



**HAL**  
open science

## Parallel implementation of k-exact finite volume reconstruction on unstructured grids

F Haider, P Brenner, B Courbet, Jean-Pierre Croisille

► **To cite this version:**

F Haider, P Brenner, B Courbet, Jean-Pierre Croisille. Parallel implementation of k-exact finite volume reconstruction on unstructured grids. High Order Nonlinear numerical schemes for evolutionary PDEs, Mar 2013, Bordeaux, France. pp.59-75, 10.1007/978-3-319-05455-1\_4 . hal-01283662

**HAL Id: hal-01283662**

**<https://hal.science/hal-01283662>**

Submitted on 5 Mar 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Parallel implementation of $k$ -exact Finite Volume Reconstruction on Unstructured Grids

F. Haider, P. Brenner, B. Courbet and J.-P. Croisille

**Abstract** We present a high-order piecewise polynomial approximation for finite volume schemes. This method is designed for general unstructured polyhedral grids. A new algorithm for the calculation of the approximants is discussed. The main geometric property of the procedure is that the direct neighborhood only is needed. This allows for an efficient parallel implementation.

## 1 Introduction

An important contribution to the design of higher-order versions of the upwind finite-volume method for gas dynamics is the one by Barth and Frederickson [3]. This suggested a high-order finite volume algorithm to approximate Euler's equations on general unstructured grids. The time-marching of their finite volume method for the Euler equations can be decomposed into three steps:

1. Step 1: high-order reconstruction in each cell,
2. Step 2: high-order numerical integration of the numerical flux,
3. Step 3: high-order time stepping.

Selecting efficient interpolants to properly handle Step 1 has been the center of attention in the CFD community for more than 20 years. This quest is motivated

---

F. Haider, B. Courbet  
ONERA, BP72 – 92322 Châtillon-sous-Bagneux, France, e-mail: florian.haider@onera.fr,  
bernard.courbet@onera.fr

P. Brenner  
ASTRIUM Space Transportation - Aerodynamics, BP3002 – 78133 Les Mureaux France, e-mail:  
pierre.brenner@astrium.eads.net

J.-P. Croisille  
Université de Lorraine – Département de Mathématiques, Institut Elie Cartan de Lorraine, UMR  
CNRS 7502, Ile du Saulcy, 57045 Metz, France e-mail: jean-pierre.croisille@univ-lorraine.fr

by the increasing need of industrial applications such as jet noise prediction, LES modeling, internal flow simulation in aerothermochemistry. Efficient simulations of such problems clearly require approximations beyond second order.

As is well known, high-order interpolation schemes for advective problems can result in unstable schemes [1]. This question was extensively studied in the framework of adaptive stencil selection (ENO/WENO schemes [12]). Other approaches include non polynomial functions such as radial basis functions (RBF) or splines [10].

The present contribution is the continuation of previous studies devoted to designing high-order centered polynomial reconstructions [7, 8, 9].<sup>1</sup>

The reconstruction step 1 above clearly belongs to the theory of approximation. It can be stated as follows: consider a smooth function  $u : \Omega \subset \mathbb{R}^d \rightarrow \mathbb{R}$  supposed to be known by its cell averages on an unstructured grid. We seek a piecewise polynomial (possibly discontinuous) approximant  $w$  to the function  $u$ . The local accuracy in each cell is subject to the *k-exact* condition: the approximant procedure  $u \mapsto w$  is said to be *(k + 1)-order accurate* [3, 5] if any locally polynomial function of degree  $k$  is invariant.

An important question when dealing with *k-exact* interpolants ( $k > 1$ ) is the following. On the one hand computational efficiency requires accessing data close to the cell (typically the Von-Neumann neighborhood, see (3)). On the other hand high-order approximations clearly require accessing data (cell averages) belonging to a wide stencil. Note that the question of designing efficient reconstruction algorithms for ENO schemes on triangular grids is addressed in [2].

The main outcome of this paper is an algorithm to efficiently calculate *k-exact* reconstructions from non local data in an efficient way.

The outline of the paper is as follows: in Section 2 we briefly review notation for finite-volume approximations on general grids as well as elements of tensor algebra useful in the sequel. In Section 3 we restate and examine rigorously the approximation method of *k-exact* polynomial reconstruction. Our algorithm, introduced in [7, 8] is expounded in Section 4. Finally we present in Section 5,6 numerical results showing the relevance of the preceding analysis.

Note that we do not address in this paper how to handle non smooth data, which is clearly an essential aspect when dealing with high-order approximation methods in the CFD context. This is the subject of ongoing work.

---

<sup>1</sup> The present study takes place within several industrial projects: the CEDRE and FLUSEPA projects are multiphysics codes respectively developed at ONERA / DSN (Châtillon-sous-Bagneux, FRANCE) and at ASTRIUM Space Transportation (Les Mureaux, FRANCE) for applications in aerothermochemistry and propulsion.

## 2 Notation

### 2.1 Finite-Volume notation

We consider a computational domain  $\Omega \subset \mathbb{R}^d$  and a *finite-volume grid* of  $\Omega$  made of polyhedral (compact) cells  $\mathcal{T}_\alpha$ .<sup>2</sup> The barycenter of  $\mathcal{T}_\alpha$  is denoted  $x_\alpha$  and  $|\mathcal{T}_\alpha|$  is its  $d$ -volume. The *internal face*  $\mathcal{A}_{\alpha\beta} = \mathcal{T}_\alpha \cap \mathcal{T}_\beta$  has a  $(d-1)$ -volume  $|\mathcal{A}_{\alpha\beta}|$  and a barycenter  $x_{\alpha\beta}$ . The surface vector of the face  $\mathcal{A}_{\alpha\beta}$  is  $a_{\alpha\beta}$  with orientation from cell  $\mathcal{T}_\alpha$  to  $\mathcal{T}_\beta$ . The local geometry nearby the cell  $\mathcal{T}_\alpha$  is described using the two vectors  $h_{\alpha\beta}$  and  $k_{\alpha\beta}$  defined by

$$h_{\alpha\beta} \triangleq x_\beta - x_\alpha; \text{ for all cells } \mathcal{T}_\alpha, \mathcal{T}_\beta \quad (1)$$

$$k_{\alpha\beta} \triangleq x_{\alpha\beta} - x_\alpha; \text{ for all adjacent cells } \mathcal{T}_\alpha, \mathcal{T}_\beta. \quad (2)$$

The  $n^{\text{th}}$  *grid neighborhood* of the cell  $\mathcal{T}_\alpha$  is defined as follows. First we define *first grid neighborhood* (or *Von Neumann neighborhood*) by

$$\mathbb{V}_\alpha^{(1)} \triangleq \{ \beta \mid \mathcal{T}_\beta \text{ is adjacent to } \mathcal{T}_\alpha \} \cup \{ \alpha \}, \quad (3)$$

then the  $n^{\text{th}}$  *grid neighborhood* is recursively defined by

$$\mathbb{V}_\alpha^{(n)} \triangleq \bigcup_{\gamma \in \mathbb{V}_\alpha^{(n-1)}} \mathbb{V}_\gamma^{(1)}. \quad (4)$$

