Congress on Numerical Methods in Engineering 2011
Coimbra, 14 to 17 June, 2011

ADAPTIVE COLLOCATION METHODS FOR THE NUMERICAL SOLUTION OF DIFFERENTIAL MODELS

Paulo Brito[1,2]* and António Portugal[1]

1: CIEPQPF, Department of Chemical Engineering
Faculty of Sciences and Technology, University of Coimbra
Pólo II, Rua Sílvio Lima, 3030-790 Coimbra - Portugal
e-mail: {brito, atp}@eq.uc.pt, web: http://www.eq.uc.pt

2: Department of Chemical and Biological Technology
School of Technology and Management
Polytechnic Institute of Bragança
Campus de Santa Apolónia, PO Box 1134, 5301-857 Bragança - Portugal
e-mail: paulo@ipb.pt

Keywords: Partial Differential Equations; Numerical Methods; Adaptive Methods; Collocation Methods; Dyadic Grids

Abstract *A PDE integration algorithm that associates a Method of Lines (MOL) strategy based on finite differences or high resolution space discretizations, with a collocation strategy based on increasing level one or two-dimensional dyadic grids is presented. It reveals potential either as a grid generation procedure for predefined steep localised functions, and as an integration scheme for moving steep gradient PDE problems, namely 1D and 2D Burgers equations. Therefore, it copes satisfactorily with an example characterized by a steep 2D travelling wave and an example characterised by a forming steep travelling shock, which confirms its flexibility in dealing with diverse types of problems, with reasonable demands of computational effort.*

1. INTRODUCTION

It can be stated that the main purpose of science is to contribute for the understanding of the physical phenomena that surround us. In order to achieve this goal, scientific researchers adopt the so called scientific method (or hypothetico-deductive model) which can be defined in the next four general steps:

- Observation – use of experience and data available for recognition of problems that need to be solved.
- Hypothesis – formulation of a potential explanation which would solve the problem detected.
- Deduction – deduce a possible prediction which arises as a consequence from the hypothesis formulated.
- Test (Experiment) – confirmation or rejection of the hypothesis formulated by testing the veracity of the prediction.

A possible explanatory hypothesis can be a model, or more precisely, a mathematical model, that translates the observed phenomena to more easily treatable relations between abstract entities trough mathematical manipulations. In the specific field of mathematical modelling, one can narrow even more the range of interest to problems defined over space-time continuous domains, where phenomena are not only affected by the values of the variables that define its state, but also by the gradients of these variables relating to the independent coordinates. In this case, the mathematical models are defined by differential (or integral) equations established over multidimensional domains, i.e., partial differential equations (PDE's). Nevertheless, the process of deducing a suitable model, or modelling, has to be completed by the not less important task of solving it effectively.

It is obvious that it is not always feasible to solve mathematical problems using analytic procedures. In these cases (usually non-linear problems), one has to resort to numerical analysis, which generally implies the study of algorithms, i.e. sequential operation schemes that usually involve a discretization of continuous defined problems. These schemes can be applied in the solution of a variety of mathematical problems, such as optimization, interpolation, computation of integrals, resolution of algebraic or differential equations, etc.

2. NUMERICAL METHODS

Our particular concern resides on the numerical methods for the solution of time dependent partial differential equations (or systems of equations) defined over one- or multidimensional space domains. The main strategy behind the development of these schemes typically implies the construction of discrete grids that cover the total domain, and the approximation of the originally continuous solution by chosen suitable basis functions. Hence, the main classes of numerical methods developed for the solution of PDE's differ between each other by the type of basis functions selected, e.g.:

- Finite Differences (FD) – Taylor expansion series.
- Finite Elements (FE) – Interpolating polynomials.

2.1. Method of Lines

Our interest reside in a general strategy for the solution of PDE's named Method of Lines (MOL)[1] which structure can hold different schemes of mesh discretization. Usually, the numerical solution of PDE's implies the approximation of the original differential continuous problem, to a system of algebraic equations defined on a discrete domain. This transformation may be done simultaneously along every independent variable. Otherwise, one may apply a sequenced strategy: first, the discretization of the original problem in all directions but one (normally time for Initial-Boundary Value Problems) and, second, the integration in the remaining direction using a standard integrator package. The initial PDE problem is approximated to a system of ordinary differential equations (ODE's), which is solved by an available ODE integrator. Therefore, one can resort to a wide variety of different basis functions to execute the discretization step, e.g., Taylor expansion approximations[1], different order polynomials, wavelets[2,3], radial basis functions[4], etc.

2.2. Adaptation Concept

The classical approach to these procedures is rigid and non-adjustable to its evolution. One way to overcome the problems that may possibly arise from the lack of flexibility of this approach is the adoption of the adaptation concept. Adaptivity implies the tuning of algorithm parameters to the specific conditions of the solution evolution. For the numerical solution of PDE´s it can assume the following general principles:

- h-refinement – grid refinement and relaxation.
- p-refinement – tuning approximating orders.
- r-refinement – assignment of nodal velocities.

These different strategies are not mutually exclusive and may be combined in mixed adaptive methods. The relevance of adaptivity in the PDE solving field has already been a subject of study for a few decades, and the variety of methods proposed is rather broad [5,6]. Nevertheless, the objectives of the adaptive procedure are always the same: the generation of grids that cluster nodes in the domain regions where the solution is more active (i.e. exhibits steeper gradients) and disperse them in the remaining areas, and/or follow efficiently the problematic features of the solution. The application of adaptivity into the wider MOL strategy concept is a fairly straightforward operation[7].

