Contents lists available at ScienceDirect

Applied Numerical Mathematics

www.elsevier.com/locate/apnum

A RBF-WENO finite volume method for hyperbolic conservation laws with the monotone polynomial interpolation method

Jingyang Guo, Jae-Hun Jung*

Department of Mathematics, State University of New York at Buffalo, Buffalo, NY 14260-2900, USA

ARTICLE INFO

Article history:
Received 29 March 2016
Received in revised form 21 July 2016
Accepted 5 October 2016
Available online 11 October 2016

Keywords:
Essentially non-oscillatory method
Weighted essentially non-oscillatory method
Radial basis function interpolation
Finite volume method
Hyperbolic conservation laws

ABSTRACT

Essentially non-oscillatory (ENO) and weighted ENO (WENO) methods are efficient high order numerical methods for solving hyperbolic conservation laws designed to reduce the Gibbs oscillations. The original ENO and WENO methods are based on the polynomial interpolation and the overall convergence rate is uniquely determined by the total number of interpolation points involved for the approximation. In this paper, we propose non-polynomial ENO and WENO finite volume methods in order to enhance the local accuracy and convergence. The infinitely smooth radial basis functions (RBFs) are adopted as a non-polynomial interpolation basis. Particularly we use the multi-quadratic and Gaussian RBFs. The non-polynomial interpolation such as the RBF interpolation offers the flexibility to control the local error by optimizing the free parameter. Then we show that the non-polynomial interpolation can be represented as a perturbation of the polynomial interpolation. To guarantee the essentially non-oscillatory property, the monotone polynomial interpolation method is introduced as a switching method to the polynomial reconstruction adaptively near the non-smooth area. The numerical results show that the developed non-polynomial ENO and WENO methods with the monotone polynomial interpolation method enhance the local accuracy and give sharper solution profile than the ENO/WENO methods based on the polynomial interpolation.

1. Introduction

Consider the hyperbolic conservation laws

$$v_t + \nabla \cdot F(v) = 0, \tag{1}$$

for the state vector $v \equiv v(t,x) : I \times \Omega \to \mathbb{R}^m$, where $I := (0, T]$ is a time interval with $T > 0$ and $\Omega \subset \mathbb{R}^d$ is an open bounded computational domain. $F(v) := [f_1(v), \cdots, f_m(v)]$ is the flux function. An initial condition $v_0(x) = v(x, 0)$ is given along with appropriate boundary conditions. Despite the smoothness of $v_0(x)$, the solution to (1) may develop a discontinuity within a finite time. High order numerical approximations of the developed discontinuity suffer from the Gibbs phenomenon yielding spurious oscillations near the discontinuity. Since the publications by Harten et al. [13] and by Jiang and Shu [15], the essentially non-oscillatory (ENO) and weighted essentially non-oscillatory (WENO) methods have been one of the

* Corresponding author.
E-mail addresses: jguo4@buffalo.edu (J. Guo), jaehun@buffalo.edu (J.-H. Jung).

http://dx.doi.org/10.1016/j.apnum.2016.10.003

Table 1
Commonly used radial basis functions $\phi(r)$, $r \geqslant 0$ with ϵ known as the shape parameter.

Infinitely smooth RBFs		Piecewise smooth RBFs	
Gaussian (GA)	$\exp(-(\epsilon r)^2)$	Polyharmonic spline	r^k, $k = 1, 3, 5, \ldots$
Multiquadratic (MQ)	$\sqrt{1+(\epsilon r)^2}$		$r^k \ln(r)$, $k = 2, 4, 6, \ldots$
Inverse quadratic (IQ)	$\frac{1}{1+(\epsilon r)^2}$		

most powerful numerical methods that can successfully deal with the Gibbs oscillations. Numerous modifications of the original ENO/WENO methods have been also developed, while resolving small scale structures accurately and efficiently. These include recent works such as WENO-M [14], WENO-Z [3], power-ENO [20], WENO-P [11], modification of the ENO basis [2] and WENO-η [8] methods, to name a few. There is no best ENO/WENO variation because all variations have their own strengths and weaknesses. However, most variations have a common ground: the polynomial reconstruction. In recent reviews of the WENO method by Shu [23], the WENO reconstruction based on non-polynomial functions such as the Fourier functions is briefly mentioned [5].

In this paper, we present a simple new type of the ENO/WENO methods based on non-polynomial interpolations. As an example of non-polynomial bases, radial basis functions (RBFs) are used. In [1], the ADER method was developed based on the polyharmonic spline, which belongs to the family of piecewise smooth RBFs. The motivation of the method presented in [1] was to adopt the WENO method efficiently for the arbitrary geometry and unstructured mesh by using the meshless feature of RBFs. So there was no undetermined shape parameter — or the shape parameter is fixed as $\epsilon = 1$. The order of convergence is overall fixed once the size of each stencil k is fixed. Our main motivation in this paper, however, is to enhance the original ENO/WENO accuracy by modifying the interpolation coefficients. For this reason, we need free parameters to optimize, which makes the presented method in this paper different from the one in [1].

RBFs are divided into two categories depending on whether there are undetermined shape parameters: piecewise smooth RBFs and infinitely smooth RBFs (see Table 1). In this paper, we first use the infinitely smooth RBFs because they are defined with a free parameter ϵ, so-called the shape parameter. Since the parameter is free yet to be determined locally, it yields the flexibility to improve the original ENO/WENO accuracy. In fact, different RBFs give equivalent interpolations. This means we will end up with the same type of reconstruction irrespective of the bases used. This is also true for the piecewise smooth RBF basis used in [1], if we regard them as a special case of the infinitely smooth RBF basis with the shape parameter fixed as $\epsilon = 1$. We can also show that the derived RBF interpolation formulas are equivalent to the perturbed polynomial interpolation. Thus one can use other non-polynomial bases rather than RBFs as long as the new basis is defined with one or more free parameters for improving the local accuracy and convergence. For the RBF interpolation, it becomes a polynomial interpolation if the shape parameter vanishes. This makes it easy to modify the existing ENO/WENO code to the proposed ENO/WENO methods. We restrict our discussion to the one-parameter perturbation although it may be possible to utilize multiple free parameters.

Unlike the polynomial interpolation, the perturbed polynomial interpolation such as the RBF interpolation is not necessarily consistent, i.e. reconstruction coefficients may not sum to unity. Such an inconsistency helps the proposed method to enhance local accuracy in the smooth area. However, if the solution contains discontinuities, the inconsistent reconstruction causes the Gibbs oscillations. To prevent the Gibbs oscillations, the monotone polynomial interpolation method by measuring the local extrema is introduced. The non-polynomial interpolation is switched into the polynomial interpolation in the non-smooth region. This can be done easily by adopting the vanishing shape parameter to reduce the method into the polynomial method [16].

The paper is composed of the following sections. In Section 2, we briefly explain the finite volume ENO/WENO methods. In Section 3, we use the case of $k = 2$ to illustrate the RBF-ENO interpolation based on the multi-quadric (MQ), Gaussian RBFs and the perturbed polynomial. In this section, the tables of the reconstruction coefficients for $k = 2$ and $k = 3$ are provided. In Section 4, we explain the monotone polynomial interpolation method in detail. In Section 5, we briefly explain the time-integration and flux schemes that are used for the numerical experiment. Then the 1D numerical examples are presented for both scalar and system problems. In Section 6, we explain the 2D ENO/WENO finite volume interpolation method based on the non-polynomial bases. In Section 7 the 2D numerical examples are presented. In Section 8, we provide a brief conclusion and our future research.

2. Finite volume ENO/WENO method

Suppose that we are given a grid with N cells such that

$$a = x_{\frac{1}{2}} < x_{\frac{3}{2}} < \cdots < x_{N-\frac{1}{2}} < x_{N+\frac{1}{2}} = b.$$

For the i-th cell $I_i = [x_{i-\frac{1}{2}},\ x_{i+\frac{1}{2}}]$, define the cell center x_i and cell size Δx_i as

$$x_i = \frac{1}{2}(x_{i-\frac{1}{2}} + x_{i+\frac{1}{2}}),\ \Delta x_i = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}},\ i = 1, 2, \cdots, N,$$

and denote the maximum cell size by

$$\Delta x = \max_{1 \le i \le N} \Delta x_i .$$

The cell average $\bar{v}_i$ of a function $v(x)$ for the i-th cell is given as

$$\bar{v}_i = \frac{1}{\Delta x_i} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} v(\xi) d\xi, \quad i = 1, \cdots, N.$$

For the classical finite volume ENO method, we seek a function $p_i(x)$ to get k-th order accurate approximation to the function $v(x)$ in I_i, that is,

$$p_i(x) = v(x) + O(\Delta x^m), \quad m = k, \quad x \in I_i.$$

In this paper, for the finite volume RBF-ENO method we want to find an approximation that yields $m \ge k$. The cell boundary values of $v(x)$ of I_i are then approximated by $p_i(x)$ as

$$v^-_{i+\frac{1}{2}} = p_i(x_{i+\frac{1}{2}}) \quad \text{and} \quad v^+_{i-\frac{1}{2}} = p_i(x_{i-\frac{1}{2}}),$$

so that they are at least k-th order accurate. Here the superscripts $+$ and $-$ denote the right hand side and left hand side limits.

For the k-th order ENO reconstruction, the stencil is based on r cells to the left and s cells to the right including I_i such that

$$r + s + 1 = k.$$

Define $S_r(i)$ as the stencil composed of those k cells including the cell I_i

$$S_r(i) = \{I_{i-r}, \cdots, I_{i+s}\}, \quad r = 0, \cdots, k-1, \tag{2}$$

and a primitive function $V(x)$ such that

$$V(x) = \int_{x_{i-r-\frac{1}{2}}}^{x} v(\xi) d\xi, \tag{3}$$

where the lower limit in the integral can be any cell boundary [22]. By the definition of $V(x)$ in (3), $V'(x) = v(x)$ [22]. Then for $i - r - 1 \leqslant l \leqslant i + s$, $V(x_{l+\frac{1}{2}})$ is given by the linear sum of cell averages

$$V(x_{l+\frac{1}{2}}) = \sum_{j=i-r}^{l} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} v(\xi) d\xi = \sum_{j=i-r}^{l} \Delta x_j \bar{v}_j = \sum_{j=i-r}^{l} \Delta x \bar{v}_j.$$

The regular ENO method constructs the polynomial interpolation of $V(x)$ based on $V(x_{l+\frac{1}{2}})$, $i - r - 1 \leqslant l \leqslant i + s$, while the non-polynomial ENO method constructs the non-polynomial interpolation of $V(x)$ such as the RBF interpolation of $V(x)$. Suppose $P(x)$ is some interpolation of $V(x)$ such that

$$P(x) = V(x) + O(\Delta x^{m+1}), \tag{4}$$

then $p(x) \equiv P'(x)$ is the function we seek to approximate $v(x)$ where

$$p(x) = v(x) + O(\Delta x^m). \tag{5}$$

Section 3 will show that whether the polynomial or non-polynomial interpolation is used, the reconstruction is given in the same form as below

$$v^{(r)-}_{i+\frac{1}{2}} \equiv p(x_{i+\frac{1}{2}}) = \sum_{j=0}^{k-1} c_{rj} \bar{v}_{i-r+j} \tag{6}$$

with slightly different reconstruction coefficients c_{rj}.

The (RBF-)WENO reconstruction is then formulated as a convex combination of all possible (RBF-)ENO reconstructions. According to (2), there are k different (RBF-)ENO reconstructions of $v^{(r)-}_{i+\frac{1}{2}}$ and the (RBF-)WENO reconstruction would take the convex combination as:

$$v^-_{i+\frac{1}{2}} = \sum_{r=0}^{k-1} w_r v^{(r)-}_{i+\frac{1}{2}}, \tag{7}$$

where

$$w_r = \frac{\alpha_r}{\sum_{s=0}^{k-1}\alpha_s}, r = 0, \cdots, k-1$$

with

$$\alpha_r = \frac{d_r}{(\epsilon + \beta_r)^2}.$$

Here d_r are the ideal weights and $\epsilon > 0$ is introduced to avoid the case that the denominator becomes 0 usually taken as 10^{-6}. β_r are the "smoothness indicators" of the stencil $S_r(i)$. See [22] for more details of d_r and β_r.

3. 1D polynomial and non-polynomial interpolation

Now we explain in detail how (6) is derived under the polynomial and RBF frameworks. Consider the case of $k = 2$, that is, two cells are used for the ENO reconstruction. For this case, three cell averages $\bar{v}_{i-1}$, $\bar{v}_i$ and $\bar{v}_{i+1}$ are available. To reconstruct the boundary values of $v^-_{i+\frac{1}{2}}$ and $v^+_{i-\frac{1}{2}}$, either $\{\bar{v}_{i-1}, \ \bar{v}_i\}$ or $\{\bar{v}_i, \ \bar{v}_{i+1}\}$ is used based on the Newton's divided difference method [22]. Suppose that for the given cell I_i the Newton's divided difference method suggests that $\{\bar{v}_i, \ \bar{v}_{i+1}\}$ should be used for the local reconstruction from the available stencil $\{\bar{v}_{i-1}, \ \bar{v}_i, \ \bar{v}_{i+1}\}$. For simplicity, the uniform grid is considered, i.e. $\Delta x_i = \Delta x, \ \forall i$. We only show the reconstruction at the cell boundary $x = x_{i+\frac{1}{2}}$. The reconstruction for the other cell boundary $x = x_{i-\frac{1}{2}}$ can be obtained in the same manner.

Let $V(x) = \int_{x_{i+\frac{1}{2}}}^x v(\xi)d\xi$. Then the primitive function at the cell boundaries are given by

$$\begin{aligned}
V_{i-\frac{3}{2}} &= V(x_{i-\frac{3}{2}}) = -\Delta x \cdot \bar{v}_{i-1} - \Delta x \cdot \bar{v}_i, \\
V_{i-\frac{1}{2}} &= V(x_{i-\frac{1}{2}}) = -\Delta x \cdot \bar{v}_i, \\
V_{i+\frac{1}{2}} &= V(x_{i+\frac{1}{2}}) = 0, \\
V_{i+\frac{3}{2}} &= V(x_{i+\frac{3}{2}}) = \Delta x \cdot \bar{v}_{i+1}.
\end{aligned} \tag{8}$$

3.1. Polynomial reconstruction

The polynomial interpolation $P(x)$ of $V(x)$ is given as

$$P(x) = \lambda_1 + \lambda_2 x + \lambda_3 x^2.$$

Let $\vec{V} = [V_{i-\frac{1}{2}}, V_{i+\frac{1}{2}}, V_{i+\frac{3}{2}}]^T$, $\vec{\lambda} = [\lambda_1, \lambda_2, \lambda_3]^T$ and the interpolation matrix be

$$A = \begin{bmatrix} 1 & x_{i-\frac{1}{2}} & x^2_{i-\frac{1}{2}} \\ 1 & x_{i+\frac{1}{2}} & x^2_{i+\frac{1}{2}} \\ 1 & x_{i+\frac{3}{2}} & x^2_{i+\frac{3}{2}} \end{bmatrix}.$$

Then the expansion coefficients λ_i are determined by solving the linear system $\vec{V} = A \cdot \vec{\lambda}$. Take the first derivative of $P(x)$ and plug $x = x_{i+\frac{1}{2}}$ to obtain

$$v^-_{i+\frac{1}{2}} = p(x_{i+\frac{1}{2}}) = \frac{1}{2} \cdot \bar{v}_i + \frac{1}{2} \cdot \bar{v}_{i+1}. \tag{9}$$

The Taylor series expansion of $v^-_{i+\frac{1}{2}}$ around $x = x_{i+\frac{1}{2}}$ yields

$$v^-_{i+\frac{1}{2}} = p(x_{i+\frac{1}{2}}) = v_{i+\frac{1}{2}} + \frac{1}{6} v''_{i+\frac{1}{2}} \Delta x^2 + \frac{1}{120} v^{(4)}_{i+\frac{1}{2}} \Delta x^4 + O(\Delta x^6). \tag{10}$$

The first term in the right hand side in (10) is the exact value of $v(x)$ at $x = x_{i+\frac{1}{2}}$. So we confirm that (9) is a 2nd order reconstruction.

