

Three-Dimensional High-Order Spectral Finite Volume Method for Unstructured Grids

Yen Liu (To receive correspondence) and **Marcel Vinokur**

NASA Ames Research Center, Mail Stop T27B-1

Moffett Field, CA 94035

liu@nas.nasa.gov

Tel: 650-604-6667

Z.J. Wang

Department of Mechanical Engineering, Michigan State University

2555 Engineering Building, East Lansing, MI 48824

zjw@egr.msu.edu

**An extended abstract submitted to 16th AIAA CFD Conference
Orlando, Florida, June 23-26, 2003**

1. INTRODUCTION

Many areas require a very high-order accurate numerical solution of conservation laws for complex shapes. This paper deals with the extension to three dimensions of the Spectral Finite Volume (SV) method for unstructured grids, which was developed to solve such problems [1-2]. We first summarize the limitations of traditional methods such as finite-difference, and finite-volume for both structured and unstructured grids. We then describe the basic formulation of the spectral finite volume method. What distinguishes the SV method from conventional high-order finite-volume methods [3-5] for unstructured triangular or tetrahedral grids is the data reconstruction. Instead of using a large stencil of neighboring cells to perform a high-order reconstruction, the stencil is constructed by partitioning each grid cell, called a spectral volume (SV), into "structured" sub-cells, called control volumes (CVs). One can show that if all the SV cells are partitioned into polygonal or polyhedral CV sub-cells in a geometrically similar manner, the reconstructions for all the SVs become *universal*, irrespective of their shapes, sizes, orientations, or locations. It follows that the reconstruction is reduced to a weighted sum of unknowns involving just a few simple adds and multiplies, and those weights are universal and can be pre-determined once for all. The method is thus very efficient, accurate, and yet geometrically flexible. The most critical part of the SV method is the partitioning of the SV into CVs. In this paper we present the partitioning of a tetrahedral SV into polyhedral CVs with one free parameter for polynomial reconstructions up to degree of precision five. (Note that the order of accuracy of the method is one order higher than the reconstruction degree of precision.) The free parameter will be determined by minimizing the Lebesgue constant of the reconstruction matrix or similar criteria to obtain optimized partitions. The details of an efficient, parallelizable code to solve three-dimensional problems for any order of accuracy are then presented. Important aspects of the data structure are discussed. Comparisons with the Discontinuous

Galerkin (DG) method [7-10] are made. Numerical examples for wave propagation problems are presented.

2. LIMITATIONS OF TRADITIONAL METHODS

2.1 Finite-difference methods

The most widely used method is the finite-difference method applied to a body-fitted curvilinear coordinate system. The limitations for very high order of accuracy implementation are:

- a. The spatial differencing is essentially one-dimensional, carried out along coordinate directions. Thus a large number of data points near the unknown to be updated are ignored. The large stencil has to be modified near boundaries, where one-sided formulas are necessary. For implicit methods, in order to maintain a necessary bandwidth, the order of accuracy must be reduced for points near the boundary.
- b. The metric terms are evaluated by numerically differencing the grid point coordinates. Since numerical grid generators are mostly only second-order accurate, the overall accuracy of the solution can be severely degraded if the grid is not sufficiently smooth. This is particularly true in highly stretched areas, or near corners or boundaries with very high curvature.
- c. The unknowns are values at grid points. While the differencing can be performed in a numerically "conservative" manner, the true integral conservation laws can only be satisfied to second-order accuracy.
- d. A single, structured grid is not feasible for very complex shapes. Calculations must be carried out over a set of patched or overlapping grids. At interface boundaries between patches, or in the overlap regions, the high accuracy is generally degraded.

2.2 Finite-volume methods for structured grids

Finite-volume methods are often employed to overcome limitations b and c above. The unknowns are cell averages over quadrilaterals (2D) or hexahedra (3D). A high order reconstruction in terms of neighboring unknowns is used to obtain values at cell boundaries, which may be modified by appropriate limiters where necessary. These are used to calculate the flux, using (approximate) Riemann solvers. In practice, the method is subject to the same limitations as the finite-difference method.

