International Journal of Engineering Applied Sciences and Technology, 2021
Vol. 6, Issue 6, ISSN No. 2455-2143, Pages 1-7
Published Online October 2021 in IJEAST (http://www.ijeast.com)

AN ADAPTIVE METHOD USING THE L-WENO RECONSTRUCTION

Mrumun C. Soomiyol
Department of Mathematics and Computer Science, Benue State University, Makurdi, Nigeria

Terhemen Aboiyar
Department of Mathematics, Statistics and Computer Science, University of Agriculture, Makurdi, Nigeria

Nathaniel M Kamoh
Department of Mathematics, University of Jos, Nigeria

Abstract-- **In this work, an adaptive formulation of the Legendre - Weighted Essentially Non-Oscillatory (L-WENO) method is used to solve some problems of two-dimensional linear conservation laws on unstructured triangular mesh. The mesh adaptivity is used to improve the performance of the method. Although the results with the L-WENO method gets better as the mesh is refined, the mesh adaptation algorithm was able to improve the quality of the numerical approximation and reduce computational cost by refining and coarsening the computational mesh based on some specified criteria.**

Keywords: **Legendre - Weighted Essentially Non-Oscillatory Method, Adaptive formulation, Mesh adaptation**

I. INTRODUCTION

Adaptive methods play a very important role in many PDE solvers. These methods are useful particularly because in many PDE simulations for conservation laws, the solution in most of the domain is smooth with discontinuities occurring over just a fraction of the domain. In such problems, an efficient solution method can be a mesh refinement approach, where the grids are refined with the much smaller mesh spacing placed only where they are needed. A good adaptive method combines high order approximations in smooth regions of the solution together with mesh refinement near the singularities. This enhances the quality of the numerical approximation and decreases computational cost Tang and Tang [1].

Mesh adaptivity is essentially important for the efficient computation of finite volume methods. The presence of discontinuities and sharp gradients in the computation require separate resolutions in different sections of the computational domain. This can be attained with the use of adaptive methods. Aboiyar *et al.* [2]. Over the past few decades, there has been some important progress in developing moving mesh methods for PDEs. The moving mesh method for Hyperbolic systems Tang and Tang [1], the adaptive mesh refinement (AMR) based on the finite difference WENO scheme Shen *et al.* [3], a high order one-step ADER-WENO scheme with AMR in multiple space dimensions Dumbser *et al.* [4]. Other adaptive methods can be found in Colella *et al.* [5], Zhang *el al.* [6] and Li [7].

In this work, we will implement an adaptive algorithm based on mesh refinement to the solution of two-dimensional conservation laws using the L-WENO reconstruction presented by Soomiyol *et al.*[8].

II. METHODS

2.1 The L-WENO Reconstruction Procedure

Consider the set $\mathcal{P}_n$ which is a vector space of dimension $N(n) = \frac{1}{2}(n+1)(n+2)$. It was assumed that the computational domain $\Omega \subset \mathbb{R}^2$ is discretized by a conforming triangulation $\mathcal{T}$, formed by the set $\mathcal{T} = \{T_\ell\}_\ell$ of triangles $T_\ell \in \Omega$, $\ell = 1, \dots, \#\mathcal{T}$. In the finite volume structure, each triangle has a cell average value

$$\bar{u}_\ell = \frac{1}{|T_\ell|} \int_{T_\ell} u(x)\, dx \qquad (1)$$

where $|T_\ell|$ is the area of the triangle T_ℓ.

In their reconstruction, the following problems were solved:

Given the space of polynomials $\mathcal{P}_n$, and cell average values $\bar{u}_{\ell_k}$, $k = 1, \dots, N$ (where $N = \dim\ \mathcal{P}_n$) of the function u on each control volume T_{ℓ_k}, find a polynomial $p \in \mathcal{P}_n$, that satisfies

$$p_{\ell_1} = \bar{u}_{\ell_1}, p_{\ell_2} = \bar{u}_{\ell_2}, \dots, p_{\ell_N} = \bar{u}_{\ell_N}$$

where the system has a unique solution if and only if the associated Vandermonde matrix is non-singular (Liu and Zhang, [9]).