On a finite-volume grid  $\cup_{1 \leq \alpha \leq N} \mathcal{T}_\alpha$  of size  $N$  we consider *grid functions*  $u$ . In practice a grid function is given by the values  $u_i$ ,

$$\mathbf{u} \triangleq (u_1, \dots, u_N) \in \mathbb{R}^N. \quad (5)$$

To any grid function  $u$  we further associate the *piecewise polynomial reconstruction* defined by

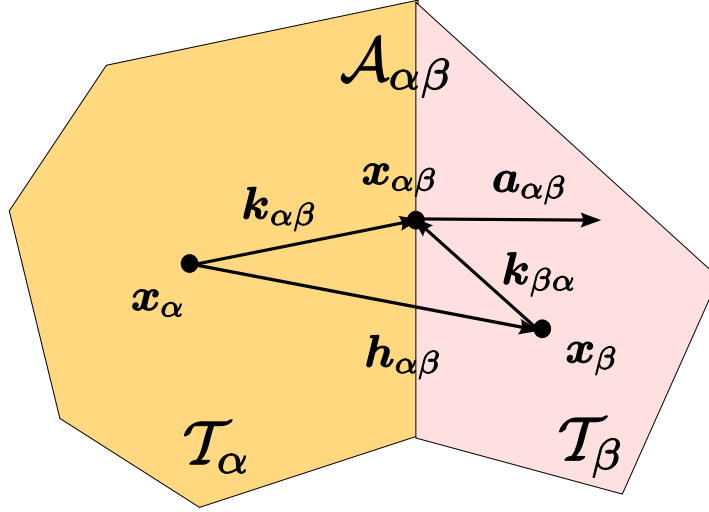
$$x \mapsto w[\mathbf{u}](x) = \sum_{\alpha=1}^N w_\alpha[\mathbf{u}](x) \chi_{\mathcal{T}_\alpha}(x), \quad (6)$$

where  $\chi_{\mathcal{T}_\alpha}$  is the characteristic function of the cell  $\mathcal{T}_\alpha$  and where the  $w_\alpha[\mathbf{u}](x)$  denote polynomial functions of the variable  $x$ . The particular case  $w_\alpha[\mathbf{u}] = u_\alpha$  corresponds to the piecewise constant function associated with  $u$  by the relation

$$x \mapsto w^0[\mathbf{u}](x) = \sum_{\alpha=1}^N u_\alpha \chi_{\mathcal{T}_\alpha}(x). \quad (7)$$

---

<sup>2</sup>  $\mathcal{T}_\alpha$  is not required to be convex, but the barycenter  $x_\alpha$  must be located in the topological interior of  $\mathcal{T}_\alpha$ .



**Fig. 1** Two cells  $\mathcal{T}_\alpha$  and  $\mathcal{T}_\beta$  sharing a common face  $\mathcal{A}_{\alpha\beta}$  in a two-dimensional polyhedral grid. The barycenters of  $\mathcal{T}_\alpha$  and  $\mathcal{T}_\beta$  are  $x_\alpha$  and  $x_\beta$ , respectively. The face  $\mathcal{A}_{\alpha\beta}$  has a barycenter  $x_{\alpha\beta}$  and a surface vector  $a_{\alpha\beta}$ .

In analogy to the finite difference setting the consistency analysis of a reconstruction is based upon analyzing how it operates on a given function  $u(x)$  of the continuous variable  $x$ : we associate with the function  $u(x)$  its matching gridfunction  $\bar{u} = (\bar{u}_1, \dots, \bar{u}_N)$  where  $\bar{u}_\alpha$  is defined by

$$\bar{u}_\alpha \triangleq \frac{1}{|\mathcal{T}_\alpha|} \int_{\mathcal{T}_\alpha} u(x) dx. \quad (8)$$

For each  $\alpha = 1, \dots, N$  the operator  $\mathfrak{R}_\alpha$  is given by

$$\mathfrak{R}_\alpha : \mathbb{R}^N \ni \mathbf{u} \mapsto w_\alpha[\mathbf{u}] \in \mathbb{P}_k(\mathbb{R}^d). \quad (9)$$

Two important properties of the reconstruction (6) are the following:

- The reconstruction is *linear* if each operator  $\mathfrak{R}_\alpha$  in (9) is linear as a function of the argument  $\mathbf{u}$ .
- The reconstruction is *conservative* if the operator  $\mathfrak{R}_\alpha$  satisfies the property

$$\frac{1}{|\mathcal{T}_\alpha|} \int_{\mathcal{T}_\alpha} w_\alpha[\bar{\mathbf{u}}](x) dx = \bar{u}_\alpha \text{ for all } \bar{\mathbf{u}} = (\bar{u}_1, \dots, \bar{u}_N) \in \mathbb{R}^N. \quad (10)$$

In terms of gridfunctions, the condition (10) can be expressed as

$$w_\alpha \overline{[\mathbf{u}]} = \bar{\mathbf{u}}. \quad (11)$$

For physical reasons, condition (10) is crucial when using piecewise polynomial reconstruction in the context of finite volume schemes for conservation laws: obviously the total density, momentum and energy must be preserved through all interpolations procedures.

## 2.2 Tensor notation

In order to handle higher-order derivatives within the multidimensional geometrical setting it appeared convenient to rely on some classical *tensor analysis*. For example to each pair of cells  $\mathcal{T}_\alpha$  and  $\mathcal{T}_\beta$  we will need the momentum  $z_{\alpha\beta}^{(k)}$  defined by

$$z_{\alpha\beta}^{(k)} \triangleq \frac{1}{|\mathcal{T}_\beta|} \int_{\mathcal{T}_\beta} (x - x_\alpha)^{\otimes k} dx. \quad (12)$$

Considering  $z_{\alpha\beta}^{(k)}$  as an element of the space  $S^m(\mathbb{R}^d)$  of symmetric tensors of rank  $m$  greatly facilitates calculations of such momenta.

Recall that a symmetric tensor  $a \in S^m(\mathbb{R}^d)$  has components  $a_{i_1 \dots i_m}$  where  $a_{i_1 \dots i_m} = a_{i_{\pi(1)} \dots i_{\pi(m)}}$  for all permutations  $\pi$ . Thus each tensor  $a \in S^m(\mathbb{R}^d)$  is specified by  $\binom{m+d-1}{m}$  independent components. For  $a, b \in S^m(\mathbb{R}^d)$  the contracted product is defined by  $a \bullet b \triangleq \sum_{i_1=1}^d \dots \sum_{i_m=1}^d a_{i_1 \dots i_m} b_{i_1 \dots i_m}$ .

