# Application of ENO technique to semi-Lagrangian interpolations

RC LACE stay report
Scientific supervisors: Petra Smolíková and Ján Mašek

Alexandra Crăciun
NMA, Romania

CHMI, Prague

09.05 - 03.06.2016

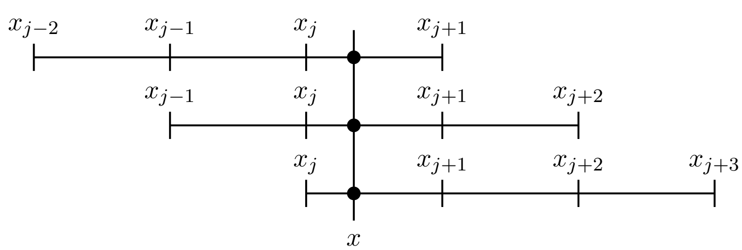
# 1 Introduction

Weighted essentially non-oscillatory scheme was first proposed by Liu, Osher and Chan in [4] and further developped by many authors, see for example [1, 2, 3]. WENO schemes are based on ENO technique with the key idea of finding the "smoothest" stencil among several candidates to interpolate to a high order accuracy and to avoid at the same time spurious oscillations near discontinuities or sharp gradients in the interpolated field. In case of WENO method instead of choosing the "smoothest" stencil possible, the weighted sum of values interpolated on several stencils is used with weights based again on smoothness evaluation. Moreover, the weights can be chosen in such a way that in smooth regions it approaches certain optimal weights to achieve a higher order of accuracy, while in regions near discontinuities, the stencils which contain discontinuities are assigned a nearly zero weight. ENO schemes are not cost effective on vector supercomputers because the stencil-choosing step involves heavy usage of logical statements, which perform poorly on such machines. On the other hand, WENO scheme completely removes the logical statements that appear in the ENO stencil choosing step. As the result, WENO scheme appear to be much faster than ENO scheme on vector machines.

After a preparatory extension of the whole interpolation stencil from 4 to 6 points, third order ENO technique was implemented in the code in the last year, see [5]. The goal of this stay was to implement the third order WENO interpolation technique based on previous work and compare it with the classical cubic Lagrange interpolation and cubic ENO scheme.

# 2 Implementation in the cycle 40t1

To construct a third order WENO interpolation of a field $f$ to the point $x$, we need to know $f$ in a set $S = \{x_{j-2}, ..., x_{j+3}\}$ of six points, with the subsets $S_k = \{x_{j-3+k}, ..., x_{j+k}\}$, $k = \overline{1,3}$.




Having the interpolation point $x$ in the central interval $[x_j, x_{j+1}]$ of the big stencil $S$, the interpolator is built as a linear combination of the interpolation polynomials on the three stencils $S_k$, through the following form:

$$P(x) = \sum_{k=1}^{3} \omega_k(x) P_k(x), \tag{1}$$

where the weights were built as follows:

$$\omega_k(x) = \frac{\tilde{\omega}_k(x)}{\tilde{\omega}_1(x) + \tilde{\omega}_2(x) + \tilde{\omega}_3(x)}, \ \tilde{\omega_k}(x) = \frac{C_k(x)}{(\beta_k(x) + \varepsilon)^p}. \tag{2}$$

This method should assign to each polynomial $P_k$ a weight $\omega_k$ according to the smoothness of the function $f$ on the corresponding sub-stencil $S_k$. In the above formula, the term $\beta_k$ is considered to be a *smoothness indicator* and is used to measure the smoothness of the solution on each sub-stencil $S_k$. There are multiple ways of defining it. If all the stencils are equally smooth according to the definition used for the smoothness indicators $\beta_k$, we want to obtain the non-linear weights $\omega_k$ in a way that leads to the highest degree approximation for the interpolated function. We can achieve fifth degree for 6 point stencil. The smoother is the function on the stencil $S_k$, the lower is the value of $\beta_k$.

The success of this method depends largely on the definition of the smoothness indicators $\beta_k$. One way of measuring the variation of the function is based on L2-norm of high-order variations of the reconstruction polynomials [3]:

$$\beta_k(x) = \sum_{l=1}^{3} \int_{x_j}^{x_{j+1}} (\Delta x)^{2l-1} (P_k^{(l)}(x))^2 dx \tag{3}$$