2.3. Dyadic Grids

The overall adaptive grid is based in a series of embedded dyadic grids of increasing level, at each time step of the integration. A k-level one-dimensional uniform dyadic grid is defined by a nodal mesh with 2^k equidistant intervals. Therefore, a higher level grid is generated by adding nodes to the previous one, at every interval centre position (vd. Figure 1). It is important to note that a grid of level k is always included in all grids of higher level. So, the principle is to construct grids that merge nodes of different resolution levels according to the estimative of the function activity over the different regions of the whole domain, accomplished by the definition of a collocation criterion.

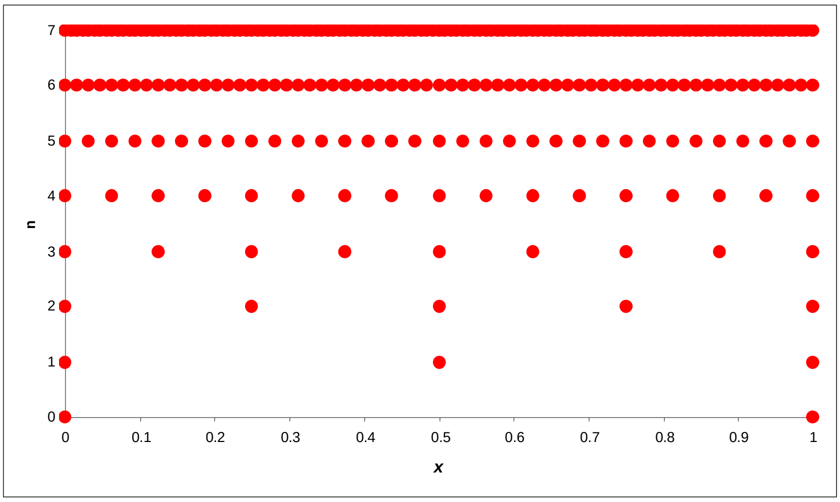


Figure 1. Uniform 1D dyadic grids of increasing level.

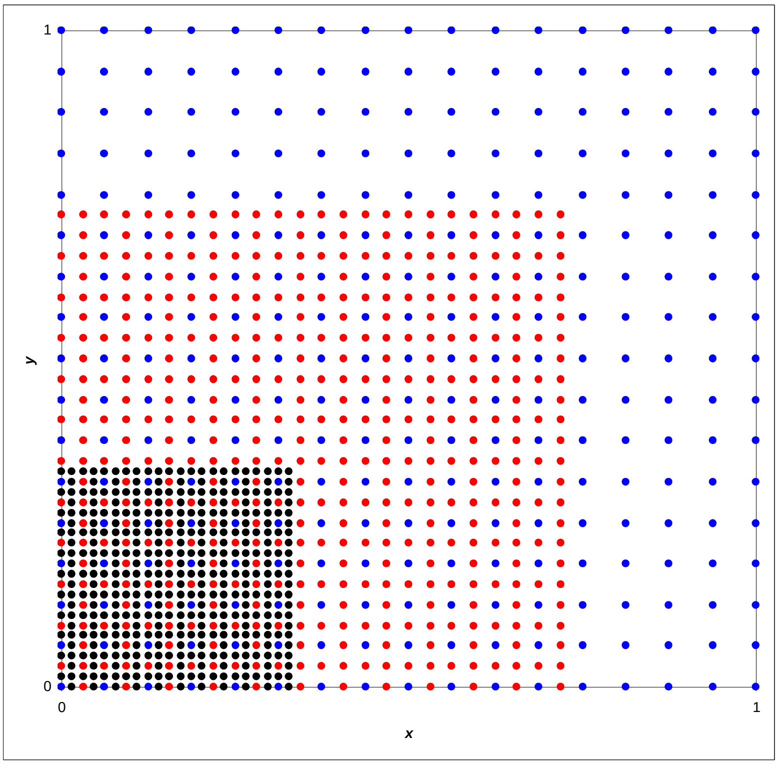


Figure 2. Uniform 2D dyadic grids of increasing level.

It is evident that the referred strategy can be extended to two-dimensional dyadic grids (vd. Figure 2), in a rather straightforward fashion. For that purpose, we establish collocation strategies that apply function dependent features, allowing the activation (or deactivation) of nodes belonging to dyadic grids ranging from a minimum resolution level (M) – the basis level; to a maximum allowed resolution level (N).

2.4. Numerical Collocation Algorithm

Using the concept of dyadic grid associated with finite differences approximations, we construct a collocation algorithm for grid generation which can be introduced in a MOL algorithm general structure for the solution of PDE's. As an example, consider a region of space domain defined by two consecutive one-dimension dyadic grids (vd. Figure 3). A collocation algorithm is developed for the activation of the required nodes by the following procedure[8-9]:

- $k = M$
- **for** $i = 1, \ldots, 2^k - 1$
- estimate U_i^n (order n derivative at node i) by finite differences
- **if** collocation criterion is met: select intermediate nodes of level k+1: $x_{2i-1}^{k+1}; x_{2i}^{k+1}; x_{2i+1}^{k+1}$
- $k = k + 1$ (repeat **for** $k = M, \ldots, N - 1$)

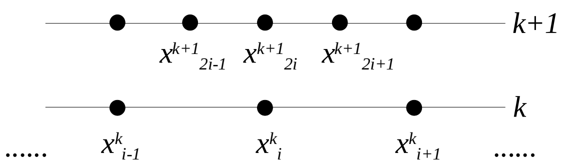


Figure 3. Representation of the connection between nodes of consecutive levels.