- a. The reconstruction is still done one-dimensionally along coordinate directions.
- b. The surface area vectors can be exactly calculated in terms of the cell vertices. But the flux integral is approximated by a one-point quadrature at the computational face center, which is equal to the face centroid only to second order. For a non-planar face in 3D, a face centroid does not even exist.

- c. While the cell volume can be precisely calculated, the unknowns are implicitly assigned to a computational cell center, which is equal to the cell centroid only to second order.
- d. If the grid is very unsmooth, and highly curved, even the second order accuracy is reduced to first order.

2.3 Finite-volume methods for unstructured grids

The unknowns are cell averages over triangles (2D) or tetrahedra (3D). A reconstruction of any desired order of accuracy is obtained in terms of unknowns at an appropriate number of neighboring cells in all directions. The flux integral for each face is evaluated using a quadrature approximation of the same order of accuracy as the reconstruction. The flux at each quadrature point is obtained using the reconstructed solution for the two cells sharing that face. In principle, one can in this manner obtain a numerical solution of any desired order of accuracy. In practice, this method has severe computational limitations.

- a. It is difficult to obtain a non-singular stencil. In general, one is faced with an overdetermined problem which requires a least square inversion. For very high order of accuracy, the number of cells, and thus the size of the matrix to be inverted, becomes prohibitively large in three dimensions.
- b. Each cell requires a different reconstruction stencil. If the inversion coefficients are stored, the memory requirements become prohibitive for 3D. On the other hand, repeating the inversion for every cell at each time step would involve impractically large CPU times.
- c. Due to the unstructured nature of the data in physical space, the data from neighboring cells required for the computation can be far apart in memory. This would hamper the efficiency of the code due to data gathering and scattering.

3. THE SPECTRAL FINITE VOLUME METHOD

3.1 Basic formulation

The main motivation behind the spectral finite volume method is to find a simple way to obtain a single non-singular stencil that can be applied to all the cells in an unstructured grid. We start with a relatively coarse unstructured grid of cells, triangles in 2D and tetrahedra in 3D, called spectral volumes (SVs). Each SV is then further subdivided into a number of “structured” sub-cells, called control volumes (CVs), that support a polynomial expansion of a desired degree of precision. The unknowns are now the cell averages over the CVs. The subdivision has a high degree of structure, making use of all the symmetries of the simplex geometry. The CVs can be polygons or polyhedra. For 3D, they can have non-triangular faces, which must be subdivided into triangular facets in order to perform the required integrations. All the SVs are partitioned in a geometrically similar manner. We thus obtain a single, universal reconstruction for all SVs.

Due to the symmetry of the subdivision, only a few distinct coefficients appear in the expansion in terms of the CV unknowns.

A CV face that lies on an SV boundary will have a discontinuity on its two sides. A Riemann solver is then necessary to compute the flux on that face. If the flux is a linear function of the unknowns, the integration can be performed analytically without invoking quadratures [5], and the result expressed as a weighted sum of all the CV unknowns in the two SVs. The weights for each type of CV face are universal numbers that are pre-calculated once and read in as input to the program. If the flux is a non-linear function of the unknowns, a quadrature approximation of the appropriate degree of precision is required [6]. The conservative variable on one side of a quadrature point can again be expressed as a weighted sum of the CV unknowns in the SV on that side. Since the quadrature points belong to just a few symmetry groups, the total number of distinct weights that need to be stored is relatively small. In order to suppress spurious numerical oscillations, it may be necessary to modify the reconstructed solution using appropriate limiters or filters.

The reconstruction within each spectral volume is continuous. Therefore a linear flux over a CV face that lies in the interior of a SV can be evaluated directly, and the weights for each type of face can be stored. For a non-linear flux, a similar procedure can be carried out for each quadrature point. Again a modification involving limiters or filters may be required in certain regions.