Similar definitions for $\beta_k$ were also tested (the difference between them being the derivatives of $P_k$ taken into account):

$$\beta_k(x) = \sum_{l=2}^{3} \int_{x_j}^{x_{j+1}} (\Delta x)^{2l-1} (P_k^{(l)}(x))^2 dx \tag{4}$$

$$\beta_k(x) = \int_{x_j}^{x_{j+1}} (\Delta x)^{3} (P_k^{(2)}(x))^2 dx \tag{5}$$

$$\beta_k(x) = \int_{x_j}^{x_{j+1}} (\Delta x)^{5} (P_k^{(3)}(x))^2 dx \tag{6}$$

Another method ([2],[4]) of estimating the smoothness of the function $f$ on stencil $S_k$ involves undivided differences:

$$\beta_k(x) = \sum_{l=1}^{3} \sum_{i=1}^{4-l} \frac{(f[j+k+i-4,l])^2}{4-l}, \tag{7}$$

where the undivided difference $f[\cdot,\cdot]$ is defined recursively as follows:

$$f[j,0] = f(x_j),\ f[j,l] = f[j+1,l-1] - f[j,l-1].$$

In the above formula, the "linear weights" $C_k$ are polynomials of degree 2 and $P_k$ are polynomials of degree 3 interpolating the function on each sub-stencil $S_k$ (for our experiments, we used cubic Lagrange polynomials for defining $P_k$). Following the definition described in [1] and assuming regular grid both in horizontal and vertical (LREGETA=.T.), the "linear weights" have the form:

$$C_1(x) = \frac{1}{20}(2-\xi)(3-\xi),\ C_2(x) = \frac{1}{-10}(2+\xi)(\xi-3),\ C_3(x) = \frac{1}{20}(2+\xi)(1+\xi), \tag{8}$$

where variable $\xi = \dfrac{x - x_j}{\Delta x}$ and $\Delta x$ is the mesh size.

In equation (2), parameter $p$ in the denominator serves to increase/decrease the dissipation of the scheme (see [3]), and $\varepsilon$ is a small positive number used to avoid/prevent division by zero. In the experiments, $\varepsilon = 10^{-6}$ and three values of p were tested, namely p=0.5, p=2 and p=3.

In order to implement this scheme, new subroutine LAIWENO was designed for computation of weights $\omega_k$. This routine is called from subroutine LAITRI_WENO (and the bidimensional version LAIDDI_WENO). The choice for the smoothness indicators $\beta_k$ can be made using variable KDER, which can get integer values from 1 to 5, depending on their definition, respectively, according to equations (3) - (7). Also, the value of $p$ used in the definition of the weights (equation 2) may be set, through the namelist parameter RALPHA. Variable PDIST is either PDLO (in longitude), PDLAT (in latitude) or PDVER (in vertical dimension) and represents the distance $\xi$ in equation (8).

The WENO scheme may be applied using the general switch LWENO, appearing in the module YOMENO and being declared in the namelist *namdyn*. Other necessary changes were made in subroutines LARCINB and LAITRE_GMV.

## 3 Experiments

For the evaluation of WENO method, several tests were performed. Preliminary 1D tests performed by Petra Smolíková include linear advection of a rectangular pulse in a periodic domain and visualisation of the weights $\omega_k$, $k = \overline{1,3}$ for each definition of the smoothness indicators (equations 3-7), for several values of variable $p$, as well as the particular case of $\beta_k = 0$, corresponding to the Lagrange interpolation of the fifth order.

Figure 1 shows results for different definitions of the smoothness indicator, compared with cubic Lagrange interpolator and ENO scheme. The over/undershoots present in cubic Lagrange solution almost disappear when ENO is used while they are completely eliminated with WENO scheme using definition (4) and (7). In Figures 2 and 3 one can see that WENO definitions lead to the expected behaviour: the stencil not containing the discontinuity has bigger value of the weight. One can notice that even if all definitions give very similar weights after one time step (see Figure 2), longer interpolation leads to different weight patterns (see Figure 3).

Having this encouraging results we proceeded to 2D experiments - classical test (Andre Robert): warm bubble with sharp boundary rising up in the field of constant potential temperature (300K) without forced horizontal advection. The reference solution uses cubic Lagrange interpolator.