3.2. Multiquadratic RBF reconstruction

For a non-polynomial reconstruction, the procedure is similar to the polynomial reconstruction [4]. First consider the MQ-RBF interpolation $P(x)$ given by

$$P(x) = \lambda_1\sqrt{1+\epsilon^2(x-x_{i-\frac{1}{2}})^2} + \lambda_2\sqrt{1+\epsilon^2(x-x_{i+\frac{1}{2}})^2} + \lambda_3\sqrt{1+\epsilon^2(x-x_{i+\frac{3}{2}})^2}.$$

Then the interpolation matrix A is given by

$$A = \begin{bmatrix} 1 & \sqrt{\Delta x^2\epsilon^2+1} & \sqrt{4\Delta x^2\epsilon^2+1} \\ \sqrt{\Delta x^2\epsilon^2+1} & 1 & \sqrt{\Delta x^2\epsilon^2+1} \\ \sqrt{4\Delta x^2\epsilon^2+1} & \sqrt{\Delta x^2\epsilon^2+1} & 1 \end{bmatrix}.$$

Again, take the first derivative of $P(x)$ and plug $x = x_{i+\frac{1}{2}}$ to obtain

$$v^-_{i+\frac{1}{2}} = p(x_{i+\frac{1}{2}}) = \frac{\sqrt{4\epsilon^2\Delta x^2+1}+1}{4\sqrt{\epsilon^2\Delta x^2+1}}\cdot\bar{v}_i + \frac{\sqrt{4\epsilon^2\Delta x^2+1}+1}{4\sqrt{\epsilon^2\Delta x^2+1}}\cdot\bar{v}_{i+1}. \tag{11}$$

Expanding $v^-_{i+\frac{1}{2}}$ around $x = x_{i+\frac{1}{2}}$ in the Taylor series yields

$$\begin{aligned} v^-_{i+\frac{1}{2}} = p(x_{i+\frac{1}{2}}) &= v_{i+\frac{1}{2}} + \left(\frac{1}{2}\epsilon^2 v_{i+\frac{1}{2}} + \frac{1}{6}v''_{i+\frac{1}{2}}\right)\Delta x^2 \\ &+ \left(-\frac{9}{8}\epsilon^4 v_{i+\frac{1}{2}} + \frac{1}{12}\epsilon^2 v''_{i+\frac{1}{2}} + \frac{1}{120}v^{(4)}_{i+\frac{1}{2}}\right)\Delta x^4 + O(\Delta x^6). \end{aligned} \tag{12}$$

Thus (11) is at least 2nd order accurate to $v_{i+\frac{1}{2}}$. If we take the value of ϵ as

$$\epsilon^2 = -\frac{1}{3}\frac{v''_{i+\frac{1}{2}}}{v_{i+\frac{1}{2}}}, \tag{13}$$

then a 4th order accurate approximation is obtained.

Notice that the coefficients of $\bar{v}_i$ and $\bar{v}_{i+1}$ in (11) are in complicated forms involving a calculation of square roots. We can alternatively derive more efficient and simpler forms that can still yield the same desired order. To do this, Taylor expansion is used again to expand the right hand sides of (11) as below

$$\begin{aligned} v^-_{i+\frac{1}{2}} = p(x_{i+\frac{1}{2}}) &= \left(\frac{1}{2} + \frac{1}{4}\epsilon^2\Delta x^2 - \frac{9}{16}\epsilon^4\Delta x^4\right)\cdot\bar{v}_i \\ &+ \left(\frac{1}{2} + \frac{1}{4}\epsilon^2\Delta x^2 - \frac{9}{16}\epsilon^4\Delta x^4\right)\cdot\bar{v}_{i+1} + O(\Delta x^6). \end{aligned} \tag{14}$$

Since we want the RBF-ENO method with $k = 2$ to give 3rd order accuracy at least, all the high order terms in (14) can be ignored, which yields

$$v^-_{i+\frac{1}{2}} = p(x_{i+\frac{1}{2}}) = \left(\frac{1}{2} + \frac{1}{4}\epsilon^2\Delta x^2\right)\cdot\bar{v}_i + \left(\frac{1}{2} + \frac{1}{4}\epsilon^2\Delta x^2\right)\cdot\bar{v}_{i+1}. \tag{15}$$

Carry out the error analysis on our new reconstructions (15) to get

$$\begin{aligned} v^-_{i+\frac{1}{2}} = p(x_{i+\frac{1}{2}}) &= v_{i+\frac{1}{2}} + \left(\frac{1}{2}\epsilon^2 v_{i+\frac{1}{2}} + \frac{1}{6}v''_{i+\frac{1}{2}}\right)\Delta x^2 \\ &+ \left(\frac{1}{12}\epsilon^2 v''_{i+\frac{1}{2}} + \frac{1}{120}v^{(4)}_{i+\frac{1}{2}}\right)\Delta x^4 + O(\Delta x^6). \end{aligned} \tag{16}$$

Thus getting rid of all the high order terms $O(\Delta x^4)$ in (14) would only exert influence on $O(\Delta x^4)$ terms in (12), while all the $O(\Delta x^2)$ terms stay the same. In another word, we can still use (13) to remove the 2nd order terms and achieve the 3rd order accuracy for the new reconstruction.

Now the problem is to calculate such values of ϵ in (13) to achieve higher order accuracy than the 2nd order. The idea is as following. Although $v^-_{i+\frac{1}{2}}$ is computed based on $\bar{v}_i$ and $\bar{v}_{i+1}$, the cell average to the left of I_i, i.e. $\bar{v}_{i-1}$ is also known — notice that it was already used when the Newton's divided difference method was applied to determine the cell averages used for the reconstruction for the cell I_i. Thus, we can approximate $v''_{i+\frac{1}{2}}$ and $v_{i+\frac{1}{2}}$ if all the given cell average information is used. Construct the Lagrange interpolation $P(x)$ of $V(x)$ based on $V_{i-\frac{3}{2}}, V_{i-\frac{1}{2}}, V_{i+\frac{1}{2}}$ and $V_{i+\frac{3}{2}}$. Then $v_{i+\frac{1}{2}}$ is approximated by $P'(x)$ at $x = x_{i+\frac{1}{2}}$

$$v_{i+\frac{1}{2}} = -\frac{1}{6}\bar{v}_{i-1} + \frac{5}{6}\bar{v}_i + \frac{1}{3}\bar{v}_{i+1} + O(\Delta x^3),$$

and $v''_{i+\frac{1}{2}}$ is approximated by $P'''(x)$ at $x = x_{i+\frac{1}{2}}$

$$v''_{i+\frac{1}{2}} = \frac{\bar{v}_{i-1} - 2\bar{v}_i + \bar{v}_{i+1}}{\Delta x^2} + O(\Delta x).$$

Then by plugging the above approximations of $v_{i+\frac{1}{2}}$ and $v''_{i+\frac{1}{2}}$ into (13), the value of ϵ^2 in terms of the cell averages, can be determined as below

$$\epsilon^2 \approx \frac{2}{\Delta x^2} \cdot \frac{-\bar{v}_{i-1} + 2\bar{v}_i - \bar{v}_{i+1}}{-\bar{v}_{i-1} + 5\bar{v}_i + 2\bar{v}_{i+1} + \epsilon_M}. \tag{17}$$

$\epsilon_M = 10^{-13}$ is a positive small number to avoid the denominator being zero. Here note that ϵ can be a complex number if ϵ^2 is negative, but it does not harm the RBF interpolation. This is because it is ϵ^2 not ϵ that is used in the RBF-ENO reconstruction and all the operations are done on real numbers. Replace ϵ^2 in (12) with that in (17) to get

$$v^-_{i+\frac{1}{2}} = p(x_{i+\frac{1}{2}}) = v_{i+\frac{1}{2}} + \frac{1}{12} v^{(3)}_{i+\frac{1}{2}} \Delta x^3 + O(\Delta x^4). \tag{18}$$

Thus we confirm that the above approximate value of ϵ in (17) actually yields the 3rd order accuracy.

3.3. Gaussian RBF reconstruction

Now consider another infinitely smooth RBF basis, i.e. the Gaussian RBF. We will show below that all the RBF reconstructions are equivalent. The Gaussian RBF interpolation $P(x)$ of $V(x)$ is

$$P(x) = \lambda_1 \exp(-\epsilon^2 (x - x_{i-\frac{1}{2}})^2) + \lambda_2 \exp(-\epsilon^2 (x - x_{i+\frac{1}{2}})^2) + \lambda_3 \exp(-\epsilon^2 (x - x_{i+\frac{3}{2}})^2).$$

Follow the same procedure as before. The exact RBF reconstruction at $x = x_{i+\frac{1}{2}}$ is

$$v^-_{i+\frac{1}{2}} = p(x_{i+\frac{1}{2}}) = \frac{2\epsilon^2 \Delta x^2 \exp(-\epsilon^2 \Delta x^2)}{1 - \exp(4\epsilon^2 \Delta x^2)} \cdot \bar{v}_i + \frac{2\epsilon^2 \Delta x^2 \exp(-\epsilon^2 \Delta x^2)}{1 - \exp(4\epsilon^2 \Delta x^2)} \cdot \bar{v}_{i+1}. \tag{19}$$

Expand the reconstruction in a Taylor series and ignore all the high order terms to get

$$v^-_{i+\frac{1}{2}} = p(x_{i+\frac{1}{2}}) = \left(\frac{1}{2} + \frac{1}{2}\epsilon^2 \Delta x^2\right) \cdot \bar{v}_i + \left(\frac{1}{2} + \frac{1}{2}\epsilon^2 \Delta x^2\right) \cdot \bar{v}_{i+1}. \tag{20}$$

Note that the reconstruction coefficients in (20) and (15) are slightly different. Expanding $\bar{v}_i$ and $\bar{v}_{i+1}$ around the boundary value yields

$$\begin{aligned} v^-_{i+\frac{1}{2}} = p(x_{i+\frac{1}{2}}) = v_{i+\frac{1}{2}} &+ \left(\epsilon^2 v_{i+\frac{1}{2}} + \frac{1}{6} v''_{i+\frac{1}{2}}\right) \Delta x^2 \\ &+ \left(\frac{1}{12}\epsilon^2 v''_{i+\frac{1}{2}} + \frac{1}{120} v^{(4)}_{i+\frac{1}{2}}\right) \Delta x^4 + O(\Delta x^6). \end{aligned} \tag{21}$$

Again, if we take

$$\epsilon^2 = \frac{1}{\Delta x^2} \cdot \frac{-\bar{v}_{i-1} + 2\bar{v}_i - \bar{v}_{i+1}}{-\bar{v}_{i-1} + 5\bar{v}_i + 2\bar{v}_{i+1}}, \tag{22}$$

then (21) becomes

$$v^-_{i+\frac{1}{2}} = p(x_{i+\frac{1}{2}}) = v_{i+\frac{1}{2}} + \frac{1}{12} v^{(3)}_{i+\frac{1}{2}} \Delta x^3 + O(\Delta x^4). \tag{23}$$

By comparing (18) and (23), we see that no matter what RBF basis is used, the Taylor series of the reconstruction results in the same expansion after the proper value of ϵ is plugged in.[1]

[1] Here note that the Gaussian RBF interpolation becomes equivalent to the Lagrange interpolation polynomial if the uniform center is used [19]. This implies that the Gaussian RBF may allow a larger size of the stencil for stability among other RBFs even though all the RBFs yield similar reconstruction form.

Table 2
Left: The polynomial reconstruction coefficients, also for the RBF interpolation with $\epsilon \to 0$. Right: The MQ-RBF reconstruction coefficients c_{rj} corresponding to (15). $\eta = \epsilon^2 \Delta x^2$.

k	r	$j=0$	$j=1$
2	-1	$\frac{3}{2}$	$-\frac{1}{2}$
	0	$\frac{1}{2}$	$\frac{1}{2}$
	1	$-\frac{1}{2}$	$\frac{3}{2}$
		$\eta = 0$	

k	r	$j=0$	$j=1$
2	-1	$\frac{3}{2} - \frac{3}{2}\eta$	$-\frac{1}{2} + \frac{1}{2}\eta$
	0	$\frac{1}{2} + \frac{1}{4}\eta$	$\frac{1}{2} + \frac{1}{4}\eta$
	1	$-\frac{1}{2} + \frac{1}{2}\eta$	$\frac{3}{2} - \frac{3}{2}\eta$
		$\eta = \epsilon^2 \Delta x^2 = \frac{2(-\bar{v}_{i-1}+2\bar{v}_i-\bar{v}_{i+1})}{-\bar{v}_{i-1}+5\bar{v}_i+2\bar{v}_{i+1}+\epsilon_M}$	

3.4. Perturbed polynomial reconstruction

Now let us consider the generalized case by perturbing the polynomial interpolation. By knowing the polynomial interpolation as (9), assume that the unknown perturbed polynomial interpolation of the 2nd order or higher is given as

$$v^-_{i+\frac{1}{2}} = p(x_{i+\frac{1}{2}}) = \left(\frac{1}{2} + a\epsilon^2\Delta x^2\right) \cdot \bar{v}_i + \left(\frac{1}{2} + b\epsilon^2\Delta x^2\right) \cdot \bar{v}_{i+1}, \tag{24}$$

where a and b are real constants and ϵ^2 is to be determined. Expand the given cell averages around $x_{i+\frac{1}{2}}$ to get

$$\begin{aligned} v^-_{i+\frac{1}{2}} = p(x_{i+\frac{1}{2}}) = v_{i+\frac{1}{2}} &+ \left[(a+b)\epsilon^2 v_{i+\frac{1}{2}} + \frac{1}{6} v''_{i+\frac{1}{2}}\right]\Delta x^2 \\ &- \frac{1}{2}(a-b)\epsilon^2 v'_{i+\frac{1}{2}}\Delta x^3 + O(\Delta x^4). \end{aligned} \tag{25}$$

To make the 2nd order error term vanish, take the unknown parameter ϵ as

$$\epsilon^2 = \frac{1}{\Delta x^2(a+b)} \cdot \frac{-\bar{v}_{i-1} + 2\bar{v}_i - \bar{v}_{i+1}}{-\bar{v}_{i-1} + 5\bar{v}_i + 2\bar{v}_{i+1}}. \tag{26}$$

Then (25) becomes

$$v^-_{i+\frac{1}{2}} = p(x_{i+\frac{1}{2}}) = v_{i+\frac{1}{2}} + \left(\frac{1}{12} v^{(3)}_{i+\frac{1}{2}} + \frac{a-b}{a+b} \frac{v'_{i+\frac{1}{2}} v''_{i+\frac{1}{2}}}{v_{i+\frac{1}{2}}}\right)\Delta x^3 + O(\Delta x^4). \tag{27}$$

There are multiple ways to determine the values of the perturbed coefficients a and b. For example,

$$a + b \neq 0, \tag{28}$$

$$a - b = 0. \tag{29}$$

The condition (28) implies that the non-polynomial reconstruction is inconsistent, since it makes the coefficients of the cell averages in (24) not sum to unity. Both (15) and (20) satisfy (28) and (29). In this way, we see that the RBF reconstruction irrespective of the basis used is equivalent to the perturbed polynomial reconstruction. That is, it is not necessary to know the exact form of the non-polynomial basis for the interpolation.