The components of the tensor  $z_{\alpha\beta}^{(k)}$  in (12) are

$$z_{\alpha\beta; i_1, \dots, i_k}^{(k)} \triangleq \frac{1}{|\mathcal{T}_\beta|} \int_{\mathcal{T}_\beta} (x_{i_1} - x_{\alpha; i_1}) \dots (x_{i_k} - x_{\alpha; i_k}) dx. \quad (13)$$

When  $\beta = \alpha$ , (12) reduces to the  $k^{\text{th}}$  momentum of the cell  $\mathcal{T}_\alpha$  namely

$$x_\alpha^{(k)} \triangleq z_{\alpha\alpha}^{(k)} = \frac{1}{|\mathcal{T}_\alpha|} \int_{\mathcal{T}_\alpha} (x - x_\alpha)^{\otimes k} dx. \quad (14)$$

In index notation (14) is expressed as

$$x_{\alpha; i_1, \dots, i_k}^{(k)} \triangleq z_{\alpha\alpha; i_1, \dots, i_k}^{(k)} = \frac{1}{|\mathcal{T}_\alpha|} \int_{\mathcal{T}_\alpha} (x_{i_1} - x_{\alpha; i_1}) \dots (x_{i_k} - x_{\alpha; i_k}) dx. \quad (15)$$

Note that when  $k = 0, 1$ , we have

$$z_{\alpha\beta}^{(0)} = 1, \quad z_{\alpha\beta}^{(1)} = h_{\alpha\beta} \quad (16)$$

and

$$x_\alpha^{(0)} = 1, \quad x_\alpha^{(1)} = h_{\alpha\alpha} = x_\alpha - x_\alpha = 0. \quad (17)$$

### 3 Algebra of $k$ -exact Reconstruction for Derivatives

This section is devoted to the general setting of  $k$ -exact reconstructions [3] for  $k \geq 1$ . A function  $u$  is called  $k$ -exact on the grid neighborhood  $\mathbb{W}_\alpha$  if the function  $u$  restricted to  $\mathbb{W}_\alpha$  is polynomial. Otherwise stated, there exists a polynomial function  $p \in \mathbb{P}_k(\mathbb{R}^d)$  such that the restriction of  $u$  and  $p$  to  $\mathcal{T}(\mathbb{W}_\alpha)$  coincide

$$u|_{\mathcal{T}(\mathbb{W}_\alpha)} = p|_{\mathcal{T}(\mathbb{W}_\alpha)}. \quad (18)$$

So the  $k$ -exact property (18) can be expressed as follows: the operator  $\mathfrak{R}_\alpha : \mathbb{R}^N \rightarrow \mathbb{P}_k(\mathbb{R}^d)$  is  $k$ -exact on the grid neighborhood  $\mathbb{W}_\alpha$  if the local approximant  $w_\alpha[\bar{u}]$  reproduces all  $k$ -exact functions  $u$  on  $\mathbb{W}_\alpha$ ,

$$w_\alpha[\bar{u}]|_{\mathcal{T}(\mathbb{W}_\alpha)} = u|_{\mathcal{T}(\mathbb{W}_\alpha)}. \quad (19)$$

Expressing the polynomial function  $p \in \mathbb{P}_k(\mathbb{R}^d)$  in terms of its successive derivatives at  $x_\alpha$  yields

$$p(x) = p(x_\alpha) + \sum_{j=1}^k \frac{1}{j!} \sum_{i_1 \dots i_j=1}^d \frac{\partial p}{\partial x_{i_1} \dots \partial x_{i_j}} \Big|_{x=x_\alpha} (x_{i_1} - x_{\alpha;i_1}) \dots (x_{i_j} - x_{\alpha;i_j}). \quad (20)$$

It is therefore the case that any  $k$ -exact reconstruction of the polynomial  $p \in \mathbb{P}_k(\mathbb{R}^d)$  is obtained by reconstructing the  $m^{\text{th}}$  derivatives of the polynomial  $p$  for  $0 \leq m \leq k$  from its cell averages. Observe that the  $m^{\text{th}}$  derivative  $D^{(m)}v(x)$  of a given smooth function  $v$  can be considered as a symmetric tensor, element of the set  $S^m(\mathbb{R}^d)$ . Our algebraic setting for  $k$ -exact reconstruction can be further expressed as follows:

**Definition 3.1 ( $k$ -exact reconstruction operator)** *Let  $\mathbb{W}_\alpha$  be a grid neighborhood of cell  $\mathcal{T}_\alpha$  and let  $k \in \mathbb{N}$  with  $m \leq k$ . A  $k$ -exact  $m^{\text{th}}$  derivative on  $\mathbb{W}_\alpha$  is a linear map  $w_\alpha^{(m|k)} : \mathbb{R}^N \rightarrow S^m(\mathbb{R}^d)$  given by coefficients  $w_{\alpha\beta;i_1 \dots i_m}^{(m|k)}$  is such that for all functions  $v$  that are  $k$ -exact on  $\mathbb{W}_\alpha$*

$$\sum_{\beta \in \mathbb{W}_\alpha} w_{\alpha\beta;i_1 \dots i_m}^{(m|k)} \bar{v}_\beta = \frac{\partial v}{\partial x_{i_1} \dots \partial x_{i_m}} \Big|_{x=x_\alpha}. \quad (21)$$

Furthermore the reconstruction stencil of  $w_\alpha^{(m|k)}$  at cell  $\mathcal{T}_\alpha$  is defined as the set of indices  $\beta \in \{1, \dots, N\}$  such that  $w_{\alpha\beta;i_1 \dots i_m}^{(m|k)} \neq 0$ .

A  $k$ -exact polynomial reconstruction at cell  $\mathcal{T}_\alpha$  is thus equivalent to a set of  $k$ -exact  $m^{\text{th}}$  derivatives  $w_\alpha^{(m|k)}$  at cell  $\mathcal{T}_\alpha$  for  $0 \leq m \leq k$ . The constant terms  $w_{\alpha\beta}^{(0|k)}$  are uniquely determined by the equation (10). Therefore, the reconstructed polynomial in cell  $\mathcal{T}_\alpha$  has the general form

$$w_\alpha [\bar{u}] (x) = \bar{u}_\alpha + \sum_{m=1}^k \frac{1}{m!} \sum_{i_1 \dots i_m} \sum_{\beta \in \mathbb{W}_\alpha} w_{\alpha\beta}^{(m|k)} \bar{u}_\beta \left[ (x_{i_1} - x_{\alpha;i_1}) \cdots (x_{i_m} - x_{\alpha;i_m}) - x_{\alpha;i_1 \dots i_m}^{(m)} \right]. \quad (22)$$

Using (12) the cell averages  $\bar{p}_\gamma$  on a cell  $\mathcal{T}_\gamma$  of the function  $p(x)$  in (20) can be expressed as

$$\bar{p}_\gamma = p(x_\alpha) + \sum_{j=1}^k \frac{1}{j!} D^{(j)} p \Big|_{x_\alpha} \bullet z_{\alpha\gamma}^{(j)} \quad (23)$$

or equivalently

$$\bar{p}_\gamma = p(x_\alpha) + \sum_{j=1}^k \frac{1}{j!} \sum_{i_1 \dots i_j=1}^d z_{\alpha\beta}^{(j)} \frac{\partial p}{\partial x_{i_1} \dots \partial x_{i_j}} \Big|_{x=x_\alpha}. \quad (24)$$