3.2 Details of the spectral finite volume method

We present further details of the formulation for a general conservation law. We employ a vector notation for brevity. A conservation law is written as

$$\frac{\partial u}{\partial t} + \nabla \cdot \mathbf{F} = 0, \quad (1)$$

where the conservative variable u can be a scalar or a vector, and the flux \mathbf{F} can be a vector or a tensor. Integrating (1) over each CV, we obtain

$$\frac{d}{dt} \int_{V_{j,i}} u dV + \sum_{k=1}^K \int_{S_{k,j,i}} d\mathbf{S} \cdot \mathbf{F} = 0, \quad (2)$$

where $V_{j,i}$ is the volume of the j^{th} CV in the i^{th} SV, and $S_{k,j,i}$ is the area of planar facet k bounding $V_{j,i}$. (In 2D, each facet is actually a line segment.) The unknowns are the volume averages of u , defined as

$$\bar{u}_{j,i} = \frac{1}{V_{j,i}} \int_{V_{j,i}} u dV. \quad (3)$$

The partitioning of each SV into CVs depends on the choice of basis functions for the reconstruction. For a complete polynomial basis, a reconstruction of degree of precision n requires a subdivision into (at least) N CVs, where

$$N = \begin{cases} (n+1) & 1D \\ (n+1)(n+2)/2 & 2D. \\ (n+1)(n+2)(n+3)/6 & 3D \end{cases} \quad (4)$$

In the present work we partition the SV into N CVs, so that the reconstruction involves the inversion of a square matrix. We also restrict ourselves to partitions involving only one free parameter. The choice of parameter for each degree of precision is determined by minimizing the Lebesgue constant of the reconstruction matrix or similar convergence criteria. We will conduct numerical experiments to determine an optimum value. Partitions valid for reconstruction up to degree of precision five for both 2D and 3D have been obtained. If the expansion of u in terms of the polynomial basis is substituted into (3), and the resulting matrix equation is solved, the result can be written in the form

$$u_i(\mathbf{r}) = \sum_j L_{j,i}(\mathbf{r}) \bar{u}_{j,i}, \quad (5)$$

where $L_{j,i}(\mathbf{r})$ are known as shape functions or cardinal basis functions. Details for obtaining the shape functions $L_{j,i}(\mathbf{r})$ will be given in the final paper.

If \mathbf{F} is a linear function of u , the flux integral in (2) for a given facet in the interior of an SV can be evaluated by substituting expression (5), and the result written as a weighted sum of the CV unknowns. The surface integrations of the shape functions per unit area are universal, which can be calculated and stored in advance. Flux integrals for facets on the SV boundaries require a Riemann solver, and the expansion is now a weighted sum of CV unknowns in both SVs sharing that facet. For non-linear flux functions, the flux integral is evaluated by a consistent quadrature approximation of the form

$$\int_{S_{k,j,i}} d\mathbf{S} \cdot \mathbf{F} = \sum_q w_q \mathbf{n} \cdot \mathbf{F}(u_i(\mathbf{r}_q)) S_{k,j,i}, \quad (6)$$

where the w_q are known quadrature weights. Using (5), we can evaluate $u_i(\mathbf{r}_q)$ as

$$u_i(\mathbf{r}_q) = \sum_j L_{j,i}(\mathbf{r}_q) \bar{u}_{j,i}. \quad (7)$$

The above equation indicates that the value of u at a quadrature point can also be evaluated as a weighted sum of the CV unknowns. These weights are the functional values of the shape functions at the quadrature point, which are also universal, can be calculated and stored in advance. For facets on the SV boundaries, the flux is replaced by a Riemann flux of the form

$$\mathbf{n} \cdot \mathbf{F}(\mathbf{u}(\mathbf{r}_q)) = F_{Riem}(\mathbf{u}_L(\mathbf{r}_q), \mathbf{u}_R(\mathbf{r}_q), \mathbf{n}). \quad (8)$$

3.3 Data Structure

There are several aspects of the data structure which can lead to a very efficient parallelizable code. The global grid data consists of face numberings, vertex numberings and locations, and cell numberings. The topology is specified by listing for each face its vertex numbers, in an order indicating its orientation, and the two adjacent cell numbers. In order to make use of the universal nature of the partitioning, all global cells are mapped into a *single* standard SV. Thus, each global face can have three possible orientations in the standard SV for 2D, and twelve for 3D. All the information connecting the local CV face numbering for each possible orientation is pre-determined and read in as input to the program. Detail of this mapping shall be given in the final paper. We thus can write a single code valid for 2D or 3D, and any desired order of accuracy.