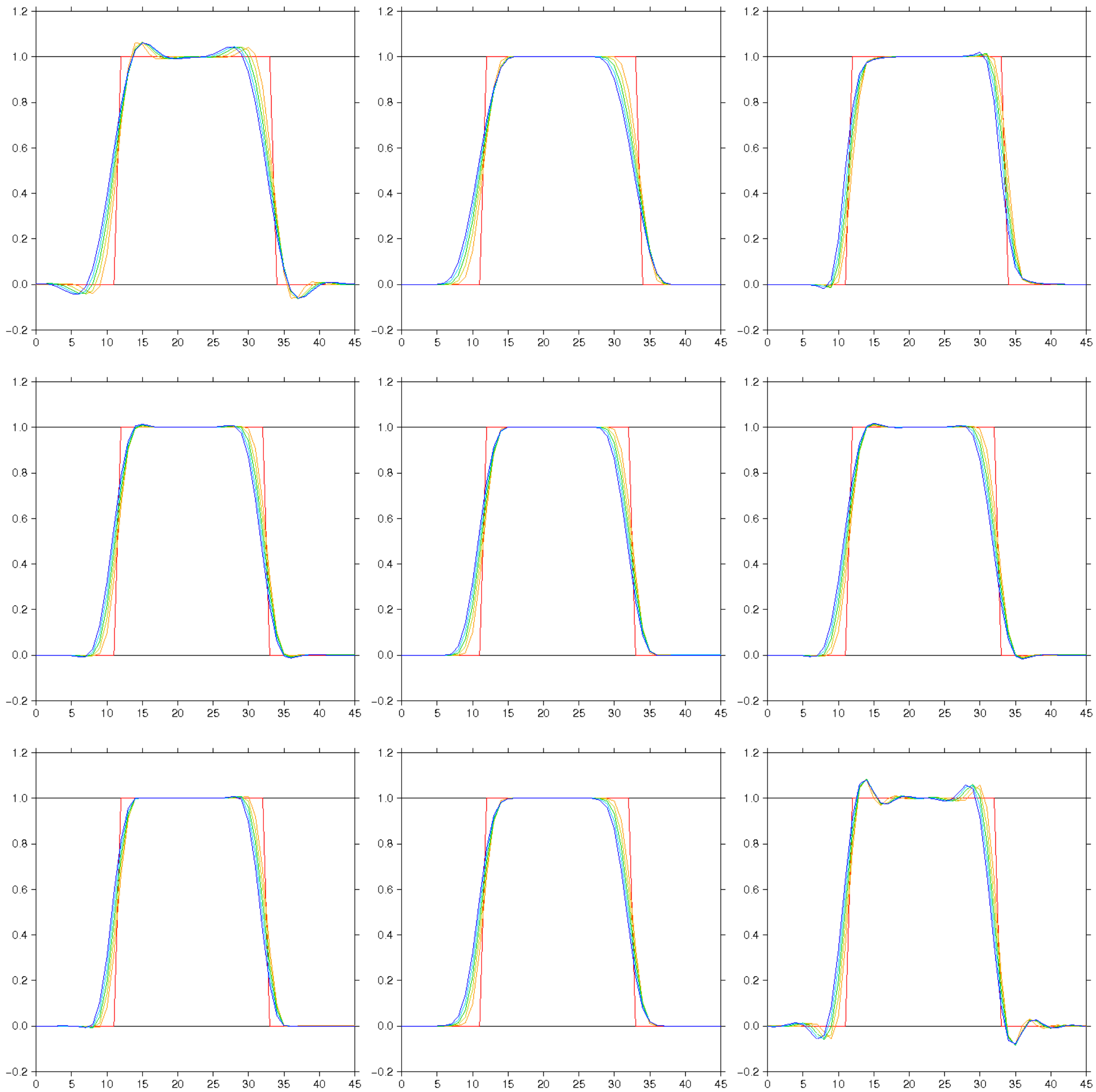


Figure 1: Linear advection of a rectangular pulse in a periodic domain, red - original function, orange - after 1 revolution, yellow - after 2 revolutions, green - after 3 revolutions, blue - after 4 revolutions, violet - after 5 revolutions. Cubic Lagrange, cubic Lagrange with quasi-monotonic version, ENO scheme (first row - from left to right), WENO scheme, p=0.5, different definitions for $\beta_k$: eq. 3, eq. 4, eq. 5 (second row - from left to right), eq. 6, eq. 7 and $\beta_k = 0$ case (third row - from left to right)