In their computation, they used three cells for linear reconstruction, six cells for quadratic reconstruction and ten cells for cubic reconstruction.

with a basis function of the form

$$P^n(x,y) = \sum_{i=0}^{n} a_i R_i(x,y) = \sum_{\ell+m=0}^{n} c_{\ell,m} x^\ell y^m (2)$$

where $R_i(x,y), i = 0, 1, 2, \ldots, \mathrm{n}$ is the i^{th} degree two-variable Legendre polynomial and P^n is a polynomial of degree n.
They have,
for $n = 1$
$P^1(x,y) = a_0R_0(x,y) + a_1R_1(x,y) \qquad = a_0 1 - a_1x + a_1y \ = c_{1,00}1 + c_{1,10}x + c_{1,01}y$
and in the same way
$P^2(x,y) = c_{2,00}1 + c_{2,10}x + c_{2,01}y + c_{2,20}x^2 + c_{2,11}xy + c_{2,02}y^2$,
$P^3(x,y) = c_{3,00}1 + c_{3,10}x + c_{3,01}y + c_{3,20}x^2 + c_{3,11}xy + c_{3,02}y^2 + c_{3,30}x^3 + c_{3,21}x^2y + c_{3,12}xy^2 + c_{3,03}y^3$. (3)
To ensure that the scheme is conservative, it needs to satisfy
$\bar{p}_\ell^k = \bar{u}_\ell \ , \ \ell = 1, \ldots, N$ (4)
where N is the stencil size.
On a cell T, the polynomial for each stencil S_i is computed
$\bar{p}_{\ell,i}^k = \frac{1}{|T_\ell|}\int_{T_\ell} P^k(x,y)dx = \bar{u}_{T_\ell}$, $k = 1, 2, 3$
$l = 1, 2, \ldots, \#N_i$ (5)
where $\#N_i$ is the number of triangles in the stencil S_i. Using (5) the coefficients for p_i^k on stencil S_i is obtained. The WENO reconstruction is the weighted sum
$\sum_i \omega_i P_i^k(x,y)$ where $i = 1,2, \ldots n$ (6)
where ω_i is the weight which is defined by
$\omega_i = \frac{(\varepsilon+\mathrm{I}_i)^{-\rho}}{\sum_{i=1}^k(\varepsilon+\mathrm{I}_i)^{-\rho}}$
where ε is a small positive number to avoid division by zero, I_i is the oscillation indicator for the polynomial in each stencil and ρ is a measure of the sensitivity of the weights with respect to the oscillation indicator. The following values were used: $\varepsilon = 10^{-6}$ and $\rho = 4$

Mesh Adaptation
The adaptivity algorithm is implemented using the L-WENO method. An error indicator for each cell T in the triangulation $\mathcal{T}$ is first identified and then the mesh adaptation is implemented, where the computational mesh is adaptively modified during the simulation based on the error indicator. This allows the optimization of storage used as well as computation time.

Error indicator
A suitable error indicator is what guides the design and implementation of any adaptive method. An error indicator is normally computed for each triangular cell T and is used to determine if a cell lies within a region where the error of approximation is large. Elliptic and parabolic problems already have well established theories for error estimation as can be seen in Nochetto *et al.*[10], but there are no standard error estimates for nonlinear conservation laws. Instead, heuristic error indicators are used to determine regions where the solution possesses any shocks or steep gradients. Kurganov *et al.* [11] and Shen *et al.* [3] have presented some error indicators for the nonlinear case.

In computing the error indicator ε of a triangulation $\mathcal{T}$, select a cell $T \in \mathcal{T}$ which has three neighbours $T_1, T_2, and\ T_3$. The error is computed for each cell T_i, i=1, 2, 3.

For every $T \in \mathcal{T}$, the cell average $\bar{u}_T$ is used in the computation. The error for each neighboring cell is given as:

$\varepsilon_{T_1} = |\bar{u}_T - \bar{u}_{T_1}|, \quad \varepsilon_{T_2} = |\bar{u}_T - \bar{u}_{T_2}|, \ \varepsilon_{T_3} = |\bar{u}_T - \bar{u}_{T_3}|,$

and in general the error indicator will be

$\varepsilon_T = \max_{i=1,2,3} \varepsilon_{T_i}.$ (7)

Mesh adaptivity
With an appropriate criterion for the error indicator, one can effectively decide on which portion of the computational mesh to refine. This is practical about any adaptive mesh method, with respect to the refinement or coarsening of certain regions. Some methods that have been used for marking cells in a triangle for coarsening or refining include the fixed function strategy in Georgoulis *et al.*[12] and the bulking strategy by Chen and Zhang [13]. A good criteria to employ in marking cells for refining or coarsening is the following definition.