There is an aspect inherent in the spectral finite volume method that permits an optimum use of cache memory, resulting in great computational efficiency on modern supercomputers. Since all unknowns in a single SV cell are packed together, when performing calculations for a given cell, all the data required from the cell is found contiguous in memory. Since data from at most two SVs is involved in any single computation, data communication between the CPU and memory is minimized. All the needed data can be located in cache memory, and may even fit into *L1* cache. This results in great reductions in memory time.

3.4 Comparison with the Discontinuous Galerkin method

The spectral finite volume method bears certain similarities to the Discontinuous Galerkin method [7-10]. We point out some of the advantages of the SV method. The DG method solves the conservation law in a weak form, rather than directly as in the SV method. The unknowns in the DG method are point values, and for time-dependent problems are coupled together, thus requiring an expensive mass matrix inversion to maintain high accuracy. Due to the weak formulation, N test functions are needed, resulting in N coupled equations instead of one. The DG method requires an integration by parts, which results in additional terms to be evaluated. In the DG method, flux calculations are carried out at quadrature points on the SV surfaces. In contrast, they are evaluated over CV faces for the SV method, which include additional faces in the interior of the SV. It appears that the SV method has more faces to evaluate the flux integrals than the DG method. However, in a $(n+1)^{\text{th}}$ order formulation, the former only involves n^{th} order surface integrations, while the latter requires $2n^{\text{th}}$ order surface and $(2n-1)^{\text{th}}$ order volume integrations. Finally, the order of accuracy of the DG method is based on the size of the SV. In contrast, the accuracy in the SV method is based on the size of the much smaller CV.

4. PARTITIONING OF THE SPECTRAL VOLUME

4.1 Details of the partitioning

The most critical part of the SFV method is the partitioning of the SV into CVs. Partitions for the 2D case up to degree 3 have been presented in [2]. Here we present the partitions for the 3D case. In the present work we restrict ourselves to partitions with one free parameter. If N , the number of CVs, is not too large, one can invert a matrix with one parameter analytically, using a symbolic language such as Mathematica. For higher values of N , the inversion is performed numerically. In the final paper we will present results of convergence studies to determine optimum values of the free parameters for various degrees of precision.

For partitions with degrees of precision no greater than three, all the CVs have at least one face on the SV boundary. There are no CVs in the interior of the SV. All the CVs then consist of the vertices of the 2D CVs for each face of the SV connected to the SV centroid with straight edges. In Fig. 1a, we show the partition of an SV into CVs for degree of precision 1. The four CVs are hexahedra, with all faces being planar quadrilaterals. The partition for degree of precision 2 is depicted in Fig. 2a. The CVs are here members of two symmetry groups. One consists of the four hexahedra at the corners of the SV. They are shown in Fig. 2b. Note that the four quadrilateral interior faces are no longer planar. When required for integration purposes, they are subdivided into two triangular facets by means of a straight line connecting the SV corner to the SV centroid. The other group consists of the six mid-edge tetrahedra. They are shown in Fig. 2c. Note that each tetrahedron consists of two exterior triangular faces and the two quadrilateral interior faces it shares with the corner CVs. Further details of the partitionings for degrees of precision up to 5 will be presented in the final paper.

