

On new strategies to control the accuracy of WENO algorithm close to discontinuities III: Conservation laws

Sergio Amat¹

Department of Applied Mathematics and Statistics, Universidad Politécnica de Cartagena (Spain)

Juan Ruiz^{1,*}

Department of Applied Mathematics and Statistics, Universidad Politécnica de Cartagena (Spain)

Chi-Wang Shu²

Division of Applied Mathematics, Brown University (USA)

Abstract

This paper is the continuation of the articles S. Amat, J. Ruiz, C.-W. Shu, On new strategies to control the accuracy of WENO algorithms close to discontinuities, SIAM J. Numer. Anal. 57 (3) (2019) 1205 – 1237 and S. Amat, J. Ruiz, C.-W. Shu, On new strategies to control the accuracy of WENO algorithm close to discontinuities II: Cell averages, (Submitted). It is devoted to the construction and analysis of new non-linear optimal weights for WENO interpolation capable of raising the order of accuracy close to shocks in the solution of conservation laws. In the references mentioned before, we showed that using these new weights it is possible to attain optimal theoretical accuracy when approximating piecewise smooth functions close to discontinuities. Our aim is to present a new algorithm for the approximation of the solution of conservation laws based on these weights.

Keywords: WENO, new optimal weights, multiresolution schemes, improved adaption to discontinuities, signal processing. **AMS(MOS) subject classifications.** 65D05, 65D17, 65M06, 65N06

1. Introduction

This article concerns the solution of hyperbolic conservation laws of the form

$$\mathbf{u}_t + \nabla \cdot f(\mathbf{u}) = 0,$$

*Corresponding author

Email addresses: `sergio.amat@upct.es` (Sergio Amat), `juan.ruiz@upct.es` (Juan Ruiz), `chi-wang_shu@brown.edu` (Chi-Wang Shu)

¹The author have been supported through the *Programa de Apoyo a la investigación de la fundación Séneca-Agencia de Ciencia y Tecnología de la Región de Murcia* 19374/PI714 and through the national research project MTM2015-64382-P (MINECO/FEDER).

²The author have been supported by NSF grant DMS-1719410.

using a modification of the WENO (weighted essentially non oscillatory) algorithm, originally proposed in [3] and lately modified in [4].

WENO schemes arouse as an improvement of ENO (essentially non oscillatory) schemes [5] for data discretized in the cell averages. The core idea of ENO schemes is to use the smoothest stencil among a group of possible stencils. The objective is to approximate the flux at the cell boundaries with a high order of accuracy assuring, at the same time, that Gibbs oscillations do not appear close to the discontinuities. Following these requirements, ENO schemes assure high order of accuracy except at the cell that contains the shock. An incomplete list of references about ENO methods is [6, 5, 7, 8, 9, 10, 11, 12, 13, 14, 15].

Liu, Osher and Chan introduced in [3] an improvement of ENO scheme and they called it WENO. Instead of using only one stencil as ENO scheme does, WENO technique consists in approximating the flux building a convex combination of all the possible stencils that ENO scheme uses. The algorithm assigns a weight to every possible stencil, that depends on the smoothness of the data. Thus, the stencils that belong to a smooth zone are assigned a weight whose value is very close to some optimal weights that assure optimal accuracy. On the other hand, close to the discontinuities the stencils that cross them are assigned weights whose values are very close to zero. In [4] Jiang and Shu introduce new smoothness indicators that are more efficient than those proposed originally in [3]. An incomplete list of references about WENO schemes is [16, 17, 18, 19, 20, 21].

With the original design of WENO schemes proposed in [3, 4], the restrictions imposed to the weights are to obtain optimal accuracy at smooth zones while providing an essentially non oscillatory result close to the discontinuities, but without demanding a strict control over the weights of the convex combination. This means that the order of approximation is not controlled if there is more than one smooth sub-stencil close to the discontinuities. In [1] we showed that this characteristic of the WENO scheme can be improved if the optimal weights, that are constant in the initial design proposed in [3, 4], are replaced by nonlinear optimal weights.

In this article we continue the path opened in [1, 2] and we propose a modification of the non-linear optimal weights introduced in [3, 4] with the aim of rising the order of accuracy when approximating the flux close to discontinuities in the solution of conservation laws. Our aim is to design a strategy that allows to keep optimal accuracy everywhere, in the sense of maximizing the presence of smooth data in the WENO convex combination. An important difference between the techniques presented in [1, 2] and the one presented in this article is due to the fact that in the approximation of conservation laws the polynomials used to approximate the flux are uncentered. In the references mentioned previously, all the polynomials used are centered and they are built from data in the point values [1] or the cell averages [2]. The reason for using uncentered polynomials is that the flux is approximated at the cell interfaces that are shifted $\Delta x/2$ with respect to the center of the stencil. This fact imposes a complete redesign of the algorithm that leads to interesting consequences in the construction.

This paper is organized as follows: In Section 2 we introduce the discretization of the data. Section 3 is dedicated to introduce the particular kind of hyperbolic partial differential equations that we will be dealing with. Section 4 talks about the classical WENO method for the solution of hyperbolic conservation laws. Section 5 introduces the new WENO algorithm for conservation laws. Section 6 analyses theoretically the accuracy that the new WENO algorithm reaches at smooth zones of the flux and close to shocks. Section 7 is dedicated to present some relevant numerical experiments. Finally, Section 8 presents the conclusions.

2. Discretization of data in the cell-averages

The cell averages discretization appears in a natural form when numerically discretizing hyperbolic conservation laws. In this section we will give some basic notes about this discretization as it will be the one used in the whole article. Let us consider piecewise smooth functions in the interval $[a, b]$ and the space of finite sequences V of length $N = J + 1$. We will denote by X to a uniform partition of the interval $[a, b]$ in J subintervals,

$$X = \{x_{i-1/2}\}_{i=0}^J, \quad x_{-1/2} = a, \quad h_i = x_{i+1/2} - x_{i-1/2}.$$

Now we consider the discretization, that is defined as,

$$\bar{f}_i = \frac{1}{h} \int_{x_{i-1/2}}^{x_{i+1/2}} f(x) dx, \quad i = 0, \dots, J-1, \quad (1)$$

for the space of absolutely integrable functions in $[a, b]$ $L^1([a, b])$ and discretized sequences of length $N = J + 1$. Now we can define the set of values $\{F_j\}$ as,

$$F(x_{j+1/2}) = \sum_{i=1}^j h \bar{f}_i = \int_{x_{-1/2}}^{x_{j+1/2}} f(y) dy, \quad (2)$$

where F can be considered the primitive of f , $F(x) = \int_{x_0}^x f(y) dy$. With this assumption, the sequence $\{F_j\}$ is a discretization in the point-values of $F(x)$. Using the previous definition, we can denote by $P(x)$ the unique polynomial of degree k that interpolates the values of the primitive. Using the derivative $P'(x)$ to approximate $f(x)$ in (1), it is clear that we can relate the values of the primitive with the cell values through the expression,

$$\bar{f}_i = \frac{1}{h} \int_{x_{i-1/2}}^{x_{i+1/2}} f(x) dx = \frac{1}{h} \int_{x_{i-1/2}}^{x_{i+1/2}} P'(x) dx = \frac{P(x_{i+1/2}) - P(x_{i-1/2})}{h} = \frac{F(x_{i+1/2}) - F(x_{i-1/2})}{h}, \quad (3)$$

where the last equality holds because $P(x)$ interpolates $F(x)$ at the grid points $x_{i+1/2}$ and $x_{i-1/2}$.

3. Conservation laws

WENO reconstructions are not directly related to numerical schemes for approximating the solution of any PDE, but they have been classically applied to the solution of hyperbolic conservation laws. In what follows we will present the equations that we are interested in and how to discretize them. We will consider systems of hyperbolic conservation laws in n space dimensions,

$$\begin{cases} \mathbf{u}_t + \sum_{i=1}^d f_i(\mathbf{u})_{x_i} = 0, \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}). \end{cases} \quad (4)$$

In the previous expressions, $\mathbf{u}_0(\mathbf{x})$ is the initial condition. Suitable boundary conditions should also be defined. Let us start by the simplest case and consider a numerical approximation of the weak solution of the following scalar hyperbolic conservation law in one dimension,

$$\begin{cases} u_t + f(u)_x = 0, \\ u(x, 0) = u_0(x). \end{cases} \quad (5)$$

We can denote by $u_j^n = u_h(x_j, t_n)$ a numerical approximation of the exact solution $u(x_j, t_n)$ of (5) defined on a computational grid $x_j = jh$ with $j \in \mathbb{Z}$, $h > 0$. In this case we are clearly considering that the data $u_j^n = u_h(x_j, t_n)$ is a discretization in the point values of the solution $u(x_j, t_n)$ of (5). Thus, our objective is to obtain a high order conservative approximation of the derivative of $f(x)$ from its point values. If we are capable of finding a function $g(x)$ such that,

$$f(u(x)) = \frac{1}{h} \int_{x-h/2}^{x+h/2} g(y) dy, \quad (6)$$

then it is clear that

$$\frac{df(u(x))}{dx} = \frac{g(x+h/2) - g(x-h/2)}{h}, \quad (7)$$

that is precisely a conservative approximation of the derivative of $f(x)$. All we need to do now is to find an approximation of g with an order of accuracy $k+1$,

$$\hat{f}_{j+\frac{1}{2}} = g(x_{j+1/2}) + O(h^{k+1}),$$

to obtain an approximation of the derivative of the flux with an order of approximation k ,

$$f(u(x_j))_x = \frac{1}{h} \left(\hat{f}(x_j + h/2) - \hat{f}(x_j - h/2) \right) + O(h^k), \quad (8)$$

where the numerical flux $\hat{f}_{j+\frac{1}{2}}$ approximates $g(x)$, that is implicitly defined by (6). The implicit definition in (6) determines that the known function $f(x)$ is just a cell average of the function $g(x)$, that is unknown. Thus, to recover an approximation of the function $g(x)$, we can just use a reconstruction via the primitive function. Obtaining the primitive of $g(x)$ we get,

$$G(x) = \int_{-\infty}^x g(y) dy, \quad (9)$$

and as a consequence,

$$G(x_{j+1/2}) = \sum_{i=-\infty}^j \int_{x_{i-1/2}}^{x_{i+1/2}} g(y) dy = h \sum_{i=-\infty}^j \bar{g}_i = h \sum_{i=-\infty}^j f_i.$$

This last expressions means that we know the primitive of $g(x)$ at the cell interfaces $x = x_{j+1/2}$. With this information it is possible to build a polynomial approximation $P(x)$ of $G(x)$ in (9), and then differentiate it and replace $x = x_j - h/2$ and $x = x_j + h/2$ as in (7), to obtain an approximation of $f(u(x_j))_x$. Thus,

$$\hat{f}(x_{j-1/2}) = P'(x_{j-1/2}). \quad (10)$$

The next section is dedicated to briefly explain how to use the WENO algorithm to recover a k -th order of accuracy approximation $P(x)$ of the primitive $G(x)$. This reconstruction will be built using the known point values of the primitive $G(x)$ at the cell interfaces in (9), and then $P'(x_{j+1/2})$ will be taken as the numerical flux $\hat{f}(x_{j+1/2})$ in (8).

As the points of interpolation of the fluxes at $x_{j-1/2}$ and $x_{j+1/2}$ differ from those used in [2], we proceed to adapt the new WENO algorithm, introduced in [1] and extended to cell averages in [2], to the context of conservation laws.

4. The classical WENO algorithm for hyperbolic conservation laws

In [3] the authors introduced a first version of the WENO algorithm for conservation laws. In [4] the authors presented new smoothness indicators that are more suitable than those presented in [3]. In this section we will briefly introduce the classical WENO algorithm for conservation laws.