Definition 1
Let $\varepsilon^* = \max_{T\in\mathcal{T}} \varepsilon_T$, and let ϑ_r, ϑ_d be two threshold values satisfying $0 < \vartheta_d < \vartheta_r < 1$. We say that a cell $T \in \mathcal{T}$ is to be refined if and only if $\varepsilon_T > \vartheta_r.\varepsilon^*$ and T is coarsened or derefined if and only if $\varepsilon_T > \vartheta_d.\varepsilon^*$.

In this numerical experiment, the threshold values

$\vartheta_r = 0.005$ and , $\vartheta_d = 0.002$ (8)

will be used. In contrast to fixed mesh computations, a new mesh is generated at each time step and so a new set of Gaussian points need to be computed. The WENO reconstruction is then performed only after the mesh adaptation is completed.

III. RESULTS

We will demonstrate the benefits of mesh adaptivity by solving the linear advection equation and the Burger's equation using mesh adaptation with mesh size $h = 1/16$, on the computational domain $\Omega = [-0.5, 0.5] \times [-0.5,0.5]$. The solution to these problems are presented, first with three mesh sizes and then with the adapted mesh.

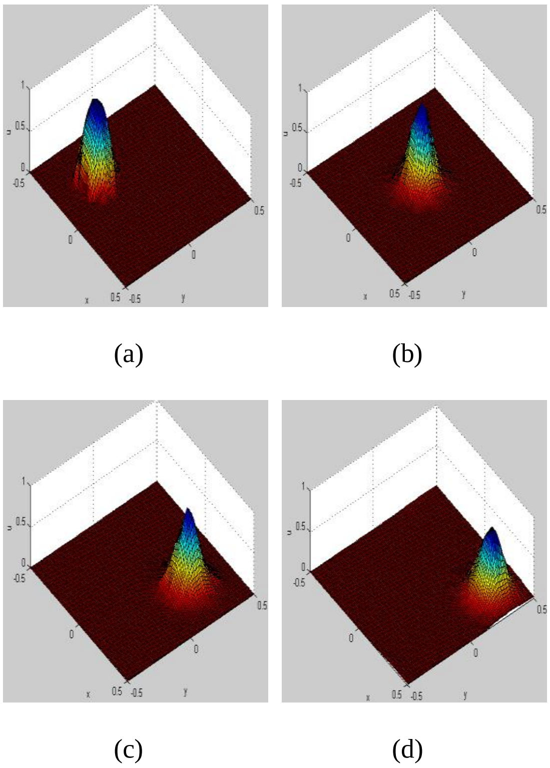
The linear advection equation

$u_t + u_x + u_y = 0$ for $u \equiv u(t, \mathbf{x})$ with $\mathbf{x} = (x,y) \in \mathbb{R}^2$ (9)

with the initial condition

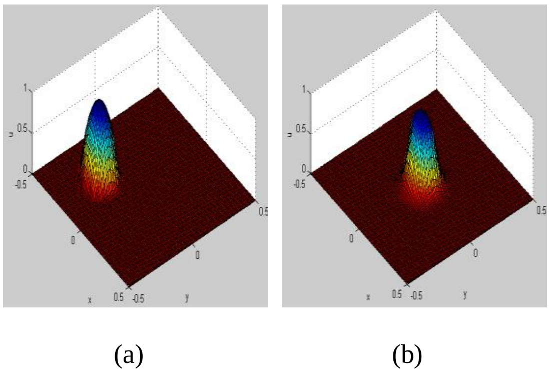
$$u(\mathbf{x}) = \begin{cases} \exp\frac{\|\mathbf{x}-\mathrm{c}\|^2}{\|\mathbf{x}-\mathrm{c}\|^2-\mathbb{R}^2}, & \|\mathbf{x}-\mathrm{c}\| < R \\ 0, & \text{otherwise} \end{cases} \quad (10)$$

is first solved on fixed meshes of $1/16$, $1/32$ and $1/64$ with a final time of $T = 0.5$, and then on the adapted mesh.