The collocation criteria obey to two different strategies. Initially, the grid size is calculated by,

$$\Delta x = \frac{x_{i+1}^k - x_{i-1}^k}{2} \tag{1}$$

Then, we define a criterion that pretends to identify oscillations on the finite difference approximation of degree n at each internal grid position i (U_i^n):

Criterion C1

- compute $\delta_1 = U_i^n \times U_{i-1}^n$ and $\delta_2 = U_{i+1}^n \times U_i^n$
- criterion verified if:

$$\left| U_i^n \times \Delta x \right| > \varepsilon_1 \qquad \text{or} \qquad \begin{cases} \delta_1 \le 0 \\ \delta_2 \le 0 \end{cases}$$

and $$\frac{\left| U_{i-1}^n \right| + \left| U_i^n \right| + \left| U_{i+1}^n \right|}{3} > \varepsilon_2$$

A similar criterion named C1σ may also be developed, by the substitution of the average evaluation presented above by the correspondent standard deviation test.
A second criterion that tracks high variations on the finite difference 1D profile is defined:

Criterion C2

- compute $\delta_1 = U_i^n - U_{i-1}^n$ and $\delta_2 = U_{i+1}^n - U_i^n$
- criterion verified if:

$$\left| U_i^n \times \Delta x \right| > \varepsilon_1 \quad \text{or} \quad \delta_1 \times \delta_2 \leq 0$$

and $$\frac{\left|\delta_1\right| + \left|\delta_2\right|}{2} > \varepsilon_2$$

ε_1 and ε_2 represent the criteria tolerances that define the sensibility of the process to the detection of non-uniformities in the solution numerical profiles. Both criteria are established to take advantage of the approximating nature of the space derivatives estimating finite difference scheme. The errors associated with the finite difference procedure induce artificial oscillations in the estimated derivative profiles especially near the steep fronts regions, which can be identified. Consequently, the grid resolution is increased on these regions by activation of higher level nodes that do not verify the more demanding collocation criteria. The assembly of all active nodes over every dyadic grid, generates the grid. One advantage of this scheme is the possibility of applying the collocation algorithm sequentially, analyzing several derivative orders by successive stages, e.g. verification of the first derivative condition and subsequent application of a second derivative analysis to the previous adaptive grid.
The collocation algorithm depicted above can be extended to 2D domains by a wide variety of strategies. We chosen to select a particular scheme that implies a sweeping of the 2D domain by a sequence of 1D procedures over the 1D-x grids for each higher resolution level y position, followed by a similar procedure on the correspondent 1D-y grids at the higher resolution x positions, as described in [8].

3. GRID GENERATION

As a first approach, we tested the performance of the collocation algorithm for the generation of grids that conform to the properties of selected one-dimensional functions.
Therefore, we apply the collocation criteria to the next function,

$$u(x,0) = \exp\left(-\frac{(x-x_0)^2}{\varepsilon}\right) \tag{2}$$

For $\varepsilon = 10^{-4}$ and $x_0 = 0.5$, which represents a steep wave centred at the domain middle position. The results for $\varepsilon_1 = \varepsilon_2 = 10^{-2}$, are presented for criteria C1 and C2 in Figures 4 and 5, respectively.

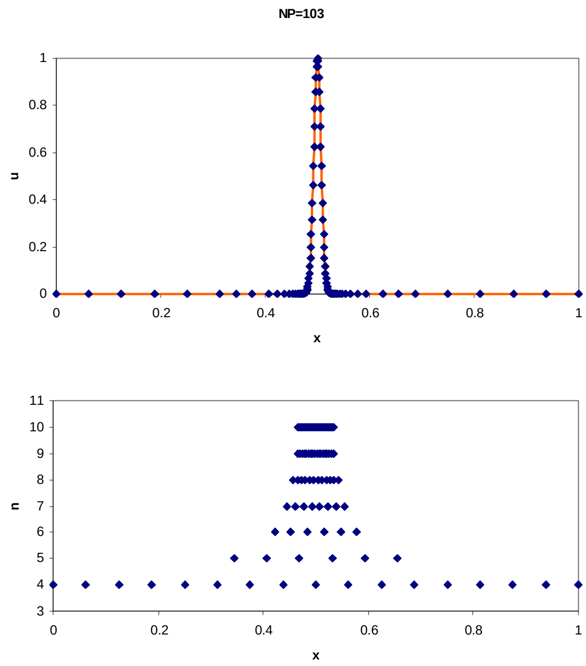


Figure 4. Grid generated for the wave example (criterion C1).

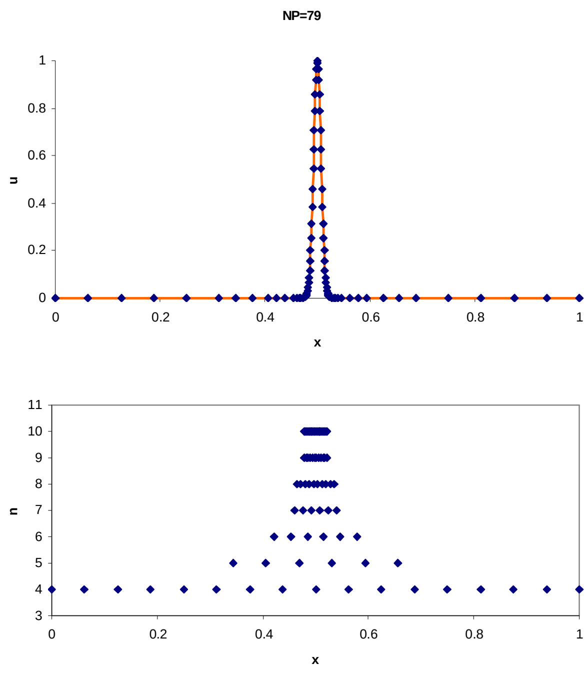


Figure 5. Grid generated for the wave example (criterion C2).