Let us represent by $S_i^m(j) = \{I_{j-m+i}, \dots, I_{j+i-1}\}$ the stencil of m cells with $I_j = [x_{j-1/2}, x_{j+1/2}] = [x_{j-h/2}, x_{j+h/2}]$. Using this notation, the WENO method uses the stencil $S_r^{2r-1}(j) = \{I_{j-r+1}, \dots, I_{j+r-1}\}$ composed of $2r-1$ cells. Following this notation, $S_k^r(j)$, $k = 1, \dots, r$, is the k^{th} sub-stencil of length r that contains the cell $I_j = [x_{j-1/2}, x_{j+1/2}]$,

$$S_k^r(j) = \{I_{j-r+k}, \dots, I_{j+k-1}\}, \quad k = 1, \dots, r, \quad (11)$$

where we want to approximate the derivative of the flux using the numerical fluxes $\hat{f}_{j-\frac{1}{2}}$ and $\hat{f}_{j+\frac{1}{2}}$, as shown in (8). As stated in Section 3, our objective is to obtain a conservative approximation of the derivative of the flux $f(u(x_j))_x$ using (8). With this configuration in mind, we can build the following convex combination,

$$q_{j-r}(x) = \sum_{k=0}^{r-1} \omega_k^{r-1}(j) p_{j-r+k}^{r-1}(x), \quad (12)$$

with $\omega_k^{r-1}(j) \geq 0$, $k = 1, \dots, r$ and $\sum_{k=1}^r \omega_k^{r-1}(j) = 1$. In (12), $p_{j-r+k}^{r-1}(x)$ is the interpolatory polynomial of degree $r-1$ written for the cell averages over the stencil $S_k^r(j)$. The prediction operator $p_{j-r+k}^{r-1}(x)$ that allows to obtain the numerical flux at the cell interfaces can be built using a polynomial interpolation in the primitive. Then, using the relation in (24) between a reconstruction in the primitive values and cell averages, we can recover the expression for the polynomial approximation in the cell averages. The expressions for $r = 3$ of $p_{j-3+k}^2(x_{j-1/2})$ for $k = 1, \dots, 3$ are,

$$\begin{aligned} p_{j-2}^2(x_{j-1/2}) &= -\frac{1}{6}\bar{f}_{j-2} + \frac{5}{6}\bar{f}_{j-1} + \frac{1}{3}\bar{f}_j, \\ p_{j-1}^2(x_{j-1/2}) &= \frac{1}{3}\bar{f}_{j-1} + \frac{5}{6}\bar{f}_j - \frac{1}{6}\bar{f}_{j+1}, \\ p_j^2(x_{j-1/2}) &= \frac{11}{6}\bar{f}_j - \frac{7}{6}\bar{f}_{j+1} + \frac{1}{3}\bar{f}_{j+2}. \end{aligned} \quad (13)$$

The expressions for $p_{j-3+k}^2(x_{j+1/2})$ are symmetrical,

$$\begin{aligned} p_{j-2}^2(x_{j+1/2}) &= \frac{1}{3}\bar{f}_{j-2} - \frac{7}{6}\bar{f}_{j-1} + \frac{11}{6}\bar{f}_j, \\ p_{j-1}^2(x_{j+1/2}) &= -\frac{1}{6}\bar{f}_{j-1} + \frac{5}{6}\bar{f}_j + \frac{1}{3}\bar{f}_{j+1}, \\ p_j^2(x_{j+1/2}) &= \frac{1}{3}\bar{f}_j + \frac{5}{6}\bar{f}_{j+1} - \frac{1}{6}\bar{f}_{j+2}. \end{aligned} \quad (14)$$

The numerical flux $\hat{f}_{j+1/2}$ is then given by,

$$\hat{f}_{j+1/2} = \sum_{k=0}^{r-1} \omega_k^{r-1}(j) p_{j-r+k}^{r-1}\left(x_{j+\frac{1}{2}}\right) + O(h^{2r-1}). \quad (15)$$

Thus, the objective of WENO algorithm is to find weights $\omega_k^{r-1}(j)$ that allow to obtain order of accuracy $2r-1$ when approximating the numerical flux $\hat{f}_{j+1/2}$ at smooth regions of the function f . Assuming that the flux is smooth on the big stencil $S_r^{2r-1} = \{I_{j-r+1}, \dots, I_{j+r-1}\}$ our objective is to find a polynomial interpolant that verifies,

$$p_{j-r+1}^{2r-2} \left(x_{j-\frac{1}{2}} \right) = \hat{f}_{j+1/2} + O(h^{2r-1}). \quad (16)$$

There also exist r interpolants of order $O(h^{r-1})$ constructed using the small stencils $S_k^r(j)$,

$$p_{j-r+k}^{r-1} \left(x_{j-\frac{1}{2}} \right) = \hat{f}_{j+1/2} + O(h^{r-1}). \quad (17)$$

At smooth zones WENO algorithm should assure that ω_k^{r-1} in (15) are very close to the values $C_k^{r-1}(j) \geq 0, \forall k$ such that $\sum_{k=0}^{r-1} C_k^{r-1}(j) = 1$, and

$$p_{j-r+1}^{2r-2} \left(x_{j-\frac{1}{2}} \right) = \sum_{k=0}^{r-1} C_k^{r-1}(j) p_{j-r+k}^{r-1} \left(x_{j-\frac{1}{2}} \right). \quad (18)$$

As shown in [4], for $r=3$ the optimal weights for $\hat{f}_{j-1/2}$ (for $\hat{f}_{j+1/2}$ are symmetrical) are,

$$C_1^2(j) = \frac{3}{10}, \quad C_2^2(j) = \frac{6}{10}, \quad C_3^2(j) = \frac{1}{10}. \quad (19)$$

In [3], the authors propose the following expression for the weights,

$$\omega_k^{r-1} = \frac{\alpha_k^{r-1}}{\sum_{k=0}^{r-1} \alpha_k^{r-1}}, \quad k = 0, \dots, r-1 \text{ where } \alpha_k^{r-1} = \frac{C_k^{r-1}}{(\epsilon + I_k^r(j, \bar{f}))^t}. \quad (20)$$

This expression for the weights satisfies that $\sum_k \omega_k^r(j) = 1$. In (20) $I_k^r(j, \bar{f})$ represents a smoothness indicator for $f(x)$ on the stencil $S_k^r(j)$. t is an integer that has the purpose of assuring the maximum order of accuracy close to the discontinuities.

We will use the smoothness indicators proposed in [4], that can be calculated through the formula,

$$I_k^r(j, \bar{f}) = \sum_{l=1}^{r-1} h^{2l-1} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \left(\frac{d^l}{dx^l} p_{j-r+k}^{r-1}(x) \right)^2 dx, \quad (21)$$

In the classical WENO algorithm these smoothness indicators take the expression,

$$\begin{aligned} I_1^2 &= \frac{13}{12}(\bar{f}_{j-2} - 2\bar{f}_{j-1} + \bar{f}_j)^2 + \frac{1}{4}(\bar{f}_{j-2} - 4\bar{f}_{j-1} + 3\bar{f}_j)^2, \\ I_2^2 &= \frac{13}{12}(\bar{f}_{j-1} - 2\bar{f}_j + \bar{f}_{j+1})^2 + \frac{1}{4}(\bar{f}_{j-1} - \bar{f}_{j+1})^2, \\ I_3^2 &= \frac{13}{12}(\bar{f}_j - 2\bar{f}_{j+1} + \bar{f}_{j+2})^2 + \frac{1}{4}(3\bar{f}_j - 4\bar{f}_{j+1} + \bar{f}_{j+2})^2. \end{aligned} \quad (22)$$

In [1] we mentioned that the value of the WENO weights (20) gives preference to the essentially non-oscillatory property over the accuracy close to the discontinuities. The consequence is an approximation which order of accuracy is not optimized if there is more than one sub-stencil free of

discontinuities. This is due to the fact that WENO algorithm uses fixed *optimal-weights* in (20) when constructing (15). The interested reader can find in Tables 1 and 3 or in Figure 5 of [22] some experiments that validate the previous statement. In [1] we introduced a nonlinear redesign of the WENO optimal weights that constitutes a possible solution for this problem in the point values. Due to the fact that the points where we want to obtain the numerical fluxes $\hat{f}_{j+1/2}$ or $\hat{f}_{j-1/2}$ are not centered, the expressions for the polynomials and the nonlinear weights are not the same as those presented in [2]. For completeness, we explain how to obtain the nonlinear optimal weights in the next section.

5. The new WENO algorithm for hyperbolic conservation laws

In [1], we explain a possible solution of the problems that WENO algorithm presents when interpolating data in the point values close to discontinuities. In particular, we propose a nonlinear modification of the optimal weights obtained through (15). In [2] we extend the new algorithm to data discretized in the cell averages. In this section we explain how to adapt the previous technique to the approximation of fluxes using finite differences. As in [1, 2], we study the particular case $r = 3$ that in this case corresponds to $n = 2r - 1 = 5$ cells ($2r$ point values of the primitive). The three sub-stencils of three cells used are $S_1^3 = \{I_{j-2}, I_{j-1}, I_j\}$, $S_2^3 = \{I_{j-1}, I_j, I_{j+1}\}$ and $S_3^3 = \{I_j, I_{j+1}, I_{j+2}\}$. The big stencil is composed of the cells $S_3^5 = \{I_{j-2}, I_{j-1}, I_j, I_{j+1}, I_{j+2}\}$. Having in mind the previous stencils, let's try to build a polynomial approximation for the numerical flux in (10) in the cell averages using the primitive function. The interpolating polynomial of degree n in the Newton form for the primitive function in (2), has the expression,

$$\begin{aligned} q_{j-r+k-1}^n(x) = & F_{j-r+k-1} + \frac{F_{j-r+k} - F_{j-r+k-1}}{h}(x - x_{j-r+k-1}) \\ & + \frac{F_{j-r+k-1} - 2F_{j-r+k} + F_{j-r+k+1}}{2h^2}(x - x_{j-r+k-1})(x - x_{j-r+k}) \\ & + \frac{-F_{j-r+k-1} + 3F_{j-r+k} - 3F_{j-r+k+1} + F_{j-r+k+2}}{6h^3}(x - x_{j-r+k-1})(x - x_{j-r+k})(x - x_{j-r+k+1}) \\ & + \dots + \frac{\delta_{j-r+k-1}^n}{n!h^n} \prod_{i=0}^{n-1} (x - x_{j-r+k-1+i}), \end{aligned} \quad (23)$$

being $\delta_{j-r+k-1}^n$ divided differences of order n and h a uniform grid spacing. The reconstruction p using the cell-averages can be written as the derivative of the interpolator q for the primitive values,

$$p(x) = \frac{d}{dx} q(x). \quad (24)$$

Applying (24) to (23), we obtain the expression for the polynomial of degree $n-1$ in the cell averages,

$$\begin{aligned} p_{j-r+k}^{n-1}(x) = & \bar{f}_{j-r+k} + \frac{\bar{f}_{j-r+k+1} - \bar{f}_{j-r+k}}{2h}((x - x_{j-r+k-1}) + (x - x_{j-r+k})) \\ & + \frac{\bar{f}_{j-r+k+2} - 2\bar{f}_{j-r+k+1} + \bar{f}_{j-r+k}}{6h^2}((x - x_{j-r+k-1})(x - x_{j-r+k}) + (x - x_{j-r+k-1})(x - x_{j-r+k+1}) \\ & + (x - x_{j-r+k})(x - x_{j-r+k+1})) + \dots + \frac{\delta_{j-r+k}^{n-1}}{n!h^{n-1}} \left(\prod_{i=0}^{n-1} (x - x_{j-r+k-1+i}) \right) \left(\sum_{m=0}^{n-1} \frac{1}{x - x_{j-r+k-1+m}} \right). \end{aligned} \quad (25)$$

For $r = 3$, we can denote the polynomials in the cell averages by $p_{j-r+k}^{r-1}(x) = p_{j-3+k}^2(x)$, such that $r - 1 = 2$ denotes the degree of the polynomial and I_{j-3+k} the cell where the substencil starts, for $k = 1, \dots, 3$. Even so, in what follows we will ease the notation and drop the dependence with j and simply write $p_k^{r-1}(x)$, as we will be constantly referring to the big stencil $S_3^5 = \{I_{j-2}, I_{j-1}, I_j, I_{j+1}, I_{j+2}\}$. All the polynomials in the primitive are evaluated at the point of interpolation $x_{j-1/2} = x_j - h/2$ or $x_{j+1/2} = x_j + h/2$, as shown in Figures 1 and 2, in order to construct the conservative expression of the flux derivative in (8). In order to obtain the new nonlinear optimal weights, for $r = 3$ we will be dealing with the polynomials in the cell averages $p_1^2(x), p_2^2(x)$ and $p_3^2(x)$ for the convex combination in (18) plus $p_1^3(x), p_2^3(x), p_3^4(x)$ for calculating the nonlinear optimal weights. All the previous polynomials will also be used to obtain smoothness indicators. The optimal weights for the classical WENO algorithm in the case $r = 3$ are those presented in (19) and can be easily obtained through (18). We can see that in this case the weights do not present symmetry, as it occurred in [1, 2]. As mentioned before, this is due to the fact that, in order to estimate $\hat{f}_{j+1/2}$ or $\hat{f}_{j-1/2}$, the interpolation point is not at the middle of the stencil, but shifted $h/2$. All the following considerations are given to obtain an approximation of $\hat{f}_{j-1/2}$. For $\hat{f}_{j+1/2}$ all the process is just symmetric. Following what was done to obtain the WENO constant optimal weights in (20), it is straightforward to prove that,

$$\begin{aligned} p_1^4(x) &= \frac{5}{3}C_1^2p_1^2(x) + \frac{5}{6}C_2^2p_2^2(x) \\ p_2^4(x) &= \frac{5}{4}C_2^2p_2^2(x) + \frac{5}{2}C_2^2p_3^2(x), \end{aligned} \quad (26)$$

with C_1^2, C_2^2 , and C_3^2 the optimal weights for $r = 3$ presented in (19). From (26), now it is straightforward to give the expression for the vectors of optimal weights $\mathbf{C}_1^3, \mathbf{C}_2^3$, each of which is suitable for a particular position of the discontinuity,

$$\begin{aligned} \mathbf{C}_1^3 &= \left(\frac{5}{3}C_1^2, \frac{5}{6}C_2^2, 0 \right), \\ \mathbf{C}_2^3 &= \left(0, \frac{5}{4}C_2^2, \frac{5}{2}C_3^2 \right). \end{aligned} \quad (27)$$