4.2 Shape functions

For a given partition, there exists a unique shape function or cardinal basis function for each CV, as defined by Eq. (5). These functions are necessary to calculate the weights used in determining the surface flux integrals. For degree one reconstruction, due to symmetry, there are only two sets of weights needed to be stored, $\{-23/52, 5/4, 5/52, 5/52\}$ and $\{29/52, 29/52, -3/52, -3/52\}$, corresponding to the boundary CV faces and the interior CV faces, respectively. Since the shape functions are functions of three variables, they cannot be easily depicted. We choose to represent them by showing contour plots on the surface of the SV. These are shown in Fig. 1b for

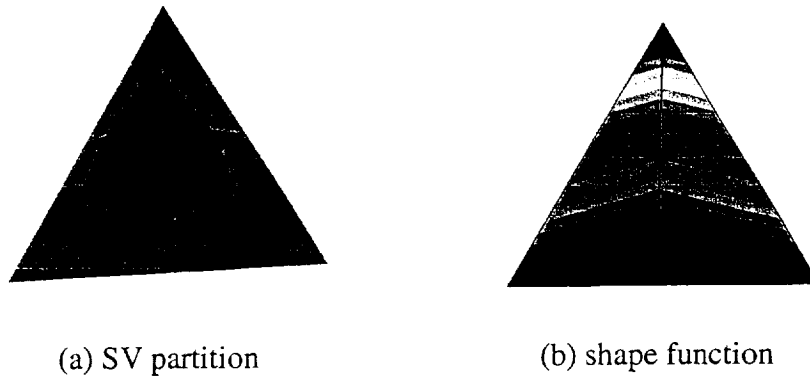


Fig. 1. SV partition and shape function for polynomial reconstruction of degree one

a representative CV for the partition of degree 1. The contour plots for the two types of CVs in the partitions of degree 2, corresponding to Figs. 2b and 2c, are plotted in Figs. 2d and 2e, respectively. Contour plots for higher degrees of precision, as well as the analytic expressions for the shape functions in terms of local coordinates, will be given in the final paper.

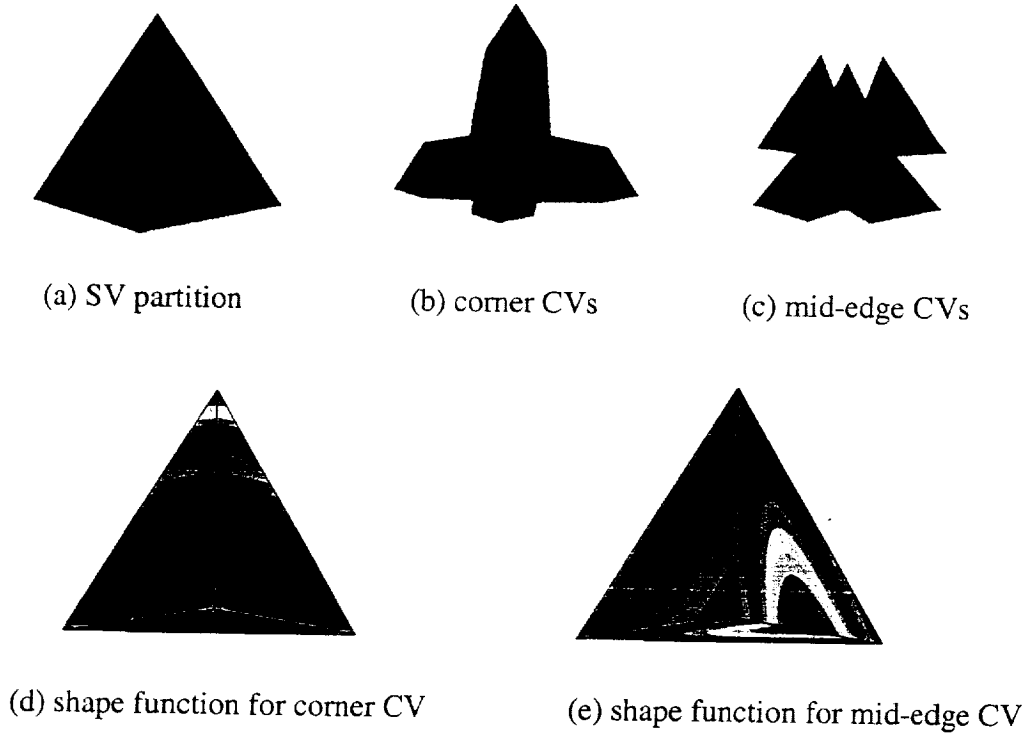


Fig. 2. SV partition and shape functions for polynomial reconstruction of degree two