Using (24), the reconstruction error of a  $k$ -exact  $m^{\text{th}}$  derivative  $w_\alpha^{(m|k)}$  on a grid neighborhood  $\mathbb{W}_\alpha$  at cell  $\mathcal{T}_\alpha$  applied to a  $j$ -exact function  $u(x)$  (with  $j > k$ ) can be expressed as

$$\sum_{\beta \in \mathbb{W}_\alpha} w_{\alpha\beta}^{(m|k)} \bar{u}_\beta - D^{(m)} u \Big|_{x=x_\alpha} = \sum_{l=k+1}^j \frac{1}{l!} \sum_{\beta \in \mathbb{W}_\alpha} w_{\alpha\beta}^{(m|k)} \left( z_{\alpha\beta}^{(l)} \bullet D^{(l)} u \Big|_{x=x_\alpha} \right). \quad (25)$$

Equation (25) can be expressed using index notation as

$$\begin{aligned} & \sum_{\beta \in \mathbb{W}_\alpha} w_{\alpha\beta}^{(m|k)} \bar{u}_\beta - \frac{\partial u}{\partial x_{i_1} \dots \partial x_{i_m}} \Big|_{x=x_\alpha} \\ &= \sum_{l=k+1}^j \frac{1}{l!} \sum_{\beta \in \mathbb{W}_\alpha} w_{\alpha\beta}^{(m|k)} \sum_{i_1 \dots i_l=1}^d z_{\alpha\beta}^{(l)} \frac{\partial u}{\partial x_{i_1} \dots \partial x_{i_l}} \Big|_{x=x_\alpha}. \end{aligned} \quad (26)$$

We refer to [7] for an in-depth mathematical study of the derivative approximation (26).

## 4 Implementing $k$ -exact reconstructions

As explained in section 3 a  $k$ -exact  $m^{\text{th}}$  derivative in cell  $\mathcal{T}_\alpha$  can be calculated using the tensor form (see (21)). This is given by the sum

$$\sum_{\beta \in \mathbb{W}_\alpha} w_{\alpha\beta}^{(m|k)} \bar{u}_\beta. \quad (27)$$



Implementing (27) directly requires the list of cells  $\mathcal{T}_\alpha$  in the reconstruction stencil  $\mathbb{W}_\alpha$  to be stored. The implementation of such an algorithm raises the following problems:

1. A parallel calculation takes advantage of the fact that an unstructured grid can easily be partitioned into subdomains each of which is handled by a different processor. Near boundary cells may have a reconstruction stencil that overlaps the opposite domain. For those cells, all data in the reconstruction stencil must be transferred from one processor to the other. The size of the overlap of reconstruction stencils varies from cell to cell, giving data packets of different size. This is a serious impairment to the efficiency of the parallel code (capability to run  $n$  times faster with  $n$  times more processors).
2. A list of neighborhoods must be created, sorted and accessed at run time. On large grids, this gives performance degradation and memory access problems.

These issues strongly motivate the search for  $k$ -exact reconstruction algorithms based upon the Von Neumann stencil (3). However,  $k$ -exact reconstruction requires a stencil size greater than  $\binom{k+d}{k}$  cells. In dimension  $d = 3$  and for second-order derivatives ( $k = 2$ ) and third-order derivatives ( $k = 3$ ) reconstructions, this number is in general larger than the Von Neumann stencil size. The next section shows how to overcome this problem.

#### 4.1 An Efficient Reconstruction Algorithm

Let  $\mathbb{W}_\alpha$  be a small grid neighborhood of cell  $\mathcal{T}_\alpha$ , typically the Von Neumann neighborhood (3). The cells in  $\mathbb{W}_\alpha \setminus \{\alpha\}$  are labelled with  $\beta \in \{\beta_1, \dots, \beta_m\}$ , where  $m \triangleq |\mathbb{W}_\alpha \setminus \{\alpha\}|$ .

The suggested algorithm is to *recursively calculate*  $k$ -exact  $k^{\text{th}}$  derivatives  $w_\beta^{(k|k)}$ . This idea is well-known in the finite-difference context. Recall for example that the standard three-point second-order derivative  $\delta_x^2$  defined by  $\delta_x^2 u_j = (u_{j+1} + u_{j-1} - 2u_j)/h^2$  is the same as  $\delta_x^2 = \delta_x \circ \delta_x$  where the notation  $\delta_x$  stands for the staggered difference operator  $\delta_x u_j = (u_{j+1/2} - u_{j-1/2})/h$ . The operator  $\delta_x^2$  is thus recursively obtained from the knowledge of  $\delta_x$ .

Coming back to our problem, suppose that  $k$ -exact  $k^{\text{th}}$  derivative operators  $w_\beta^{(k|k)}$  at cells  $\mathcal{T}_\beta$  for all  $\beta$  are known.

Let us introduce the operator  $\mathfrak{J}_{\mathbb{W}_\alpha}^{(k+1)}$ ,

$$\mathfrak{J}_{\mathbb{W}_\alpha}^{(k+1)} : \mathbf{S}^{k+1}(\mathbb{R}^d) \longrightarrow \left( \mathbf{S}^k(\mathbb{R}^d) \right)^m. \quad (28)$$

The image of the symmetric tensor  $b \in \mathbf{S}^{k+1}$  is

$$\left( \mathfrak{J}_{\mathbb{W}_\alpha}^{(k+1)}(b) \right)_\beta \in \left( \mathbf{S}^k(\mathbb{R}^d) \right)^m. \quad (29)$$

The  $\beta^{\text{th}}$  component of (29) is defined by

$$\begin{aligned} \left( \mathfrak{J}_{\mathbb{W}_\alpha}^{(k+1)}(b) \right)_\beta &\triangleq h_{\alpha\beta} \cdot b \\ &+ \frac{1}{(k+1)!} \sum_{\gamma \in \mathbb{W}_\beta^{(k)}} w_{\beta\gamma}^{(k|k)} \left( z_{\beta\gamma}^{(k+1)} \bullet b \right) - \frac{1}{(k+1)!} \sum_{\gamma \in \mathbb{W}_\alpha^{(k)}} w_{\alpha\gamma}^{(k|k)} \left( z_{\alpha\gamma}^{(k+1)} \bullet b \right). \end{aligned} \quad (30)$$

Using index notation (30) can be expressed as

$$\begin{aligned} \mathfrak{J}_{\mathbb{W}_\alpha}^{(k+1)}(b)_{\beta; i_1 \dots i_k} &\triangleq \sum_{j=1}^d h_{\alpha\beta; j} b_{i_1 \dots i_k j} \\ &+ \frac{1}{(k+1)!} \sum_{\gamma \in \mathbb{W}_\beta^{(k)}} w_{\beta\gamma}^{(k|k)} \sum_{i_1 \dots i_{k+1}=1}^d z_{\beta\gamma; i_1 \dots i_{k+1}}^{(k+1)} b_{i_1 \dots i_{k+1}} \\ &- \frac{1}{(k+1)!} \sum_{\gamma \in \mathbb{W}_\alpha^{(k)}} w_{\alpha\gamma}^{(k|k)} \sum_{i_1 \dots i_{k+1}=1}^d z_{\alpha\gamma; i_1 \dots i_{k+1}}^{(k+1)} b_{i_1 \dots i_{k+1}}. \end{aligned} \quad (31)$$