3.5. Reconstruction coefficients for $k = 2$ and $k = 3$

The left table in Table 2 shows the coefficients of the polynomial reconstruction that yield 2nd order convergence. The right table provides the coefficients with the RBF reconstruction corresponding to the MQ-RBF interpolation (15) which yields the 3rd order convergence. The left table is obtained by taking $\epsilon \to 0$ in the RBF reconstruction. Notice that the RBF reconstruction is the same as the polynomial reconstruction except for the perturbation terms added. Table 3 shows the coefficients for $k = 3$.

4. Switching condition of ϵ

The regular ENO method helps to avoid using the cell averages across the discontinuity by adaptively choosing the proper cells from the Newton's divided difference method. This essentially yields the non-oscillatory reconstruction near the discontinuity as $N \to \infty$. Although the non-polynomial ENO reconstruction also utilizes the Newton's divided difference method to determine the proper cells, the value of ϵ^2 used in the interpolation coefficients is determined by the cell averages across the discontinuity as the WENO reconstruction. This is why the non-polynomial ENO reconstruction achieves higher order accuracy in the smooth area. However, it may not necessarily be non-oscillatory near the discontinuity for the same reason.

Table 3
Left: The polynomial reconstruction coefficients, also for the RBF interpolation with $\epsilon \to 0$. Right: The MQ-RBF reconstruction coefficients c_{rj}, $\eta = \epsilon^2 \Delta x^2$.

k	r	$j=0$	$j=1$	$j=2$
3	-1	$\frac{6}{11}$	$-\frac{7}{6}$	$\frac{1}{3}$
	0	$\frac{1}{3}$	$\frac{5}{6}$	$-\frac{1}{6}$
	1	$-\frac{1}{6}$	$\frac{5}{6}$	$\frac{1}{3}$
	2	$\frac{1}{3}$	$-\frac{7}{6}$	$\frac{11}{6}$
		$\eta = 0$		

k	r	$j=0$	$j=1$	$j=2$
3	-1	$\frac{6}{11} - \frac{9}{2}\eta$	$-\frac{7}{6} + 6\eta$	$\frac{1}{3} - \frac{3}{2}\eta$
	0	$\frac{1}{3} + \frac{5}{6}\eta$	$\frac{5}{6} - \frac{2}{3}\eta$	$-\frac{1}{6} - \frac{1}{6}\eta$
	1	$-\frac{1}{6} - \frac{1}{6}\eta$	$\frac{5}{6} - \frac{2}{3}\eta$	$\frac{1}{3} + \frac{5}{6}\eta$
	2	$\frac{1}{3} - \frac{3}{2}\eta$	$-\frac{7}{6} + 6\eta$	$\frac{11}{6} - \frac{9}{2}\eta$
		$\eta = \epsilon^2 \Delta x^2 = \frac{\bar{v}_{i-1} - 3\bar{v}_i + 3\bar{v}_{i+1} - \bar{v}_{i+2}}{\bar{v}_{i-1} - 15\bar{v}_i + 15\bar{v}_{i+1} - \bar{v}_{i+2} + \epsilon_M}$		

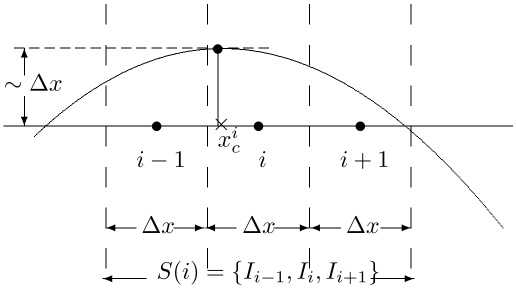

Fig. 1. A possible reconstruction in a quadratic form, $p_i(x)$ in $S(i)$ composed of three cells, I_{i-1}, I_i, I_{i+1}. The critical point of $p_i(x)$ is indicated as x_c^i.

One possible way to deal with the oscillations is to make the RBF-ENO reconstruction become the regular ENO reconstruction in the non-smooth area. This can be easily done by adopting the vanishing shape parameter ϵ^2 in the non-smooth area. To achieve this, first we identify which cell contains the discontinuity using the edge detection method. The edge detection algorithm has been utilized for the construction of hybrid methods such as the spectral-WENO method [6,7] and the Fourier-continuation WENO method [21,18]. For those hybrid methods, the edge detection algorithm identifies the edge location and the neighboring stencils containing the edge cells are treated by the WENO method and the other cells are treated by the spectral or Fourier methods.

Most edge detection methods focus on finding edges as accurately as possible using as many cell information as possible such as the multi-resolution analysis [12]. Thus those methods use cell values even outside the stencils. A new and simple algorithm for finding the WENO stencils is proposed in this paper. The key idea of our proposed algorithm is to utilize the monotone polynomial interpolation method for each stencil. Since the proposed method uses the cell values within the stencils of interest only, the method is optimal and is efficient with minimum cost of computation in identifying the non-smooth areas.

4.1. Monotone interpolation and critical point analysis

Consider the stencil $S(i)$ composed of three cells including I_i

$$\begin{aligned} S(i) &= S_0(i) \cup S_1(i) \\ &= \{I_{i-1},\ I_i\} \cup \{I_i,\ I_{i+1}\} \\ &= \{I_{i-1},\ I_i,\ I_{i+1}\}, \end{aligned}$$

where $S_0(i)$ and $S_1(i)$ are defined in (2). With the primitive function $V(x)$ defined as (3), let the unique polynomial $P_i(x)$ of degree at most 3 interpolate the pointwise values of $V(x)$ in (8). Let $p_i(x) = P_i'(x)$. Then $p_i(x)$ is a polynomial of degree at most 2.

A possible $p_i(x)$ from $V(x)$ within $S(i)$ is illustrated in the schematic figure in Fig. 1. When $p_i(x)$ is a polynomial of degree 2, the critical point of $p_i(x)$, where $p_i'(x) = 0$, is indicated as x_c^i in Fig. 1. For this example, the critical point x_c^i exists inside the stencil $S(i)$ while for other cases it can be outside of $S(i)$. If x_c^i exists and is inside $S(i)$, the reconstruction p_i is not monotone. Otherwise, the reconstruction is monotone. Without loss of generality, choose the coordinate of the cell boundaries as below

$$x_{i-\frac{3}{2}} = 0, \quad x_{i-\frac{1}{2}} = \Delta x, \quad x_{i+\frac{1}{2}} = 2\Delta x, \quad x_{i+\frac{3}{2}} = 3\Delta x.$$

Then the monotonicity of $p_i(x)$ is easily determined by checking whether x_c^i exists inside $S(i)$. That is, $p_i(x)$ is monotone if $x_c \le 0$ or $x_c \ge 3\Delta x$ and $p_i(x)$ is oscillatory if $0 < x_c < 3\Delta x$ in the sense that the derivative of $p_i(x)$ changes its sign within the domain. This oscillation originates either from the Gibbs oscillations or is real. For the case of $k = 2$, it is enough to check whether x_c^i is within $S(i)$ or not. While for the case of $k = 3$, we have to check $S(i-1)$, $S(i)$ and $S(i+1)$ to cover the entire stencil. If any of these substencils contains the critical point, it will result in an oscillatory WENO reconstruction. After simple calculation, we get

$$x_c^i = \frac{-2\bar{v}_{i-1} + 3\bar{v}_i - \bar{v}_{i+1}}{-\bar{v}_{i-1} + 2\bar{v}_i - \bar{v}_{i+1}} \Delta x. \tag{30}$$

Let x_c^{i-1}, x_c^i, x_c^{i+1} denote the critical points from $S(i-1)$, $S(i)$, $S(i+1)$, respectively. Each of them can be easily obtained using the formula (30). Then the cell I_i should be recognized as the WENO cell as below

$$I_i = \begin{cases} \text{a possible WENO cell} & \text{if } x_c^{i-1} \in \mathbb{I}, \text{ or } x_c^i \in \mathbb{I}, \text{ or } x_c^{i+1} \in \mathbb{I}, \\ \text{RBF cell} & \text{otherwise}, \end{cases} \tag{31}$$

where $\mathbb{I}$ is the open interval of $\mathbb{I} = (0, 3\Delta x)$ and "$I_i =$ *WENO cell*" means that the given I_i needs to be treated by the original ENO/WENO methods, and "$I_i =$ *RBF cell*" means that the given cell I_i is smooth enough to employ the RBF-ENO/WENO methods.

Here note that the condition (31) not only detects the discontinuous area but also the area where the function has a critical point. To eliminate the case of the genuine critical point, consider the following alternative way to impose the condition (31).

Notice that the formula (30) demands us to look into the case when the denominator vanishes. The vanishing denominator in fact reveals an important aspect of the proposed method. To explain this, suppose we have a constant function. In this case, both the denominator and numerator will vanish. Further suppose that the cell average values are slightly perturbed. Although the perturbation is negligible, (30) may still provide a finite value of the critical point x_c. This means that even with slight perturbation, the given cell I_i can be turned into the WENO cell if x_c exists in $\mathbb{I}$. That is, even for the reconstruction of the constant function, the proposed method will adopt the WENO reconstruction if the condition (31) is applied, which is obviously not necessary. To avoid this problem, one more condition is added by directly measuring the degree of oscillations. For $S(i)$, the polynomial reconstruction of $p_i(x)$ is given by

$$p_i(x) = ax^2 + bx + c, \quad x \in \mathbb{I} = (0,\ 3\Delta x),$$

where

$$a = \frac{1}{2} \cdot \frac{\bar{v}_{i-1} - 2\bar{v}_i + \bar{v}_{i+1}}{\Delta x^2},$$
$$b = -\frac{2\bar{v}_{i-1} - 3\bar{v}_i + \bar{v}_{i+1}}{\Delta x},$$
$$c = \frac{11}{6}\bar{v}_{i-1} - \frac{7}{6}\bar{v}_i + \frac{1}{3}\bar{v}_{i+1}.$$

Here a can be zero. If $a \neq 0$, the critical point is given by $x_c^i = -\frac{b}{2a}$ as in (30). If $a = 0$, the condition (31) can not be applied directly but the adaption is not necessary because the reconstruction is monotone. If $a \neq 0$, consider two cases, 1) $|a|$ is large enough for the condition (31) to be well defined for the discontinuity and 2) $|a|$ is small enough so that the quadratic oscillation may be ignored compared to the WENO reconstruction error. The second case can happen, as explained above, when the oscillation is due to the WENO oscillations from the neighbor cells for the WENO reconstruction of the constant function near the discontinuity. If the condition (31) is applied, the region of the WENO cells near the discontinuity can become broaden with time. Thus we need to apply the condition (31) if $|a|$ is not small.

If the critical point is real, the magnitude of the local maxima does not decay with N. But if the critical point is due to the Gibbs oscillations, it will decay with N for the WENO reconstruction. Thus we adopt the WENO reconstruction if the magnitude of the local maxima is larger than the constant multiple of Δx. Otherwise the RBF interpolation is kept, that is, when

$$|p_i(x)| = |a(x - x_1)(x - x_2)| < D\Delta x,$$

where D is a positive constant and x_1 and x_2 are roots of $p_i(x)$. Since the necessary condition of the adaption to the WENO method is that the critical point x_c^i exists within the stencil, the condition will be

$$|p_i(x_c^i)| \leq |a|\Delta x^2 < D\Delta x,$$

and therefore a is bounded as[2]

$$|a| < D\Delta x^{-1}. \tag{32}$$

[2] In general, the condition for RBF cell is $|a| < D\Delta x^{-p}$, where $p \geq 0$.

Table 4
Pseudo code for the RBF-ENO/WENO methods with $k = 3$.

```
$C_i = 0$, $\forall i = 1, \cdots, N$ $(\textit{indicating } I_i \textit{ is an RBF cell})$
For $i = 1, \cdots, N$
    Given $I_i$
        $\mathcal{A} = |-\bar{v}_{i-1} + 2\bar{v}_i - \bar{v}_{i+1}|/2$
        if $\mathcal{A} > \Delta x$
            $x_c = \frac{-2\bar{v}_{i-1}+3\bar{v}_i-\bar{v}_{i+1}}{-\bar{v}_{i-1}+2\bar{v}_i-\bar{v}_{i+1}}\Delta x$
            if $0 < x_c < 3\Delta x$
                $C_i = 1$ $(\textit{indicating } I_i \textit{ is a WENO cell})$
For $i = 1, \cdots, N$
    Given $I_i$
        if $(C_{i-1} + C_i + C_{i+1}) \geq 1$
            Use the ENO/WENO reconstruction for $v^-_{i+\frac{1}{2}}$ and $v^+_{i-\frac{1}{2}}$
        if $(C_{i-1} + C_i + C_{i+1}) = 0$
            Use the RBF-ENO/WENO reconstruction for $v^-_{i+\frac{1}{2}}$ and $v^+_{i-\frac{1}{2}}$
```

Let

$$\mathcal{A} = \frac{|-\bar{v}_{i-1} + 2\bar{v}_i - \bar{v}_{i+1}|}{2}.$$

Then $a = \frac{\mathcal{A}}{\Delta x^2}$ and the condition to indicate RBF cell (31) is equivalent to $\mathcal{A} < D\Delta x$. In another word, the condition to indicate WENO cell is $\mathcal{A} > D\Delta x$. Thus the modified algorithm of finding the WENO cell is as below

$$I_i = \begin{cases} \text{WENO cell} & \text{if } x_c^{i-1} \in \mathbb{I}, \text{ or } x_c^i \in \mathbb{I}, \text{ or } x_c^{i+1} \in \mathbb{I}, \text{ and } \mathcal{A} > \Delta x, \\ \text{RBF cell} & \text{otherwise,} \end{cases} \tag{33}$$

if the parameter D is taken as 1 for instance.

Table 4 shows a pseudo-code for the case $k = 3$, where we use the auxiliary grid function C_i. If $C_i = 0$, I_i is smooth enough to be a RBF cell while if $C_i = 1$, I_i is recognized as a WENO cell. Altogether, for the given stencil, $\{S(i-1), S(i), S(i+1)\}$, if any of cells is a WENO cell, i.e. if $C_{i-1} + C_i + C_{i+1} \geq 1$, then the original ENO/WENO reconstruction is adopted for the reconstruction of the boundary values of $v^-_{i+\frac{1}{2}}$ and $v^+_{i-\frac{1}{2}}$ for the cell I_i.

