 & 20 \end{bmatrix}.$$

We note the close relationship between Q^0 and Q^2. Note also that row and column sums are zero, as is expected by consistency since there operators are associated with the derivative of a polynomial.

In agreement with (11.10), we can rewrite these smoothness indicators as

$$\beta_0=\frac{13}{12}\left(\bar{v}_j-2\bar{v}_{j+1}+\bar{v}_{j+2}\right)^2+\frac{1}{4}\left(3\bar{v}_j-4\bar{v}_{j+1}+\bar{v}_{j+2}\right)^2,$$

$$\beta_1=\frac{13}{12}\left(\bar{v}_{j-1}-2\bar{v}_j+\bar{v}_{j+1}\right)^2+\frac{1}{4}\left(\bar{v}_{j-1}-\bar{v}_{j+1}\right)^2,$$

$$\beta_2=\frac{13}{12}\left(\bar{v}_{j-2}-2\bar{v}_{j-1}+\bar{v}_j\right)^2+\frac{1}{4}\left(\bar{v}_{j-2}-4\bar{v}_{j-1}+3\bar{v}_j\right)^2.$$

■

While the accuracy of the WENO scheme is designed to be $\mathcal{O}(h^{2m-1})$ in smooth regions of the solution and its accuracy reduces to $\mathcal{O}(h^m)$ near discontinuties, one can question if an intermediate accuracy is possible, i.e., will the non-linear weights take values that improves the accuracy in the neighborhood of a discontinuity.

To gain some insight into this, let us express the non-linear weights (11.8) as

$$\omega_r^{r_0}=d_r^{r_0}\left(\sum_{s=0}^{m-1}d_s^{r_0}\left(\frac{\beta_r+\varepsilon}{\beta_s+\varepsilon}\right)^{2p}\right)^{-1}.$$

If all stencils are smooth, then (11.11) immediately yields the linear weights and the optimal order of accuracy is recovered as expected.

Let us instead assume that one stencil, e.g., $r=0$ is non-smooth. In this case, $\beta_0=\mathcal{O}(1)$ and we observe that $\omega_0^{r_0}=\mathcal{O}(h^{2p})$ while

$$\omega_r^{r_0}=\frac{d_r^{r_0}}{\sum_{s=1}^{m-1}d_s^{r_0}}.$$

Hence, the $m-2$ stencils are combined linearly away from the non-smooth stencil but with weights that are different from the optimal linear weights. Hence, the WENO scheme will switch abruptly in accuracy from $\mathcal{O}(h^{2m-1})$ in smooth regions of the solution to $\mathcal{O}(h^m)$ is areas where at least one stencil encounters a discontinuity.

$$[\Delta_\alpha \mathscr{A} - \mathscr{A}\Delta_\alpha]\, g(x(\alpha)) =$$
$$\int_{\alpha-h/2}^{\alpha+h/2} g(x(\beta+h/2))\big[x'(\beta+h/2)-x'(\beta)\big] - g(x(\beta-h/2))\big[x'(\beta-h/2)-x'(\beta)\big]\, d\beta.$$

For this to vanish for all functions g, we must require that

$$x'(\beta \pm h/2) \mp x'(\beta) = 0,$$

for all values of β and h. This is clearly possible only when $x'(\alpha)$ is a constant. Hence, assumption (11.21) holds only for equidistant grids.

The unfortunate consequence of this result is that we cannot pursue the development of high-order accurate finite difference formulations on uneven grids as we will loose either conservation or accuracy. In [25], examples of special smoothly varying grids that allow one to overcome this restriction are discussion. However, for general non-smooth grids, conservative finite difference schemes with a local flux cannot exceed second order accuracy [32].

Let us now return to the details of the finite volume scheme on uneven grids. To enable this, we only need to modify the expression of three sets of coefficients: c_{ir}^m to allow the reconstruction in both ENO and WENO schemes, and $d_r^{r_0}$ and β_r to enable the computation of the nonlinear weights in the WENO scheme. These coefficients become locally varying functions and depend on the grid size of all neighboring elements. The approach discussed previously generalizes directly to this case, and we shall not pursue the details, some of which can be found in [31].

11.5 ▪ Beyond one dimension

In the following we discuss the extension of essentially non-oscillatory schemes to the case of two spatial dimensions. To keep things simple, we restrict ourselves to equidistant Cartesian grids. In this case, as we discussed at length in Chap. 7, the extension to problems beyond one dimension is particularly simple.

As we are now aiming at methods of arbitrary accuracy, we need to revisit the developments in Chap. 7. In general, from the integral form of the conservation laws, we recover that

$$\bar{u}_{ij}^{n+1} = \bar{u}_{ij}^{n} - \frac{1}{h_x h_y} \int_{t^n}^{t^{n+1}} \int_{x_{i-1/2}}^{x_{i+1/2}} \int_{y_{j-1/2}}^{y_{j+1/2}} \nabla \cdot f(u)\, dx dy,$$

where the cell average is defined as

$$\bar{u}_{ij}^{n} = \frac{1}{h_x h_y} \int_{x_{i-1/2}}^{x_{i+1/2}} \int_{y_{j-1/2}}^{y_{j+1/2}} u(x,y,t^n)\, dx dy.$$

We rewrite the flux term as

$$\int_\Omega \nabla \cdot f(u)\, dV = \int_{\partial\Omega} \hat{\boldsymbol{n}} \cdot f(u)\, dS,$$

where Ω represents the control volume $[x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}]$.

In Chapter 7, we used the midpoint rule to approximate the four surface integrals as