We observe that both criteria deals rather satisfactorily with the difficulties posed by the example studied. The algorithm is able to detect and describe the steep wave and its curvature on the edges, and to represent the low activity regions at both sides of the wave. For equivalent tolerances the C1 criterion seems to be more sensitive than criterion C2, putting a larger amount of nodes for a similar support of the wave configuration.

4. SIMULATION EXPERIMENTS

The node collocation procedure is incorporated in an algorithm for the resolution of one-dimensional time-dependent PDE´s. This strategy is based on the conjugation of a MOL algorithm where the space derivatives are approximated by finite differences formulas or high resolution schemes(HRS), with grid generation procedure at specified times that reformulate the space grid according to the solution evolution. At these intermediate times the solution profiles are reconstructed through an interpolation scheme. The time integration is performed by the ODE integrators DASSL[10] or RKF-45[11]. Therefore the presented algorithm can be included in the class of h-refinement PDE solution adaptive procedures.

The finite difference coefficients are computed using the recursive method of Fornberg[12] and the HRS schemes are based on the NVSF method[13] associated to flux limiting strategies, like the SMART or MINMOD procedures[14].

4.1. 1D Burgers Equation

The first test model is the widely studied 1-D general Burgers equation[4],

$$\frac{\partial u}{\partial t} = -u\frac{\partial u}{\partial x} + \nu\frac{\partial^2 u}{\partial x^2} \tag{3}$$

defined in the domain $x \in [0,1]$, with the Dirichlet boundary conditions:

$$u(0,t) = u(1,t) = 0 \tag{4}$$

This problem corresponds to an advection-diffusion problem, which may present some interesting challenges, depending of the initial condition applied. Hence, for the particular sinusoidal initial condition,

$$u(x,0) = \sin(2\pi x) + \frac{1}{2}\sin(\pi x) \tag{5}$$

as the advection velocities represent the solution itself, the problem develops from a smooth sinus type profile to a rather steep front, which forms at $x \approx 0{,}60$ by $t \approx 0{,}20$. From this instant on, the front moves on the positive direction of x until it eventually crashes onto the right boundary (x=1) and gradually fades away. The moving front thickness depends on the significance of the diffusion term, i.e., it is proportional to the magnitude of the diffusion coefficient (ν). Therefore, we test the performance of both collocation criteria by fixing the algorithm run conditions shown in Table 1 (for $\nu = 10^{-3}$). The simulation results for criteria C1 and C2 are presented in Figures 6 and 7, respectively. We observe that, in both cases, the algorithm effectively follows the formation and movement of the steep front, with hardly any difficulty. The results obtained using the two collocation criteria seem to be very similar.

Now, the Burgers' equation is tested in more challenging circumstances, by decreasing the diffusivity by a factor of 10, fixing the parameter $\nu = 10^{-4}$. In these conditions, we apply again the sequential first and second derivative analysis, associated with criterion C1. On the other hand, the maximum resolution level is increased to N=12, in order to account the reduced thickness of the moving steep front. The general conditions chosen for this implementation with integrator DASSL are resumed in Table 2.

Collocation criterion	C1 or C2
Derivative order for collocation	n=1 and 2; or n=1
Time step	10^{-3}
Finite Difference approximation	5 nodes centred - uniform grid
Interpolation strategy	Cubic splines with 9 nodes
Time integrator tolerances	10^{-6}
Dyadic grids levels	M=4; N=10
$\varepsilon_1 = \varepsilon_2 = 10^{-2}$	

Table 1. Simulation parameters for 1D Burgers equation ($\nu = 10^{-3}$).

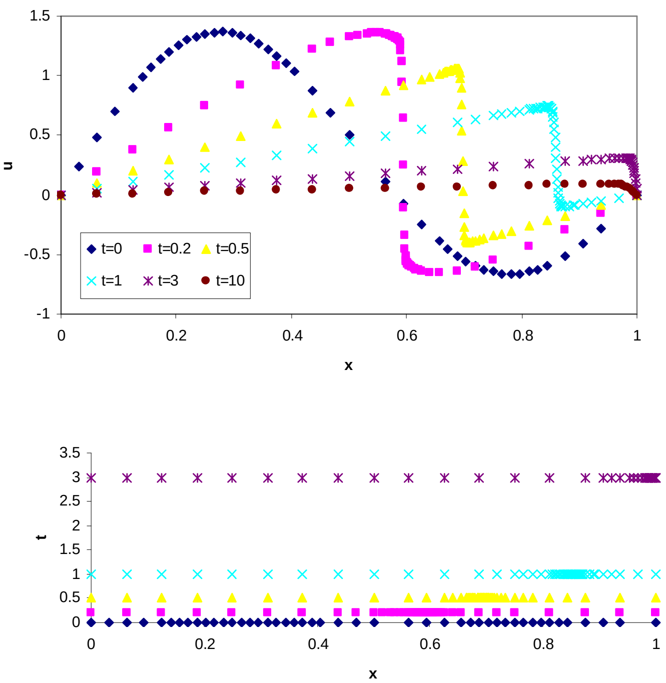
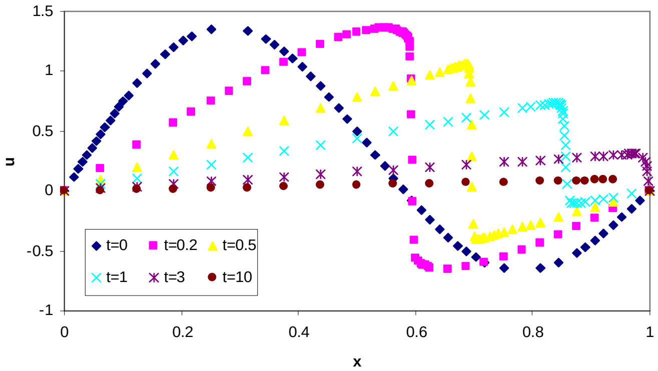


Figure 6. Numerical implementation for 1D Burgers equation (criterion C1; $\nu = 10^{-3}$).



