Application of Adaptive Methods Based on Finite Difference Discretizations in the Simulation of a Tubular Reactor System

P. M. P. Brito and A. A. T. G. Portugal *

Departamento de Engenharia Química, Largo Marquês de Pombal, 3000 Coimbra, Portugal

Abstract

In this paper two adaptive algorithms are presented for the solution of systems of evolutive one-dimensional Partial Differential/Algebraic Equations (PDAEs). The temporal integration is coupled with a spatial adapting strategy. The identification of the spatial subdomains, where a regridding technique is introduced, is done through the comparison of the solutions computed with two fixed grids of different sizes. The subproblems generated are solved by two adaptive strategies: the Grid Refinement Method (GRM), that promotes the refinement of the subgrids detected in the previous step, and the Moving Mesh Method (MMM) includes an additional differential equation for the nodal mobility.

The two algorithms proposed were successfully applied to the solution of an nonisothermal tubular reactor pseudo-homogeneous model described by two PDEs referring to reagent concentration and system temperature dynamics. The performance of each algorithm is compared to the results obtained by [3], based on the application of a formulation of the Moving Finite Elements Method, with cubic Hermite polynomials approximations.

Keywords: Adaptive methods, partial differential equations, finite difference approximations, tubular reactor

AMS classification: 65M50

1 Introduction

Several problems in Engineering can be properly simulated by the solution of evolutive Differential/Algebraic Systems where the influence of diffusional/convective phenomena is very important. In the case of hyperbolic systems, the weight of the convective terms is dominant, which may lead to the development of steep moving waves or shocks on the solution profiles.

The numerical strategies used in this work are based on the Method of Lines: the derivatives over one of the independent variables (generally the spatial one) are estimated by algebraic formulas (in this case, finite difference approximations). The original PDE problem is transformed in a complex system of ODEs, that is integrated over the remaining independent variable (the temporal one) by a numerical integrator software (the DASSL implicit BDF formula package [6]). The finite difference weights are estimated by a recursive scheme developed by Fornberg [4], for arbitrarily spaced grids, and by a strategy inspired by Schiesser [8], for the evaluation of weights associated with Neumann boundary conditions.

When the solution developes large non-static spatial gradients, the overall grid has to be very dense, to reproduce accurately the numerical results without introducing numerical instability, which leads to unreasonable computational times. Such problems can be overcome by the introduction of a mobility criteria for the positions of the nodes in the grid, on the regions

*Author to whom correspondence should be addressed: e-mail: eq1atp@eq.uc.pt, phone: 351-39-828392, fax: 351-39-841138

of the domain where the spatial solution gradients are larger and the advance of the temporal integration is more difficult. Therefore, the grid adapts itself to the specific characteristics of the solution in each region of the domain.Thus, these type of algorithms are designated by Adaptive Methods.

In this paper, two adaptive algorithms are developed, that basically apply two important regridding techniques, widely studied by several authors:

- Grid Refinement and Relaxation [5] - Introduction and elimination of nodes from an initial grid. Based on a estimation discretization error procedure in each time step, several grids are constructed with various sizes or refinement levels through all the spatial domain, over which the problem is solved. Nodes are added in the areas of major solution activity (Grid Refinement) and removed from regions where the spatial gradients are lower (Mesh Relaxation).

- Dynamical Node Displacement [7] - Using selected properties of the solution, additional partial differential equations are deduced, that describe the movement of the nodal positions during the solution of the problem. These equations are computed together with the original differential system, that obviously has to be transposed to the related dynamical coordinate set of independent variables.

2 Adaptive Numerical Algorithms

In this paper, two adaptive mesh algorithms are described for one-dimensional evolutive systems of Algebraic-Differential Equations that can be resumed by the following general model:

$$F(\underline{u}_t, \underline{u}, \underline{u}_z, \underline{u}_{zz}) = 0 \tag{2.1}$$

$$G(\underline{u}) = 0 \tag{2.2}$$

subjected to the boundary conditions: $\underline{u}(z^L, t) = \underline{u}^L(t)$ and $\underline{u}(z^R, t) = \underline{u}^R(t)$,
and the initial condition: $\underline{u}(z, 0) = \underline{u}^0(z)$; $z \in [z^L, z^R]$.

Both algorithms are described in [1, 2] and can be structured in two main stages: estimation of the discretization error and identification of the adaptive subdomains; and solution of the subproblems generated in the first stage, by the introduction of an adaptive grid technique.

2.1 *Stage I - Discretization Error Estimation*

This stage is similar in both algorithms and it is based on the comparison of the solution obtained by solving the original problem on two different grids: a fine and a coarse grid (Grids of level 2 and 1, respectively). Initially, the fine grid is constructed by the bissection of each interval of the coarse one. The nodes in the level 1 grid, that do not satisfy the error criterium, are grouped together with the level 2 nodes placed between them, to form the subdomains over which the adaptive subproblems are generated and then solved.