(a) (b)

(c) (d)

Fig 1: Solution of the linear advection equation at times (a) t=0, (b) t=0.2, (c) t=0.4 and (d) t=0.5 using the L-WENO scheme on mesh size h=1/16


(a) (b)

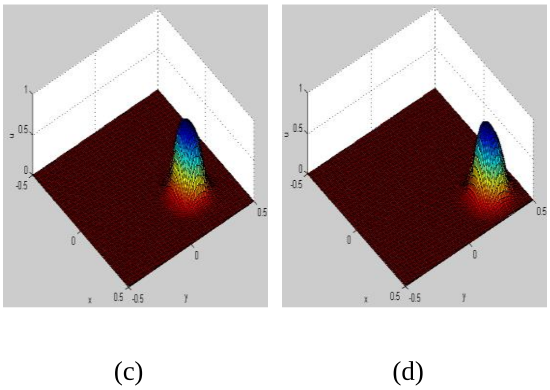
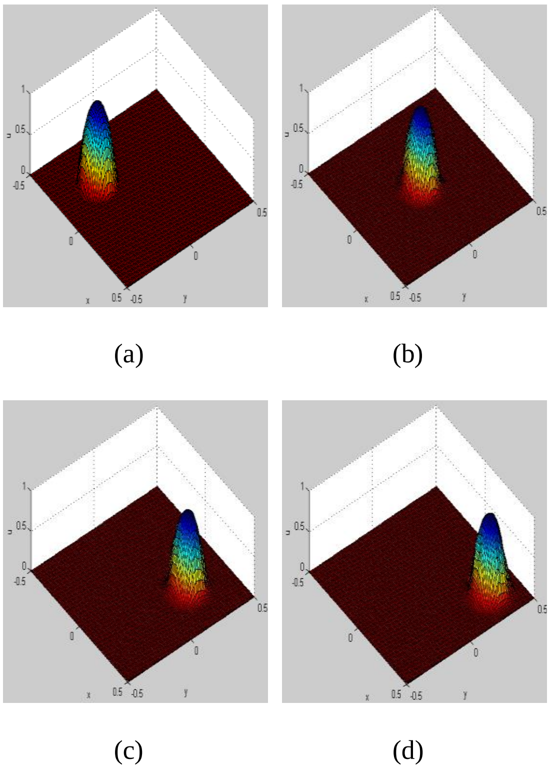
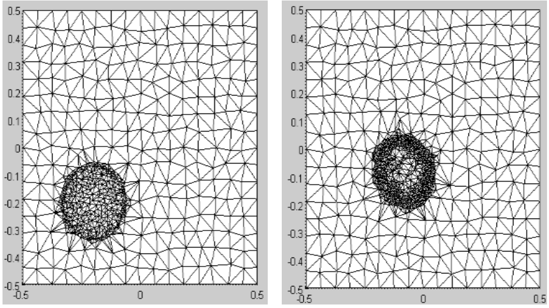

(c) (d)

Fig 2: Solution of the linear advection equation at times (a) t=0, (b) t=0.2, (c) t=0.4 and (d) t=0.5 using the L-WENO scheme on mesh size h=1/32

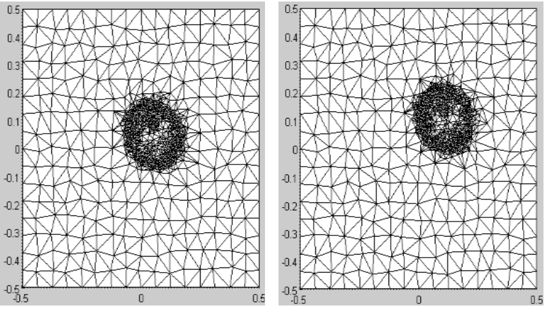

(a) (b)

(c) (d)

Fig 3: Solution of the linear advection equation at times (a) t=0, (b) t=0.2, (c) t=0.4 and (d) t=0.5 using the L-WENO scheme on mesh size h=1/64