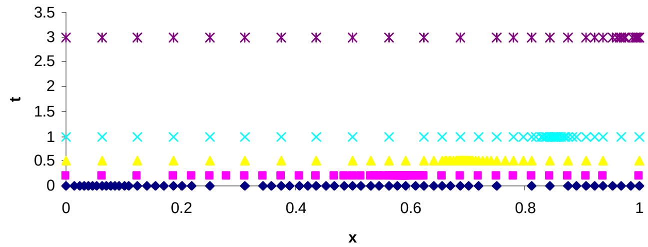


Figure 7. Numerical implementation for 1D Burgers equation (criterion C2; $\nu = 10^{-3}$).

Collocation criterion	C1
Derivative order for collocation	n=1 and 2
Time step	10^{-3}
Finite Difference approximation	5 nodes centred - uniform grid
Interpolation strategy	Cubic splines with 9 nodes
Time integrator tolerances	10^{-6}
Dyadic grids levels	M=4; N=12
$\varepsilon_1 = \varepsilon_2 = 10^{-2}$	

Table 2. Simulation parameters for 1D Burgers equation ($\nu = 10^{-4}$).

In Figure 8, we condense the simulation numerical results for the conditions of Table 2. It is obvious that due to the smooth characteristics of the initial profile, the grid is relatively coarse at the start. However, the situation changes radically for *t*=0.20 (vd. Figure 8). At this instant, the front is fully developed, and the procedure has to take advantage of the maximum level nodes generating a localized high resolution grid, which adequately conforms to the front and its positive and negative edges.
After the formation of the steep front, the algorithm shows its ability to follow the movement of the front without introducing numerical distortions on the edges. The algorithm also proves its suitability by providing a adequately simulation of the front crash at the right boundary. In general, we may conclude that the simulation is successfully carried out.

The evolution of the size of the adapted grids for the three examples presented is monitorized along each problem execution (vd. Figure 9) in order to access the relative computational effort demand. It is clear that the steepest example ($\nu = 10^{-4}$) is more exigent promoting the generation of heavier grids to account the formation of a steeper moving front. In similar conditions the criterion C1 seems to be more demanding than criterion C2 leading to the generation of relatively denser grids. The procedures tend to generate lighter grids, as the front slowly fades out through the right boundary, until it eventually disappears all together.

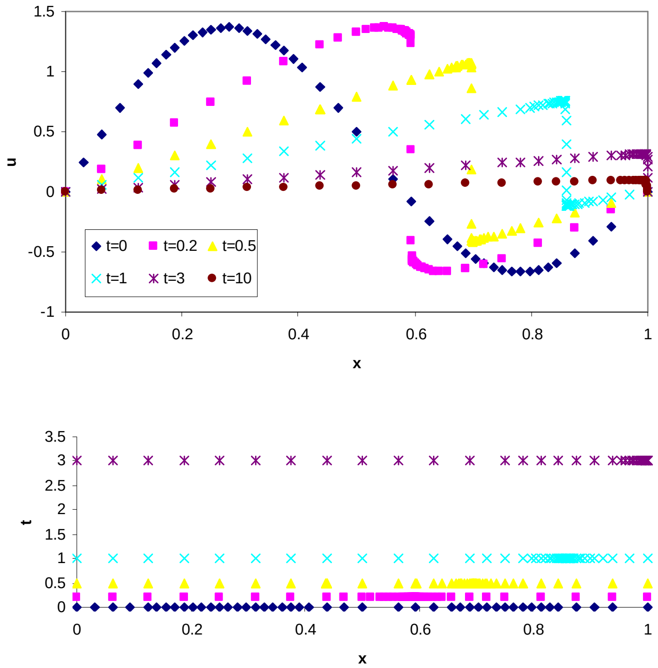


Figure 8. Numerical implementation for 1D Burgers equation (criterion C1; $\nu = 10^{-4}$).

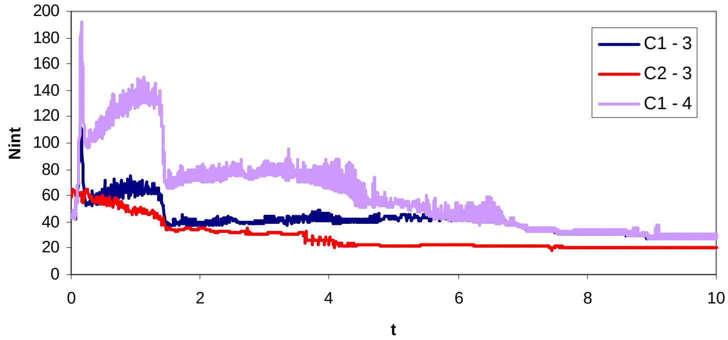


Figure 9. Time evolution of the dimension of the adaptive grid generated by the 1D Burgers equation numerical implementation.