Now, following the procedure explained in [1], we use smoothness indicators for the polynomials of 3, 4 and 5 cells that arise from the big stencil, that is composed of 5 cells. We denote by $\tilde{\omega}_k^n$ the quotients,

$$\tilde{\omega}_1^3 = \frac{\tilde{\alpha}_1^3}{\tilde{\alpha}_1^3 + \tilde{\alpha}_2^3}, \quad \tilde{\omega}_2^3 = \frac{\tilde{\alpha}_2^3}{\tilde{\alpha}_1^3 + \tilde{\alpha}_2^3}, \quad (28)$$

with,

$$\tilde{\alpha}_1^3 = \frac{\bar{C}_1^3}{(\epsilon + I_1^3)^t}, \quad \tilde{\alpha}_2^3 = \frac{\bar{C}_2^3}{(\epsilon + I_2^3)^t}. \quad (29)$$

The constant weights $\bar{C}_1^3, \bar{C}_2^3$ arise from the fact that the addition of vectors of weights (27) is not a multiple of (C_1^2, C_2^2, C_3^2) , that is the vector of weights that we want to obtain at smooth zones. This fact will impose the use of the new constant optimal weights $\bar{C}_n^m$ in (29) with the value,

$$\bar{C}_1^3 = 3/5, \quad \bar{C}_2^3 = 2/5. \quad (30)$$

Now, it is clear that

$$\bar{C}_1^3 \mathbf{C}_1^3 + \bar{C}_2^3 \mathbf{C}_2^3 = \mathbf{C}_1^4 = (C_1^2, C_2^2, C_3^2),$$

and if we establish that the nonlinear optimal weights take the expression,

$$(\tilde{C}_1^2, \tilde{C}_2^2, \tilde{C}_3^2) = \tilde{\omega}_1^3 \mathbf{C}_1^3 + \tilde{\omega}_2^3 \mathbf{C}_2^3, \quad (31)$$

it is easy to check that at smooth zones we obtain that $(\tilde{C}_1^2, \tilde{C}_2^2, \tilde{C}_3^2) \approx (C_1^2, C_2^2, C_3^2) = \mathbf{C}_1^4$, that are the original constant optimal weights of WENO algorithm presented in (19). The nonlinear optimal weights $\tilde{C}_k^r$ are used in place of the optimal weights C_k^r in the expression of the WENO weights given in (20). The smoothness indicators that appear in (20) are those presented in (22) and use three cells. This part of the WENO algorithm remains untouched and we only modify the optimal weights, that now are nonlinear. In what follows we introduce a generalization of the algorithm and analyze the smoothness indicators that appear in (29).

5.1. Generalization to a higher order

The previous algorithm can also be interpreted as a WENO algorithm applied in several scales each time to two substencils. For example, the procedure to obtain the WENO algorithm presented in previous sections can also be interpreted as follows:

1. From (27) we can see that the vector of weights $\mathbf{C}_1^3$ is just the application of WENO to the first two stencils of three cells S_1^3 and S_2^3 with weights $\frac{5}{3}C_1^2$ and $\frac{5}{6}C_1^2$. It is also clear that $\mathbf{C}_2^3$ is the application of WENO to the stencils S_2^3 and S_3^3 with weights $\frac{5}{4}C_2^2$ and $\frac{5}{2}C_3^2$. If the stencil is smooth, the result will be the two polynomials of 4 cells presented in (26).
2. With the resulting polynomials in (26), we can just apply WENO again, but this time with the weights $\bar{C}_1^3$ and $\bar{C}_2^3$ in (30) to obtain the final result.
3. The process described here for polynomials of 5 cells can be extended very easily to construct WENO algorithms of $2r - 1$ cells, as the process would be iterative.

5.2. Smoothness indicators

An immediate option for I_1^3, I_2^3 in the expressions in (29), that are smoothness indicators of 4 cells and I_1^4 is a smoothness indicator of 5 cells, is to obtain them through (21). They have the expression,

$$\begin{aligned} I_1^3 &= -\frac{1261}{120} \bar{f}_{j+1} \bar{f}_j + \frac{961}{120} \bar{f}_{j+1} \bar{f}_{j-1} - \frac{247}{120} \bar{f}_{j+1} \bar{f}_{j-2} - \frac{2983}{120} \bar{f}_j \bar{f}_{j-1} + \frac{267}{40} \bar{f}_j \bar{f}_{j-2} - \frac{821}{120} \bar{f}_{j-1} \bar{f}_{j-2} \\ &\quad + \frac{547}{240} \bar{f}_{j+1}^2 + \frac{3443}{240} \bar{f}_j^2 + \frac{2843}{240} \bar{f}_{j-1}^2 + \frac{89}{80} \bar{f}_{j-2}^2 \\ I_2^3 &= \frac{89}{80} \bar{f}_{j+2}^2 + \frac{2843}{240} \bar{f}_{j+1}^2 + \frac{3443}{240} \bar{f}_j^2 + \frac{547}{240} \bar{f}_{j-1}^2 - \frac{821}{120} \bar{f}_{j+2} \bar{f}_{j+1} + \frac{267}{40} \bar{f}_{j+2} \bar{f}_j - \frac{247}{120} \bar{f}_{j+2} \bar{f}_{j-1} \\ &\quad - \frac{2983}{120} \bar{f}_{j+1} \bar{f}_j + \frac{961}{120} \bar{f}_{j+1} \bar{f}_{j-1} - \frac{1261}{120} \bar{f}_j \bar{f}_{j-1} \end{aligned} \quad (32)$$

In [1] we mentioned that the computational cost of the previous option is high. It can be reduced if we use as smoothness indicators of high order a function of those presented in (22). A good option, that provides the same accuracy results, could be for example,

$$\begin{aligned} I_1^3 &= I_1^2 \cdot I_2^2, \\ I_2^3 &= I_2^2 \cdot I_3^2, \end{aligned} \quad (33)$$

The reason is that at smooth zones, it is easy to see that a Taylor expansion around x_j of (22) [4] (and also (32)) gives ,

$$I_k^n = (h\bar{f}'_j)^2 \cdot (1 + O(h^2)), \quad n = 2, 3. \quad (34)$$

In the cases that $\bar{f}'(j) = 0$, then,

$$I_k^n = \frac{13}{12} (h^2 \bar{f}''_j)^2 \cdot (1 + O(h^2)), \quad n = 2, 3. \quad (35)$$

And for the indices in (33),

$$I_k^n = \left((h\bar{f}'_j)^2 \cdot (1 + O(h^2)) \right)^2 = (h\bar{f}'_j)^4 \cdot (1 + O(h^2)), \quad n = 2, 3. \quad (36)$$

In the cases that $\bar{f}'(j) = 0$, then,

$$I_k^n = \left(\frac{13}{12} (h^2 \bar{f}''_j)^2 \cdot (1 + O(h^2)) \right)^2 = \left(\frac{13}{12} \right)^2 (h^2 \bar{f}''_j)^4 \cdot (1 + O(h^2)), \quad n = 2, 3. \quad (37)$$

These results will be useful in the next section in order to prove the accuracy of the algorithm. As the constants that go with $(1 + O(h^2))$ are the same for each kind of smoothness indicator, the proofs will be exactly the same.

6. Accuracy of the new algorithm

In what follows we will analyze the accuracy of the new algorithm in all the possible cases that we can find when the stencil crosses a discontinuity when $r = 3$ and the big stencil is composed of $2r - 1$ cells:

- The first case that we will consider is when none of the sub-stencils S_k^n , $n = 2, 3$, (three of three cells and two of four cells) crosses a discontinuity and $f'_j \neq 0$ in (36). In this case all of the smoothness indicators are $I_k^n = (hf'_j)^2 \cdot (1 + O(h^2))$, $n = 2, 3$ (as shown in (36)). Replacing (29) in (28), the nonlinear weights in (28) can be written as,

$$\tilde{\omega}_k^n = \frac{\tilde{\alpha}_k^n}{\tilde{\alpha}_1^3 + \tilde{\alpha}_2^3} = \frac{\bar{C}_k^n}{(\epsilon + I_k^n)^t} \frac{1}{\frac{\bar{C}_1^3}{(\epsilon + I_1^3)^t} + \frac{\bar{C}_2^3}{(\epsilon + I_2^3)^t}}, \quad (38)$$

At zones that present enough regularity, replacing I_j^3 by (36) in (38) and supposing that ϵ is small enough,

$$\tilde{\omega}_k^n = \bar{C}_k^n \frac{(1 + O(h^2))^t}{(1 + O(h^2))^t}.$$

In order to simplify the previous expression, we know that $(1 + O(h^2))^t = 1 + O(h^2)$ and $\frac{1}{(1 + O(h^2))^t} = 1 + O(h^2)$, so,

$$\tilde{\omega}_k^n = \bar{C}_k^n \frac{(1 + O(h^2))^t}{2(1 + O(h^2))^t} = \bar{C}_k^n (1 + O(h^2)).$$

Now, using the values of the $\bar{C}_k^n$ in (30), (31) can be written as,

$$\begin{aligned} (\tilde{C}_1^2, \tilde{C}_2^2, \tilde{C}_3^2) &= (1 + O(h^2)) (\bar{C}_1^3 \mathbf{C}_1^3 + \bar{C}_2^3 \mathbf{C}_2^3) \\ &= \left(\frac{3}{5} \left(\frac{5}{3} C_1^2, \frac{5}{6} C_2^2, 0 \right) + \frac{2}{5} \left(0, \frac{5}{4} C_2^2, \frac{5}{2} C_3^2 \right) \right) (1 + O(h^2)) \\ &= (C_1^2, C_2^2, C_3^2) + O(h^2), \end{aligned} \quad (39)$$

that are the original optimal weights $C_k^r(j)$ in (18) and proposed in [4] plus an $O(h^2)$ perturbation. Let us check now the value of the WENO weights in (20), using as optimal weights the ones obtained in (39),

$$\omega_k^r = \frac{\tilde{\alpha}_k^r}{\sum_{i=0}^{r-1} \tilde{\alpha}_i^r} = \frac{C_k^r + O(h^2)}{(\epsilon + I_k^r)^t} \frac{1}{\sum_{i=0}^{r-1} \frac{C_i^r + O(h^2)}{(\epsilon + I_i^r)^t}},$$

Replacing I_i^r by (36) in the previous expression and choosing a small enough ϵ , we obtain that the WENO weights for this case are,

$$\omega_k^r = \frac{C_k^r + O(h^2)}{(1 + O(h^2))^t} \frac{(1 + O(h^2))^t}{1 + O(h^2)} = C_k^r + O(h^2). \quad (40)$$

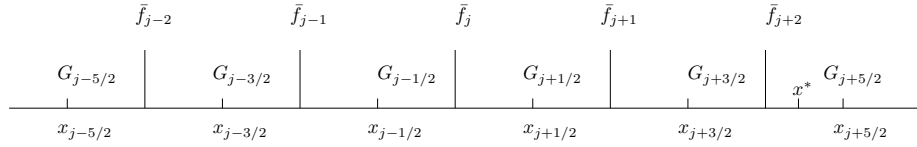


Figure 1: Representation of a discontinuity in the flux placed at x^* in the interval $(x_{j+3/2}, x_{j+5/2})$, that affects the cell average value $\bar{f}_{j+2}$. We have also represented the primitive values $G_{j-n/2}$. We can observe that in this particular case there are two smooth stencils of three cells.