Suppose now that  $\mathbb{W}_\alpha$  is a grid neighborhood of cell  $\mathcal{T}_\alpha$ . Let  $w_\beta^{(k|k)}$  for  $\beta \in \mathbb{W}_\alpha$  be a family of  $k$ -exact  $k^{\text{th}}$  derivatives at cell  $\mathcal{T}_\beta$  with respective reconstruction stencils  $\mathbb{W}_\beta^{(k)}$  and let the function  $u$  be  $(k+1)$ -exact on  $\mathbb{U}_\alpha \triangleq \bigcup_{\beta \in \mathbb{W}_\alpha} \mathbb{W}_\beta^{(k)}$ . Then we have the following identity

$$\mathfrak{J}_{\mathbb{W}_\alpha}^{(k+1)} \left( D^{(k+1)} u \Big|_{x_\alpha} \right) = \left( w_{\beta_1}^{(k|k)} [\bar{u}] - w_\alpha^{(k|k)} [\bar{u}], \dots, w_{\beta_m}^{(k|k)} [\bar{u}] - w_\alpha^{(k|k)} [\bar{u}] \right). \quad (32)$$

The *main claim* is the following:

*Claim.* Assume that the operator  $\mathfrak{J}_{\mathbb{W}_\alpha}^{(k+1)}$  defined by (30) has a left inverse  $\mathfrak{D}_{\mathbb{W}_\alpha}^{(k+1)}$ . Then in that case the linear operator  $w_\alpha^{(k+1|k+1)} : \mathbb{R}^N \rightarrow \mathbb{S}^{k+1}(\mathbb{R}^d)$  defined by

$$w_\alpha^{(k+1|k+1)} [\bar{u}] \triangleq \mathfrak{D}_{\mathbb{W}_\alpha}^{(k+1)} \left( w_{\beta_1}^{(k|k)} [\bar{u}] - w_\alpha^{(k|k)} [\bar{u}], \dots, w_{\beta_m}^{(k|k)} [\bar{u}] - w_\alpha^{(k|k)} [\bar{u}] \right) \quad (33)$$

is a  $(k+1)$ -exact  $(k+1)^{\text{th}}$  derivative on the grid neighborhood  $\bigcup_{\beta \in \mathbb{W}_\alpha} \mathbb{W}_\beta^{(k)}$ .

The claim holds for the following reason: let the function  $u$  be  $(k+1)$ -exact on  $\bigcup_{\beta \in \mathbb{W}_\alpha} \mathbb{W}_\beta^{(k)}$ . Therefore  $u$  and the family of  $w_\beta^{(k|k)} [\bar{u}]$  satisfy the identity (32). Application of the operator  $\mathfrak{D}_{\mathbb{W}_\alpha}^{(k+1)}$  to both sides of (32) gives the desired result

$$w_\alpha^{(k+1|k+1)} [\bar{u}] = D^{(k+1)} u \Big|_{x_\alpha}. \quad (34)$$

The construction of the  $(k+1)$ -exact  $(k+1)^{\text{th}}$  derivative (33) thus depends on the existence of a left inverse of (28). Unfortunately, an abstract proof of existence of a left inverse of (28) is a *difficult question on arbitrary unstructured grids*. Nevertheless, we show in the next section how to take advantage of (33) to define a  $k$ -exact  $m^{\text{th}}$  derivatives  $w_\alpha^{(m|k)}[\bar{\mathbf{u}}]$  for  $1 \leq m \leq k-1$ .

## 4.2 Practical implementation

In this section we specify a  $k$ -exact reconstruction algorithm based on the  $k$ -exact  $k^{\text{th}}$  derivative defined by the recurrence relation (33).

Since the operator  $\mathfrak{J}_{\mathbb{V}_\alpha}^{(k)}$  (28) is in general not surjective, we need to properly define a specific inverse  $\mathfrak{D}_{\mathbb{V}_\alpha}^{(k)}$ . Selecting this inverse is based upon numerical efficiency. In Theorem 3.14 in [9] it is proven that the Moore Penrose inverse  $J^\dagger$  of a matrix  $J$  is the left inverse of  $J$  that minimizes all *unitarily invariant* matrix norms. These norms have the property that  $\|UJV\| = \|J\|$  for all unitary matrices  $U, V$ . This includes the Frobenius norm, the spectral norm, the trace norm, the Ky-Fan norms, etc. We found that selecting the Moore Penrose inverse is a suitable choice for minimizing numerical errors. In practice the Moore-Penrose inverse is evaluated using the SVD decomposition [6]. This brings us to the following

- Algorithm 1 (Calculating  $k$ - exact  $k$ -th derivative)** 1. Calculate a 1-exact  $1^{\text{st}}$  derivative  $w_\beta^{(1|1)}[\bar{\mathbf{u}}]$  from the cell averages on the Von Neumann neighborhood  $\mathbb{V}_\alpha^{(1)}$  (3) of each cell  $\mathcal{T}_\alpha$  using least squares. [9].
2. Iterate the following step from  $m = 1$  to  $m = k - 1$ : using (33), compute in each cell  $\mathcal{T}_\alpha$  a  $(m+1)$ -exact  $(m+1)^{\text{th}}$  derivative  $w_\alpha^{(m+1|m+1)}[\bar{\mathbf{u}}]$  from the  $m$ -exact  $m^{\text{th}}$  derivatives  $w_\beta^{(m|m)}[\bar{\mathbf{u}}]$  for  $\beta \in \mathbb{V}_\alpha^{(1)}$ .
3. Use recursively error formula (25) to calculate the  $k$ -exact  $m^{\text{th}}$  derivatives  $w_\alpha^{(m|k)}[\bar{\mathbf{u}}]$  for  $1 \leq m \leq k-1$ .

Note that Algorithm 1 closely emulates the differentiation procedure of a smooth function where the  $(m+1)^{\text{th}}$  pointwise derivative is obtained from the  $m^{\text{th}}$  derivative on a neighborhood of that point.

In the three-dimensional case, the Von Neumann neighborhood on tetrahedral grids consists of five cells. Numerical evidence [9] shows that 1-exact reconstruction on this stencil leads to ill-conditioned problems. It was observed that overcoming this problem can be obtained by increasing the width of the reconstruction stencil. In practice steps 1 and 2 in Algorithm 1 are modified as follows: after calculating the  $m$ -exact  $m^{\text{th}}$  derivative  $w_\alpha^{(m|m)}[\bar{\mathbf{u}}]$  in each cell  $\mathcal{T}_\alpha$ , replace  $w_\alpha^{(m|m)}[\bar{\mathbf{u}}]$  with the smoothed value

$$\tilde{w}_\alpha^{(m|m)}[\bar{\mathbf{u}}] = \frac{\sum_{\beta \in \mathbb{V}_\alpha^{(1)}} |\mathcal{T}_\beta| w_\beta^{(m|m)}[\bar{\mathbf{u}}]}{\sum_{\gamma \in \mathbb{V}_\alpha^{(1)}} |\mathcal{T}_\gamma|}. \quad (35)$$

It can easily be shown that (35) again is a  $m$ -exact  $m^{\text{th}}$  derivative at cell  $\mathcal{T}_\alpha$  on the grid neighborhood  $\widetilde{\mathbb{W}}_\beta^{(m)} \triangleq \bigcup_{\beta \in \mathbb{V}_\alpha^{(1)}} \mathbb{W}_\beta^{(m)}$  where  $\mathbb{W}_\beta^{(m)}$  is the reconstruction stencil of  $w_\beta^{(m)}[\bar{u}]$ .