4.2. 2D Burgers Equation

Now, we study the 2D version of the Burgers equation in its inviscid form (with a vanished viscosity coefficient ($\nu_x = \nu_y = 0$) in both spatial directions):

$$\frac{\partial u}{\partial t} = -\left(\frac{\partial f(u)}{\partial x} + \frac{\partial f(u)}{\partial y}\right) \tag{6}$$

with $f(u) = u^2/2$; completed by the boundary and initial conditions:

$$u(-1, y, t) = u(1, y, t) = u(x, -1, t) = u(x, 1, t) = 0 \tag{7}$$

$$u(x, y, 0) = \begin{cases} 1, & (x+0.5)^2 + (y+0.5)^2 \le 0.4^2 \\ 0, & else \quad in \quad [x, y] \in [-1,1]^2 \end{cases} \tag{8}$$

which stands for an unitary cylindrical pulse, centred on the [0.5;0.5] position of the spatial 2D domain $[-1,1]^2$. The execution is implemented using the time integrator RKF-45 and the numerical parameters resumed in Table 3. We conclude that the grid generation procedure is successful in representing the steep gradients that characterise the initial pulse solution with a reasonable number of nodes (vd. Figure 10).

Collocation criterion	C1σ or C2
Derivative order for collocation	n=1; or n=1
Time step	10^{-2}
Finite Difference approximation	5 nodes centred - uniform grid
Discretization scheme	NVSF; SMART/MINMOD
Interpolation strategy	Cubic splines with 7 nodes
Time integrator tolerances	10^{-5}
Dyadic grids levels	M_x=M_y=4; N_x=N_y=8
$\varepsilon_1 = \varepsilon_2 = 10^{-1}$	

Table 3. Simulation parameters for 2D Burgers equation.

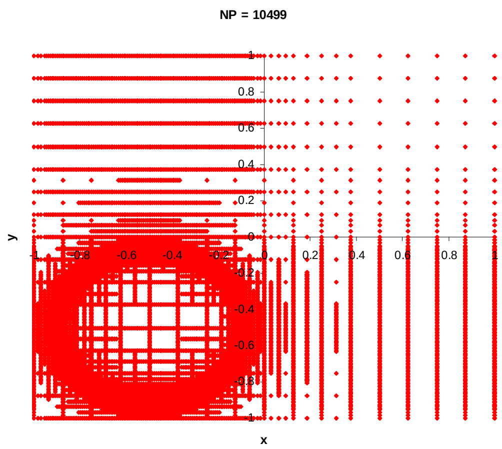


Figure 10. Initial grid generated for the 2D Burgers equation (NVSF discretization with a SMART limiter and collocation criterion C1σ).

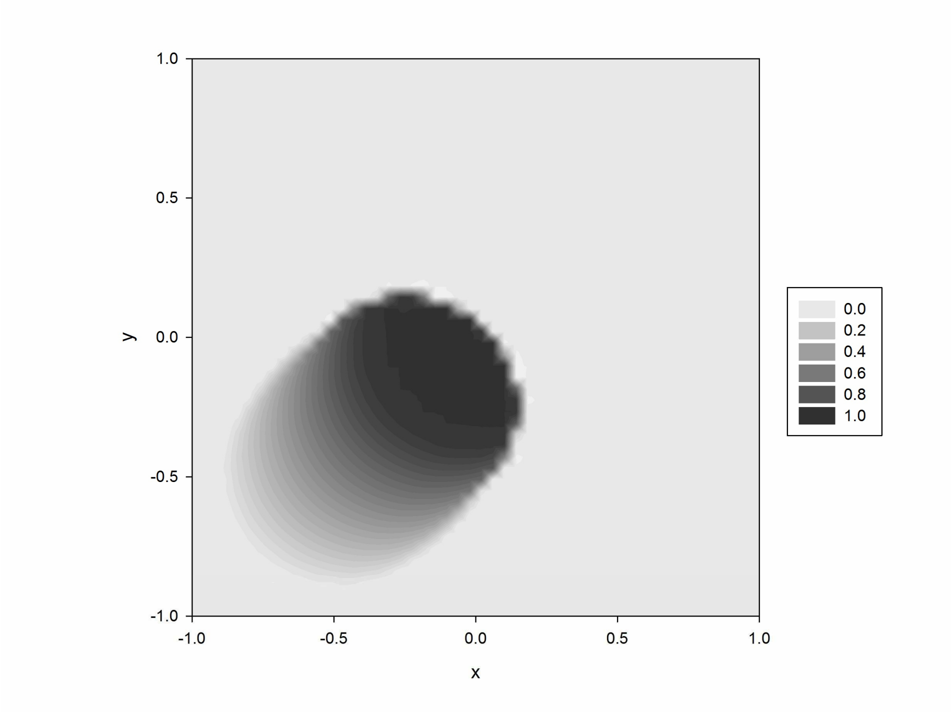


Figure 11. Contour numerical solution profiles (t = 0.5) for the 2D Burgers equation (NVSF discretization with a SMART limiter and collocation criterion C1σ).