- Figures 1 and 2 present two cases that are symmetric. Thus, we will only analyze the one in Figure 1. A discontinuity in the flux contained in the interval $(x_{j+3/2}, x_{j+5/2})$ affects the smoothness indicator I_2^3 in (32) resulting in a value $O(1)$. $I_1^3 = O(h^2)$ in (32), as it is not affected by the discontinuity. Thus, the weights in (29) take the expression

$$\begin{aligned} \tilde{\alpha}_1^3 &= \frac{1}{(\epsilon + O(h^2))^t}, \\ \tilde{\alpha}_2^3 &= \frac{1}{(\epsilon + O(1))^t}. \end{aligned}$$

Replacing these expressions in (38) and assuming again that ϵ takes a value that is small

enough, we obtain that the weights are,

$$\begin{aligned}\tilde{w}_1^3 &= \frac{\bar{C}_1^3}{(\epsilon + I_1^3)^t} \frac{1}{\frac{\bar{C}_1^3}{(\epsilon + I_1^3)^t} + \frac{\bar{C}_2^3}{(\epsilon + I_2^3)^t}} = \frac{\bar{C}_1^3}{(\epsilon + I_1^3)^t} \frac{1}{\frac{\bar{C}_1^3}{(\epsilon + I_1^3)^t} (1 + O(h^{2t}))} = \frac{1}{1 + O(h^{2t})} \\ &= 1 + O(h^{2t}), \\ \tilde{w}_2^3 &= \frac{1}{\frac{\bar{C}_1^3}{(\epsilon + I_1^3)^t} + \frac{\bar{C}_2^3}{(\epsilon + I_2^3)^t}} = \frac{\bar{C}_2^3}{(\epsilon + I_2^3)^t} \frac{1}{\frac{\bar{C}_1^3}{(\epsilon + I_1^3)^t} (1 + O(h^{2t}))} = O(h^{2t}).\end{aligned}$$

In this case, the adapted optimal weights take the form,

$$(\tilde{C}_1^2, \tilde{C}_2^2, \tilde{C}_3^2) = \mathbf{C}_1^3 + O(h^{2t}) = \left(\frac{5}{3}C_1^2, \frac{5}{6}C_2^2, 0 \right) + O(h^{2t}). \quad (41)$$

If the discontinuity is placed in the interval (x_{j-3}, x_{j-2}) , the conclusions would be exactly the same but

$$(\tilde{C}_1^2, \tilde{C}_2^2, \tilde{C}_3^2) = \mathbf{C}_2^3 + O(h^{2t}) = \left(0, \frac{5}{4}C_2^2, \frac{5}{2}C_3^2 \right) + O(h^{2t}). \quad (42)$$

Now, we can analyze the result of the WENO algorithm with the adapted optimal weights that we have obtained instead of the original optimal weights that appear in (20). A jump in the interval (x_{j+1}, x_{j+2}) produces that $I_1^2 = O(h^2)$, $I_2^2 = O(h^2)$ and $I_3^2 = O(1)$. Assuming again that ϵ is small enough, using the nonlinear optimal weights presented in (41) that are $(\tilde{C}_1^2, \tilde{C}_2^2, \tilde{C}_3^2) = (\frac{5}{3}C_1^2, \frac{5}{6}C_2^2, 0) + O(h^{2t})$ and having into account (19) then, $\tilde{C}_1^2 + \tilde{C}_2^2 + \tilde{C}_3^2 = \frac{5}{3}C_1^2 + \frac{5}{6}C_2^2 + O(h^{2t}) = 1 + O(h^{2t})$, and,

$$\begin{aligned}\omega_1^2 &= \frac{\tilde{C}_1^2}{(\epsilon + I_1^2)^t} \frac{1}{\sum_{i=0}^{r-1} \frac{\tilde{C}_i^2}{(\epsilon + I_i^2)^t}} = \frac{\tilde{C}_1^2}{(\epsilon + I_1^2)^t} \frac{1}{\frac{1}{(\epsilon + I_1^2)^t} (\tilde{C}_1^2 + \tilde{C}_2^2 (1 + O(h^2)) + O(h^{2t}))} \\ &= \frac{\tilde{C}_1^2}{1 + O(h^2)} = \tilde{C}_1^2 + O(h^2), \\ \omega_2^2 &= \frac{\tilde{C}_2^2}{(\epsilon + I_2^2)^t} \frac{1}{\sum_{i=0}^{r-1} \frac{\tilde{C}_i^2}{(\epsilon + I_i^2)^t}} = \frac{\tilde{C}_2^2}{(\epsilon + I_2^2)^t} \frac{1}{\frac{1}{(\epsilon + I_2^2)^t} (\tilde{C}_1^2 (1 + O(h^2)) + \tilde{C}_2^2 + O(h^{2t}))} \\ &= \frac{\tilde{C}_2^2}{1 + O(h^2)} = \tilde{C}_2^2 + O(h^2), \\ \omega_3^2 &= \frac{\tilde{C}_3^2}{(\epsilon + I_3^2)^t} \frac{1}{\sum_{i=0}^{r-1} \frac{\tilde{C}_i^2}{(\epsilon + I_i^2)^t}} = \frac{\tilde{C}_3^2}{(\epsilon + I_3^2)^t} \frac{1}{\frac{1}{(\epsilon + I_1^2)^t} (\tilde{C}_1^2 + \tilde{C}_2^2 (1 + O(h^2)) + O(h^{2t}))} \\ &= \frac{\tilde{C}_3^2}{O(1)} \frac{1}{O(h^{2t}) (1 + O(h^2))} = O(h^{2t}).\end{aligned} \quad (43)$$

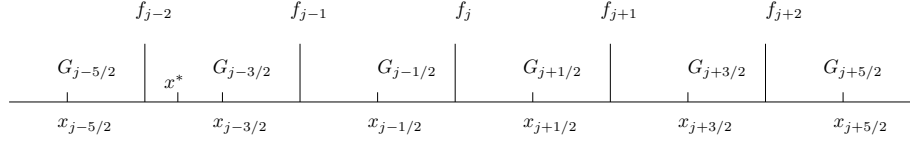


Figure 2: Representation of a discontinuity in the flux placed at x^* in the interval $(x_{j-5/2}, x_{j-3/2})$, that affects the cell average value $\bar{f}_{j-2}$. We have also represented the primitive values $G_{j-n/2}$. We can observe that in this particular case there are two smooth stencils of three cells.

- If there is a jump in the function affecting the flux in the intervals $(x_{j+1/2}, x_{j+3/2})$ or $(x_{j-3/2}, x_{j-1/2})$, the best order of accuracy that can be obtained is $O(h^3)$ as there is only one smooth stencil. This is precisely the order that the classical WENO algorithm reaches in this case. The adapted optimal weights (28) for this case are,

$$\tilde{\omega}_1^3 = O(1), \quad \tilde{\omega}_2^3 = O(1). \quad (44)$$

In what follows we analyze the case when the discontinuity is placed in the interval $(x_{j+1/2}, x_{j+3/2})$. When the discontinuity falls in the interval $(x_{j-3/2}, x_{j-1/2})$ we have the symmetric case. Replacing (44) in (31) we obtain that $(\tilde{C}_1^2, \tilde{C}_2^2, \tilde{C}_3^2) = (O(1), O(1), O(1))$. Now, if ϵ is small enough, the WENO weights in (20) are,

$$\begin{aligned} \omega_1^2 &= \frac{\tilde{C}_1^2}{(\epsilon + I_1^2)^t} \frac{1}{\sum_{i=0}^{r-1} \frac{\tilde{C}_i^2}{(\epsilon + I_i^2)^t}} = \frac{\tilde{C}_1^2}{(\epsilon + I_1^2)^t} \frac{1}{\frac{\tilde{C}_1^2}{(\epsilon + I_1^2)^t} (1 + O(h^{2t}))} = 1 + O(h^{2t}), \\ \omega_2^2 &= \frac{\tilde{C}_2^2}{(\epsilon + I_2^2)^t} \frac{1}{\sum_{i=0}^{r-1} \frac{\tilde{C}_i^2}{(\epsilon + I_i^2)^t}} = \frac{\tilde{C}_2^2}{(\epsilon + I_2^2)^t} \frac{1}{\frac{\tilde{C}_1^2}{(\epsilon + I_1^2)^t} (1 + O(h^{2t}))} \\ &= \frac{\tilde{C}_2^2}{O(1)} \frac{1}{\frac{\tilde{C}_1^2}{O(h^{2t})} (1 + O(h^{2t}))} = O(h^{2t}), \\ \omega_3^2 &= \frac{\tilde{C}_3^2}{(\epsilon + I_3^2)^t} \frac{1}{\sum_{i=0}^{r-1} \frac{\tilde{C}_i^2}{(\epsilon + I_i^2)^t}} = \frac{\tilde{C}_3^2}{(\epsilon + I_3^2)^t} \frac{1}{\frac{\tilde{C}_1^2}{(\epsilon + I_1^2)^t} (1 + O(h^{2t}))} \\ &= \frac{\tilde{C}_3^2}{O(1)} \frac{1}{\frac{\tilde{C}_1^2}{O(h^{2t})} (1 + O(h^{2t}))} = O(h^{2t}). \end{aligned} \quad (45)$$

This result means that the two last substencils of the WENO algorithm receive a weight that is close to zero while the first one receives a weight that is very close to 1.

- Finally, the case depicted in Figure 3 results in a loss of accuracy due to the fact that the discontinuity is placed at the central interval of the stencil and can not be avoided unless a subcell-resolution like strategy is used.

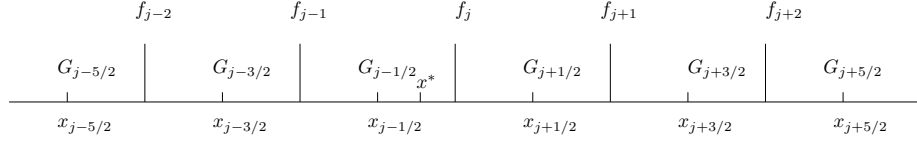


Figure 3: Representation of a discontinuity in the flux placed at x^* in the interval $(x_{j-1/2}, x_{j+1/2})$, that affects the cell average value $\bar{f}_j$. We have also represented the primitive values $G_{j-n/2}$. In this case we can observe that all the stencils of three, four and five cells are affected by the discontinuity.

Let us consider the stencil $S_r^{2r} = \{x_{j-5/2}, x_{j-3/2}, x_{j-1/2}, x_{j+1/2}, x_{j+3/2}, x_{j+5/2}\}$ and the cell average values $\{\bar{f}_{j-2}, \bar{f}_{j-1}, \bar{f}_j, \bar{f}_{j+1}, \bar{f}_{j+2}\}$. Now, we can prove the following theorem about the weights, that will also provide us information about the value of t and how small ϵ must be.

Theorem 1. *Let us assume that $r = 3$, $t \geq 1$, $\epsilon \leq h^2$ and that the grid spacing h is small enough such that there is only one discontinuity in the considered big stencil. The following cases must be taken into account regarding the accuracy of the new algorithm when approximating the derivative of the flux:*

- If the nonlinear optimal weights satisfy the following relation at smooth zones where,

$$(\tilde{C}_1^2, \tilde{C}_2^2, \tilde{C}_3^2) = \tilde{\omega}_1^3 \mathbf{C}_1^3 + \tilde{\omega}_2^3 \mathbf{C}_2^3 = \mathbf{C}_1^4 + O(h^2),$$

with $\mathbf{C}_1^4 = (C_1^2, C_2^2, C_3^2)$, then $\sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) = \hat{f}(x_{j-1/2}) + O(h^5)$ and $f(u(x))_x = \frac{1}{h}(\hat{f}(x_{j+1/2}) - \hat{f}(x_{j-1/2})) + O(h^4)$.

- If there is a discontinuity in the interval $(x_{j-5/2}, x_{j-3/2})$ and the nonlinear optimal weights satisfy the following relation,

$$(\tilde{C}_1^2, \tilde{C}_2^2, \tilde{C}_3^2) = \tilde{\omega}_1^3 \mathbf{C}_1^3 + \tilde{\omega}_2^3 \mathbf{C}_2^3 = \mathbf{C}_2^3 + O(h^2),$$

with $\mathbf{C}_2^3 = (0, \frac{5}{4}C_2^2, \frac{5}{2}C_3^2)$, then $\sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) = \hat{f}(x_{j-1/2}) + O(h^4)$ and $f(u(x))_x = \frac{1}{h}(\hat{f}(x_{j+1/2}) - \hat{f}(x_{j-1/2})) + O(h^3)$.

- If there is a discontinuity in the interval $(x_{j+3/2}, x_{j+5/2})$ and the nonlinear optimal weights satisfy the following relation,

$$(\tilde{C}_1^2, \tilde{C}_2^2, \tilde{C}_3^2) = \tilde{\omega}_1^3 \mathbf{C}_1^3 + \tilde{\omega}_2^3 \mathbf{C}_2^3 = \mathbf{C}_1^3 + O(h^2),$$

with $\mathbf{C}_1^3 = (\frac{5}{3}C_1^2, \frac{5}{6}C_2^3, 0)$, then $\sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) = \hat{f}(x_{j-1/2}) + O(h^4)$ and $f(u(x))_x = \frac{1}{h}(\hat{f}(x_{j+1/2}) - \hat{f}(x_{j-1/2})) + O(h^3)$.

- If there is a discontinuity in the intervals $(x_{j-3/2}, x_{j-1/2})$ or $(x_{j+1/2}, x_{j+3/2})$, then the nonlinear optimal weights satisfy the following relation,

$$(\tilde{C}_1^2, \tilde{C}_2^2, \tilde{C}_3^2) = \tilde{\omega}_1^3 \mathbf{C}_1^3 + \tilde{\omega}_2^3 \mathbf{C}_2^3 = (O(1), O(1), O(1)),$$

and $\sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) = \hat{f}(x_{j-1/2}) + O(h^3)$ and $f(u(x))_x = \frac{1}{h}(\hat{f}(x_{j+1/2}) - \hat{f}(x_{j-1/2})) + O(h^2)$.

Thus, the new WENO algorithm is at least as good as classical WENO algorithm close to discontinuities.

Proof.