## 5 Numerical convergence of $k$ -exact reconstructions

In this paper we present numerical results only for the grid convergence of the approximation error  $\|w[\bar{u}] - u\|_{\mathcal{L}_\infty(\Omega)}$ .

The test case considered consists of calculating the derivatives of the two functions defined on the unit cube by

$$u(x, y, z) = \sin(2\pi x) \sin(2\pi y) \sin(2\pi z) \quad (36)$$

$$u(x, y, z) = 30 \exp(-40(x^2 + y^2 + z^2)). \quad (37)$$

Periodic boundary conditions are applied in both cases. We use three kinds of grid with different shapes: a tetrahedral grid, a Cartesian grid and a grid made of general polyhedra. In the latter case, cells can have up to 20 faces.

Let  $w[\bar{u}]$  denote the approximant (6) to the function  $u$ . We assume that cell averages are given with at least  $k$ -order accuracy. In practice this is obtained by using cubature formulas exact for polynomials of degree  $k = 3$ . The approximation error  $\|w[\bar{u}] - u\|_{\mathcal{L}_\infty(\Omega)}$  is numerically evaluated as follows. Denote by  $V_{\alpha\beta}$  the set of vertices spanning the face  $\mathcal{A}_{\alpha\beta}$  and define for each cell  $\mathcal{T}_\alpha$  the set of vertices

$$X_\alpha \triangleq \{x_\alpha\} \cup \bigcup_{\beta \in \mathbb{V}_\alpha^{(1)}} \{x_{\alpha\beta}\} \cup \bigcup_{\beta \in \mathbb{V}_\alpha^{(1)}} V_{\alpha\beta}. \quad (38)$$

The estimate of the approximation error is thus defined as

$$\|w[\bar{u}] - u\|_{\mathcal{L}_\infty(\Omega)} \approx \sup_{1 \leq \alpha \leq N} \sup_{x \in X_\alpha} |w_\alpha[\bar{u]}(x) - u(x)|. \quad (39)$$

Tables 1, 2 and 3 display the convergence rate of the approximation error (39) as a function of the maximum cell diameter  $h_{\max}$  on the three different grid shapes for the function (36). Tables 4, 5 and 6 display the same error for the function (37). Each row except the last one represents a grid with the column  $N$  displaying the number of cells and the column  $h_{\max}$  the maximum cell diameter. The remaining columns show for each approximation method the convergence rate of the error when the grid is refined from the grid of the row immediately above to the grid of the current row. The last row of each table shows the convergence rate when the grid is refined from the coarsest to the finest grid.

The tests show results for piecewise polynomial approximation of degree  $k = 2$  and  $k = 3$  calculated using the *Coupled Least Squares* Algorithm 1. We use the nomenclature CLS  $Dk(n)$ . The values  $k = 2, 3$  denote the reconstruction order and  $n$

indicates the effective reconstruction stencil as defined by (4). Note that on tetrahedral grids, Algorithm 1 is modified along the smoothing formula (35). Direct least squares reconstruction results are displayed for comparison. The results are called DLS  $Dk(n)$  for  $k = 1, 2, 3$  and  $n$  specifying the size of the reconstruction stencil.

**Table 1** Convergence on tetrahedral grids for the function (36)

$h_{\max}$	$N$	DLS D1(2)	DLS D2(3)	CLS D2(4)	DLS D3(4)	CLS D3(6)
0.23632	4271					
0.18863	12817	1.2493	4.3705	3.6689	3.6081	4.2008
0.15720	22493	2.1638	2.3196	0.90776	3.5680	4.3848
0.12751	39518	2.0075	2.5284	4.0729	5.3495	4.3547
0.099903	77770	1.0628	2.3637	1.7146	2.0623	3.0495
0.079988	192972	3.1449	2.0579	2.5916	5.9683	5.1498
		1.8966	2.7426	2.6210	4.0738	4.1966

**Table 2** Convergence on Cartesian grids for the function (36)

$h_{\max}$	$N$	DLS D1(1)	CLS D2(2)	DLS D2(2)	CLS D3(3)	DLS D3(3)
0.10825	4096					
0.078730	10648	1.9930	3.0350	2.7834	4.1130	4.0078
0.061859	21952	2.0918	3.0251	3.0007	4.0649	4.0170
0.050943	39304	2.0121	3.0177	2.9003	4.1084	4.0440
0.043301	64000	2.0688	3.0127	3.0146	4.0631	4.0268
0.037653	97336	2.0135	3.0096	2.9368	4.0559	4.0482
0.033309	140608	2.0554	3.0089	3.0174	3.9505	3.9696
0.029863	195112	2.0125	3.0061	2.9559	4.0147	4.0865
		2.0338	3.0202	2.9246	4.0671	4.0249

**Table 3** Convergence on polyhedral grids for the function (36)

$h_{\max}$	$N$	DLS D1(1)	CLS D2(2)	DLS D2(2)	CLS D3(3)	DLS D3(3)
0.084952	12083					
0.080261	22983	4.9195	12.521	10.677	6.9120	10.221
0.053768	52487	1.2017	1.7349	2.1258	2.8336	2.4397
0.042933	80995	2.5688	2.1967	2.0627	4.6916	3.9082
0.040684	135609	2.9787	8.7158	6.7513	7.7399	7.3488
0.027591	238993	1.0183	1.3725	1.9126	2.4008	2.4448
		1.6845	2.5807	2.6924	3.4964	3.3630

**Table 4** Convergence on tetrahedral grids for the function (37)

$h_{\max}$	$N$	DLS D1(2)	DLS D2(3)	CLS D2(4)	DLS D3(4)	CLS D3(6)
0.23632	4271					
0.18863	12817	1.8947	3.9364	2.2439	5.5639	5.2849
0.15720	22493	-0.66561	2.4186	2.5729	2.2471	2.3350
0.12751	39518	3.1658	2.0958	4.0293	2.5890	2.3286
0.099903	77770	0.68881	3.2331	2.2069	3.6160	3.6302
0.079988	192972	2.1892	3.4054	2.3709	4.9739	5.6719
		1.4983	3.0579	2.6617	3.8710	3.9239

**Table 5** Convergence on Cartesian grids for the function (37)

$h_{\max}$	$N$	DLS D1(1)	CLS D2(2)	DLS D2(2)	CLS D3(3)	DLS D3(3)
0.10825	4096					
0.078730	10648	1.9049	3.2080	2.2145	4.1410	3.7255
0.061859	21952	1.9113	2.8185	2.9336	3.8385	4.0430
0.050943	39304	1.9455	3.0864	2.8520	3.9288	4.1242
0.043301	64000	1.9628	3.0955	2.7264	3.9679	4.1314
0.037653	97336	1.9740	2.9818	3.0550	3.9878	4.2836
0.033309	140608	1.9802	3.2079	2.9015	3.9861	4.0752
0.029863	195112	1.9844	2.9120	2.8733	4.0017	4.3373
		1.9410	3.0530	2.7224	3.9874	4.0422