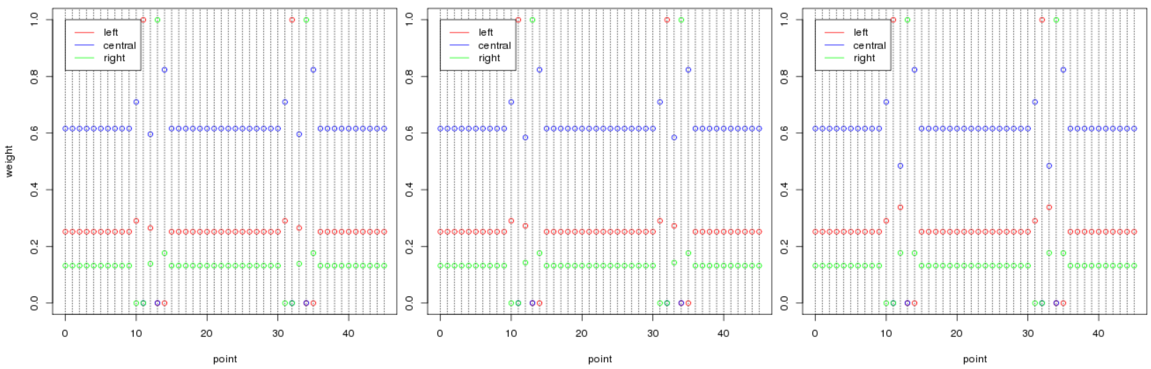



Figure 2: Linear advection of a rectangular pulse after 1 time step, WENO weights $\omega_1$ (red), $\omega_2$ (blue), $\omega_3$ (green), $p = 0.5$, different definitions for $\beta_k$: eq. 3 (left), eq. 4 (center), eq. 7 (right)

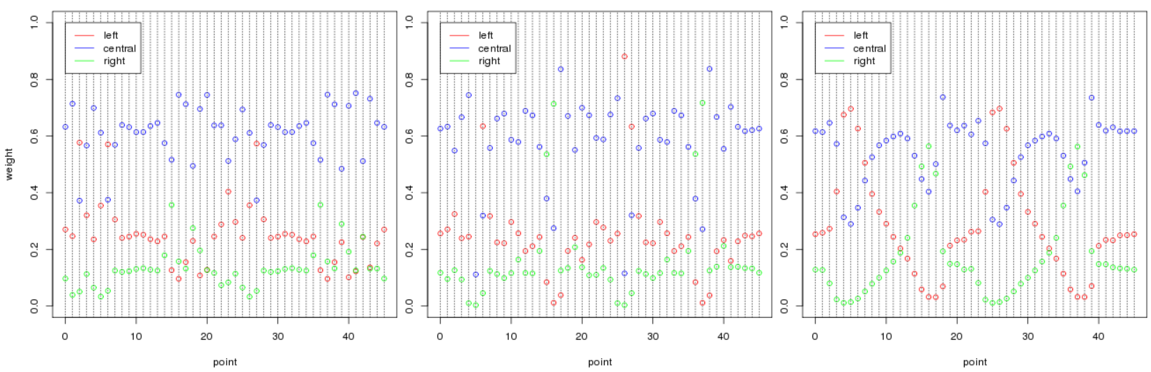



Figure 3: The same as Figure 2 after 280 time steps (5 revolutions).

As expected, the results shown in Figures 4, 5 and 6 prove that when WENO scheme is applied, the solution becomes smoother than the one which uses cubic Lagrange interpolator. Smoothness indicators computed as in equation (3), (4) and (7) lead to quite similar solution and better accuracy than for the cases of $\beta_k$ defined in equation (5) (not shown) and (6), when some of the vortices in the reference solution disappear.

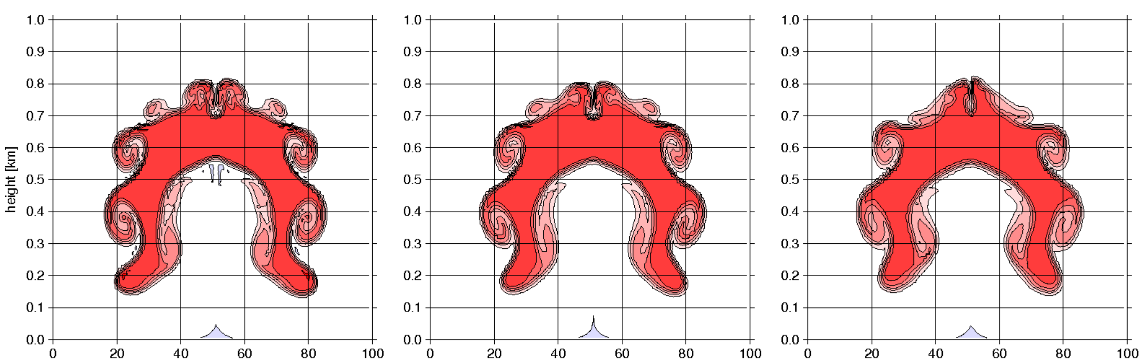



Figure 4: Sharp warm bubble: cubic Lagrange interpolator (left), quasi - monotonic version of cubic Lagrange (center) and ENO (right)