5. Numerical Results

In order to demonstrate the high accuracy of the method, it was decided in this initial phase to choose problems for which there exist exact solutions. To this end we solve the electromagnetic wave equations. We present solutions involving two and three space variables. Those for two space variables are actually three-dimensional, since they involve electromagnetic field components in all three directions. We first carry out a convergence study by calculating the propagation of a wave through a square region at 45 degrees. A non-reflecting boundary condition is applied at the four boundaries of the square. The coarsest grid consists of 200 triangles, and each successive refinement multiplies the number of triangles by four. Fig. 3a shows contour plots of E_z for the coarsest grid, and Fig. 3b shows the solution after two grid refinements. A partition of degree of precision of 1, leading to second order accuracy, has been used. The solutions shown are after the wave has propagated through two periods in time. Note the excellent solutions at the open boundaries demonstrating the effectiveness of the non-

reflecting boundary condition. In the final paper we will verify the order of accuracy by calculating the errors for five successive grid refinements. Similar calculations will be presented for higher orders of accuracy.

In Fig.4, we present a three-dimensional solution for a plane wave propagating through a rectangular parallelepiped, which has been discretized by a tetrahedral grid. Only the second order result for the coarsest grid is shown. Results for various orders of accuracy as well as grid refinements will be given in the final paper.

We next present results for a plane wave incident on a perfectly conducting circular cylinder. The grid used is shown in Fig. 5. The averaged grid size is about 1/12 of a wavelength. Figures. 6 show contours of E_z for a TM wave. The exact solution is presented in Fig.6a. The numerical solution, for second order (linear reconstruction) is shown in Fig. 6b, and for fourth order (cubic reconstruction) in Fig. 6c. Fig. 7 shows analogous results for H_z for a TE wave. In the final paper we will present results up to sixth order, as well as quantitative comparisons with the exact solution. We will also show results for a plane wave incident on a sphere.

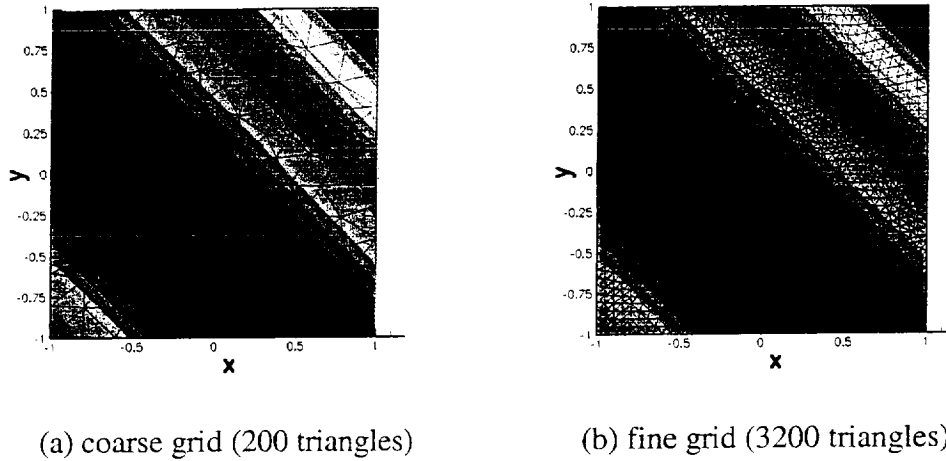


Fig. 3. Contour plot of E_z for a plane wave propagating through a square region (second order)

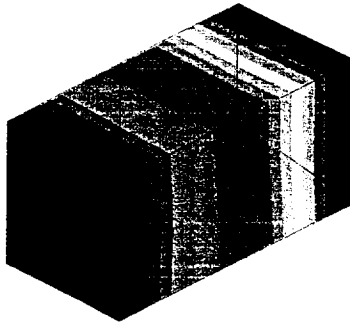


Fig. 4. Contour plot of E_z for a plane wave propagating through a rectangular parallelepiped

