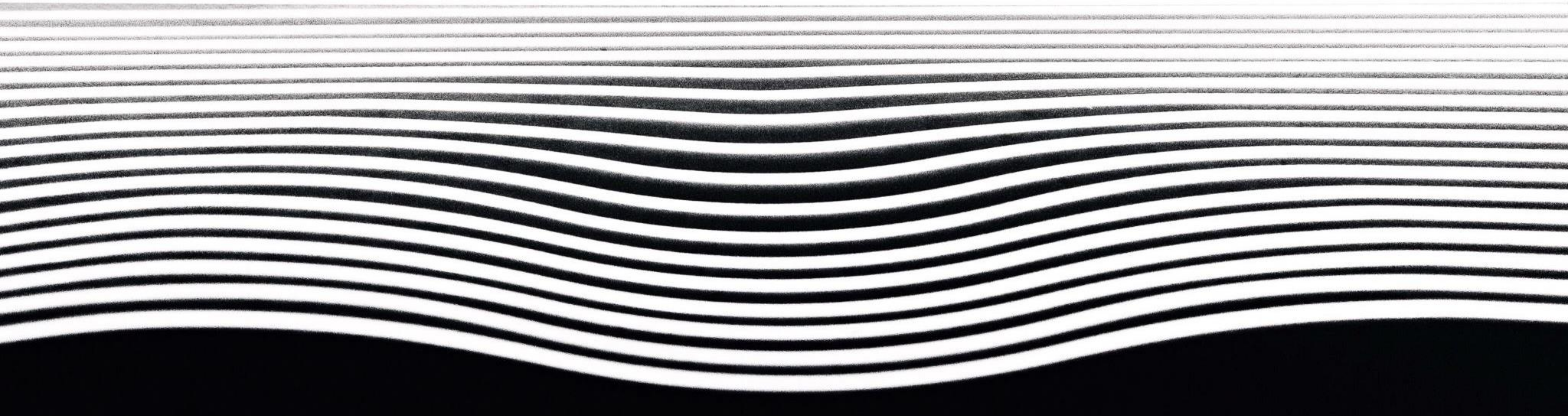
Numerical methods for conservation laws
16: Weighted Essentially non-oscillatory schemes



Recall

- We have introduced essentially non-oscillatory (ENO) schemes

- For these schemes, we adaptively choose a stencil of width m that we can shift to the left or the right.

- **Weighted** ENO schemes: instead of picking one stencil, we more generally can pick a weighted combination of those stencils.

What are some possible issues with ENO schemes?

- We have a range of $2m-1$ cells, but only get approximation order $O(h^m)$ because we discard about a half of the cell values.

  In smooth regions, we would like to get $O(h^{2m-1})$ by using all stencils.

- Stencil selection leads to discontinuous numerical fluxes. Continuous fluxes are more desirable.

  If the divided differences are close, then rounding errors decide the stencil. We expect that in smooth regions

- However, close to discontinuous of u, we would like to pick only one stencil that avoids the jump (as best as possible)

How do mitigate these issues?

- Close to jumps, we still select only one stencil
- Away from jumps, we choose a weighted linear combination of the stencils.

This leads to weighted ENO (WENO) schemes.

Main issue: how should we compute the weights?

First, suppose we are in a smooth region.

Let m be the stencil size, $r_0 \in \{-1,0\}$.
We search coefficients $d_r^{r_0}$ such that

$$U_{j+1/2} = \sum_{r=r_0}^{r_0+m-1} d_r^{r_0} U_{j+\frac{1}{2}}^{(r)}$$

Recall that r_0 determines the family of the stencils:

$r_0 = 0$: all stencils contain x_j

$r_0 = -1$: all stencils contain x_{j+1}

We want to achieve coefficients such that

$$U_{j+1/2} = u\left(x_{j+\frac{1}{2}}\right) + O(h^{2m-1})$$

These coefficients are computed by solving a linear system. (Hesthaven, Ch11.1) Given the uniform mesh size, we can just precompute these coefficients at the start. They are non-negative and sum up to one:

$$d_r^{r_0} \geq 0, \qquad \sum_{r=r_0}^{r_0+m-1} d_r^{r_0} = 1$$

So we have a convex combination of stencil values. In particular, using those weights as such as a linear scheme. It is of higher order, but problems show up near discontinuities.