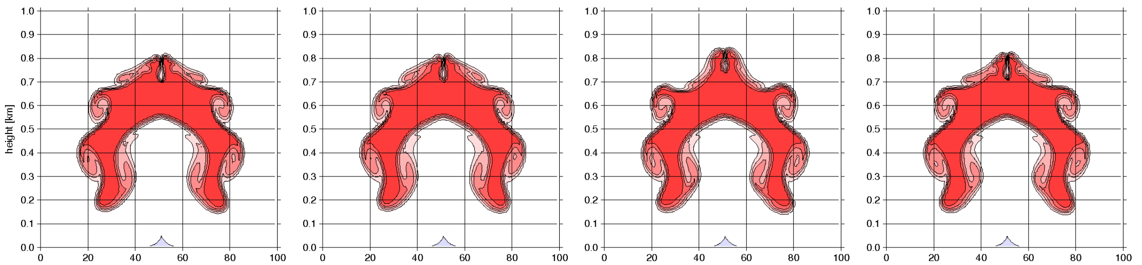



Figure 5: Sharp warm bubble: WENO scheme, p=2, different definitions for $\beta_k$: eq. (3), (4), (6) and (7), from left to right

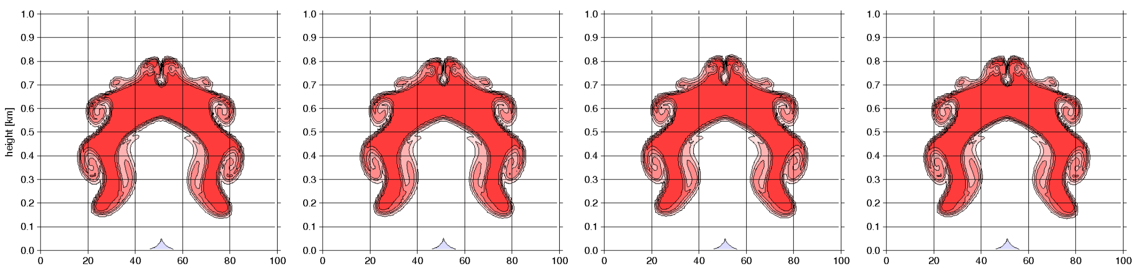



Figure 6: The same as Figure 5 but with p=0.5.

For the same definitions of smoothness indicators, setting value of p either 2 or 3 gives similar results; we show results only for p=2. Moreover, both values of this parameter prove to be too smoothing in comparison with the solution provided by p=0.5 (Figure 6), which seems to be the most similar to the reference solution. Besides, p=0.5 and $\beta_k$ computed according to definition (3) or (7) remain the best candidates also when the scheme is applied only in horizontal dimension.

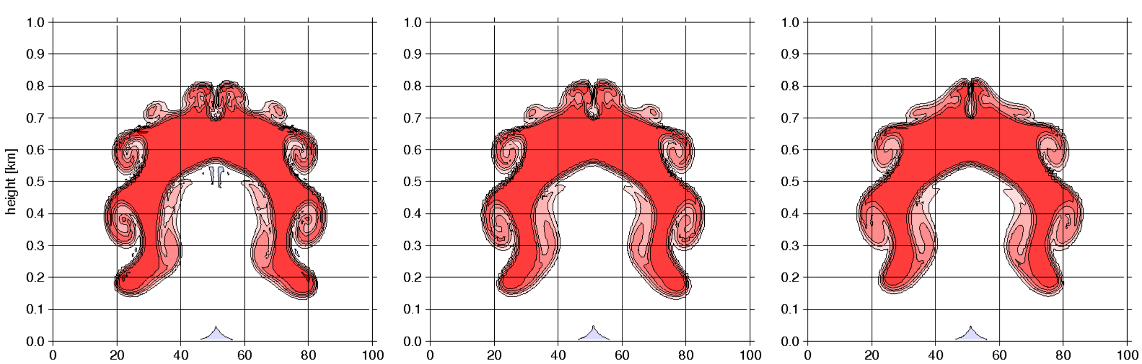



Figure 7: Sharp warm bubble: cubic Lagrange (left), WENO scheme applied only in horizontal, $\beta_k$ as in eq. (3): p=0.5 (center), p=2 (right)

Another step in the evaluation was to visualize the way this scheme chooses the interpolation stencil. The plots below (Figure 8) show the stencils corresponding to the highest value of weights $\omega_k$ (k=1,2,3), in horizontal dimension (latitude) in the second layer of the whole 3D-interpolation grid used (see [5] for details). Parameter IWLOC (in subroutine LAIWENO) is equal to 1 when the highest weight was obtained on the central stencil (white color), 2 for the right stencil (red color) and 3 for the left stencil (blue color). It can be seen in Figure 8 that the WENO scheme behaves as expected near discontinuities, for example, the right stencil is chosen mostly on the left side of the bubble's vortices, the left one on the right side and central stencil elsewhere. One can see that for WENO with p=2 lateral stencils are used more often then for WENO with p=0.5 and cubic Lagrange interpolator (which apparently uses always the central stencil).