- We can start proving the first statement of the theorem. In (27) we can see that the vector $\mathbf{C}_1^4$ has as coordinates the $C_k^r(j)$ in (19). The error of the interpolation in (12) at $x_{j-1/2}$ is,

$$\begin{aligned} \sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2} &= \sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2} + \sum_{k=1}^r C_k^r p_k^r(x_{j-1/2}) \\ &\quad - \sum_{k=1}^r C_k^r p_k^r(x_{j-1/2}), \end{aligned}$$

with C_k^r being the WENO optimal weights in (19). After simple algebraic manipulations of the previous expression we obtain,

$$\begin{aligned} \sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2} &= \sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) - \sum_{k=1}^r C_k^r p_k^r(x_{j-1/2}) \\ &\quad + \sum_{k=1}^r C_k^r p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2} \\ &= \sum_{k=1}^r (\omega_k^r - C_k^r) p_k^r(x_{j-1/2}) + O(h^{2r-1}). \end{aligned}$$

The fact that $\sum_{k=1}^r \omega_k^r = \sum_{k=1}^r C_k^r = 1$, allows us to write,

$$\begin{aligned} \sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2} &= \sum_{k=1}^r (\omega_k^r - C_k^r) p_k^r(x_{j-1/2}) + O(h^{2r-1}) \\ &\quad + \sum_{k=1}^r (\omega_k^r - C_k^r) \hat{f}_{j-1/2} \\ &= \sum_{k=1}^r (\omega_k^r - C_k^r) (p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2}) + O(h^{2r-1}) \end{aligned}$$

The expression in (40) assures that $(\omega_k^r - C_k^r) = O(h^m)$ with $m = 2$, so

$$\sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2} = O(h^m) O(h^r) + O(h^{2r-1}) = O(h^{\min(m+r, 2r-1)}). \quad (46)$$

Then, it is clear that for $r = 3$, we obtain accuracy $O(h^5)$ at smooth zones when approximating $\hat{f}_{j-1/2}$, that is optimal as we are using five cells. When applying the expression in (8) we typically obtain $O(h^4)$ accuracy for the approximation of the derivative of the flux $f(u(x))_x$, the same as classical WENO algorithm.

- Let us continue with the second statement. The expression in (27) determines the value of the vector of weights in this case, $\mathbf{C}_2^3 = (0, \frac{5}{4}C_2^2, \frac{5}{2}C_3^2) = (0, \frac{3}{4}, \frac{1}{4})$. The proof for the order accuracy of the approximation at $\hat{f}_{j-1/2}$ is similar to the one exposed in the previous point,

$$\begin{aligned} \sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2} &= \sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2} + \sum_{k=1}^r \mathbf{C}_2^3(k) p_k^r(x_{j-1/2}) \\ &- \sum_{k=1}^r \mathbf{C}_2^3(k) p_k^r(x_{j-1/2}). \end{aligned}$$

After some algebraic manipulations we obtain,

$$\begin{aligned} \sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2} &= \sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) - \sum_{k=1}^r \mathbf{C}_2^3(k) p_k^r(x_{j-1/2}) \\ &+ \sum_{k=1}^r \mathbf{C}_2^3(k) p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2} \\ &= \sum_{k=1}^r (\omega_k^r - \mathbf{C}_2^3(k)) p_k^r(x_{j-1/2}) + O(h^{2r-2}). \end{aligned}$$

Now, as $\sum_{k=1}^r \omega_k^r = \sum_{k=1}^r \mathbf{C}_2^3(k) = 1$ we obtain,

$$\begin{aligned} \sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2} &= \sum_{k=1}^r (\omega_k^r - \mathbf{C}_2^3(k)) (p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2}) + O(h^{2r-2}) \\ &= O(h^m) O(h^r) + O(h^{2r-2}) = O(h^{\min(m+r, 2r-2)}). \end{aligned} \tag{47}$$

From the expressions in (42) and (43), we reach the conclusion that with $t \geq 1$ we obtain that $(\omega_k^r - \mathbf{C}_2^3(k)) = O(h^m)$ with $m = 2$. For $r = 3$, the precision of the approximation of $\hat{f}_{j-1/2}$ is $O(h^4)$ that is optimal when the stencil presents four smooth cells. If we apply the expression in (8) we typically obtain $O(h^3)$ accuracy for the approximation of the derivative of the flux $f(u(x))_x$, while classical WENO algorithm typically obtains $O(h^2)$ order of accuracy.

- The second and third statements of the theorem are symmetric, so following the previous point it is straightforward to obtain the proof of the third statement.
- The fourth statement of the theorem considers a discontinuity in the interval $(x_{j-3/2}, x_{j-1/2})$ or $(x_{j+1/2}, x_{j+3/2})$. Both cases are symmetric. As only one substencil of three cells is smooth, the classical WENO algorithm obtains optimal accuracy. In what follows, we consider a discontinuity in the interval (x_j, x_{j+1}) . The expression in (45) leads us to the vector of weights $\mathbf{C} = (1, 0, 0)$. Reproducing the steps followed in the previous points,

$$\begin{aligned} \sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2} &= \sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2} + \sum_{k=1}^r \mathbf{C}(k) p_k^r(x_{j-1/2}) \\ &- \sum_{k=1}^r \mathbf{C}(k) p_k^r(x_{j-1/2}). \end{aligned}$$

After some algebraic manipulations we arrive to,

$$\begin{aligned}
\sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2} &= \sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) - \sum_{k=1}^r \mathbf{C}(k) p_k^r(x_{j-1/2}) \\
&+ \sum_{k=1}^r \mathbf{C}(k) p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2} \\
&= \sum_{k=1}^r (\omega_k^r - \mathbf{C}(k)) p_k^r(x_{j-1/2}) + O(h^r).
\end{aligned}$$

As $\sum_{k=1}^r \omega_k^r = \sum_{k=1}^r \mathbf{C}(k) = 1$, we can write,

$$\begin{aligned}
\sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2} &= \sum_{k=1}^r (\omega_k^r - \mathbf{C}(k)) (p_k^r(x_{j-1/2}) - \hat{f}_{j-1/2}) + O(h^r) \\
&= O(h^m) O(h^{r+1}) + O(h^r) = O(h^{\min(m+r, r)}).
\end{aligned} \tag{48}$$

The expression in (45) assures that $m = 2t$ and for $r = 3$, with $t \geq 1$ we obtain accuracy $O(h^3)$ for the approximation of $\hat{f}_{j-1/2}$ if we find a discontinuity in the intervals $(x_{j-3/2}, x_{j-1/2})$ or $(x_{j+1/2}, x_{j+3/2})$. Of course, applying (8) we obtain $O(h^2)$ accuracy for the approximation of the derivative of the flux $f(u(x))_x$, the same as classical WENO algorithm.

□

Remark 1. As mentioned in [8], in practice, in the prediction of the flux at the cell interfaces obtained through $\sum_{k=1}^r \omega_k^r p_k^r(x_{j-1/2}) = \hat{f}(x_{j-1/2}) + O(h^k)$ the term $O(h^k)$ is usually smooth, so the difference of the conservative derivative of the flux in (8) gives an extra h , so the one in the denominator of (8) would cancel out. This means that the prediction of the flux at the cell interfaces is in practice $O(h^k)$, the same as the derivative of the flux at the middle of the cells. Thus, in practice, we would obtain one extra order of accuracy in all the statements of Theorem 1 for the derivative of the flux $f(u(x))_x$.

Remark 2. A small enough value of ϵ in (20) and in (29) is a value of order $O(h^2)$, as this is the minimum value of the smoothness indicators (22), (32), that is reached at smooth zones, as can be seen from (36).

In the case of the smoothness indicators in (33), following the same process but using these order indices, the same result is obviously attained. In this case ϵ should be the minimum value obtained by the smoothness indicator, that is $\epsilon = O(h^4)$ in this case.

Theorem 2. The new algorithm satisfies the ENO property for $t \geq 2$, satisfying at the same time Theorem 1.

Proof. For $t \geq 1$ Theorem 1 is satisfied, so for $t \geq 2$ it is also satisfied. From (40), (43) and (45) we can see that for $t \geq 2$:

- If the function f is smooth at the stencil S_k^r , then the weight related to this stencil will verify $w_k^r = O(1)$.
- If the function f has a singularity at the stencil S_k^r , then the corresponding w_k^r will verify $w_k^r \leq O(h^r)$.

That is the ENO property.

□

7. Numerical experiments

7.1. Advection equation

In what follows, we present some results for the advection equation. We try to solve the linear equation,

$$\begin{aligned} u_t + u_x &= 0, & x_l < x < x_r, \\ u(x, 0) &= u_0(x). \end{aligned} \quad (49)$$

We compare the results of the classical WENO algorithm with the results obtained by the WENO algorithm plus the new weights. In Table 1 we show the errors for the periodic initial condition $u_0(x) = \sin(\pi x)$. Table 2 presents the results for the the periodic initial condition $u_0(x) = \sin^4(\pi x)$.

Method	N	L_∞ error	L_∞ order	L_1 error	L_1 order
New WENO	10	0.0022472	-	0.0012545	-
	20	6.2591e-05	5.1660	2.9669e-05	5.4020
	40	1.6583e-06	5.2382	8.1232e-07	5.1908
	80	4.8948e-08	5.0823	2.3915e-08	5.0861
	160	1.4283e-09	5.0989	7.2596e-10	5.0419
	320	4.311e-11	5.0501	2.2365e-11	5.0206
WENO	10	0.0022978	-	0.001283	-
	20	6.3346e-05	5.1809	2.9852e-05	5.4255
	40	1.6717e-06	5.2439	8.1232e-07	5.1996
	80	4.9305e-08	5.0834	2.3915e-08	5.0861
	160	1.4373e-09	5.1003	7.2596e-10	5.0419
	320	4.3355e-11	5.0510	2.2365e-11	5.0206

Table 1: Grid refinement analysis for the unidimensional advection equation with the periodical initial condition $u_0(x) = \sin(\pi x)$.

The results obtained in Tables 1 and 2 show that at smooth zones the new algorithm behaves essentially the same as WENO: the errors are very similar and both attain optimal accuracy. Now we can try to check what happens when the initial condition presents singularities. An initial analysis of the behavior that we can expect can be done now. Given a set of discrete data, the new algorithm is designed to deal with sharp discontinuities with the aim of reaching the maximum theoretical accuracy. If the data is highly polluted due to the diffusion introduced by the time solver, then we can not expect to obtain much better results than WENO: the results should be similar for both algorithms. As mentioned before, the diffusion is introduced by the time solver so less time steps should imply less diffusion. Thus we can expect to obtain better results with the new algorithm when the CFL number is close to instability. In fact, the best results that we obtain are for $CFL = 0.9$. For smaller CFL numbers the results for both algorithms are similar and we do not obtain a real advantage using the new algorithm. Having these considerations in mind, when the initial condition presents discontinuities, we can expect to obtain better results at the beginning of the simulation and then lose the inherent advantage of the new algorithm as the diffusion pollutes the data. As a consequence, rising the spatial resolution fixing the CFL condition will produce that the region of improvement will compress close to the beginning of the simulation, as there is one time instant when the diffusion is so high that it is not possible to obtain any order improvement close to the

Method	N	L_∞ error	L_∞ order	L_1 error	L_1 order
New WENO	10	0.17667	-	0.073823	-
	20	0.011121	3.9897	0.0044103	4.0651
	40	0.0020331	2.4515	0.00057164	2.9477
	80	0.00012639	4.0077	2.2848e-05	4.6450
	160	4.2155e-06	4.9060	6.1679e-07	5.2111
	320	9.1772e-08	5.5215	1.5433e-08	5.3207
WENO	10	0.17129	-	0.069965	-
	20	0.010733	3.9963	0.0043483	4.0081
	40	0.0019159	2.4860	0.00055172	2.9784
	80	0.0001244	3.9450	2.257e-05	4.6115
	160	4.2135e-06	4.8838	6.1659e-07	5.1940
	320	9.177e-08	5.5209	1.5433e-08	5.3202

Table 2: Grid refinement analysis for the unidimensional advection equation with the periodical initial condition $u_0(x) = \sin^4(\pi x)$.

discontinuities. If the initial condition presents a discontinuity, the infinity norm of the error is dominated by the errors at the discontinuity, so we will not find any improvement in this norm. The main improvement can be observed in the L_1 norm.

In all the experiments that we present in this section, we will use a total time of simulation equal to $t_f = 14$ seconds. We have set $\epsilon = 10^{-6}$ for both algorithms in order to obtain the weights (20), but $\epsilon = 10^{-8}$ in (29) for the new WENO algorithm in order to obtain the nonlinear optimal weights. Even though, the results do not present much variation if we set all the epsilons to 10^{-6} . We obtain worse results for the classical WENO algorithm if $\epsilon < 10^{-6}$ in (20). In what follows we present some results obtained for the advection equation with the configuration explained before for the new algorithm and the classical WENO. In all the cases, if the spatial domain is not specified in the experiment, we have solved (49) with $x_l = -1$ and $x_r = 6\pi$ with periodical boundary conditions. We have obtained similar results for all the initial conditions with discontinuities that we have tested.

Let us start by with the initial condition in (50),

$$f(x) = \begin{cases} 1, & \text{if } -0.5 \leq x < 0.5 \\ 0, & \text{other case .} \end{cases} \quad (50)$$

In Figure 4 we can see how the L^1 norm of the error for the new WENO algorithm and the classical WENO behaves for different spatial resolutions as the time advances. We can see that the new algorithm attains a smaller L^1 norm of the error at the beginning of the simulations, exactly as it was described before. As we rise the spatial resolution, the gain obtained gets reduced. Figure 5 shows the L^1 norm of the error for both algorithms and different CFL conditions. The configuration is the same as before but now we vary the CFL condition fixing the spatial resolution to $n = 200$ points. We can see how the new algorithm only obtains better results when the CFL condition is set to 0.9. For the rest of the cases, we obtain similar results.