**Table 6** Convergence on polyhedral grids for the function (37)

$h_{\max}$	$N$	DLS D1(1)	CLS D2(2)	DLS D2(2)	CLS D3(3)	DLS D3(3)
0.084952	12083					
0.080261	22983	9.7162	13.505	11.200	19.321	17.962
0.053768	52487	1.5343	2.1122	2.0169	2.8475	2.8623
0.042933	80995	1.3044	1.6592	1.6647	2.2736	2.5098
0.040684	135609	7.5454	10.826	10.398	15.185	15.018
0.027591	238993	0.96482	1.4292	1.3625	1.8798	1.9781
		1.9923	2.7777	2.5850	3.8202	3.8302

The values of the convergence rate depend on the grid type and the approximated function. Theoretical considerations indicate that  $k$ -exact reconstruction should lead to a convergence rate of  $k + 1$ . Observe that this value is effectively obtained for the Cartesian grids. On the tetrahedral and polyhedral grids, the measured convergence rates vary much more from one row to the next than on the Cartesian grids. Note that the convergence rate from the coarsest to the finest grid that is displayed in the last row of each table is more significant because the difference in  $h_{\max}$  is larger. On the unstructured grids, this rate is observed to be in  $[k + \frac{1}{2}, k + 1]$ .

Since the main purpose of Algorithm 1 is to replace the direct approximation method DLS, the absolute values of the convergence rate seem less important to the authors than the following comparative results:

1. The CLS method defined by Algorithm 1 produces convergence rates that are close to those of the direct approximation method DLS for all three shapes of grid.
2. Increasing the degree of the polynomial reconstruction by one leads to a roughly identical increase in the convergence rate of the numerical error.

## 6 Isothermal vortex test-case

In this section we consider the bidimensional Euler equations of hydrodynamics discretized by the cell centered finite volume method. The reconstruction procedure using Algorithm 1 with  $k = 3$  expounded in Section 4 has been implemented combined with classical upwind fluxes such as the Roe numerical flux [11]. The test-case aims to show how the cubic reconstruction procedure compares with the classical piecewise affine reconstruction. It consists of an isothermal Gaussian vortex moving across a periodic box at constant velocity. Since we only focus on the accuracy of the spatial approximation, no slope limitation was used. The calculation was operated in parallel on 16 processors. The specific characteristics of the vortex are as follows:

$$\left\{ \begin{array}{l} \text{gas constant: } R = 8.314 \text{ J mol}^{-1} \text{ K}^{-1}, \\ \text{molar mass: } M_g = 28.013410^3 \text{ kg mol}^{-1}, \\ \sigma = 0.001, \quad a = \frac{1}{2} \left( \frac{0.005}{\sigma} \right) \frac{M_g}{RT_0} \simeq 0.000140393, \\ \text{constant temperature: } T = T_0 = 300 \text{ K}, \\ \text{Gaussian pressure distribution: } p = p_0 \left( 1 - ae^{-\frac{r^2}{2\sigma^2}} \right) \text{ Pa}, \\ p_0 = 10^5 \text{ Pa}, \text{ (gives } p_{min} \simeq 99985.96 \text{ Pa)}, \\ \text{vortex angular velocity: } v_\theta = \sqrt{\frac{2aRT_0/M_g}{1 - ae^{-\frac{r^2}{2\sigma^2}}} e^{-\frac{r^2}{2\sigma^2}}}, \\ \text{advection velocity: } v_x = 50 \text{ ms}^{-1}, \quad v_y = 20 \text{ ms}^{-1}, \\ t_{start} = 0, \quad t_{end} = 0.004, \\ 10 \text{ periods along the } x \text{ direction,} \\ 4 \text{ periods along the } y \text{ direction,} \\ \Delta t = 5.10^{-8} \text{ s, CFL on finest grid } \simeq 0.2. \end{array} \right. \quad (40)$$

This vortex is an analytical solution of the Euler equations. Different grids were tested:

- coarse Cartesian grid ( $80 \times 80$ , span of the vortex: 8 cells), fine Cartesian grid ( $200 \times 200$ , span of the vortex: 20 cells)

- coarse unstructured grid (span of the vortex:  $\simeq 10$  cells), fine unstructured grid (span of the vortex:  $\simeq 15$  cells).

The reconstruction procedure follows the lines of Algorithm 1 with cubic reconstruction ( $k = 3$ ). The RK4 time-stepping scheme is used. We report on Figs. 3–6 the numerical damping between the classical second-order (piecewise affine reconstruction) and the new fourth-order (cubic reconstruction) finite-volume schemes. The results clearly show the drastic influence offered by the cubic reconstruction even with the optional smoothing step (35), see Fig. 6.