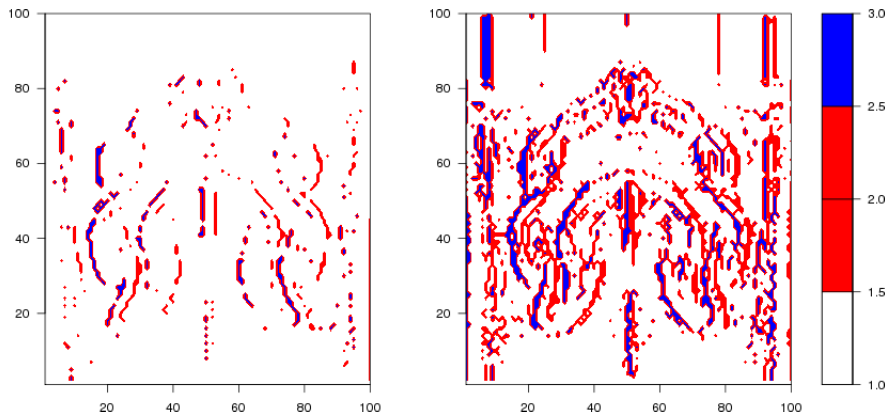



Figure 8: Stencil choice, WENO scheme after 80 steps, $\beta_k$ as in eq. (3): p=0.5 (left) and p=2 (right)

Furthermore, in order to assess where under/overshootings appear for distinct interpolators, parameter IPARQM was introduced in subroutine LAIWENO (and LAITRI, for the case of cubic Lagrange interpolator). This parameter is set to 1, when the interpolator overshoots, 2 when it undershoots and 0 when the interpolator stays within the bounds given by values of the function to be interpolated, and was obtained using a similar approach as for the case of quasi-monotonous treatment of interpolations, found in subroutine LAITRI. With WENO scheme applied in both horizontal and vertical directions, IPARQM was plotted either for the vertical direction only (see Figures 9, 10 and 11 top) or the horizontal direction only, for the second layer of the six used in the whole 3D-interpolation grid (see Figure 11 bottom).

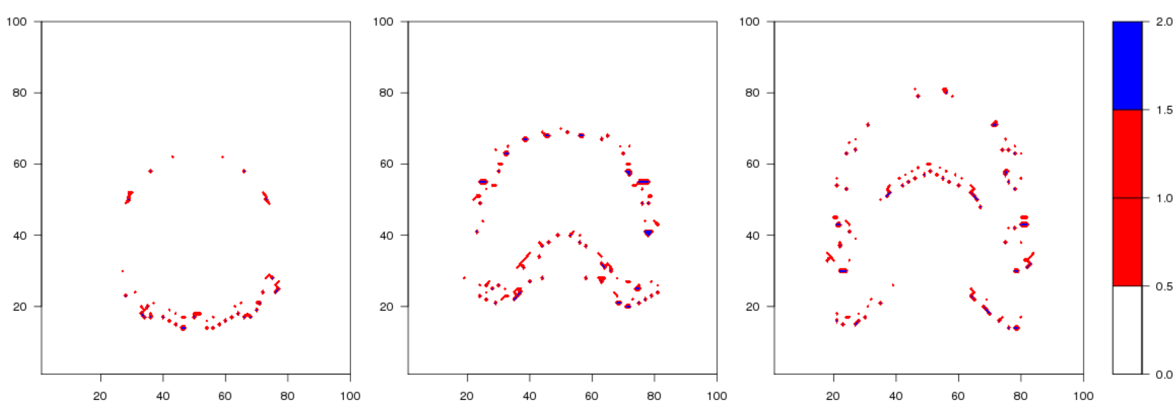



Figure 9: IPARQM (vertical direction), sharp warm bubble, cubic Lagrange: after 40 steps (left), after 60 steps (center) and after 80 steps (right)