Let us continue with the initial conditions shown in (51), (52), and (53),

$$f(x) = \begin{cases} 1 + 2x, & \text{if } -0.5 \leq x < 0 \\ 1 - 2x, & \text{if } 0 \leq x < 0.5. \end{cases} \quad (51)$$

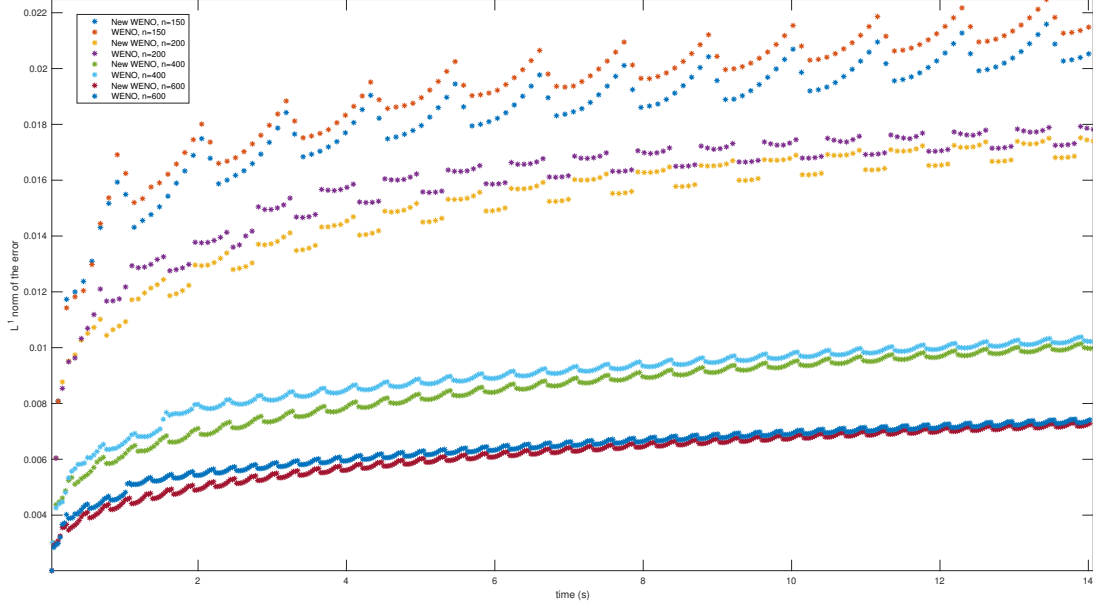


Figure 4: L^1 norm of the error for the new WENO algorithm and the classical WENO for different spatial resolutions as the time advances in the simulation of the advection equation for the initial condition in (50). We have set $CFL = 0.9$ and the final time of the simulation to $t = 14$ seconds.

$$f(x) = \begin{cases} 2 + 2x, & \text{if } -0.5 \leq x < 0 \\ 2 + 2x, & \text{if } 0 \leq x < 0.5. \end{cases} \quad (52)$$

$$f(x) = \begin{cases} \sin(2x), & \text{if } 0 \leq x < \frac{\pi}{2} \\ 0, & \text{other case.} \end{cases} \quad (53)$$

The results for the L^1 norm versus time of previous initial conditions using the new algorithm and the classical implementation of the WENO algorithm can be observed in Figures 6, 7, and 8 respectively. The conclusions are similar to those obtained in the first experiment. Finally, Figure 9 shows where the new algorithm outperforms the results obtained by the classical implementation of WENO at a particular time instant. In this case we have set $x_l = -1$ and $x_r = 6$ in (49), the final time to $t_f = 4$, $CFL = 0.9$ and $n = 50$ points in order to force the errors of approximation to show at a glance. We can see how we obtain a better performance of the new algorithm close to the discontinuities.

7.2. Yang's artificial compression

In this Subsection we analyze the effect of including the Yang's artificial compression in the simulations. We can expect to obtain some improvement of the results obtained by the new algorithm, as the Yang's artificial compression is oriented to conserve the discontinuities sharp. Figures 10, 11, 12, 14, 15, 16 show the equivalent results to those presented in previous section (Figures 4, 5, 6, 7, 8, 9) but using the Yang's artificial compression. Figure 13 has been also included due to the good

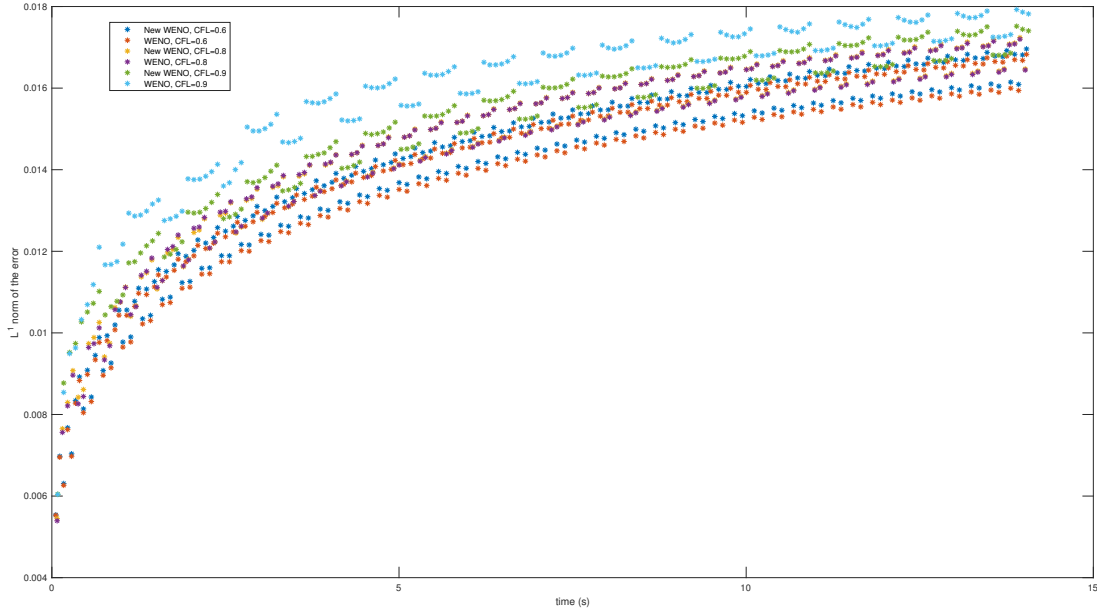


Figure 5: In this plot we present the L^1 norm of the error for the new WENO algorithm and the classical WENO for different CFL conditions. As in Figure 4 we have used the initial condition in (50) to solve the advection equation. All the configuration is the same but now we vary the CFL condition fixing the spatial resolution to $n = 200$ points.

results obtained for $CFL = 0.9$. We can see how the results are improved due to the preservation of discontinuities.

7.3. Advection in 2D

The results obtained when obtaining the numerical solution of the advection equation in 2D,

$$\begin{aligned} u_t + u_x + u_y &= 0, & x_l < x < x_r, & \quad y_p < y < y_d, \\ u(x, y, 0) &= u_0(x, y). \end{aligned} \tag{54}$$

are very similar to the ones obtained in previous sections, so the conclusions are similar.

8. Conclusions

In this article we have presented and analyzed an strategy that allows to improve the results obtained by WENO algorithm when approximating the solution of hyperbolic conservation laws. This strategy consists in a nonlinear redesign of the WENO optimal weights. This new strategy allows to control the order of accuracy of the interpolation close to the discontinuity but not in the interval that contains it. The new strategy can be designed in a way that an algorithm of $2r - 1$ cells can be obtained easily. The numerical experiments presented confirm all the theoretical conclusions reached.

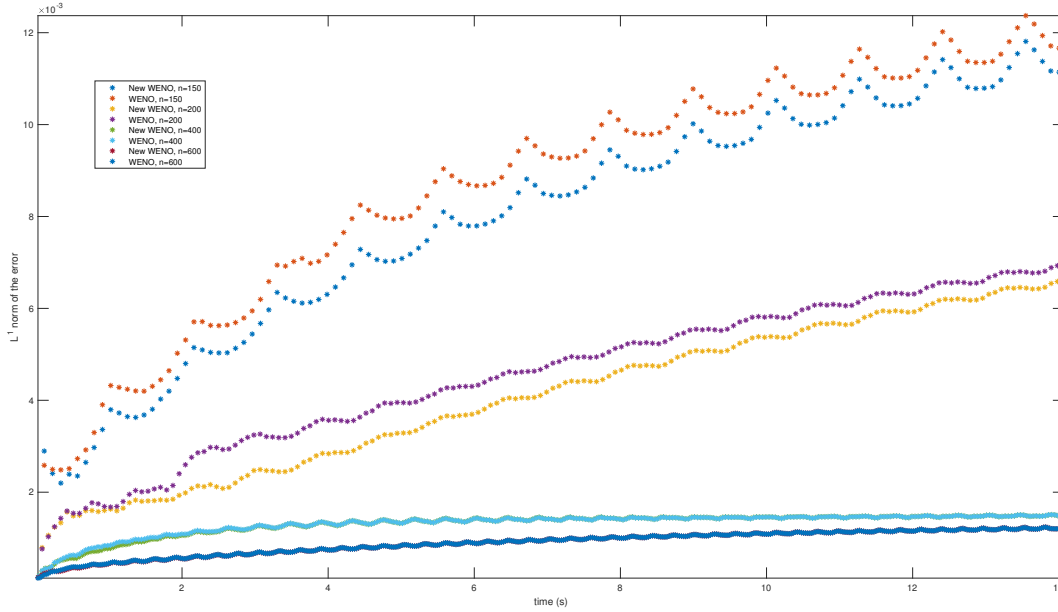


Figure 6: L^1 norm of the error for the new WENO algorithm and the classical WENO for different spatial resolutions as the time advances in the simulation of the advection equation for the initial condition in (51). We have set $CFL = 0.9$ and the final time of the simulation to $t = 14$ seconds.

References

- [1] S. Amat, J. Ruiz, C.-W. Shu, On new strategies to control the accuracy of WENO algorithms close to discontinuities, *SIAM J. Numer. Anal.* 57 (3) (2019) 1205 – 1237.
- [2] S. Amat, J. Ruiz, C.-W. Shu, On new strategies to control the accuracy of WENO algorithm close to discontinuities II: Cell averages, (Submitted).
- [3] X.-D. Liu, S. Osher, T. Chan, Weighted essentially non-oscillatory schemes, *J. Comput. Phys.* 115 (1) (1994) 200 – 212.
- [4] G. Jiang, C. Shu, Efficient implementation of weighted ENO schemes, *J. Comput. Phys.* 126 (1) (1996) 202 – 228.
- [5] A. Harten, B. Engquist, S. Osher, S. R. Chakravarthy, Uniformly high order accurate essentially non-oscillatory schemes, III, *J. Comput. Phys.* 71 (2) (1987) 231 – 303.
- [6] A. Harten, S. Osher, Uniformly high-order accurate nonoscillatory schemes. I, *SIAM J. Numer. Anal.* 24 (2) (1987) 279–309.
- [7] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, *J. Comput. Phys.* 77 (2) (1988) 439 – 471.
- [8] C.-W. Shu, *High Order ENO and WENO Schemes for Computational Fluid Dynamics*, Springer, Berlin, Heidelberg, 1999, pp. 439–582.

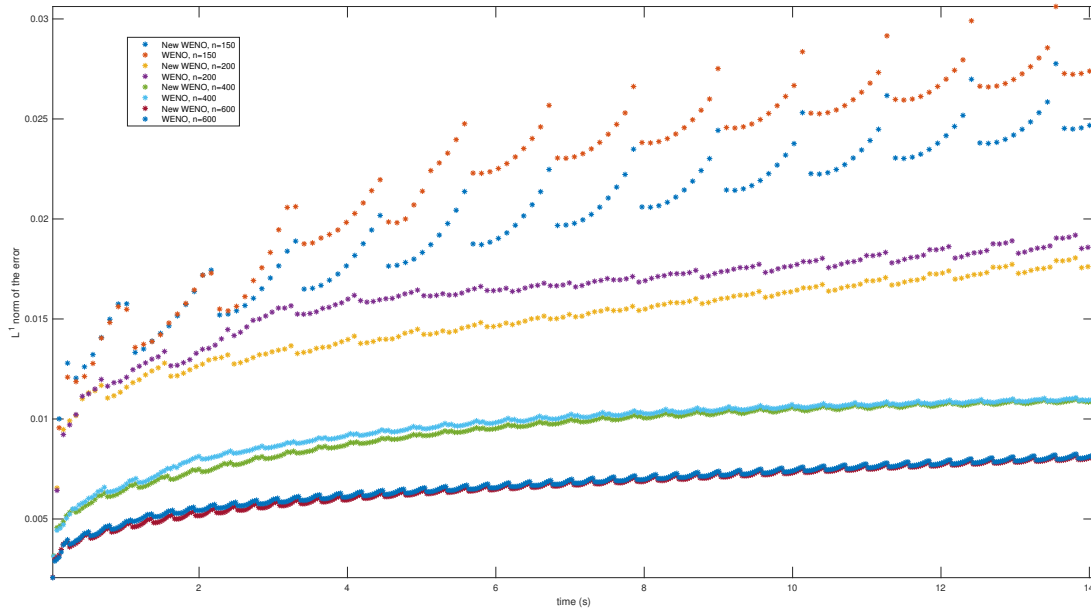


Figure 7: L^1 norm of the error for the new WENO algorithm and the classical WENO for different spatial resolutions as the time advances in the simulation of the advection equation for the initial condition in (52). We have set $CFL = 0.9$ and the final time of the simulation to $t = 14$ seconds.