5. 1D numerical experiments

Now the developed method is used to solve hyperbolic conservation laws (1). For the given cell average $\bar{v}_i$ for the cell I_i, we seek the solution by solving the following numerically

$$\frac{d\bar{v}_i}{dt} = -\frac{f(v(x_{i+\frac{1}{2}})) - f(v(x_{i-\frac{1}{2}}))}{\Delta x}. \tag{34}$$

For the numerical solution, the exact flux f in (34) is replaced with the numerical flux function h and the boundary values are computed by the non-polynomial ENO reconstruction or non-polynomial WENO reconstruction. The Lax–Friedrichs flux scheme is used for the flux

$$h(a, b) = \frac{1}{2}[f(a) + f(b) - \alpha(b - a)],$$

where $\alpha = \max_u |\frac{\partial f}{\partial u}|$ is a constant taken over the relevant range of u. For the time integration, we use the TVD RK-3 method [24].

In the following, let the WENO-JS denote the WENO method with the original smoothness indicators developed by Jiang and Shu [15] and the RBF-WENO-JS denote the corresponding method using the RBF reconstruction instead of the polynomial reconstruction. Similarly, the RBF-WENO-Z is the one where we use the RBF basis under the WENO-Z framework.

5.1. Reconstruction order

First we check the reconstruction error by the RBF-ENO/WENO methods applied to a smooth function $v(x) = \sin(\pi x)$ with $k = 2$ and $k = 3$ to confirm the desired order of accuracy. Fig. 2 shows the L_2 errors versus N in logarithmic scale. The blue, red, black and green lines denote the ENO, RBF-ENO, WENO-JS and RBF-WENO-JS methods, respectively. The accuracy of the RBF-ENO is always one order higher than the accuracy of the regular ENO. The RBF-WENO-JS method yields the same order of accuracy as the regular WENO-JS method, but it is slightly more accurate than the regular WENO because each reconstruction of the RBF-ENO is more accurate than the reconstruction by the regular ENO.

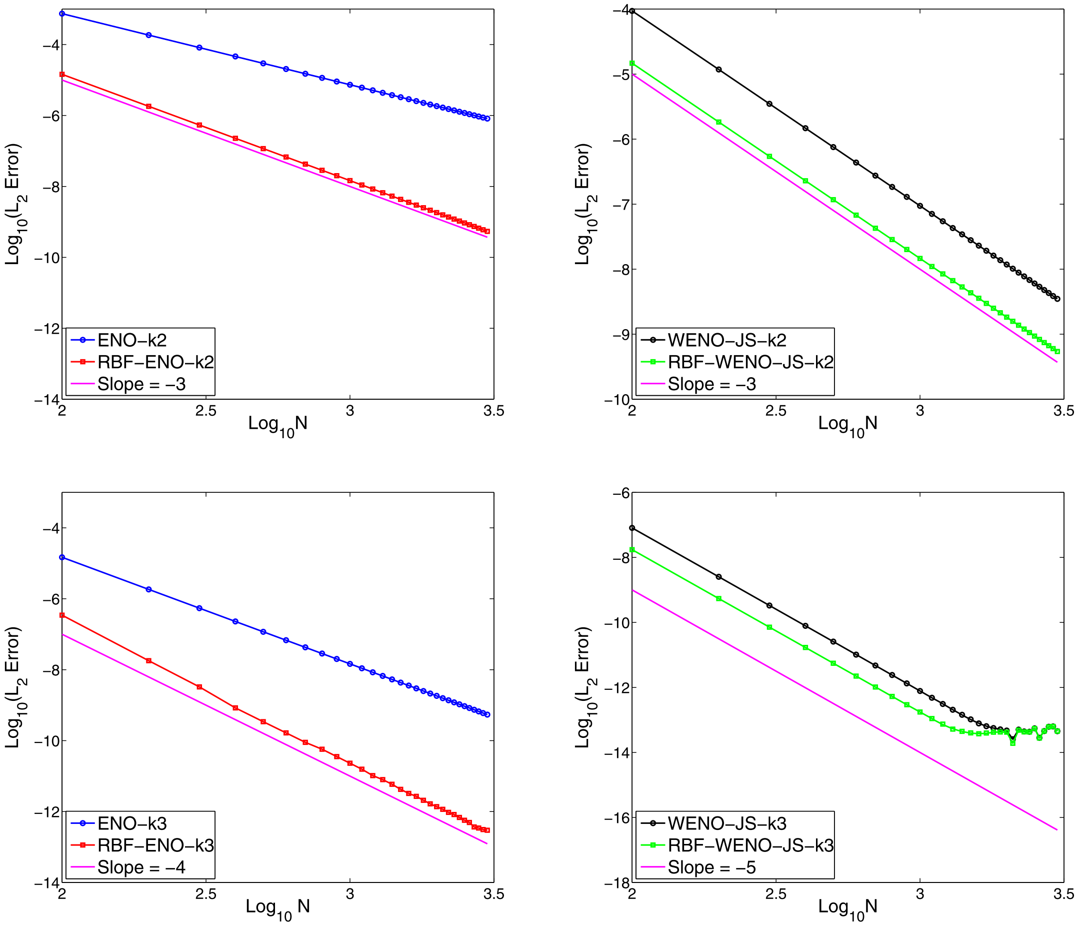

Fig. 2. L_2 errors versus N in logarithmic scale by the ENO (blue), RBF-ENO (red), WENO-JS (black) and RBF-WENO-JS (green) methods. $k = 2$ (top) and $k = 3$ (bottom). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5.2. Monotone interpolation method

To demonstrate how the monotone polynomial interpolation method performs on detecting discontinuities, the following four examples are used with $k = 3$.

- Advection equation (36) with the discontinuous initial condition (37) with the final time $T = 0.5$. For this case the discontinuity exists from $t = 0$ and propagates with a constant speed. Thus the monotone method needs to identify the discontinuity for all $t \geq 0$.
- Burgers equation (39) with the smooth initial condition $v(x, 0) = -\sin(\pi x)$ with the final time $T = 0.4$. The shock forms at $t = \frac{1}{\pi}$. Until the shock forms, the solution $v(x, t)$ has two critical points. Thus the monotone method needs to distinguish the critical point from the real shock while it needs to identify the real shock after it forms.
- Sod problem (40) with the initial condition (41) with the final time $T = 0.2$. For this problem, the density and pressure are discontinuous at $t = 0$ and multiple discontinuities develop with time for the density, pressure and velocity. The monotone method needs to identify those multiple discontinuities with time.
- Lax problem (40) with the following initial condition

$$(\rho, u, P) = \begin{cases} (0.445, 0.698, 3.528) & x \leq 0 \\ (0.5, 0, 0.571) & x > 0 \end{cases} \tag{35}$$

with the final time $T = 0.13$. This problem is similar to the Sod problem and the monotone method needs to identify the multiple shocks with time.

Fig. 3 shows the identified WENO cells with time by the monotone polynomial interpolation method for the above examples. The WENO cells identified with time for each case are shown in the figure from top to bottom and left to right, respectively. The top left figure for the advection equation shows that the initial discontinuity is well detected for all t by the proposed

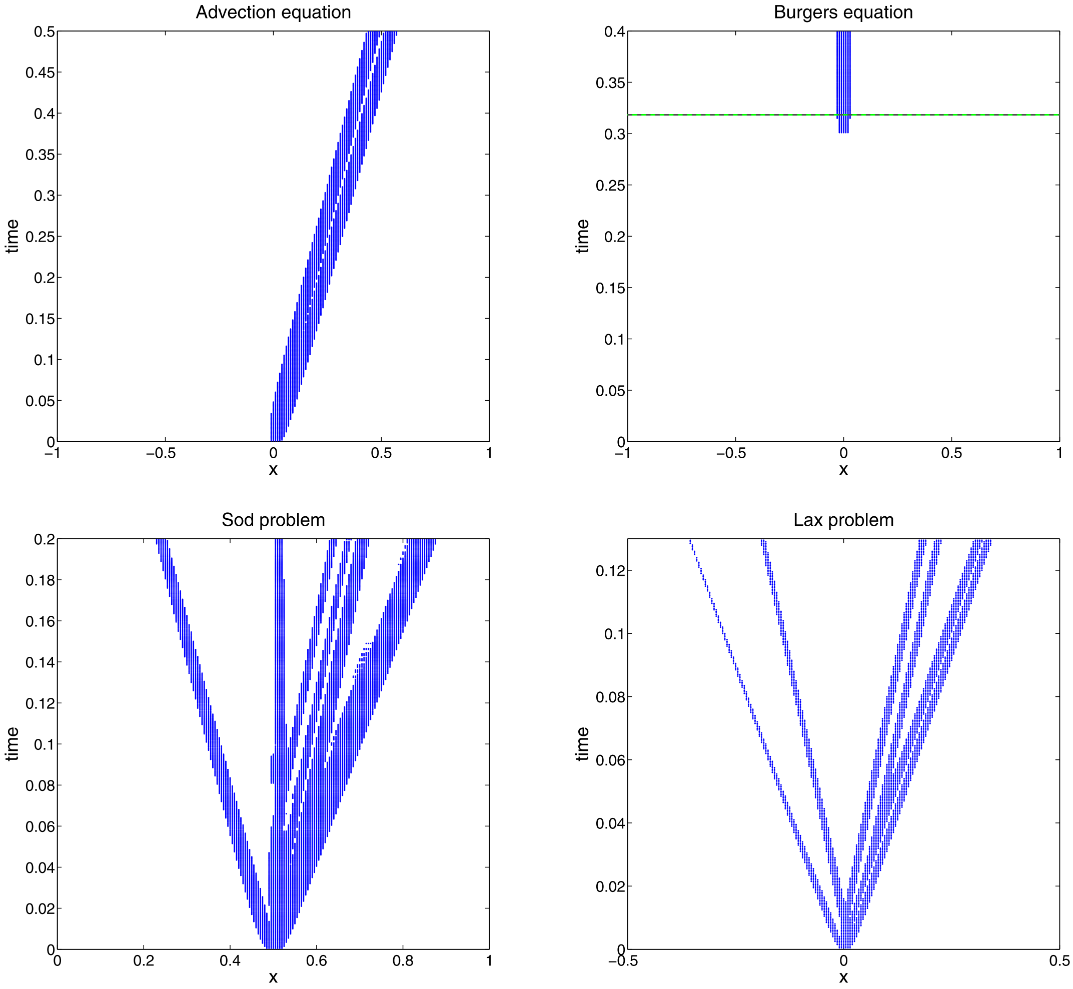

Fig. 3. Identification of the WENO cells with the monotone polynomial interpolation method. Advection equation with the discontinuous initial condition (top left), Burgers equation with the smooth initial condition (top right), Sod problem (bottom left), Lax problem (bottom right). The green line in the top right figure indicates the time at which the shock forms. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

method. The top right figure for the Burgers' equation shows that the monotone method can also distinguish the critical points from the real shock. The green line in the figure indicates the time t_s at which the shock forms. The figure shows that the WENO cells are identified as the time approaching t_s while the critical points are excluded. Bottom two figures show the results for the Sod and Lax problems, respectively. As shown in the figure, the multiple discontinuities appearing with time are all well detected with the monotone method. For the consistency of this paper, we leave the comparison of the monotone interpolation method with other edge detection method such as the multi-resolution analysis [12] in a separate paper [9]. Our preliminary results show that the proposed monotone method performs comparable to or better than the existing methods [9].

5.3. Advection equation

We solve the smooth linear scalar equation given by the following advection equation for $x \in [-1, 1]$

$$v_t + v_x = 0, \quad t > 0, \tag{36}$$

with the initial condition $v(x, 0) = \sin(\pi x)$ and the periodic boundary condition. The CFL condition is given by $\Delta t = C\Delta x$ with $C = 0.1$.

Tables 5 and 6 show the L_1, L_2 and L_∞ errors for each method at the final time $T = 0.5$ with $k = 2$ and $k = 3$, respectively. It is clear that for both $k = 2$ and $k = 3$, the convergence rate of the RBF-ENO method is one order higher than the regular ENO, while the convergence rate of the RBF-WENO-JS is similar or slightly better than the regular WENO-JS. Also we observe that for $k = 2$ the RBF-ENO is even better than the WENO-JS in terms of accuracy while both of their convergence

Table 5
L_1, L_2 and L_∞ errors for the linear advection equation, (36) with $k = 2$ at $T = 0.5$.

Method	N	L_1 error	L_1 order	L_2 error	L_2 order	L_∞ error	L_∞ order
ENO, $k = 2$	10	1.09E−1	–	1.38E−1	–	2.18E−1	–
	20	4.59E−2	1.25	5.30E−2	1.38	9.39E−2	1.22
	40	1.37E−2	1.74	1.78E−2	1.57	4.03E−2	1.22
	80	3.80E−3	1.85	5.69E−3	1.65	1.68E−2	1.26
	160	1.02E−3	1.90	1.80E−3	1.66	6.91E−3	1.28
	320	2.70E−4	1.92	5.69E−4	1.66	2.81E−3	1.30
RBF-ENO, $k = 2$	10	1.76E−2	–	2.31E−2	–	4.17E−2	–
	20	2.47E−3	2.84	2.64E−3	3.13	3.61E−3	3.53
	40	3.17E−4	2.96	3.43E−4	2.94	4.78E−4	2.92
	80	4.05E−5	2.97	4.42E−5	2.96	6.25E−5	2.94
	160	5.17E−6	2.97	5.60E−6	2.98	7.97E−6	2.96
	320	6.51E−7	2.99	7.05E−7	2.99	1.00E−6	2.99
WENO-JS, $k = 2$	10	8.94E−2	–	1.07E−1	–	1.69E−1	–
	20	2.90E−2	1.62	3.23E−2	1.74	5.47E−2	1.63
	40	4.80E−3	2.59	6.38E−3	2.36	1.37E−2	2.00
	80	6.42E−4	2.90	9.42E−4	2.74	2.60E−3	2.40
	160	7.79E−5	3.04	1.26E−4	2.94	3.96E−4	2.72
	320	9.54E−6	3.03	1.52E−5	3.00	5.24E−5	2.92
RBF-WENO, $k = 2$	10	2.20E−2	–	2.27E−2	–	3.52E−2	–
	20	2.65E−3	3.05	2.74E−3	3.05	3.74E−3	3.23
	40	3.27E−4	3.01	3.58E−4	2.94	5.08E−4	2.88
	80	4.05E−5	3.01	4.50E−5	2.99	6.61E−5	2.94
	160	5.09E−6	2.99	5.63E−6	3.00	8.27E−6	3.00
	320	6.39E−7	2.99	7.03E−7	3.00	1.00E−6	3.05

Table 6
L_1, L_2 and L_∞ errors for the linear advection equation, (36) with $k = 3$ at $T = 0.5$.