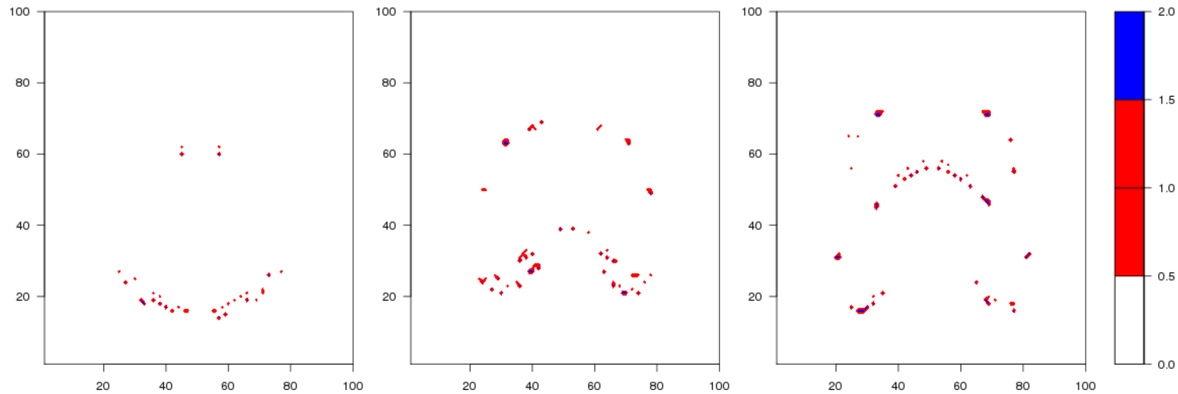



Figure 10: IPARQM (vertical direction), sharp warm bubble, WENO scheme, $\beta_k$ as in eq. (3), p=2: after 40 steps (left), after 60 steps (center) and after 80 steps (right)

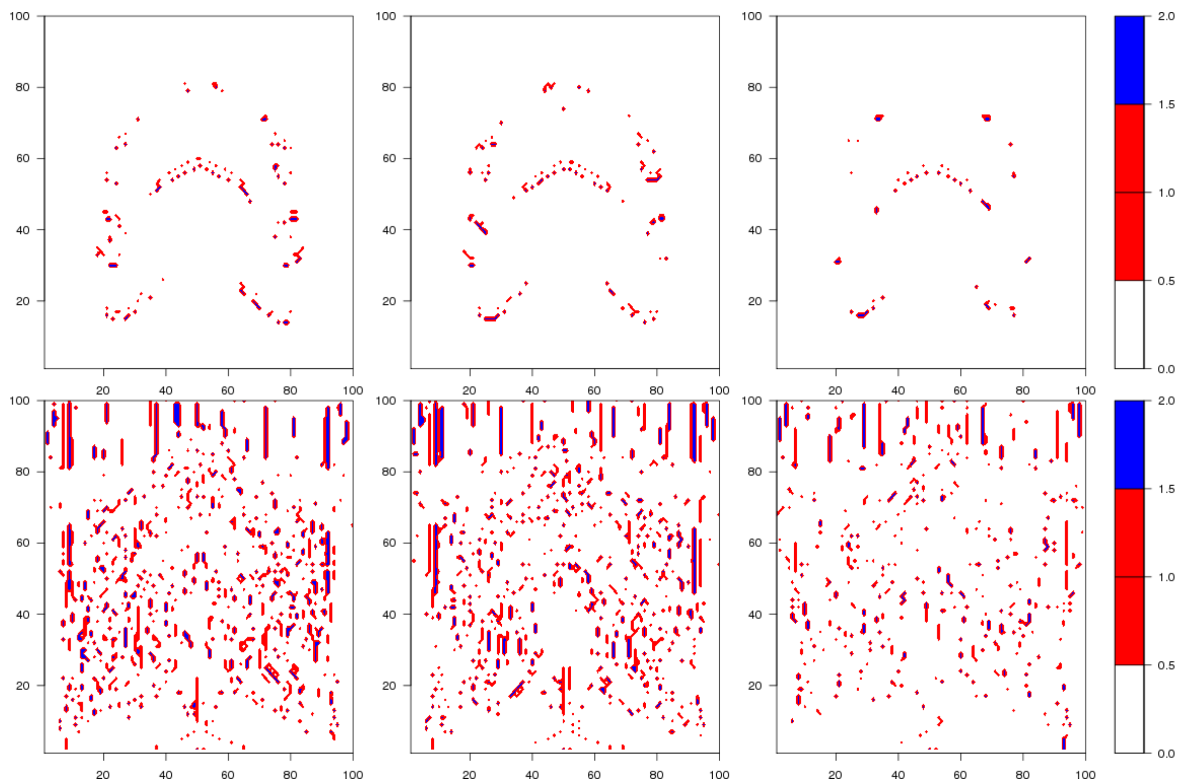



Figure 11: IPARQM (in vertical - first row, in horizontal - second row), sharp warm bubble after 80 steps: cubic Lagrange (left), WENO scheme, $\beta_k$ as in eq. (3), p=0.5 (center), and p=2 (right)

It can be seen that both schemes lead to overshooths in specific areas of the bubble. Moreover, p=0.5 leads to comparable results to cubic Lagrange interpolation, for overshoots computed either in vertical or in horizontal direction. In addition, the values of these under/overshootings were analysed. These were computed under variable ZOVER, in subroutine LAIWENO (and LAITRI, for the case of cubic Lagrange interpolator). Figure 12 shows the values of ZOVER computed in horizontal direction, for the second layer of the six used in the whole 3D-interpolation grid (with WENO scheme applied in both horizontal and vertical directions). used in the whole 3D-interpolation grid (with WENO scheme applied in both horizontal and vertical directions).