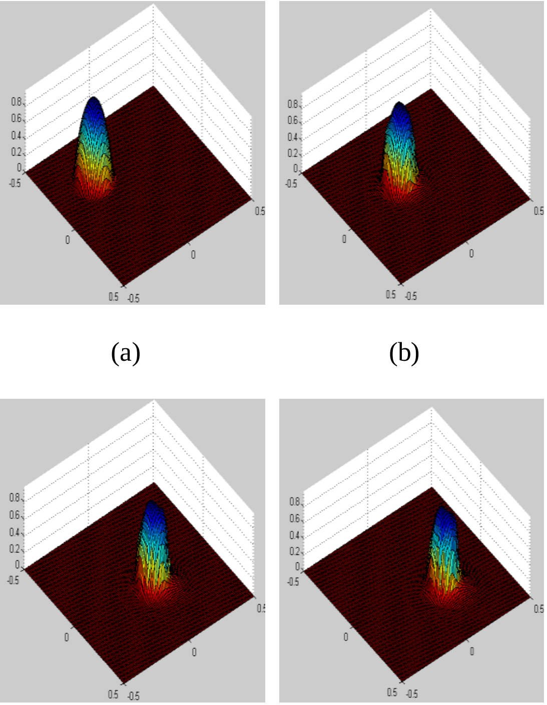


(a) (b)



(c) (d)

Fig 4: Adapted mesh for the solution of the linear advection equation at times (a) t=0, (b) t=0.2, (c) t=0.4 and (d) t= 0.5 using L-WENO method.



(a) (b)

(c) (d)

Fig 5: Solution of linear advection equation at times (a) t=0, (b) t=0.2, (c) t=0.4 and (d) t= 0.5 using L-WENO method on the adapted mesh.

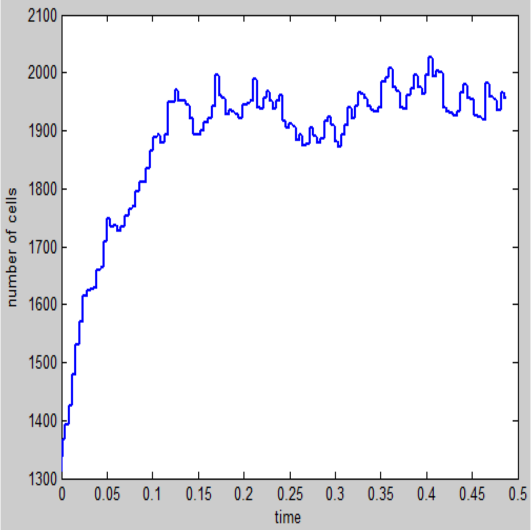


Fig 6: Number of cells for the adapted mesh during the simulation of the linear advection equation

Table - 1 Comparing minimum and maximum values, maximum number of cells and elapsed time for the solution of the linear advection equation on the fixed meshes and on the adapted mesh

Mesh	**Minimum(u)**	**Maximum(u)**	**Maximum number of cells**	**Elapsed time**
1/16	-2.141447 x 10^{-4}	1.121321	512	314.19s
1/32	-7.870207 x 10^{-4}	0.998503	2048	2541.38s
1/64	-0.002089	1.000562	8192	20086.05s
adapted mesh	-1.421939 x 10^{-4}	1.001099	2026	6289.52s

The Burger's equation

$$u_t + \left(\frac{1}{2}u^2\right)_x + \left(\frac{1}{2}u^2\right)_y = 0 \qquad (11)$$

with the initial condition (10) is also solved on the computational domain $\Omega = [-0.5, 0.5] \times [-0.5, 0.5]$ on fixed meshes of $1/16$, $1/32$ and $1/64$ with a final time of $T = 0.5$, and then on the adapted mesh.