Method	N	L_1 error	L_1 order	L_2 error	L_2 order	L_∞ error	L_∞ order
ENO, $k = 3$	10	2.88E−2	–	2.50E−2	–	3.59E−2	–
	20	2.78E−3	3.04	3.03E−3	3.05	4.45E−3	3.01
	40	3.36E−4	3.05	3.68E−4	3.03	5.47E−4	3.02
	80	4.12E−5	3.03	4.54E−5	3.02	6.76E−5	3.02
	160	5.10E−6	3.01	5.65E−6	3.01	8.53E−6	3.00
	320	6.34E−7	3.01	7.03E−7	3.01	1.06E−6	3.01
RBF-ENO, $k = 3$	10	1.76E−2	–	1.93E−2	–	2.88E−2	–
	20	1.91E−3	3.20	2.36E−3	3.03	4.18E−3	2.79
	40	1.44E−4	3.73	2.08E−4	3.50	4.98E−4	3.07
	80	8.79E−6	4.04	1.56E−5	3.73	5.09E−5	3.29
	160	4.81E−7	4.19	1.05E−6	3.89	4.59E−6	3.47
	320	2.76E−8	4.12	7.27E−8	3.86	4.23E−7	3.44
WENO-JS, $k = 3$	10	9.57E−3	–	1.12E−2	–	1.60E−2	–
	20	3.99E−4	4.58	4.62E−4	4.59	7.80E−4	4.36
	40	1.18E−5	5.08	1.38E−5	5.06	2.47E−5	4.98
	80	3.70E−7	4.99	4.28E−7	5.01	7.82E−7	4.98
	160	1.34E−8	4.78	1.51E−8	4.82	2.67E−8	4.87
	320	6.56E−10	4.36	7.30E−10	4.37	1.13E−9	4.57
RBF-WENO, $k = 3$	10	2.69E−3	–	2.93E−3	–	4.19E−3	–
	20	8.92E−5	4.91	1.05E−4	4.80	1.94E−4	4.43
	40	2.53E−6	5.14	3.00E−6	5.13	6.26E−6	4.96
	80	7.52E−8	5.07	8.56E−8	5.14	1.55E−7	5.34
	160	2.35E−9	5.00	2.63E−9	5.02	4.95E−9	4.97
	320	7.39E−11	4.99	8.32E−11	4.98	1.76E−10	4.81

rates are about the 3rd order. It is interesting to see how similar they are due to the fact that we already take advantage of all the available information at each step. This could be observed again in the following example.

Consider the same advection equation (36) but with the discontinuous initial condition

$$v(x,0) = \begin{cases} 1 & x \le 0 \\ -1 & x > 0 \end{cases} \tag{37}$$

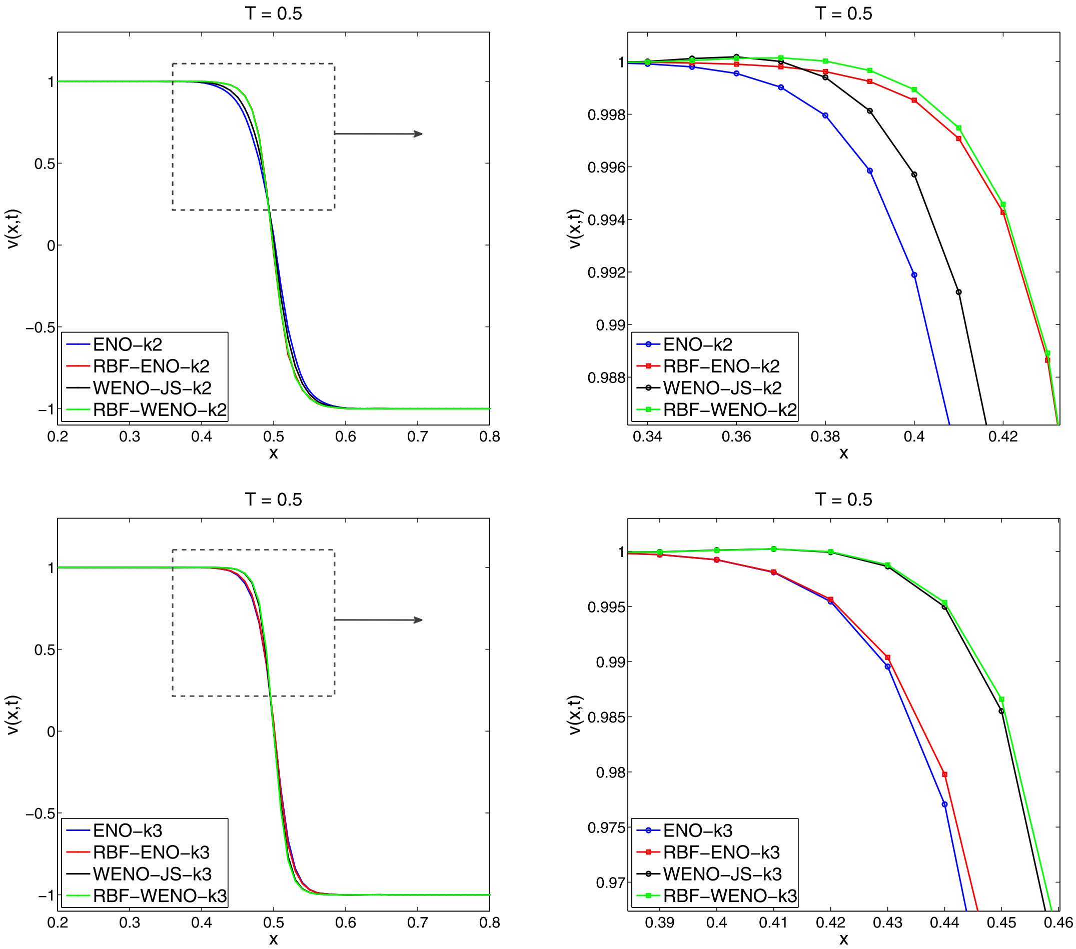

Fig. 4. Solutions to (36) at $T = 0.5$ with the discontinuous initial condition, (37) by the ENO (blue), RBF-ENO (red), WENO-JS (black) and RBF-WENO-JS (green) methods. $k = 2$ (top) and $k = 3$ (bottom). $N = 200$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

and the boundary condition $v(-1, t) = 1$ for $t > 0$. With this example, we want to check how the methods behave near the discontinuity.

Fig. 4 shows the solution profiles at $T = 0.5$ by each method with $N = 200$. The top two figures show the solutions with $k = 2$ and the bottom two figures with $k = 3$. The blue, red, black and green lines denote the ENO, RBF-ENO, WENO-JS and RBF-WENO-JS methods, respectively. As shown in the figures, the RBF-ENO solutions for both $k = 2$ and $k = 3$ are non-oscillatory. For the case of $k = 2$, the RBF-ENO solution is superior to the regular ENO and WENO-JS solutions. The ENO or RBF-ENO methods give non-oscillatory solutions due to the adaptive stencil at each reconstruction step while the WENO-JS or RBF-WENO-JS methods have no such guarantee since they use a combination of all the possible ENO reconstruction including the oscillatory ones. This is why when the RBF-ENO, WENO-JS and RBF-WENO-JS methods have the similar 3rd order convergence for $k = 2$ and the RBF-ENO gives the best non-oscillatory profile.

For the case of $k = 3$, the RBF-ENO is better than the regular ENO because it has convergence of one order higher, while the RBF-WENO-JS is slightly better than the WENO-JS in terms of accuracy and both of them have 5th order convergence.

5.4. Burgers' equation

Consider the Burgers' equation for $x \in [-1, 1]$

$$v_t + \left(\frac{1}{2}v^2\right)_x = 0, \quad t > 0, \tag{38}$$

$$v(x, 0) = -\sin(\pi x). \tag{39}$$

For this problem, a shock forms at $(x, t) = (0, \frac{1}{\pi})$. We first check if each method achieves its desired order when the solution is still smooth at $T = 0.2$. Tables 7 and 8 show the L_1, L_2 and L_∞ errors by the ENO, RBF-ENO, WENO-JS and RBF-WENO-JS

Table 7
L_1, L_2 and L_∞ errors for the Burgers' equation, (38) with $k = 2$ at $T = 0.2$.

Method	N	L_1 error	L_1 order	L_2 error	L_2 order	L_∞ error	L_∞ order
ENO, $k = 2$	10	9.58E−2	–	1.29E−1	–	2.84E−1	–
	20	3.01E−2	1.67	4.54E−2	1.51	1.12E−1	1.33
	40	9.03E−3	1.74	1.49E−2	1.61	5.68E−2	0.99
	80	2.49E−3	1.86	4.33E−3	1.78	1.64E−2	1.79
	160	6.77E−4	1.88	1.23E−3	1.81	4.17E−3	1.80
	320	1.78E−4	1.92	3.51E−4	1.81	1.52E−3	1.63
RBF-ENO, $k = 2$	10	5.40E−2	–	9.82E−2	–	2.18E−1	–
	20	9.56E−3	2.50	2.15E−2	2.19	7.47E−2	1.55
	40	1.46E−3	2.70	3.55E−3	2.60	1.65E−2	2.18
	80	1.85E−4	2.98	4.96E−4	2.84	2.58E−3	2.68
	160	2.28E−5	3.03	6.20E−5	3.00	3.69E−4	2.81
	320	2.78E−6	3.04	7.55E−6	3.04	4.51E−5	3.03
WENO-JS, $k = 2$	10	7.45E−2	–	1.13E−1	–	2.83E−1	–
	20	2.24E−2	1.73	3.77E−2	1.58	1.02E−1	1.47
	40	4.46E−3	2.33	8.17E−3	2.21	2.99E−2	1.77
	80	6.31E−4	2.82	1.13E−3	2.85	4.12E−3	2.86
	160	8.07E−5	2.97	1.44E−4	2.97	5.13E−4	3.01
	320	1.01E−5	2.99	1.79E−5	3.00	6.28E−5	3.03
RBF-WENO-JS, $k = 2$	10	5.44E−2	–	9.82E−2	–	2.18E−1	–
	20	9.67E−3	2.49	2.15E−2	2.19	7.55E−2	1.53
	40	1.44E−3	2.75	3.54E−3	2.61	1.66E−2	2.18
	80	1.86E−4	2.95	4.95E−4	2.85	2.60E−3	2.68
	160	2.27E−5	3.03	6.18E−5	3.00	3.66E−4	2.83
	320	2.77E−6	3.04	7.53E−6	3.04	4.50E−5	3.02

Table 8
L_1, L_2 and L_∞ errors for the Burgers' equation, (38) with $k = 3$ at $T = 0.2$.

Method	N	L_1 error	L_1 order	L_2 error	L_2 order	L_∞ error	L_∞ order
ENO, $k = 3$	10	4.32E−2	–	8.33E−2	–	2.49E−1	–
	20	9.68E−3	2.16	2.02E−2	2.04	7.24E−2	1.79
	40	1.47E−3	2.72	3.11E−3	2.70	1.44E−2	2.33
	80	2.23E−4	2.72	4.99E−4	2.64	2.47E−3	2.54
	160	3.11E−5	2.84	7.05E−5	2.83	3.63E−4	2.77
	320	4.31E−6	2.84	9.60E−6	2.88	4.45E−5	3.03
RBF-ENO, $k = 3$	10	3.45E−2	–	6.61E−2	–	1.94E−1	–
	20	7.76E−3	2.16	1.91E−2	1.79	6.57E−2	1.57
	40	1.24E−3	2.65	4.47E−3	2.01	2.90E−2	1.18
	80	8.64E−5	3.84	3.71E−4	3.68	3.09E−3	3.23
	160	8.39E−6	3.36	3.37E−5	3.46	3.83E−4	3.01
	320	6.14E−7	3.77	1.66E−6	4.34	1.75E−5	4.45
WENO-JS, $k = 3$	10	3.25E−2	–	7.21E−2	–	2.21E−1	–
	20	4.24E−3	2.94	1.33E−2	2.43	5.83E−2	1.92
	40	4.19E−4	3.34	1.46E−3	3.19	8.75E−3	2.74
	80	2.45E−5	4.10	9.00E−5	4.02	6.06E−4	3.85
	160	9.42E−7	4.70	3.39E−6	4.73	2.50E−5	4.60
	320	2.94E−8	5.00	1.08E−7	4.98	8.18E−7	4.93
RBF-WENO-JS, $k = 3$	10	3.45E−2	–	5.90E−2	–	1.55E−1	–
	20	3.77E−3	3.19	9.28E−3	2.67	3.87E−2	2.01
	40	3.17E−4	3.57	9.66E−4	3.26	5.06E−3	2.79
	80	1.86E−5	4.09	5.60E−5	4.11	3.66E−4	3.94
	160	9.62E−7	4.28	2.57E−6	4.45	1.39E−5	4.71
	320	2.86E−8	5.07	8.21E−8	4.97	4.67E−7	4.90

methods for $k = 2$ and $k = 3$. Fig. 5 shows the pointwise errors of the solution at $T = \frac{1}{\pi}$ with $k = 2$ and $k = 3$. The blue, red, black and green lines denote the ENO, RBF-ENO, WENO-JS and RBF-WENO-JS methods, respectively. The RBF-ENO method and the RBF-WENO-JS method yield almost the same accurate results due to the fact that we already took advantage of all the available information. For the case of $k = 3$, the RBF-WENO-JS method yields the most accurate results. From this figure, we observe that the RBF-ENO method gives smaller pointwise errors than the regular ENO method, which confirms that the RBF-ENO method much improves the original ENO accuracy in the smooth area.

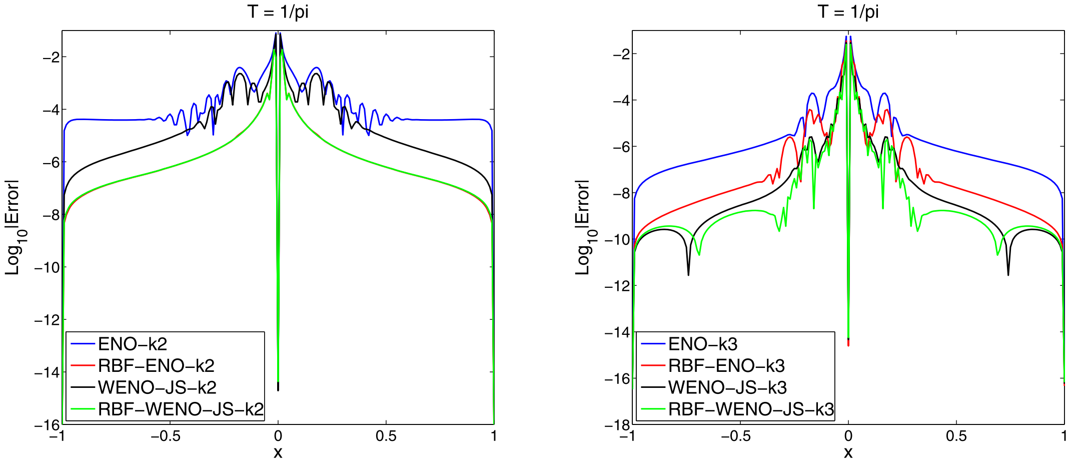

Fig. 5. Pointwise errors for Burgers' equation at $T = \frac{1}{\pi}$ by the ENO (blue), RBF-ENO (red), WENO-JS (black) and RBF-WENO-JS (green) methods. $N = 200$. Left: $k = 2$. Right: $k = 3$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5.5. *Euler equations*

Finally we consider the one-dimensional Euler equations for gas dynamics

$$U_t + F(U)_x = 0, \tag{40}$$

where the conservative state vector U and the flux function F are given by

$$U = (\rho,\ \rho u,\ E)^T, \quad F(U) = (\rho u,\ \rho u^2 + P,\ (E+P)u)^T.$$

Here ρ, u, P and E denote density, velocity, pressure and total energy, respectively. The equation of state is given by

$$P = (\gamma - 1)\left(E - \frac{1}{2}\rho u^2\right),$$

where $\gamma = 1.4$ for the ideal gas.