- [9] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes II, J. Comput. Phys. 83 (1) (1989) 32 – 78.
- [10] F. Arandiga, A. Cohen, R. Donat, N. Dyn, B. Matei, Approximation of piecewise smooth functions and images by edge-adapted (ENO-EA) nonlinear multiresolution techniques, Appl. Comput. Harmon. Anal. 24 (2) (2008) 225 – 250, Special Issue on Mathematical Imaging – Part II.
- [11] S. Amat, F. Arandiga, A. Cohen, R. Donat, G. Garcia, M. von Oehsen, Data compression with ENO schemes: A case study, Appl. Comput. Harmon. Anal. 11 (2) (2001) 273 – 288.
- [12] S. Serna, A. Marquina, Power ENO methods: a fifth-order accurate weighted power ENO method, J. Comput. Phys. 194 (2) (2004) 632–658.
- [13] A. Cohen, N. Dyn, B. Matei, Quasi linear subdivision schemes with applications to ENO interpolation, Appl. Comput. Harmon. Anal. 15 (2003) 89–116.
- [14] S. Amat, S. Busquier, J. C. Trillo, On multiresolution schemes using a stencil selection procedure: applications to ENO schemes, Numer. Algorithms 44 (1) (2007) 45–68.
- [15] S. Amat, F. Arandiga, A. Cohen, R. Donat, G. Garcia, M. von Oehsen, Data compression with ENO schemes: A case study, Appl. Comput. Harmon. Anal. 11 (2) (2001) 273 – 288.
- [16] F. Arandiga, A. Baeza, A. M. Belda, P. Mulet, Analysis of WENO schemes for full and global accuracy, SIAM J. Numer. Anal. 49 (2) (2011) 893–915.

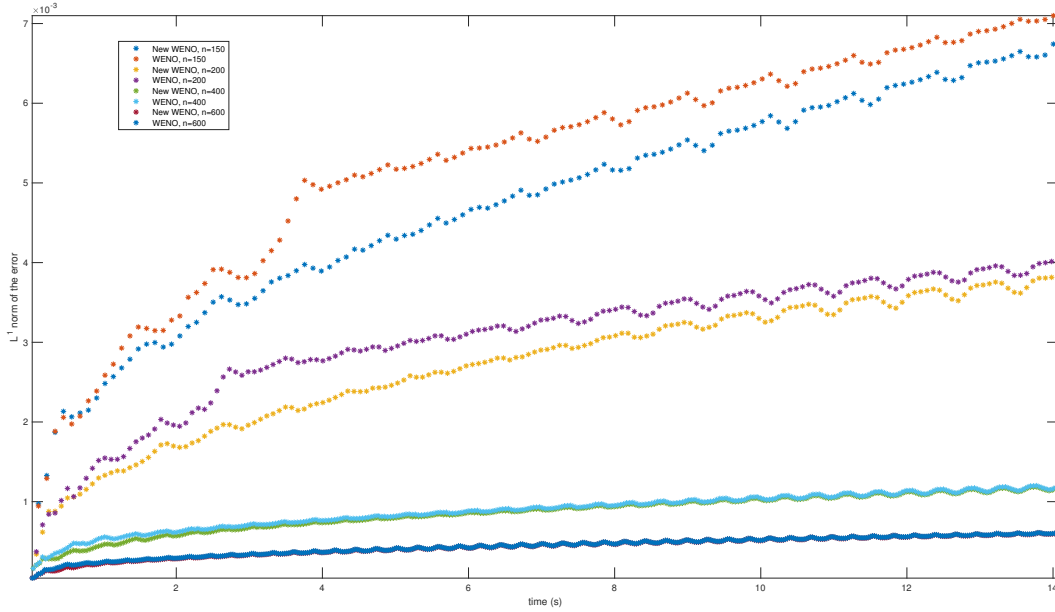


Figure 8: L^1 norm of the error for the new WENO algorithm and the classical WENO for different spatial resolutions as the time advances in the simulation of the advection equation for the initial condition in (53). We have set $CFL = 0.9$ and the final time of the simulation to $t = 14$ seconds.

- [17] F. Aràndiga, A. Belda, P. Mulet, Point-value WENO multiresolution applications to stable image compression, *J. Sci. Comput.* 43 (2) (2010) 158–182.
- [18] A. K. Henrick, T. D. Aslam, J. M. Powers, Mapped weighted essentially non-oscillatory schemes: Achieving optimal order near critical points, *J. Comput. Phys.* 207 (2) (2005) 542 – 567.
- [19] M. Castro, B. Costa, W. S. Don, High order weighted essentially non-oscillatory WENO-Z schemes for hyperbolic conservation laws, *J. Comput. Phys.* 230 (5) (2011) 1766 – 1792.
- [20] C.-W. Shu, *Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws*, Springer, Berlin, Heidelberg, 1998, pp. 325–432.
- [21] C.-W. Shu, High order weighted essentially nonoscillatory schemes for convection dominated problems, *SIAM Review* 51 (1) (2009) 82–126.
- [22] S. Amat, J. Ruiz, New WENO smoothness indicators computationally efficient in the presence of corner discontinuities, *Journal of Scientific Computing* 71 (3) (2017) 1265–1302.

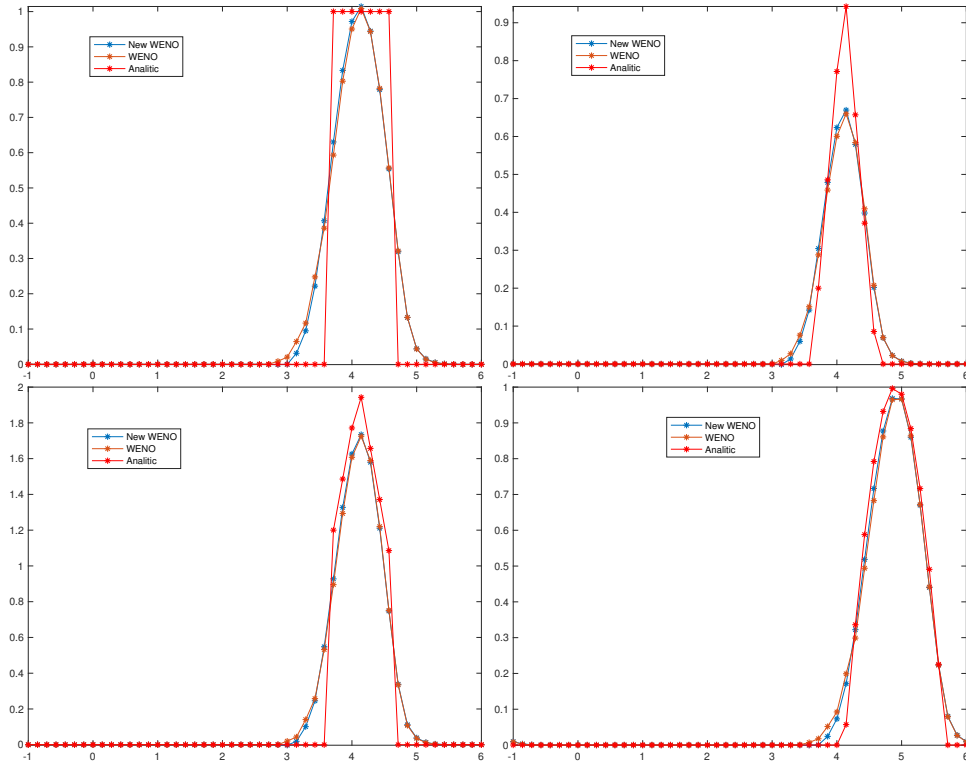


Figure 9: Numerical approximation of the solution of the advection equation for the new WENO algorithm and the classical WENO, for the initial conditions in (50) (top left), (51) (top right), (52) (bottom left), and (53) (bottom right). We have set $CFL = 0.9$ and the final time of the simulation to $t = 4$ seconds and $n = 50$ points.

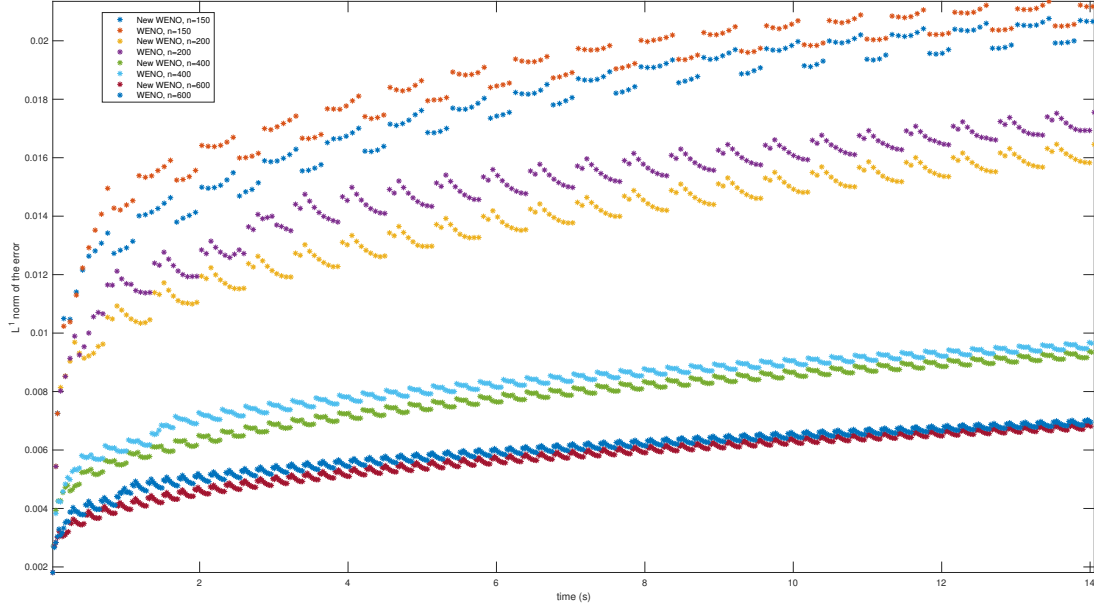


Figure 10: L^1 norm of the error for the new WENO algorithm and the classical WENO plus the Yang's artificial compression with $\alpha = 33$, for different spatial resolutions as the time advances in the simulation of the advection equation for the initial condition in (50). We have set $CFL = 0.9$ and the final time of the simulation to $t = 14s$.

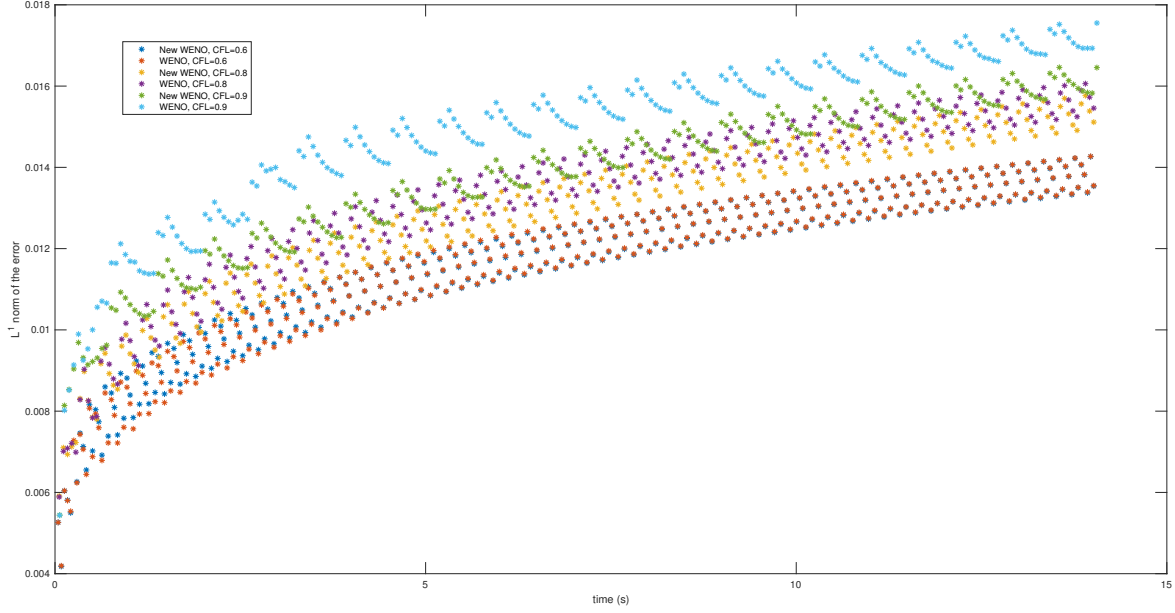


Figure 11: L^1 norm of the error for the new WENO algorithm and the classical WENO plus the Yang's artificial compression with $\alpha = 33$, for different CFL conditions. As in Figure 10 we have used the initial condition in (50) to solve (49). All the configuration is the same but now we vary the CFL fixing the spatial resolution to $n = 200$ points.