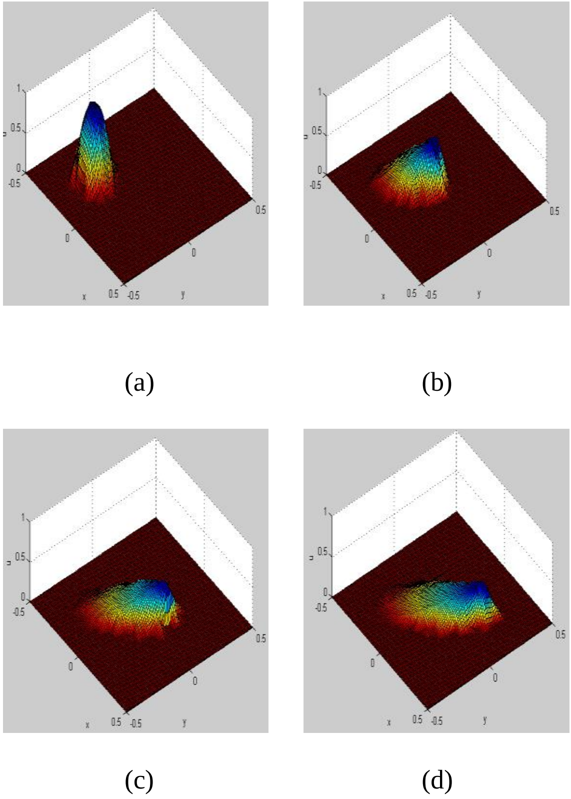

(a) (b) (c) (d)

Fig 7: Solution of the Burgers equation at times (a) t=0, (b) t=0.4, (c) t=0.8 and (d) t=1.2 using the L-WENO scheme on mesh size h=1/16.

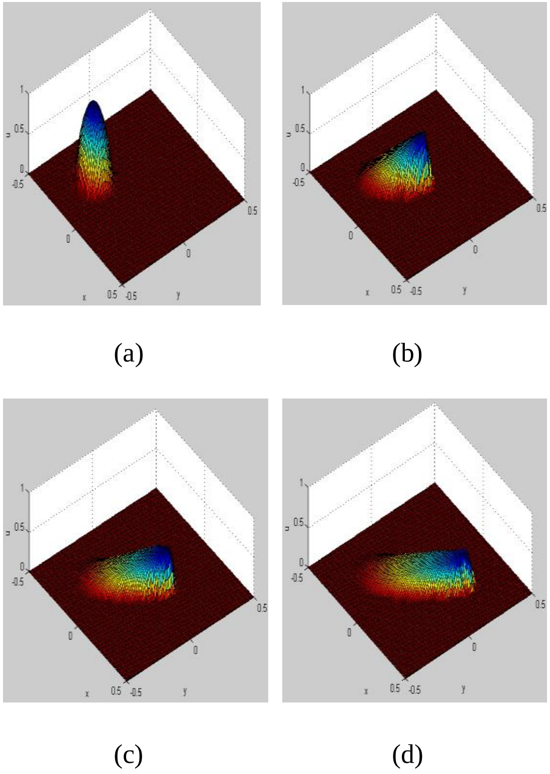

(a) (b) (c) (d)

Fig 8: Solution of the Burgers equation at times (a) t=0, (b) t=0.4, (c) t=0.8 and (d) t=1.2 using the L-WENO scheme on mesh with h=1/32.

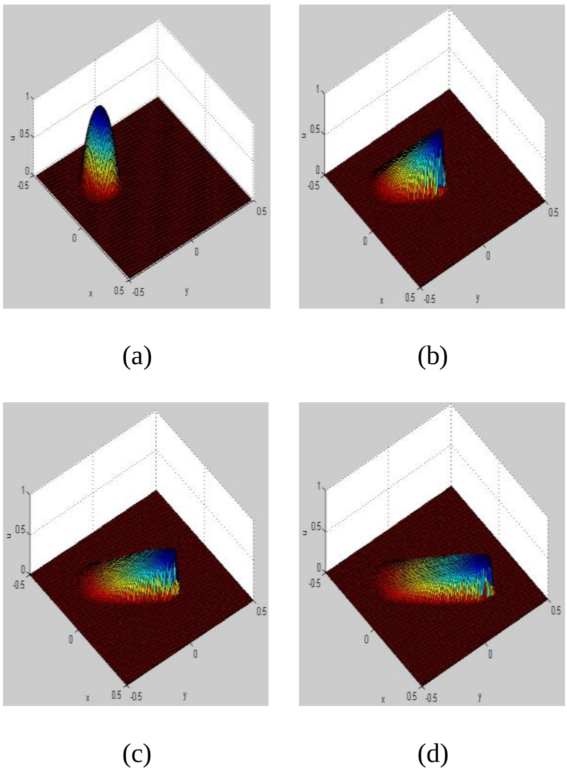

(a) (b) (c) (d)

Fig 9: Solution of the Burgers equation at times (a) t=0, (b) t=0.4, (c) t=0.8 and (d) t=1.2 using the L-WENO scheme on mesh size h=1/64.