First consider the Sod problem [26] with the initial condition

$$(\rho, u, P) = \begin{cases} (1, 0, 1) & x \le 0 \\ (0.125, 0, 0.1) & x > 0 \end{cases}. \tag{41}$$

The Sod problem is solved with $N = 600$ and the CFL number $C = 0.1$. Fig. 6 shows the density profile at $T = 0.2$ for $k = 2$ by each method. The blue, red, black and green lines denote the ENO, RBF-ENO, WENO-JS and RBF-WENO-JS methods, respectively. The exact solution on 2000 grid points is also provided. The top left figure shows the overall density profile and the other figures show the detailed density profiles for the regions of A, B and C indicated in the top left figure. In all areas of A, B and C, the RBF-ENO method yields better (sharper) density profiles than the ENO and WENO-JS methods, while it is almost the same as the RBF-WENO-JS. We observe that the WENO-JS solution is slightly oscillatory while the RBF-ENO and ENO solutions are not oscillatory.

Fig. 7 shows the same density profiles for $k = 3$. For this case, the RBF-ENO method yields 4th order accuracy in the smooth area while the ENO and WENO-JS methods yield 3rd order and 5th order accuracy, respectively. Thus unlike the case of $k = 2$, it is reasonable to observe that the RBF-WENO-JS method yields the best result among those three methods. The RBF-ENO solution is, however, still better than the regular ENO solution. Near the shock area, all three methods yield almost similar profiles but the RBF-WENO-JS solution is sharper than the others and the RBF-ENO solution is sharper than the regular ENO method.

In Table 9, we show the CPU timing for the Sod problem using each method. "Speed up" is computed as the ratio of the CPU time of the RBF-ENO/WENO methods to that of the regular ENO/WENO methods. Due to the fact the RBF-ENO method has extra terms in the reconstruction coefficients compared to regular ENO method (see Tables 2 and 3), the speedup is always larger than 1. This means the RBF reconstruction is more expensive than the original polynomial reconstruction. However, the speedup is close to 1, especially when the resolution is high. This means the extra work is minor compared to all the necessary ENO/WENO steps such as deciding which stencil to use from Newton's divided difference or computing smoothness indicators. To sum up, there is a trade-off between accuracy and computational cost, but the additional computational cost to get better convergence and accuracy is not significant.

Now consider the shock entropy wave interaction problem [25] with the following initial condition

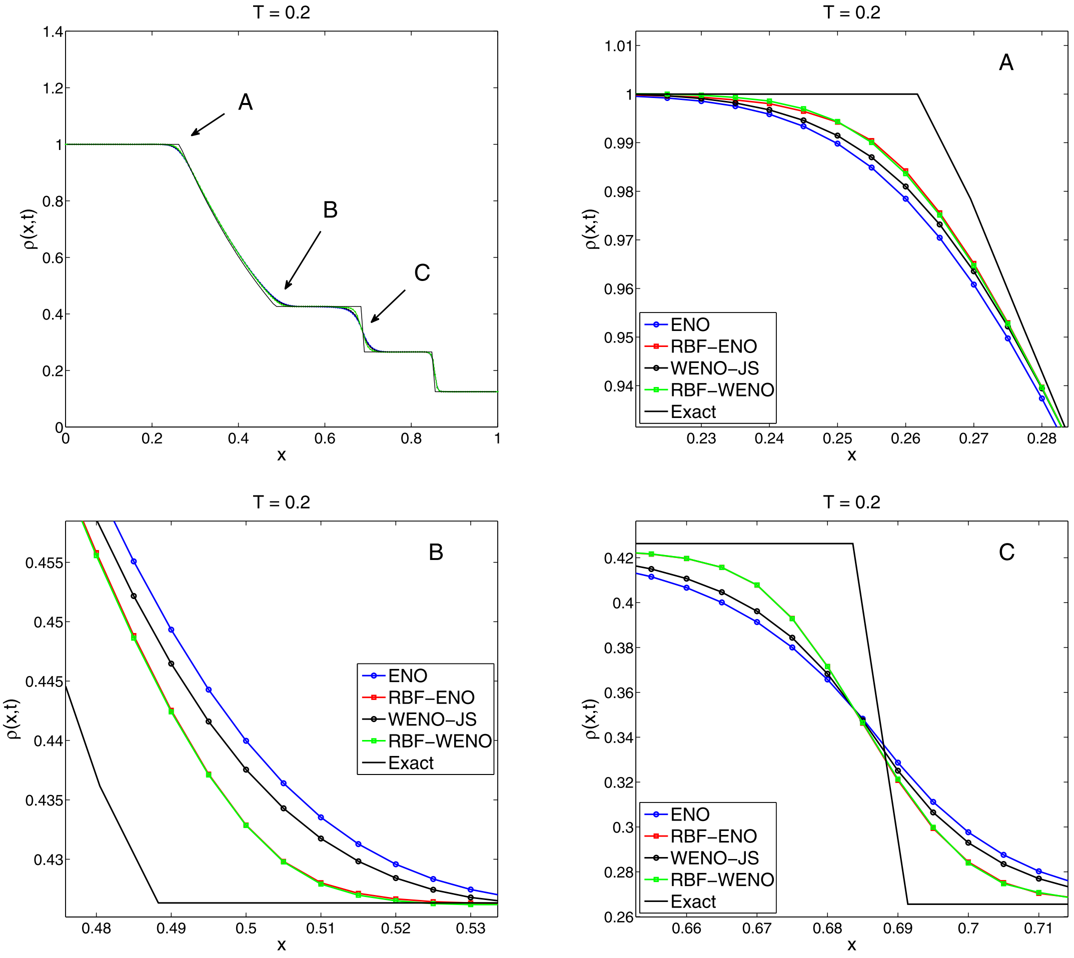

Fig. 6. Density profile for the Sod problem by the ENO (blue), RBF-ENO (red), WENO-JS (black), RBF-WENO-JS (green) methods and the exact solution at $T = 0.2$ with $N = 600$. $k = 2$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 9
The CPU timing of and speedup from the original ENO/WENO methods to the ENO/WENO methods for the Sod problem at various resolutions and at final time $T = 0.2$.

N	ENO-k2	RBF-ENO-k2	Speedup	WENO-JS-k2	RBF-WENO-JS-k2	Speedup
400	22.19	23.40	1.0545	26.04	26.10	1.0023
800	87.42	90.59	1.0363	102.32	102.48	1.0016
1600	346.83	358.22	1.0328	406.39	407.09	1.0017
2400	786.05	805.46	1.0247	912.15	912.15	1.0000

N	ENO-k3	RBF-ENO-k3	Speedup	WENO-JS-k3	RBF-WENO-JS-k3	Speedup
400	27.02	29.05	1.0751	29.89	32.39	1.0836
800	106.88	114.27	1.0691	118.03	129.43	1.0966
1600	423.71	453.96	1.0714	470.17	486.86	1.0355
2400	950.58	1019.71	1.0727	1058.71	1059.13	1.0004

$$(\rho, u, P) = \begin{cases} (\frac{27}{7}, \frac{4\sqrt{35}}{9}, \frac{31}{3}) & x \leq -9.5 \\ (e^{-\epsilon \sin(k(x-9.5))}, 0, 1) & x > -9.5 \end{cases},$$

where $\epsilon = 0.01$, $k = 13$.

Fig. 8 shows the solution profiles solved by the WENO-JS (blue), RBF-WENO-JS (black), WENO-Z (red) and RBF-WENO-Z (green) with $k = 3$. For either the low resolution $N = 1500$ or the high resolution $N = 2560$, adopting RBF basis yields a less dissipative solution for the WENO-JS or WENO-Z framework. This makes the RBF-WENO-Z (green) the best among all the four methods.

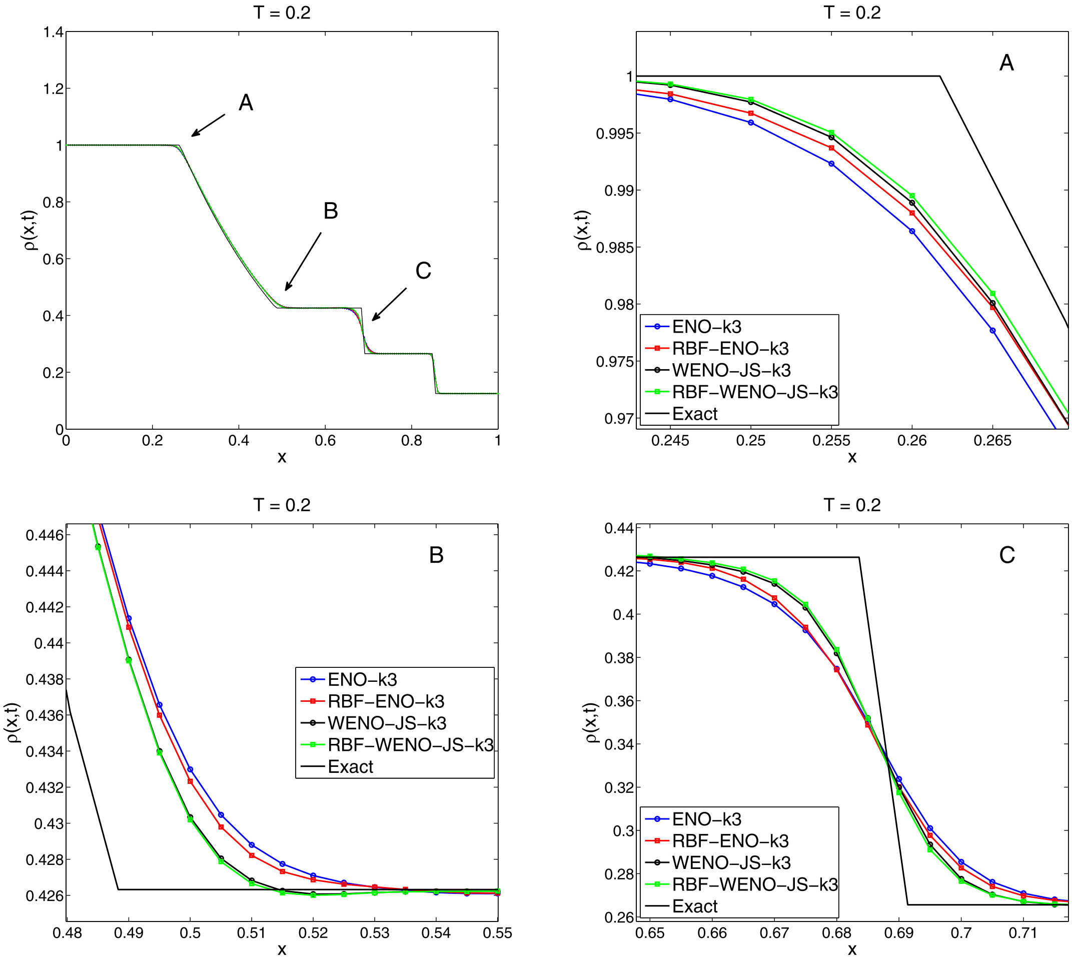

Fig. 7. Density profile for the Sod problem by the ENO (blue), RBF-ENO (red), WENO-JS (black), RBF-WENO-JS (green) methods and the exact solution at $T = 0.2$ with $N = 600$. $k = 3$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

6. 2D polynomial and non-polynomial interpolation

To illustrate the 2D reconstruction [17], consider the case with $k = 2$, for which at least $\frac{1}{2}k(k+1) = 3$ cells are needed for the ENO reconstruction among the available 5 cell averages, $\bar{v}_{i-1,j}$, $\bar{v}_{i,j}$, $\bar{v}_{i+1,j}$, $\bar{v}_{i,j-1}$ and $\bar{v}_{i,j+1}$ (see Fig. 9). For simplicity, the uniform grid is used, i.e. $\Delta x_i = \Delta x,\ \forall i$ and $\Delta y_i = \Delta y,\ \forall j$. For the illustration, assume that Newton's divided difference method determined to use the following 3 cell averages, $\bar{v}_{i,j}$, $\bar{v}_{i+1,j}$ and $\bar{v}_{i,j-1}$. We only show the reconstruction at the cell boundary of $(x, y) = (x_{i+\frac{1}{2}}, y_j)$. The reconstruction at other cell boundaries can be achieved similarly.

6.1. Polynomial reconstruction

Assume that the function $s(x, y)$ that interpolates the given 3 cell averages $\bar{v}_{i,j}$, $\bar{v}_{i+1,j}$, $\bar{v}_{i,j-1}$ is

$$s(x, y) = \lambda_1 + \lambda_2 x + \lambda_3 y.$$

Let $\vec{V} = [\bar{v}_{i,j}, \bar{v}_{i+1,j}, \bar{v}_{i,j-1}]^T$, $\vec{\lambda} = [\lambda_1, \lambda_2, \lambda_3]^T$ and the interpolation matrix be

$$A = \begin{bmatrix} 1 & x_i & y_j + dy \\ 1 & x_i & y_j \\ 1 & x_i + dx & y_j \end{bmatrix}.$$

Then the expansion coefficients λ_i are given by solving the linear system $\vec{V} = A \cdot \vec{\lambda}$. After plugging λ_i in $s(x, y)$ at $(x, y) = (x_{i+\frac{1}{2}}, y_j)$, we obtain

$$v^-_{i+\frac{1}{2},j} = s(x_{i+\frac{1}{2}}, y_j) = \frac{1}{2} \cdot \bar{v}_{i,j} + \frac{1}{2} \cdot \bar{v}_{i+1,j} + 0 \cdot \bar{v}_{i,j-1}. \tag{42}$$