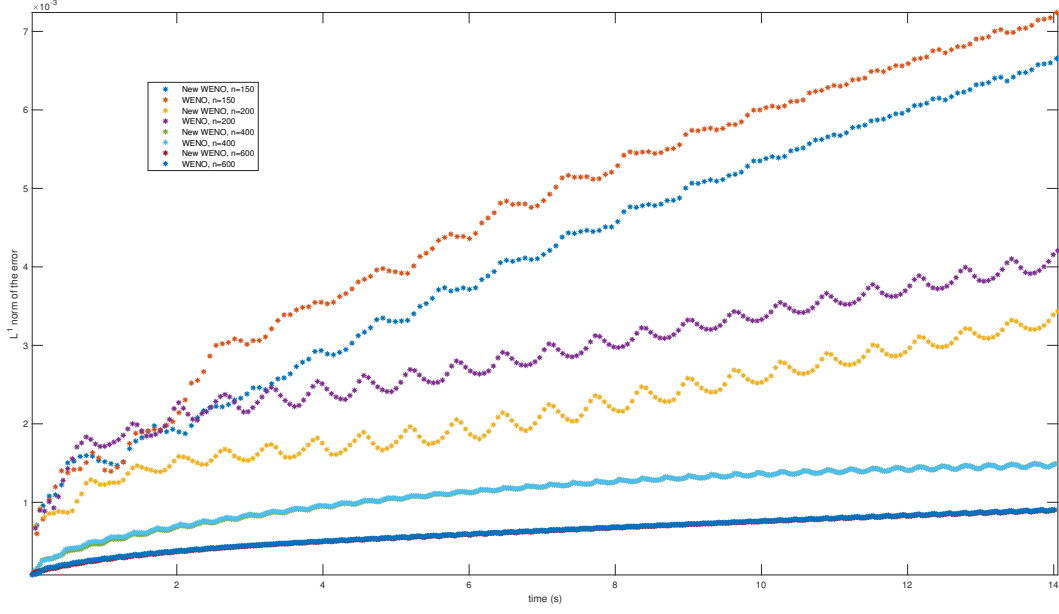


Figure 12: L^1 norm of the error for the new WENO algorithm and the classical WENO plus the Yang's artificial compression with $\alpha = 33$, for different spatial resolutions as the time advances in the simulation. We have used the initial condition in (51) to solve (49). We have set $CFL = 0.9$ and the final time of the simulation to $t = 14s$.

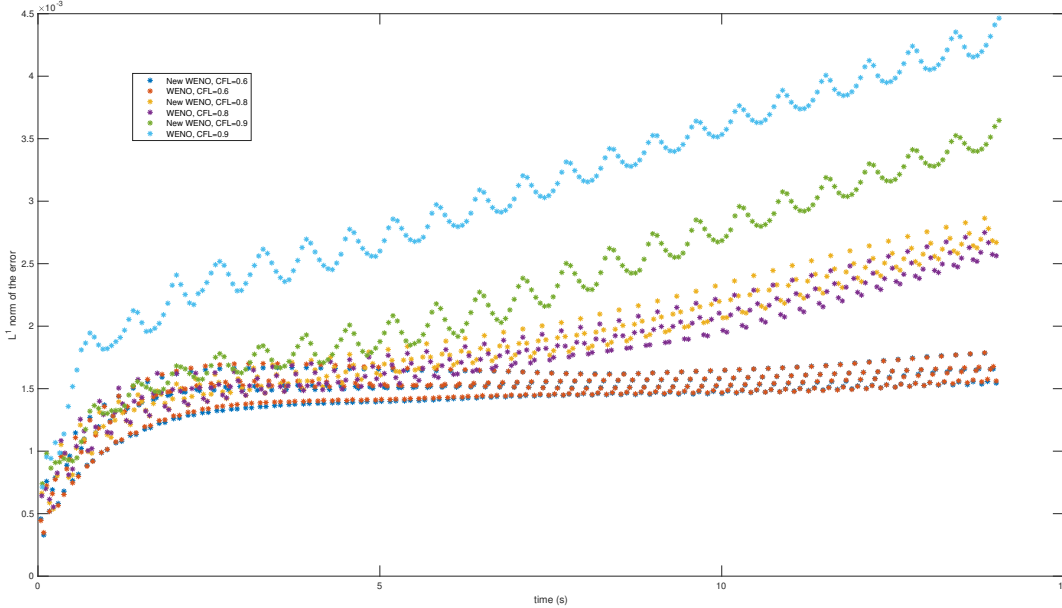


Figure 13: L^1 norm of the error for the new WENO algorithm and the classical WENO plus the Yang's artificial compression with $\alpha = 33$, for different CFL conditions. As in Figure 10 we have used the initial condition in (51) to solve (49). All the configuration is the same but now we vary the CFL fixing the spatial resolution to $n = 200$ points.

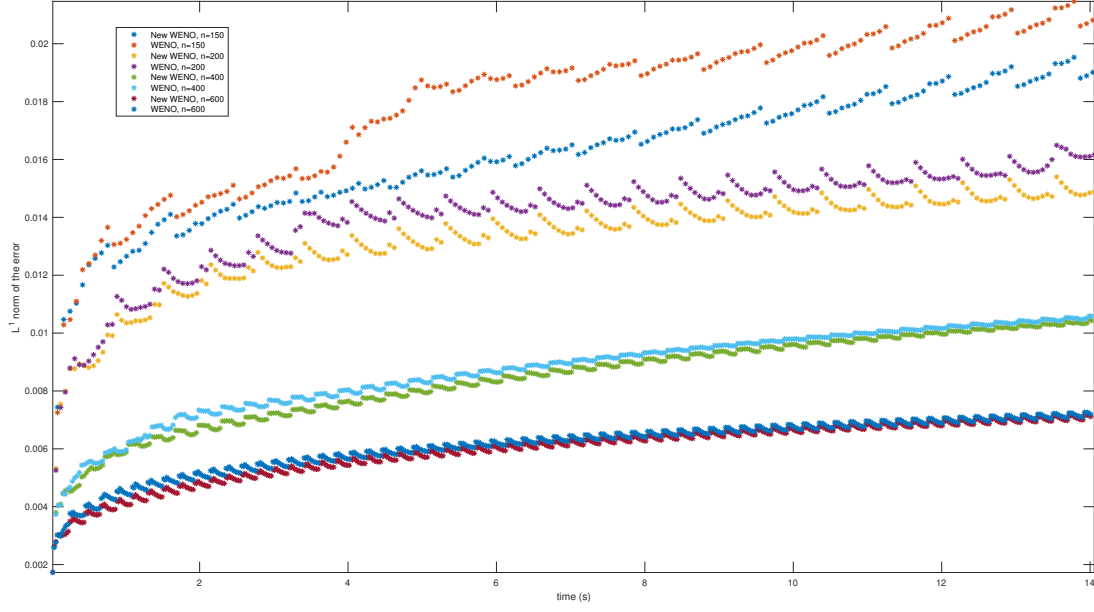


Figure 14: L^1 norm of the error for the new WENO algorithm and the classical WENO plus the Yang's artificial compression with $\alpha = 33$, for different spatial resolutions as the time advances in the simulation. We have used the initial condition in (52) to solve (49). We have set $CFL = 0.9$ and the final time of the simulation to $t = 14s$

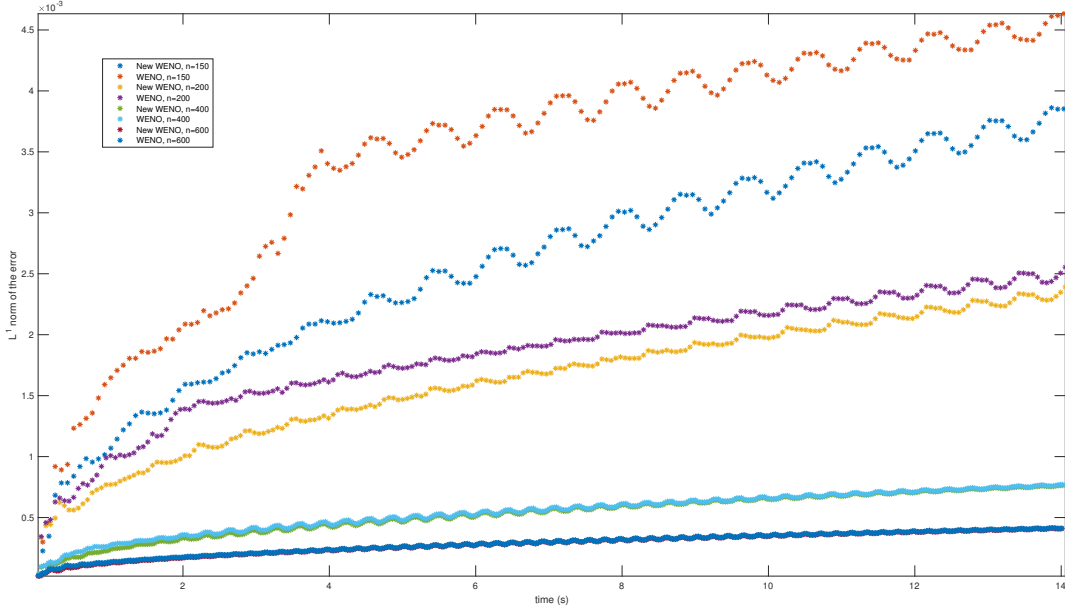


Figure 15: L^1 norm of the error for the new WENO algorithm and the classical WENO plus the Yang's artificial compression with $\alpha = 33$, for different spatial resolutions as the time advances in the simulation. We have used the initial condition in (53) to solve (49). We have set $CFL = 0.9$ and the final time of the simulation to $t = 14s$.

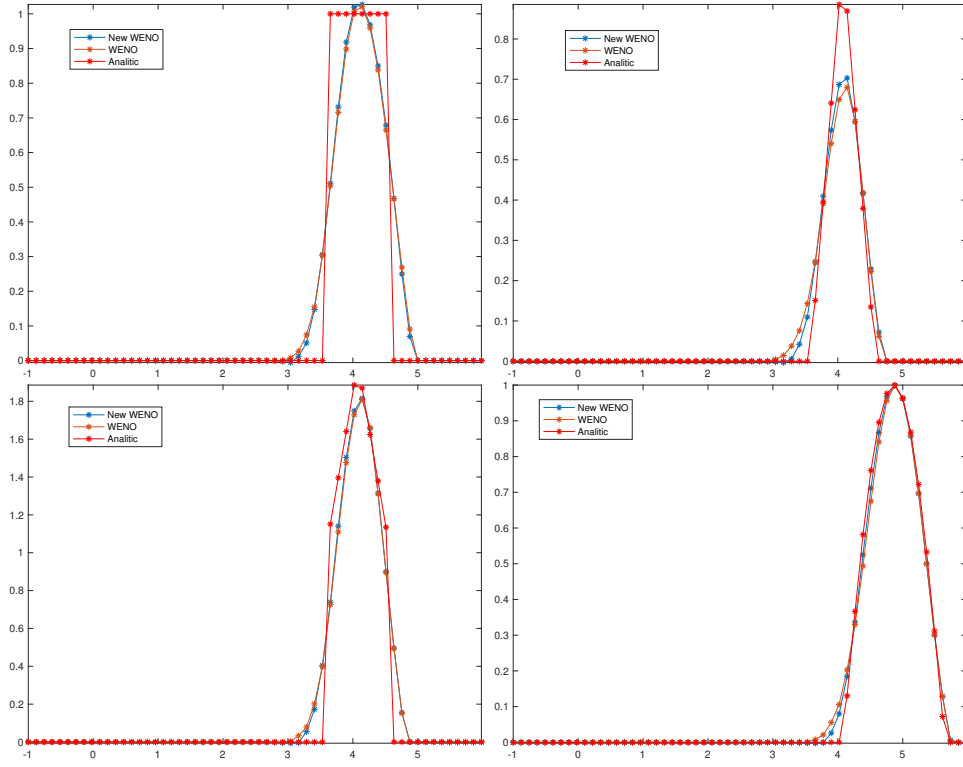


Figure 16: Numerical approximation of the solution of the advection equation for the new WENO algorithm and the classical WENO plus the Yang's artificial compression with $\alpha = 33$, for the initial conditions in (50) (top left), (51) (top right), (52) (bottom left), and (53) (bottom right). We have set $CFL = 0.9$ and the final time of the simulation to $t = 4$ seconds and $n = 50$ points.