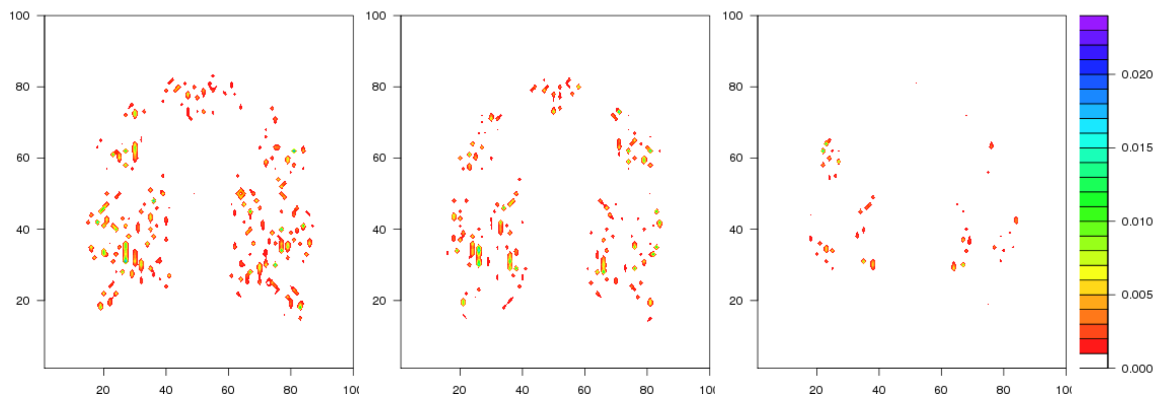



Figure 12: Sharp warm bubble after 80 steps: cubic Lagrange (left), WENO scheme, $\beta_k$ as in eq. 3, p=0.5 (center) and p=2 (right)

It can be seen in Figure 12 that after 80 steps, the two values of p lead to different results. When p is set to 2, overshoots have smaller values (analogous result could be also observed in Figure 5). At the same time, the case with p=0.5 presents slightly less overshoots with cubic Lagrange interpolation.

## 4 Conclusion

The WENO (Weighted essencially non-oscillatory) scheme was implemented to semi-Lagrangian interpolations of the model ALADIN with several possible definitions. Overall, the results show that the WENO scheme, for all definitions of smoothness indicators and values of parameter $p$ that were tested, produces slightly smoother solution than cubic Lagrange interpolator. To conclude, the best choice for the smoothness indicators was the one in equation (3), with all derivatives (up to third order) of the reconstruction polynomials taken into consideration. The case with p=0.5 showed increased accuracy than p=2 and p=3. Besides, the implementation gives fifth order Lagrange interpolation in case $\beta_k$ are set to zero.

We may say that interpolations are subject of a trade off between accuracy and noise production near discontinuities. More smoothing schemes give less over/undershoots while more accurate results suffer from noise created near sharp gradients or discontinuities in the interpolated field.

It seems to us that slight improvement in the production of over/undershoots observed for the best behaving choice of the $\beta_k$ and p parameters ($\beta_k$ according to definition (3) and p=0.5) does not compensate the increase in the computational cost of the new WENO scheme compared to the classical cubic Lagrange solution.

*Acknowledgements:* I would like to thank my supervisors for their useful advice and suggestions.

## References


[1] E. Carlini, R. Ferretti and G. Russo, *A Weighted Essentially Nonoscillatory, Large Time-Step Scheme for Hamilton–Jacobi Equations*. SIAM J. Sci. Comput., 27(3), 1071-1091, 2005

[2] G.-S. Jiang, C.-W. Shu, *Efficient Implementation of Weighted ENO Schemes*. Journal of Computational Physics, Volume 126, Issue 1, June 1996, Pages 202-228

[3] C.-Y. Jung, T. B. Nguyen, *A New Adaptive Weighted Essentially Non-Oscillatory WENO-$\theta$ Scheme for Hyperbolic Conservation Laws*, arXiv:1504.00731, 2015

[4] X.-D. Liu, S. Osher, T. Chan, *Weighted Essentially Non-oscillatory schemes*. Journal of Computational Physics, Volume 115, Issue 1, November 1994, Pages 200-212

[5] http://www.rclace.eu/?page=13 *Application of ENO techniques to SL interpolations*. RC-LACE stay report, 2015