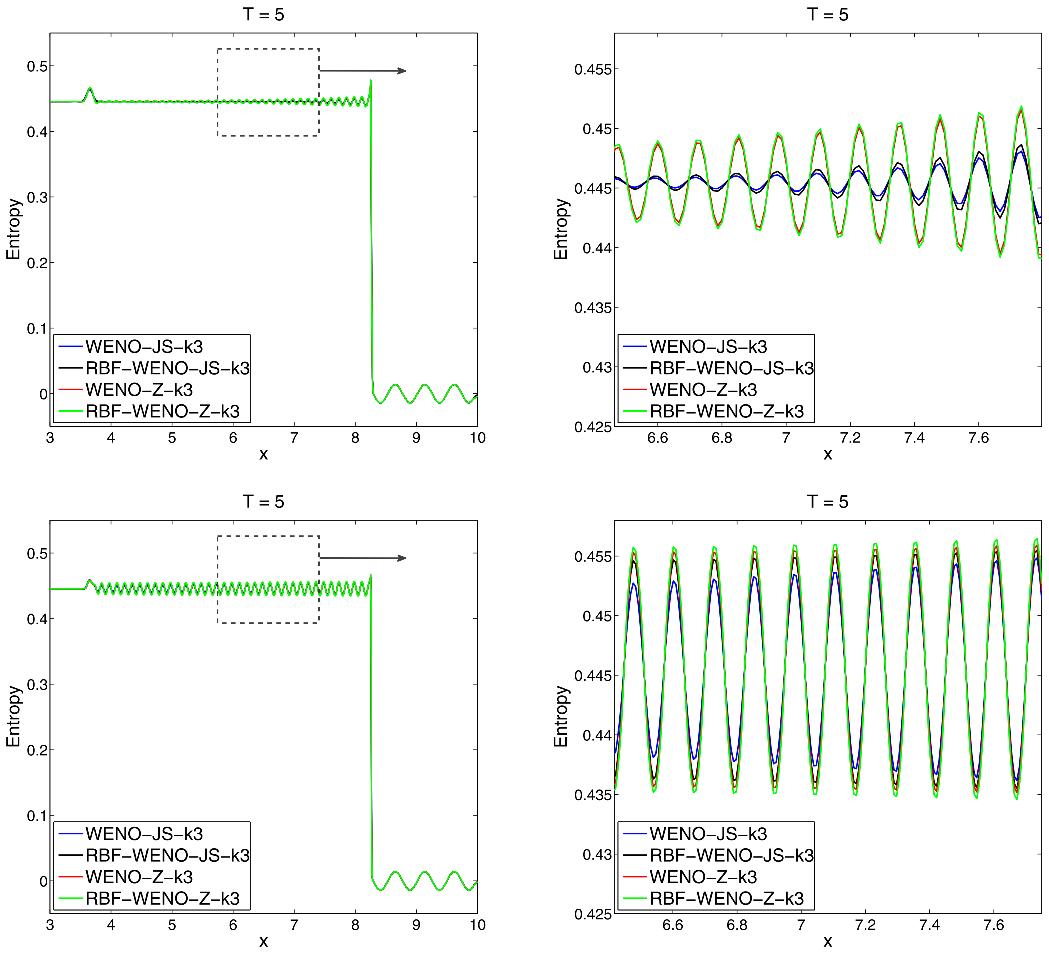

Fig. 8. Density profile for the shock entropy interaction problem by the WENO-JS-k3 (blue), RBF-WENO-k3 (black), WENO-Z-k3 (red), RBF-WENO-Z-k3 (green) methods at $T = 5$ with $N = 1500$ (top) and $N = 2560$ (bottom). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

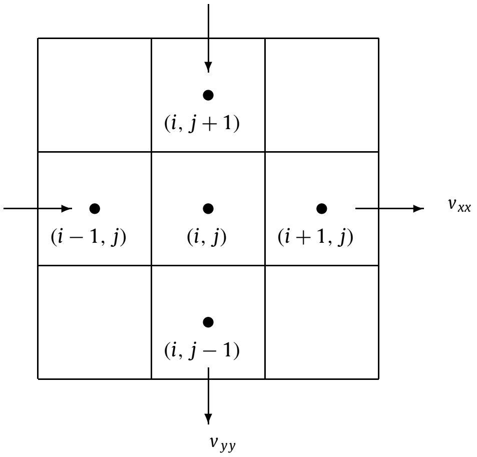

Fig. 9. Five cell averages involved for the reconstruction at the cell boundaries for $k = 2$.

Expanding $v^-_{i+\frac{1}{2},j}$ around $x = x_{i+\frac{1}{2}}$ and $y = y_j$ in a Taylor series yields

$$v^-_{i+\frac{1}{2},j} = v(x_{i+\frac{1}{2}}, y_j) + \frac{1}{6} v_{xx}(x_{i+\frac{1}{2}}, y_j)\Delta x^2 + \frac{1}{24} v_{yy}(x_{i+\frac{1}{2}}, y_j)\Delta x^2 + O(\Delta x^3) + O(\Delta y^3). \tag{43}$$

$v(x_{i+\frac{1}{2}}, y_j)$ in (43) is the exact value of $v(x)$ at $x = x_{i+\frac{1}{2}}$ and $y = y_j$. So we confirm that (42) is a 2nd order reconstruction.

6.2. Perturbed polynomial reconstruction

The 2D RBF reconstruction can be obtained as a straightforward extension of the 1D RBF reconstruction. However, the calculation can be complicated due to the double integrals of RBFs. For this reason, we consider this problem from a different perspective. That is, instead of directly constructing the 2D RBF interpolation with a specific RBF basis function, we modify the polynomial reconstruction by adding small perturbation terms containing the shape parameter as we did in Section 3.4. As in the 1D reconstruction, we then optimize the shape parameter to increase the order of accuracy.

Mimicking the 1D RBF interpolation, we assume that the perturbed 2D polynomial interpolation, (42), is given by the following form

$$v^-_{i+\frac{1}{2},j} = \left(\frac{1}{2} + c_1\epsilon^2 h^2\right)\cdot \bar{v}_{i,j} + \left(\frac{1}{2} + c_2\epsilon^2 h^2\right)\cdot \bar{v}_{i+1,j} + \left(0 + c_3\epsilon^2 h^2\right)\cdot \bar{v}_{i,j-1}, \tag{44}$$

where $h = h(\Delta x, \Delta y)$. Expanding $v^-_{i+\frac{1}{2},j}$ around $x = x_{i+\frac{1}{2}}$ and $y = y_j$ in a Taylor series yields

$$\begin{aligned} v^-_{i+\frac{1}{2},j} &= v(x_{i+\frac{1}{2}}, y_j) + \frac{1}{6}v_{xx}(x_{i+\frac{1}{2}}, y_j)\Delta x^2 + \frac{1}{24}v_{yy}(x_{i+\frac{1}{2}}, y_j)\Delta y^2 \\ &+ (c_1 + c_2 + c_3)\epsilon^2 h^2 v(x_{i+\frac{1}{2}}, y_j) \\ &+ O(\Delta x^3) + O(\Delta y^3). \end{aligned} \tag{45}$$

Thus if we take the value of ϵ as below

$$\epsilon^2 = -\frac{\frac{1}{6}v_{xx}(x_{i+\frac{1}{2}}, y_j)\Delta x^2 + \frac{1}{24}v_{yy}(x_{i+\frac{1}{2}}, y_j)\Delta y^2}{(c_1 + c_2 + c_3)v(x_{i+\frac{1}{2}}, y_j)h^2}, \tag{46}$$

then we obtain a 3rd order accurate approximation. Again, the exact values of $v_{xx}(x_{i+\frac{1}{2}}, y_j)$, $v_{yy}(x_{i+\frac{1}{2}}, y_j)$ and $v(x_{i+\frac{1}{2}}, y_j)$ in (46) are not available, so we replace them with their approximations based on the given cell averages. In fact, it is possible to approximate all those quantities using the given 5 cell averages. For example, $v_{xx}(x_{i+\frac{1}{2}}, y_j)$ and $v_{yy}(x_{i+\frac{1}{2}}, y_j)$ can be easily approximated using $\{\bar{v}_{i-1,j},\ \bar{v}_{i,j},\ \bar{v}_{i+1,j}\}$ and $\{\bar{v}_{i,j-1},\ \bar{v}_{i,j},\ \bar{v}_{i,j+1}\}$, respectively with a 2nd order accuracy (see Fig. 9). Then for this case, ϵ^2 is approximated by

$$\epsilon^2 \approx \frac{\bar{v}_{i,j+1} + 4\bar{v}_{i-1,j} - 10\bar{v}_{i,j} + 4\bar{v}_{i+1,j} + \bar{v}_{i,j-1}}{4(c_1 + c_2 + c_3)(-2\bar{v}_{i-1,j} + \bar{v}_{i,j} - 5\bar{v}_{i+1,j})h^2 + \epsilon_M}. \tag{47}$$

To check whether this approximation of ϵ^2 still achieves the 3rd order accuracy, we expand $v^-_{i+\frac{1}{2},j}$ around $x = x_{i+\frac{1}{2}}$ and $y = y_j$ in a Taylor series. After a small calculation, we confirm that

$$v^-_{i+\frac{1}{2},j} = s(x_{i+\frac{1}{2}}, y_j) = v(x_{i+\frac{1}{2}}, y_j) + O(\Delta x^3) + O(\Delta y^3).$$

Here note that for the 3rd order approximation, we used up all the given cell averages. The polynomial interpolation, even in the case that all the cell averages are used, still yields a 2nd order approximation because only 5 cell averages are used. The perturbation terms in (44) indeed yield the flexibility to use all the possible approximation from the given 5 cell averages.

7. 2D numerical experiments

For the 2D numerical examples, we consider the 2D hyperbolic conservation laws

$$v_t + f(v)_x + g(v)_y = 0.$$

In each cell $I_{i,j}$, we have

$$\int_{I_{i,j}} v_t(x,y,t)dxdy = -\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} f(v(x_{i+\frac{1}{2}}, y)) - f(v(x_{i-\frac{1}{2}}, y))dy - \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} g(v(x, y_{j+\frac{1}{2}})) - g(v(x, y_{j-\frac{1}{2}}))dx. \tag{48}$$

Thus the 2D finite volume ENO/WENO methods involve the approximation to integrals and the overall order provided by the method depends not only on the value of k, but also on the number of the Gaussian quadrature points used for the integrals. If the method is only of 2nd order accurate (polynomial interpolation with $k = 2$), then one quadrature point at the cell boundaries is enough for the integrals to maintain the same order. The RBF-ENO reconstruction is of the 3rd order accurate, so we need to use at least two quadrature points for the desired order. If one quadrature point is used as the

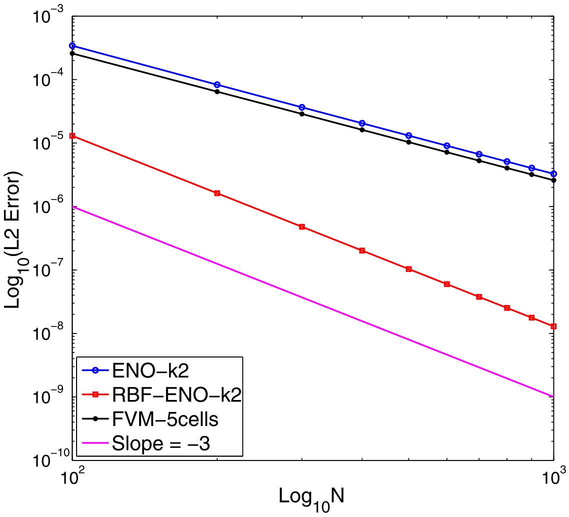

Fig. 10. L_2 errors versus N in logarithmic scale with $k = 2$. ENO method (blue), RBF-ENO method (red), 5-point Finite Volume method (black). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

regular ENO method instead, however, the RBF-ENO still maintains the 2nd order accuracy, but it yields higher accuracy than the regular ENO method because the RBF-ENO reconstruction is already 3rd order accurate. Since we only want to modify the existing 2nd order regular ENO code with the minimum changes, we still use only one quadrature point and yet improve accuracy.

7.1. *Reconstruction order*

First we check the reconstruction error by the RBF-ENO method applied to a smooth function $v(x) = \sin(2\pi(x + y))$ with $k = 2$ to confirm the desired order of convergence. If five cell averages, $\bar{v}_{i-1,j}$, $\bar{v}_{i,j}$, $\bar{v}_{i+1,j}$, $\bar{v}_{i,j-1}$, and $\bar{v}_{i,j+1}$ are available, the regular ENO method chooses three of them by the Newton's divided difference method, which ends up with the 2nd order convergence. The RBF-ENO method also chooses three cell averages from the Newton's divided difference, but the five cell averages are used to optimize the shape parameter. This will provide the 3rd order accuracy for the smooth problem. For comparison, we also use the polynomial reconstruction using all the five cell averages. If we assume the uniform grid with $\Delta x = \Delta y$, the possible interpolation using the 5 cells is given by

$$s(x, y) = c_0 + c_1 x + c_2 y + c_3 x^2 + c_4 y^2, \tag{49}$$

where c_i, $i = 0, \cdots, 4$, are all constants determined by the given five cell averages. Here note that there is no cross term, xy, due to the axial symmetry. Thus we only expect at most the 2nd order convergence although those 5 cell averages are all used while the RBF-ENO can yield the 3rd order convergence.

Fig. 10 shows the L_2 errors versus N in logarithmic scale for the regular ENO (blue circle), RBF-ENO (red square) and the 5-cell finite volume method (black filled circle) reconstructions. For the 5-cell finite volume method, (49) is used. The solid line in magenta is the reference line of order 3. As shown in the figure, the RBF-ENO reconstruction yields the desired 3rd order convergence and is much more accurate than the regular ENO method or the 5-cell finite volume method. The regular ENO and the 5-cell finite volume methods yield only the 2nd order convergence. We also note that the difference between the 5-cell finite volume reconstruction with (49) and the regular ENO reconstruction is not significant.

7.2. *2D advection equation*

We solve the advection equation for $(x, y) \in [0, 1] \times [0, 1]$

$$v_t + v_x + v_y = 0, \quad t > 0, \tag{50}$$

with the initial condition $v(x, 0) = \sin(2\pi(x + y))$ and the periodic boundary condition. The CFL condition is given by $\Delta t = C \max(\Delta x, \Delta y)$ with $C = 0.1$.

Table 10 shows the L_1, L_2, and L_∞ errors at the final time, $T = 0.5$ for the regular ENO method (blue), the RBF-ENO method (red) and the 5-cell finite volume (black). For the integral in the RHS of (48), we used one quadrature point at the mid point of each cell boundary, which makes all those methods yield the 2nd order convergence at most. From the table, we observe that the RBF-ENO is more accurate than the regular ENO or the 5-cell finite volume method. This is again because the reconstruction at the quadrature points by the RBF-ENO is 3rd order accurate.

Table 10
The L_1, L_2 and L_∞ errors for the advection equation, (50) with $k = 2$ at $T = 0.5$.

Method	N	L_1 error	L_1 order	L_2 error	L_2 order	L_∞ error	L_∞ order
ENO	10	9.90E−2	–	1.10E−1	–	1.74E−1	–
	20	3.86E−2	1.36	4.26E−2	1.37	7.99E−2	1.12
	40	1.08E−2	1.84	1.40E−2	1.60	3.43E−2	1.22
	80	3.00E−3	1.85	4.47E−3	1.65	1.43E−2	1.26
	160	8.01E−4	1.91	1.42E−3	1.65	5.91E−3	1.28
	320	2.12E−4	1.92	4.52E−4	1.65	2.40E−3	1.30
5 point finite volume	10	5.86E−2	–	5.94E−2	–	7.53E−2	–
	20	1.43E−2	2.03	1.51E−2	1.98	2.03E−2	1.89
	40	3.48E−3	2.04	3.77E−3	2.00	5.19E−3	1.97
	80	8.58E−4	2.02	9.42E−4	2.00	1.31E−3	1.98
	160	2.13E−4	2.01	2.35E−4	2.00	3.30E−4	1.99
	320	5.30E−5	2.01	5.87E−5	2.00	8.28E−5	2.00
RBF-ENO	10	1.31E−2	–	1.41E−2	–	2.20E−2	–
	20	3.52E−3	1.90	3.77E−3	1.90	5.70E−3	1.95
	40	8.57E−4	2.04	9.40E−4	2.01	1.30E−3	2.13
	80	2.10E−4	2.03	2.31E−5	2.02	3.21E−4	2.02
	160	5.20E−5	2.02	5.74E−6	2.01	8.08E−5	1.99
	320	1.29E−5	2.01	1.43E−7	2.01	2.02E−5	2.00

Now consider the same linear advection equation (50) with the following discontinuous initial condition

$$v(x, y, 0) = \begin{cases} 1 & \text{if} \quad y \in [0, 0.5] \\ -1 & \text{if} \quad y \in (0.5, 1] \end{cases}, \tag{51}$$

and the boundary condition $v(x, 0, t) = 1$, $t > 0$. With this example, we check how the RBF-ENO solution behaves near the discontinuity. To handle the discontinuous solution, we apply the same monotone method as in 1D. In Fig. 11, the solutions by the regular ENO (blue) and RBF-ENO (red) methods are given with time at $x = 0.5$. The figure clearly shows that the RBF-ENO yields non-oscillatory solutions for all time. Also RBF-ENO method yields much sharper solution profile than the regular ENO method, which is more obvious in the bottom right close-up view of the solution at $T = 0.5$.