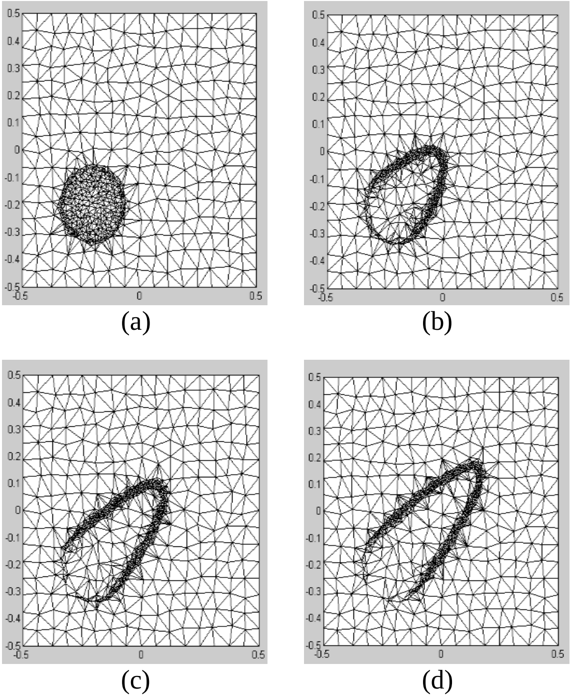
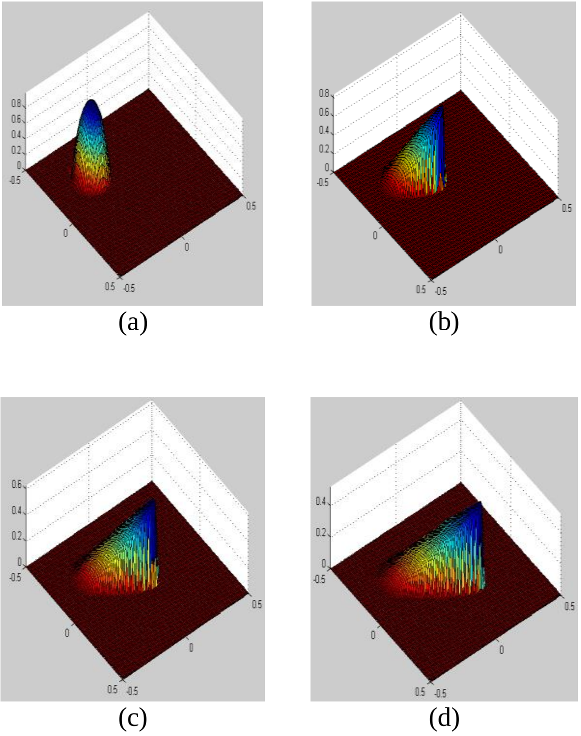

(a) (b) (c) (d)

Fig 10: Adapted mesh for the solution of the Burger's equation at times (a) t=0, (b) t=0.4, (c) t=0.8 and (d) t= 1.2 using L-WENO method.



(a) (b)

(c) (d)

Fig 11: Solution of Burger's equation at times (a) t=0, (b) t=0.4, (c) t=0.8 and (d) t= 1.2 using L-WENO method on the adapted mesh.

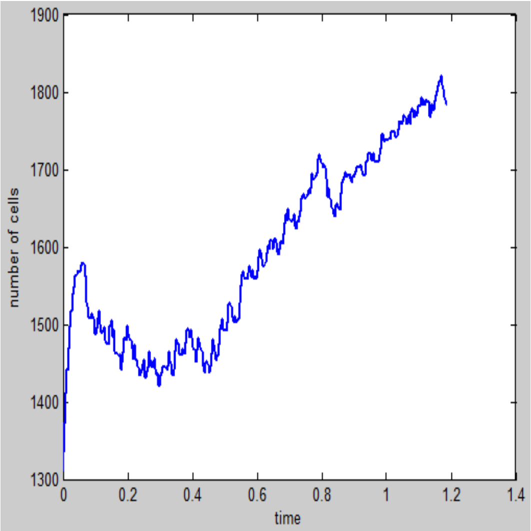


Fig 12: Number of cells for the adapted mesh during the simulation of the Burgers' equation.