Consequently, instead we search for coefficients adaptively such that in smooth regions $d_r^{r_0} \approx \omega_r^{r_0}$. What are reasonable requirements to get there?

$$U_{j+1/2} = \sum_{r=r_0}^{r_0+m-1} \omega_r^{r_0} U_{j+\frac{1}{2}}^{(r)}$$

1) They describe a convex combination:

$$\omega_r^{r_0} \geq 0, \qquad \sum_{r=r_0}^{r_0+m-1} \omega_r^{r_0} = 1$$

2) In non-smooth regions, we want $\omega_r^{r_0} \in \{0,1\}$ near jumps to enforce the selection of a single stencil (namely the smoothest one).

3) Approximation in smooth regions: $\omega_r^{r_0} = d_r^{r_0} + O(h^{m-1})$

We want to attain the accuracy under such small coefficient perturbations. We observe

$$u(x_{j+1/2}) - \sum_{r=r_0}^{r_0+m-1} \omega_r^{r_0} U_{j+\frac{1}{2}}^{(r)} = \underbrace{u(x_{j+1/2}) - \sum_{r=r_0}^{r_0+m-1} d_r^{r_0} U_{j+\frac{1}{2}}^{(r)}}_{O(h^{2m-1})} + \sum_{r=r_0}^{r_0+m-1} (d_r^{r_0} - \omega_r^{r_0}) U_{j+1/2}^{(r)}$$

We then compute

$$\sum_{r=r_0}^{r_0+m-1} (d_r^{r_0} - \omega_r^{r_0}) U_{j+1/2}^{(r)} = \sum_{r=r_0}^{r_0+m-1} (d_r^{r_0} - \omega_r^{r_0}) U_{j+1/2}^{(r)} + \sum_{r=r_0}^{r_0+m-1} (d_r^{r_0} - \omega_r^{r_0}) u(x_{j+1/2})$$

$$= \sum_{r=r_0}^{r_0+m-1} (d_r^{r_0} - \omega_r^{r_0}) \left(U_{j+1/2}^{(r)} - u(x_{j+1/2})\right) \quad \in O(h^{m-1})\, O(h^m)$$

How can we construct $\omega_r^{r_0}$ in practice? There are different possibilities.

We can try:

$$\omega_r^{r_0} = \frac{\alpha_r}{\sum \alpha_r}, \qquad \alpha_r = \frac{d_r^{r_0}}{(\epsilon + \beta_r)^{2p}}$$

Where the parameters $\beta_r \geq 0$ are still to be computed (adaptively), and the other parameters are fixed.

- $\beta_r \geq 0$ measures the roughness
- $\epsilon \approx 10^{-6}$ is a minimum roughness (avoids division by too tiny numbers).
- p controls the influence of roughness measures

How do we pick those roughness measures?

$$\omega_r^{r_0} = \frac{\alpha_r}{\sum \alpha_r}, \qquad \alpha_r = \frac{d_r^{r_0}}{(\epsilon + \beta_r)^{2p}}$$

A possible choice are (the absolute values of) the divided differences introduced above. We discuss another variant that can be found in the literature:

$$\beta_r = \sum_{1 \leq l < m} \int_{x_{j-1/2}}^{x_{j+1/2}} h^{2l-1} \left(\partial_x^l \pi_r\right)^2$$

where π_r is the reconstructed polynomial with shift parameter r.

Example For $m=2$, the smoothness indicators are

$$\beta_0 = (u_{j+1} - u_j)^2 \qquad \beta_1 = (u_j - u_{j-1})^2$$

Example For $m=3$, the indicators are

$$\beta_0 = \frac{13}{12}(u_j - 2u_{j+1} + u_{j+2})^2 + \frac{1}{4}(3u_j - 4u_{j+1} + u_{j-1})^2$$

$$\beta_1 = \frac{13}{12}(u_{j-1} - 2u_j + u_{j+1})^2 + \frac{1}{4}(u_{j-1} - u_{j+1})^2$$

$$\beta_2 = \frac{13}{12}(u_{j-2} - 2u_{j-1} + u_j)^2 + \frac{1}{4}(u_{j-2} - 4u_{j-1} + 3u_j)^2$$

We remember

$$\pi_r(x) = h \sum_{i=0}^{m} \sum_{q=0}^{i-1} \overline{U}_{q+j-r} \cdot \frac{d}{dx} l_i^{(j-r-1/2)}(x)\,.$$