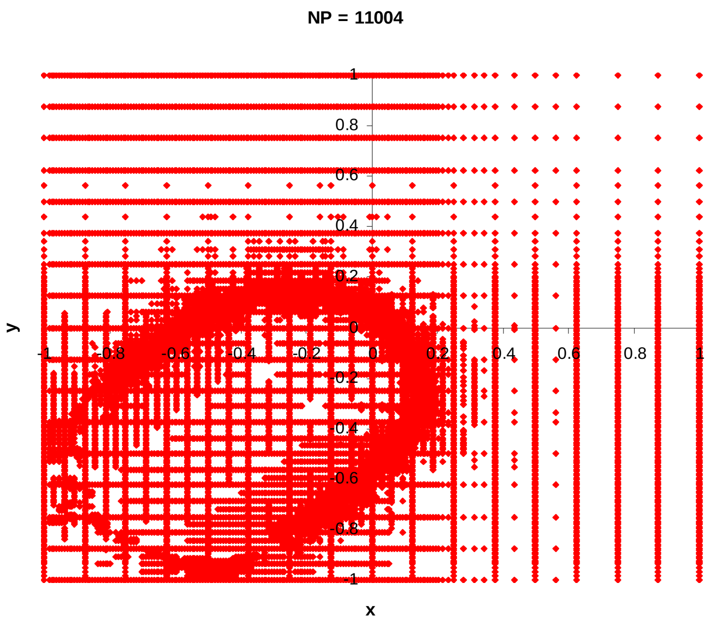


Figure 12. Grid generated (t = 0.5) for the 2D Burgers equation (NVSF discretization with a SMART limiter and collocation criterion C1σ).

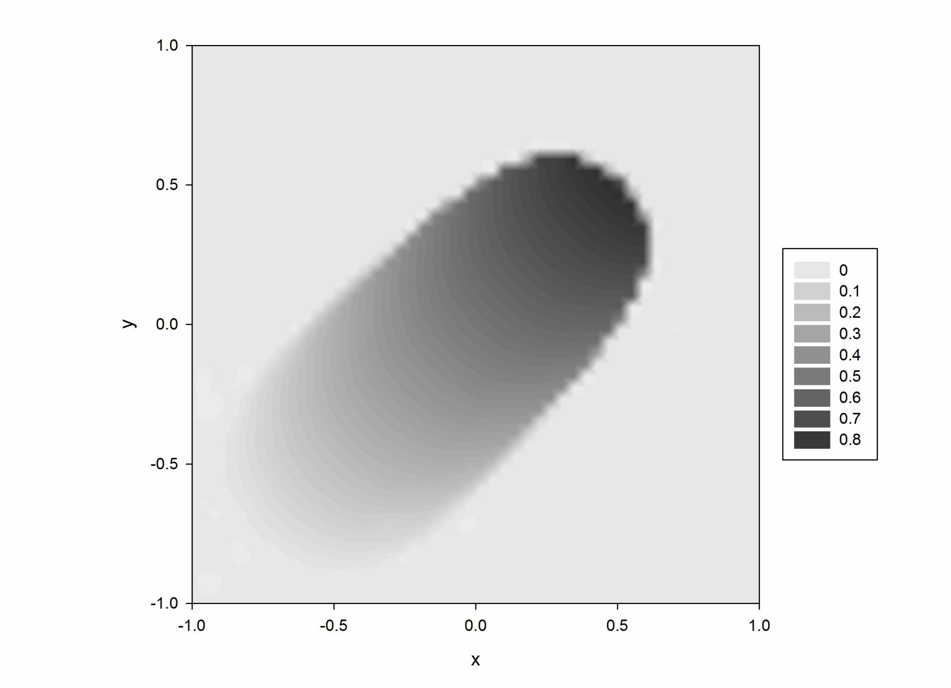


Figure 13. Contour numerical solution profiles (t = 1.5) for the 2D Burgers equation (NVSF discretization with a SMART limiter and collocation criterion C1σ).

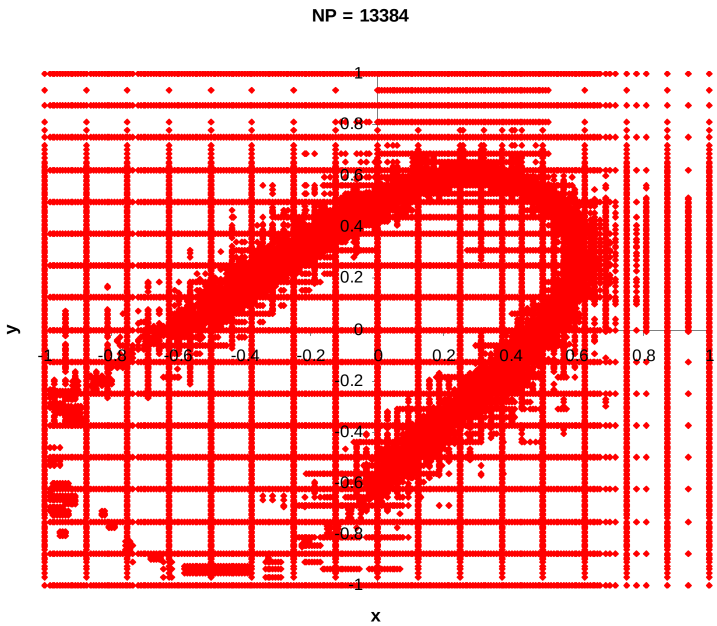


Figure 14. Grid generated (t = 1.5) for the 2D Burgers equation (NVSF discretization with a SMART limiter and collocation criterion C1σ).

The problem solution exhibits a migration of the initial pulse in the positive direction of *x* and *y*, simultaneously (vd. Figure 10). However, the movement speed of each front that constitutes the pulse is different. The downwind front has to force its movement through the plateau in its path which remains static (for the Burgers equation, the displacement characteristic velocity associated to each node coincides with the solution itself). On the other hand, the upwind front faces no resistance to its movement at the upper edge, but it is kept fixed to the plateau at the lower edge. Therefore the solution is characterized by the propagation of a steep downwind front which is eventually caught by the faster fading upwind front. The analysis of Figures 11 to 14 (for SMART limiter with collocation criterion C1σ) demonstrates the algorithm ability to follow the movement of the two fronts without introducing numerical distortions on its edges. The collocation procedure proves its suitability by providing an adequately simulation of the solution numerical profiles. The results obtained for the alternative runs (vd. Figure 15) are equivalent to the data presented above. So, we may conclude that the simulation is successfully carried out.