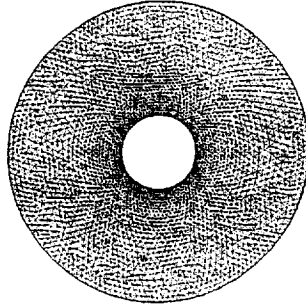
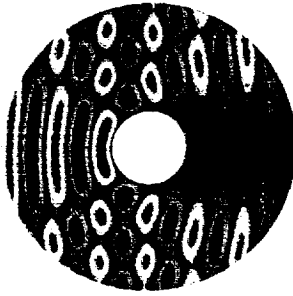
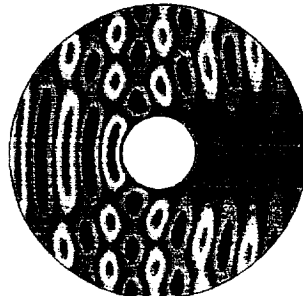


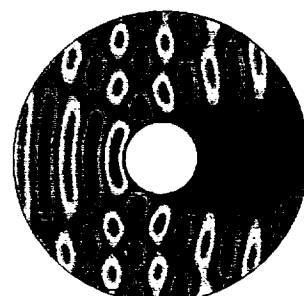
Fig. 5. Grid for a region exterior to a circular cylinder



(a) exact solution

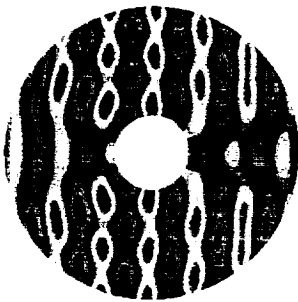


(b) second order

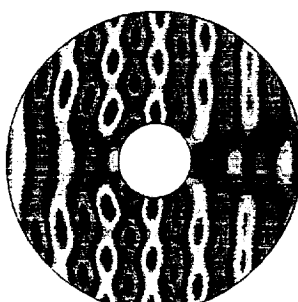


(c) fourth order

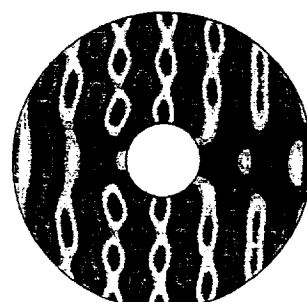
Fig. 6. Contour plot of E_z for a TM plane wave incident on a perfectly conducting cylinder



(a) exact solution



(b) second order



(c) fourth order

Fig. 7. Contour plot of H_z for a TE plane wave incident on a perfectly conducting cylinder

6. References

1. Z.J. Wang, Spectral (finite) volume method for conservation laws on unstructured grids: basic formulation, *J. Comput. Phys.* **178**, 210-251 (2002)
2. Z.J. Wang and Yen Liu, Spectral (finite) volume method for conservation laws on unstructured grids II: extension to two-dimensional scalar equation, *J. Comput. Phys.* **179**, 665-697 (2002)
3. T.J. Barth and P.O. Frederickson, High-order solution of the Euler equations on unstructured grids using quadratic reconstruction," AIAA Paper No. 90-0013, 1990.
4. M. Delanaye and Yen Liu, Quadratic reconstruction finite volume schemes on 3D arbitrary unstructured polyhedral grids, AIAA Paper No. 99-3259-CP, 1999.
5. Yen Liu, A generalized finite volume algorithm for solving Maxwell's equations on arbitrary grids, Proceedings of 10th Annual Review of Progress in Applied Computational Electromagnetics, 1994.
6. Yen Liu and M. Vinokur, Exact integration of polynomials and symmetric quadrature formulas over arbitrary polyhedral grids, *J. Comput. Phys.* **140**, 122-147 (1998).
7. B. Cockburn and C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework, *Mathematics of Computation* **52**, 411-435 (1989).
8. B. Cockburn, S.-Y. Lin and C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one-dimensional systems, *J. Comput. Phys.* **84**, 90-113 (1989).
9. B. Cockburn, S. Hou and C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: the multidimensional case, *Mathematics of Computation* **54**, 545-581 (1990).
10. B. Cockburn and C.-W. Shu, The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems, *J. Comput. Phys.* **141**, 199 - 224, (1998).