Table-2 Comparing minimum and maximum values, maximum number of cells and elapsed time for the solution of the Burgers equation on fixed meshes and on the adapted mesh

Mesh	**Minimum(u)**	**Maximum(u)**	**Maximum Number of cells**	**Elapsed time**
1/16	**-8.690456×10^{-8}**	**1.010109**	**512**	**720.45s**
1/32	**-3.280590×10^{-7}**	**0.998503**	**2048**	**5844.4s**
1/64	**-1.907082×10^{-6}**	**0.999363**	**8192**	**48488.6s**
adapted mesh	**-2.050222×10^{-4}**	**1.003887**	**1820**	**14238.13s**

IV. CONCLUSION

In this work, adaptivity was used as a tool to improve the L-WENO method of Soomiyol et al., the linear advection equation and the Burger's equation was used to validate the performance of the mesh adaptivity. The values on Table 1 and Table 2 show that the quality of the numerical approximation using the L-WENO method has been improved using the mesh adaptation algorithm. A large mesh size was used, and the procedure was able to refine the mesh at the points of discontinuity only, by coarsening the computational mesh over some portions of the computational domain based on some criterion thereby producing very fine results at a minimal computational time and cost.

V. REFERENCES

[1] Tang H. and Tang T. (2003). Adaptive Mesh Methods for One-and Two- dimensional Hyperbolic Conservation Laws. *Society for Industrial and Applied Mathematics Journal on Numerical Analysis*, 41(2): (pp. 487-515).

[2] Aboiyar T., Georgoulis E. H. and Iske A. (2010). Adaptive ADER Methods Using Kernel Based Polyharmonic Spline WENO Reconstruction. *Society For Industrial and Applied Mathematics Journal on Scientific Computing,* 32(6): (pp. 3251-3277).

[3] Shen C., Qiu J.M. and Christlieb A. (2011). Adaptive Mesh Refinement Based on High Order Finite Difference WENO Scheme for Multi-scale Simulations. *Journal of Computational Physics*, 230(10): (pp. 3780-3802).

[4] Dumbser M., Zanotti O., Hidalgo A. and Balsara D. S. (2013). ADER-WENO finite volume schemes with space–time adaptive mesh refinement. *Journal of Computational Physics, 248* (pp. 257-286).

[5] Colella P., Dorr M., Hittinger J., Martin D. F. and McCorquodale P. (2009). High-order Finite-volume Adaptive Methods on Locally Rectangular Grids. In: *Journal of Physics: Conference Series*, 180(1): (p. 012010).

[6] Zhang Q., Johansen H. and Colella P. (2012). A Fourth-order Accurate Finite-volume Method with Structured Adaptive Mesh Refinement for Solving the Advection-diffusion Equation. *Society for Industrial and Applied Mathematics Journal on Scientific Computing, 34*(2): (pp. B179-B201).

[7] Li R. (2005). On multi-mesh h-adaptive methods. *Journal of Scientific Computing*, *24*(3), (pp. 321-341).

[8] Soomiyol M.C., Aboiyar T., and Shior M. (2020). The Legendre-WENO Method. *IOSR Journal of Mathematics*, 16(3): (pp. 14-23).

[9] Liu Y. and Zhang Y.T. (2013). A Robust Reconstruction for Unstructured WENO Schemes. *Journal of Scientific Computing*, 54(2-3): (pp. 603-621).

[10] Nochetto R., Schmidt A. and Verdi C. (2000). A Posteriori Error Estimation and Adaptivity for Degenerate Parabolic Problems. *Mathematics of Computation of the American Mathematical Society*, 69(229): (pp. 1-24).

[11] Kurganov A., Petrova G. and Popov B. (2007). Adaptive Semidiscrete Central-upwind Schemes for Nonconvex Hyperbolic Conservation Laws. *Society for Industrial and Applied Mathematics Journal on Scientific Computing*, 29(6): (pp. 2381-2401).

[12] Georgoulis E.H., Hall E. and Houston P. (2009). Discontinuous Galerkin Methods on hp-anisotropic Meshes II: A posteriori Error Analysis and Adaptivity. *Applied Numerical Mathematics*, 59(9): (pp. 2179-2194).

[13] Chen L. and Zhang C.S. (2007). A coarsening algorithm on adaptive grids by newest vertex bisection and its applications. *Journal of Computations Mathematics,* 28(6): (pp. 767-789).