Thus

$$\frac{d^l}{dx^l} \pi_r(x) = h \sum_{i=0}^{m} \sum_{q=0}^{i-1} \overline{U}_{q+j-r} \cdot \frac{d^{l+1}}{dx^{l+1}} l_i^{(j-r-1/2)}(x) = h \sum_{q=0}^{m-1} \overline{U}_{q+j-r} \left(\sum_{i=q+1}^{m} \cdot \frac{d^{l+1}}{dx^{l+1}} l_i^{(j-r-1/2)}(x) \right)$$

We rewrite this as

$$\frac{d^l}{dx^l} \pi_r(x) = h \sum_{q=0}^{m-1} \overline{U}_{q+j-r} Q_{q,r,j}^{l}(x)$$

Note: the parameter j enters $Q_{q,r,j}^{l}(x)$ only via translation x_j. Except for this translation, these polynomial terms look the same for each local problem.

We observe

$$\int_{x_{j-1/2}}^{x_{j+1/2}} \left(\partial_x^l \pi_r\right)^2 = h \sum_{q=0}^{m-1} \sum_{q'=0}^{m-1} \bar{U}_{q+j-r} \bar{U}_{q'+j-r} \int_{x_{j-1/2}}^{j+1/2} Q_{q,r,j}^l(x)\, Q_{q',r,j}^l(x)$$

The last integrals do not depend on j and can be computed in advance. In particular, this is a quadratic form in the stencil vector:

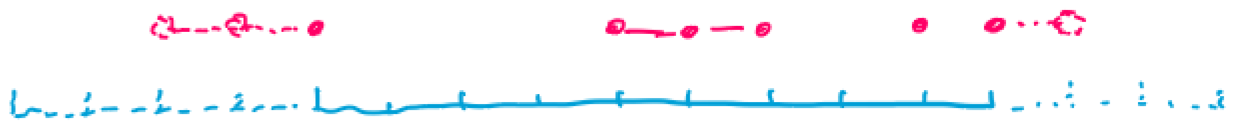
$$\int_{x_{j-1/2}}^{x_{j+1/2}} \left(\partial_x^l \pi_r\right)^2 = \bar{U}_{j,r}^T \cdot Q \cdot U_{jr}$$

What is known theoretically?

- It is possible to get accuracy $O(h^{2m-1})$ for smooth regions.
- Unfortunately, little is know for stability theory
- There are different variants in the literature

Coda: what are some practical issues with ENO and WENO schemes?

1) For boundary value problems, we need to extend the computational domain via ghost nodes.

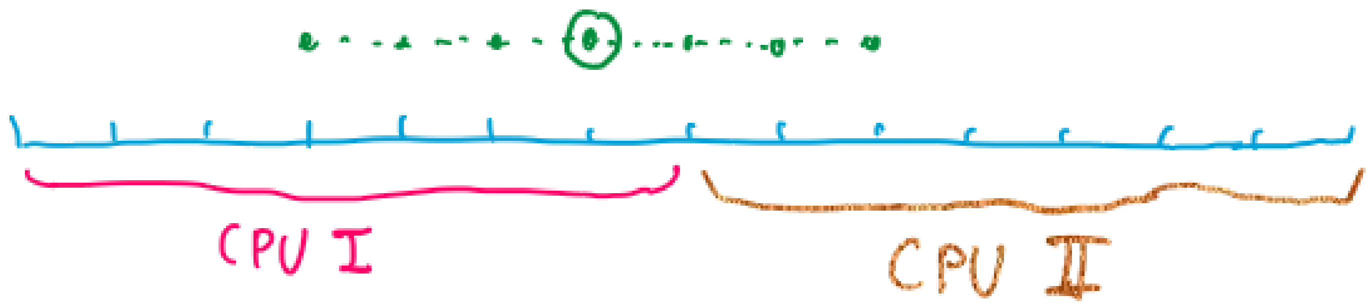


This is similar to setting up the finite difference methods for the Neumann problem:

$$-u'' = f, \qquad u'(0) = 0, \qquad u'(1) = 0$$

Coda: what are some practical issues with ENO and WENO schemes?

2) The higher the order, the larger the stencil, the less the locality. That is an issue with parallel computing.



3) Lastly, the computational effort can become quite high for large stencil sizes.