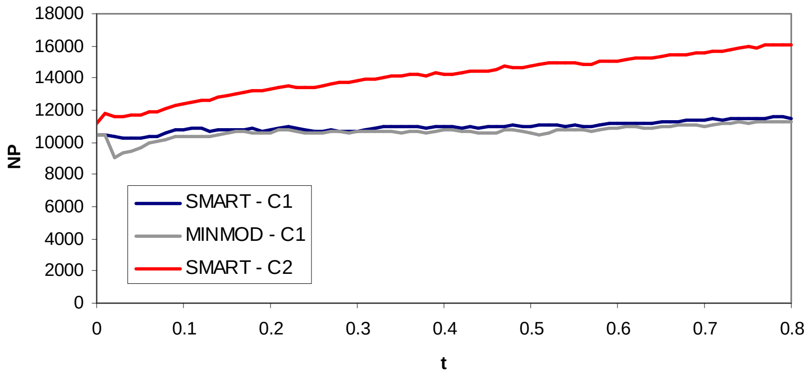


Figure 15. Time evolution of the dimension of the adaptive grid generated by the 2D Burgers equation numerical implementation.

Again, the progression of the size of the adapted grids for each example tested is monitorized along the algorithm implementation (vd. Figure 15). Now, the criterion C2 tends to generate a relatively denser grid than the corresponding criterion C1. The procedures present similar performances for limiters MINMOD and SMART conjugated with collocation criterion C1.

5. CONCLUSIONS

We conclude that an integration MOL algorithm that conjugates finite differences and/or high resolution space discretizations, with a collocation scheme based on increasing level 1D or 2D dyadic grids, reveals potential either as a grid generation procedure for predefined functions, and as an integration scheme for moving steep gradient PDE problems. It copes satisfactorily with an example characterized by a steep 2D travelling wave and an example characterised by a forming steep travelling shock, which proves its flexibility in dealing with different types of problems, with reasonable demands of computational effort.

REFERENCES

[1] W. E. Schiesser, *The numerical Method of Lines: integration of partial differential equations*, Academic Press, San Diego, (1991).

[2] J.C. Santos, P. Cruz, F.D. Magalhães e A. Mendes, "2-D Wavelet-based adaptive-grid method for the resolution of PDEs", *AIChe J.*, Vol. **49**, pp. 706-717, (2003).

[3] P. Cruz, M.A. Alves, F.D. Magalhães e A. Mendes, "Solution of hyperbolic PDEs using a stable adaptive multiresolution method", *Chem. Eng. Sci.*, Vol. **58**, pp. 1777-1792, (2003).

[4] T.A. Driscoll e A.R.H. Heryudono, "Adaptive residual subsampling methods for radial basis function interpolation and collocation problems", *Comput Math. Appl.*, Vol. **53**, pp. 927-939, (2007).

[5] D.F. Hawken, J.J. Gottlieb e J.S. Hansen, "Review of some adaptive node-movement techniques in finite-element and finite-difference solutions of partial differential equations", *J. Comput. Phys.*, Vol. **95**, pp. 254-302, (1991).

[6] P.M.P. Brito, *Aplicação de métodos numéricos adaptativos na integração de sistemas algébrico-diferenciais caracterizados por frentes abruptas*, MSc. Thesis, DEQ-FCTUC, Coimbra, Portugal, (1998).

[7] A. Vande Wouwer, Ph. Saucez, and W. E. Schiesser (eds.), *Adaptive Method of Lines*, Chapman & Hall/CRC Press, Boca Raton, (2001).

[8] P.M.P. Brito, *Métodos numéricos adaptativos para a resolução de modelos multidimensionais em Engenharia Química*, PhD Thesis, DEQ-FCTUC, Coimbra, Portugal, (2010).

[9] P. Brito e A. Portugal, *Adaptive collocation methods for the solution of partial differential equations*. T. Sobh, e K. Elleithy eds. *Int. Conf. on Syst., Comput. Sci. and Software Engineer. (SCSS 09), Int. Joint Conf. on Comp., Inf., and Systems Sci., and Engineer. (CISSE 2009)* – Vol. **2**: *Innovations in Computing Sciences and Software Engineering*, Springer (2010), pp. 499-504.

[10] L.R. Petzold, *A description of DASSL: a differential/algebraic system solver*, Sandia Tech. Rep. 82-8637, (1982).

[11] L. Shampine, H. Watts e S. Davenport, "Solving non-stiff ordinary differential equations - The state of the art", *SIAM Rev.*, Vol. **18**, pp. 376-411, (1976).

[12] B. Fornberg, "Generation of finite difference formulas on arbitrarily spaced grids", *Math. Comput.*, Vol. **51**, pp. 699-706, (1988).

[13] M.S. Darwish e F. Moukalled, "Normalized variable and space formulation methodology for high-resolution schemes", *Numer. Heat Transfer, Part B*, Vol. **26**, pp. 79-96, (1994).

[14] M.A. Alves, P.J. Oliveira e F.T. Pinho, "A convergent and universally bounded interpolation scheme for the treatment of advection", *Int. J. Numer. Meth. Fluids*, Vol. **41**, pp. 47-75 (2003).