7.3. 2D Burgers' equation

We consider the 2D Burgers' equation for $(x, y) \in [0, 1] \times [0, 1]$

$$v_t + \left(\frac{1}{2}v^2\right)_x + \left(\frac{1}{2}v^2\right)_y = 0, \quad t > 0, \tag{52}$$

with the initial condition

$$v(x, 0) = \sin(2\pi(x + y)). \tag{53}$$

Fig. 12 shows the RBF-ENO solutions at various times at $x = 0.5$ (left figure) and the pointwise errors by the ENO (blue) and RBF-ENO (red) methods with $k = 2$ and $M = N = 100$ at $t = 1/4\pi$. The left figure clearly shows that the RBF-ENO solution is not oscillatory yet yielding a sharp shock profile near the boundaries. The right figure shows that the RBF-ENO method yields more accurate results than the regular ENO method in the smooth region.

8. Conclusion

In this paper, we developed a finite volume non-polynomial ENO/WENO method for solving hyperbolic conservation laws. As an example of non-polynomial basis, RBFs are used. The formulation based on the non-polynomial basis yields the flexibility of improving the original ENO/WENO accuracy. The key idea of the developed method lies in the adaptation of the shape parameters in the expansion with a non-polynomial basis that can make the leading error term vanish or at least become small in the local interpolation. The new non-polynomial ENO/WENO method improves local accuracy and convergence if the underlying solution is smooth. For the non-smooth solution such as a shock, we proposed the monotone polynomial interpolation method so that the non-polynomial ENO/WENO reconstruction is reduced into the regular ENO/WENO reconstruction resulting in the suppression of the Gibbs oscillations. The proposed monotone method identifies the non-smooth area well efficiently using the given cell information only within the given stencil. The numerical results show that the non-polynomial ENO method is superior to the regular ENO method and even better than the WENO-JS method for $k = 2$. While for $k = 3$, the non-polynomial ENO method yields 4th order accuracy while the regular ENO solution is only 3rd order accurate in the smooth region. The WENO method based on the non-polynomial ENO reconstruction is also better than the regular WENO method in terms of accuracy although they have the same order of convergence. The

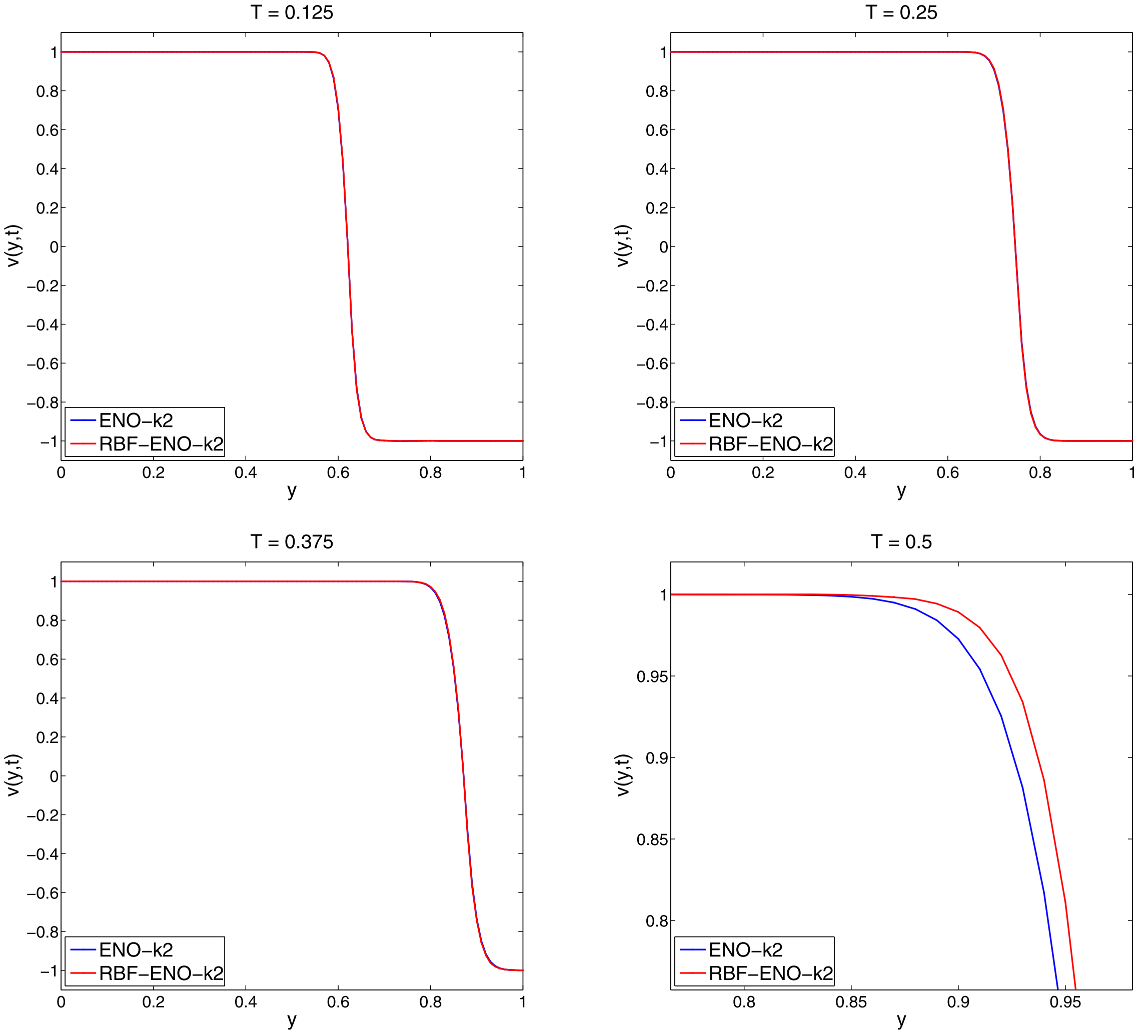

Fig. 11. Solutions to (50) at $x = 0.5$ with the discontinuous initial condition, (51) for the ENO (blue) and RBF-ENO (red) methods with $k = 2$ and $M = N = 100$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

non-polynomial ENO method is slightly more costly than the regular ENO method because it has a procedure of computing the optimal shape parameter values. But it is less expensive than the WENO method for the given value of k. For some cases, the non-polynomial ENO method achieves the same level of accuracy as the WENO method or even better accuracy than the WENO method while its computational cost is less demanding than the WENO method. The 2D non-polynomial finite volume interpolation is more beneficial than the 2D polynomial interpolation. We showed that the non-polynomial interpolation can raise the order which can not be obtained with the polynomial interpolation even though all the cell averages are used.

The non-polynomial ENO formulation for higher values of k will be considered in our future work, as well as dealing with nonuniform and unstructured mesh. The proposed idea will be applied to the construction of the finite difference RBF-WENO method [10]. As mentioned in Introduction, the meshless feature of RBFs was combined with the WENO method in [1] where the shape parameter was globally fixed for the reconstruction. It will be interesting to investigate how the optimization of the shape parameter can be realized with the meshless properties of RBFs on the unstructured mesh.

Acknowledgements

The authors thank W.-S. Don for his useful comments on the construction of the RBF-ENO/WENO method. The second author thanks Grady Wright for his useful comments on the RBF interpolation. The authors also thank the anonymous reviewers for their helpful comments.

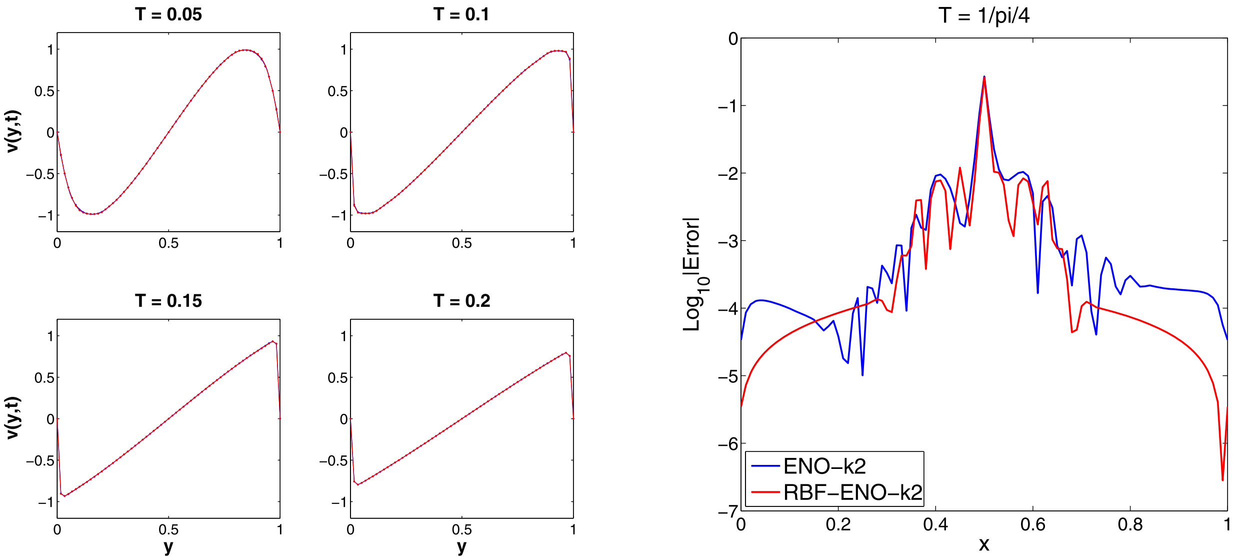

Fig. 12. Left: The RBF-ENO solutions to (52) at various time. Right: The pointwise errors at $t = \frac{1}{4\pi}$ by the ENO (blue) and RBF-ENO (red) in logarithmic scale with $k = 2$ and $M = N = 100$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

References

[1] T. Aboiyar, E.H. Georgoulis, A. Iske, Adaptive ADER methods using kernel-based polyharmonic spline WENO reconstruction, SIAM J. Sci. Comput. 32 (2010) 3251–3277.

[2] M. Berzins, Nonlinear data-bounded polynomials approximations and their applications in ENO methods, Numer. Algorithms 55 (2010) 171–189.

[3] R. Borges, M. Carmona, B. Costa, W.-S. Don, An improved WENO scheme for hyperbolic conservation laws, J. Comput. Phys. 227 (2008) 3191–3211.

[4] M.D. Buhman, Radial Basis Functions — Theory & Implementations, Cambridge University Press, Cambridge, 2003.

[5] S. Christofi, The study of building blocks for essentially non-oscillatory (ENO) schemes, Ph.D. thesis, Division of Applied Mathematics, Brown University, 1996.

[6] B. Costa, W.-S. Don, Multi-domain hybrid spectral-WENO methods for hyperbolic conservation laws, J. Comput. Phys. 224 (2007) 970–991.

[7] B. Costa, W.-S. Don, D. Gottlieb, R. Sendersky, Two-dimensional multi-domain hybrid spectral-WENO methods for the conservation laws, Commun. Comput. Phys. 1 (2006) 550–577.

[8] P. Fan, Y.Q. Shen, B.L. Tian, C. Yang, A new smoothness indicator for improving the weighted essentially nonoscillatory scheme, J. Comput. Phys. 269 (2014) 329–354.

[9] J. Guo, J.-H. Jung, A numerical study of the local monotone polynomial edge detection for the hybrid WENO method, 2016, submitted for publication.

[10] J. Guo, J.-H. Jung, Radial basis function ENO and WENO finite difference methods based on the optimization of shape parameters, J. Sci. Comput. (2016), http://dx.doi.org/10.1007/s10915-016-0257-y, in press.

[11] Y. Ha, C.H. Kim, Y.J. Lee, J. Yoon, An improved weighted essentially non-oscillatory scheme with a new smoothness indicator, J. Comput. Phys. 232 (2013) 68–86.

[12] A. Harten, Multiresolution representation of data: a general framework, SIAM J. Numer. Anal. 33 (1996) 1205–1256.

[13] A. Harten, B. Engquist, S. Osher, S.R. Chakravarthy, Uniformly high order accurate essentially non-oscillatory schemes III, J. Comput. Phys. 71 (1987) 231–303.

[14] 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 (2005) 542–567.

[15] G.-S. Jiang, C.-W. Shu, Efficient implementation of weighted ENO schemes, J. Comput. Phys. 126 (1996) 202–228.

[16] E. Larsson, B. Fornberg, Theoretical and computational aspects of multivariate interpolation with increasingly flat radial basis functions, Comput. Math. Appl. 49 (2005) 103–130.

[17] Y. Lee, J. Yoon, C. Micchelli, On convergence of flat multivariate interpolation by translation kernels with finite smoothness, Constr. Approx. 40 (2014) 37–60.

[18] P. Li, Z. Gao, W.-S. Don, S. Xie, Hybrid Fourier-continuation method and weighted essentially non-oscillatory finite difference scheme for hyperbolic conservation laws in a single-domain framework, J. Sci. Comput. 64 (2015) 670–695.

[19] R.B. Platte, T.A. Driscoll, Eigenvalue stability of radial basis function discretizations for time-dependent problems, Comput. Math. Appl. 8 (2006) 1251–1268.

[20] S. Serna, A. Marquina, Power-ENO methods: a fifth-order accurate weighted power ENO method, J. Comput. Phys. 194 (2004) 632–658.

[21] K. Shahbazi, N. Albin, O.P. Bruno, J.S. Hesthaven, Multi-domain Fourier-continuation/WENO hybrid solver for conservation laws, J. Comput. Phys. 230 (2011) 8779–8796.

[22] C.-W. Shu, Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws, in: A. Quarteroni (Ed.), Advanced Numerical Approximation of Nonlinear Hyperbolic Equations, in: Lecture Notes in Mathematics, vol. 1697, Springer-Verlag, Berlin, Heidelberg, 1998, pp. 325–432.

[23] C.-W. Shu, High order weighted essentially nonoscillatory schemes for convection dominated problems, SIAM Rev. 51 (2009) 82–126.

[24] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, J. Comput. Phys. 77 (1988) 439–471.

[25] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes II, J. Comput. Phys. 83 (1989) 32–78.

[26] G.A. Sod, Survey of several finite-difference methods for systems of non-linear hyperbolic conservation laws, J. Comput. Phys. 27 (1978) 1–31.