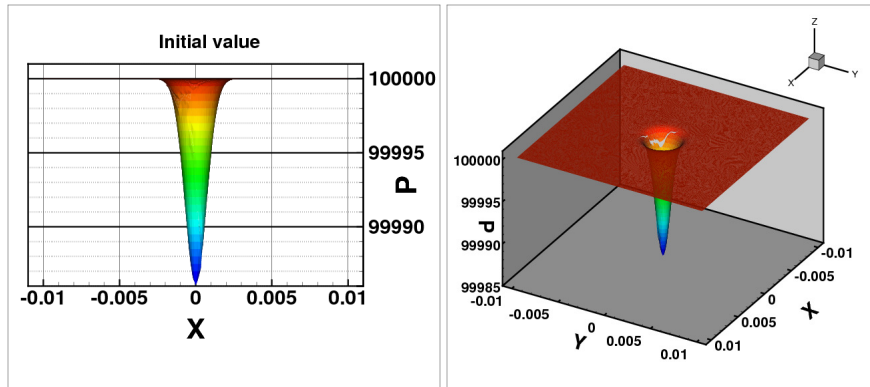


Fig. 2 Convection of the isothermal vortex (40) - Initial condition

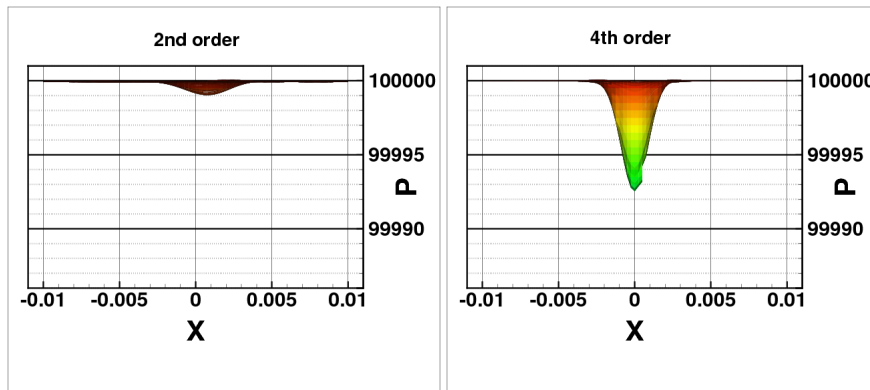
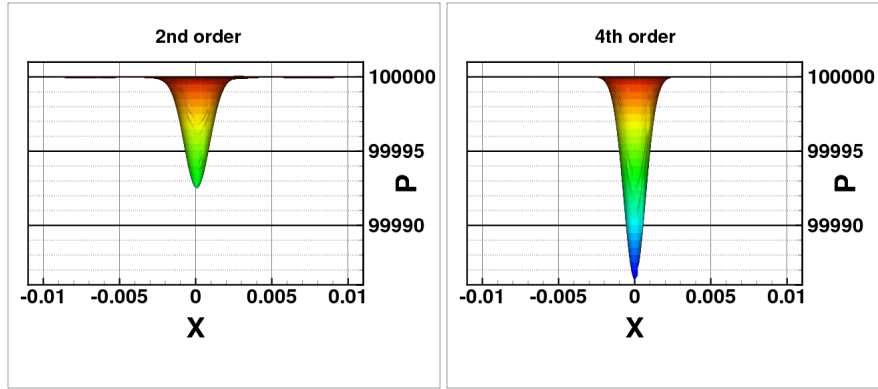
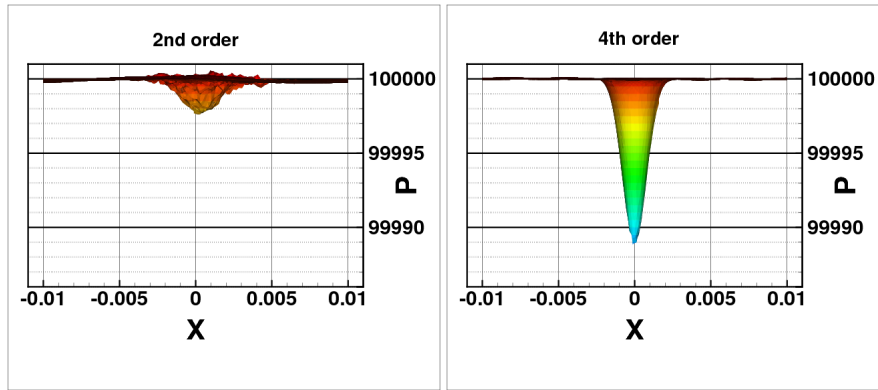


Fig. 3 The isothermal vortex (40) - Cartesian coarse grid - Final time

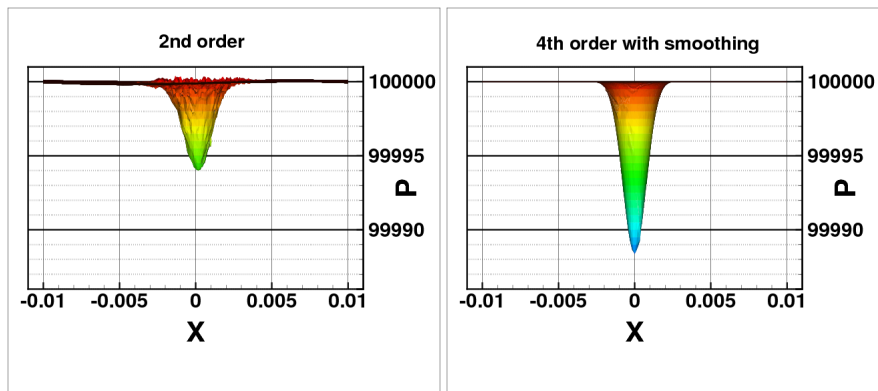




**Fig. 4** Convection of the isothermal vortex (40) - Cartesian fine grid - Final time



**Fig. 5** Convection of the isothermal vortex (40) - Unstructured coarse grid - Final time



**Fig. 6** Convection of the isothermal vortex (40) - Unstructured fine grid - Final time

## 7 Conclusion

This paper presents the mathematical foundation and the numerical implementation of a new algorithm for the calculation of  $k$ -exact polynomial approximants. This algorithm can be used for coding  $(k + 1)$ -order accurate finite volume schemes as introduced in [3]. It is well suited for parallel architectures since the approximation procedure in a grid cell uses only data from adjacent cells. By operating in several steps, the algorithm accesses data from grid cells beyond the adjacent cells in an implicit and natural way. The paper contains a thorough numerical validation of the new algorithm on Cartesian and unstructured grids showing promising results regarding applications of complex fluid flows. More mathematical details of the procedure can be found in the forthcoming [7].

**Acknowledgements** The authors wish to acknowledge P.L. Roe for his help in preparing the final version of this paper.

## References

1. Abgrall R.: On Essentially Non-Oscillatory Schemes on Unstructured Meshes: Analysis and Implementation, *J. Comput. Phys.*, 114, 1994, 45–58.
2. Abgrall R. and Sonar T.: On the use of Mühlbach expansions in the recovery step of ENO methods, *Numerische Mathematik*, 76, 1997, 1–25.
3. Barth, T.J., Frederickson, P.O.: Higher order solution of the Euler equation on unstructured grids using quadratic reconstruction. In: AIAA 90, AIAA-90-0013, pp. 1–12. AIAA, Reno Nevada (1990)
4. Barth, T.J.: Recent developments in high order  $k$ -exact reconstruction on unstructured meshes. In: AIAA 93, AIAA-93-0668, pp. 1–15. AIAA, Reno Nevada (1993)
5. Delaney M. and Essers J.A.: Quadratic reconstruction finite volume scheme for compressible flows on unstructured grids, *AIAA Jou.*, 35(4), 631–639, (1997).
6. Golub G.H. and Van Loan C.F., *Matrix Computations*, John Hopkins University Press, 3-rd. edition, 1996.
7. Haider F., High order approximation on unstructured grids: Theory and implementation, Preprint, submitted.
8. Haider, F., Brenner, P., Courbet, B. and Croisille, J.P.: Efficient implementation of high order reconstruction in finite volume methods. In: F.C. Dell’Acqua, M. Reizakis, R. Herbin, J. Halama, J. Fürst (eds.) Sixth International Symposium on Finite Volumes for Complex Applications, Springer (2011), 553–562.
9. Haider F., Croisille J.-P. and Courbet B., Stability analysis of the cell centered finite-volume MUSCL method on unstructured grids, *Numerische Mathematik*, 113, 2009, 555–600.
10. Iske A. and Sonar T.: On the Structure of Function Spaces in Optimal Recovery of Point Data for ENO-Schemes by Radial Basis Functions, *Numerische Mathematik*, 74, 1996, 177–201.
11. Roe, P.L.: Approximate Riemann Solvers, Parameter Vectors and Difference Schemes, *J. Comput. Phys.*, 43, 1981, 357–372.
12. Shu C.-W., High order ENO and WENO schemes, in *High-Order Methods for Computational Physics*, Lect. Notes in Comp. Sci. Eng., (1999), Barth, T.J. and Deconinck, H. eds, Springer, 439–582.