2.2 *Stage II - Adaptive Integration of the Subproblems*

The second stage refers to the actual adaptive procedure that is different for each algorithm studied in this work.

2.2.1 Grid Refinement Method (GRM)

This adaptive procedure is based on the work presented by Guiné [5], that generates subproblems of increasing refinement level, by repeating the procedure described in Stage I, until every node in every grid verifies the tolerance condition associated with the error estimated by:

$$EU^i_{j,k+1} = Wh^i_{j,k+1} - W2h^i_{j,k+1}; \qquad j = 1, \cdots, NP_{n-1}, \qquad i = 1, \cdots, NPDE \tag{2.3}$$

In this case, $EU^i_{j,k+1}$ represents the approximation to the spatial error, in a node j of a grid of refinement level n; $Wh^i_{j,k+1}$ and $W2h^i_{j,k+1}$ are the approximations to the component i of the solution, obtained through integration between the times t_k and t_{k+1}, on the finer (level n) and the coarser (level n-1) grids, respectively; NP_{n-1} is the number of nodes in the grid of level n-1; and $NPDE$ is the number of partial differential equations of the problem.

The subdomains of level n+1 are obtained by joining all nodes n-1 that satisfy the condition:

$$| EU^i_{j,k+1} | > TOL_i; \qquad i = 1, \cdots, NPDE \tag{2.4}$$

In each refinement procedure, the profiles of the solution are computed by interpolation of the profiles of level 2, at all the intermediary positions.

The algorithm is coupled with a strategy for the treatment of boundary conditions in the refinement subproblems that simply defines fixed Dirichlet conditions on each internal bound. The positions of each bound, for the refinement level n+1 (for $n = 2, \cdots, N_{MAX} - 1$, where N_{MAX} is the maximum refinement level) are coincident with the positions of the first nodes of level n-1 that verify the specified tolerance. The constant value of the boundary conditions is given by the solution obtained in the integration over the level n-1 grid. This kind of procedure is very simple and prevents discontinuities on the overall profiles but tends to introduce significant errors in the solution.

2.2.2 Moving Mesh Method (MMM)

In this method, the subproblems are generated in Stage I and solved by a two step procedure:

1. Conversion of the problem to a moving set of coordinates by the relation:

$$\dot{u} = u_\tau + u_z \cdot \dot{z} \tag{2.5}$$

2. Inclusion of a moving grid differential equation in the transformed problem, that originates the dynamical problem that we want to solve.

The Moving Mesh Equations used here were deduced by Petzold [7]. In this case, the velocities $\dot{z}$ are chosen to minimize the time rate of change of u and z in the new coordinates. The nodal movement is smoothed by the addition of a penalty function to the minimization, which attempts to give neighbouring nodes nearly equal velocities. Thus $\dot{z}$ for the node j satisfies:

$$\min_{\dot{z}_j} \left[\|\dot{u}_j\|_2^2 + \alpha \cdot \|\dot{z}_j\|_2^2 + \lambda \cdot \left(\left\| \frac{\dot{z}_j - \dot{z}_{j-1}}{z_j - z_{j-1}} \right\|_2^2 + \left\| \frac{\dot{z}_{j+1} - \dot{z}_j}{z_{j+1} - z_j} \right\|_2^2 \right) \right] \tag{2.6}$$

The quadratic equation in $\dot{z}$ can be minimized in each mesh point. Therefore, for $\lambda > 0$, (2.6) leads to

$$\alpha \cdot \dot{z}_j + \dot{u}_j \bullet u_{zj} + \lambda \cdot \left(\frac{\dot{z}_j - \dot{z}_{j-1}}{(z_j - z_{j-1})^2} - \frac{\dot{z}_{j+1} - \dot{z}_j}{(z_{j+1} - z_j)^2} \right) = 0 \qquad (2.7)$$

Here, α is a positive scaling parameter, usually set to 1. The effect of the penalty term is similar to an extra diffusional factor that smoothes out differences in the mesh velocities and tries to keep points from crossing.

It is introduced an adjustement of the time step to prevent node crossings and a final redefinition strategy of the level 2 base grid that locally refines the intervals where the spatial step exceeds a predetermined value: $\Delta z > \Delta z_{MAX}$, by equidistributing two additional nodes, or moves away nodes that get too close from one another: $\Delta z < \Delta z_{MIN}$.

Additionally, it is adopted a procedure that allows a semi-free evolution of the solution on the internal subproblems' boundaries, which is only constricted by the spatial derivatives estimation operation that uses the time evolution of the solution on the externally adjacent nodes to the subdomains, computed on the static integration step of the original problem. The temporal profiles are approximated by linear interpolation. This procedure does not garantee the continuity of the overall profile. So, this is actually an iterative procedure, where convergence is only reached when a specific tolerance is verified on both boundaries of each dynamic subproblem.

The algorithms briefly described in this section are resumed in Figure 2.1.

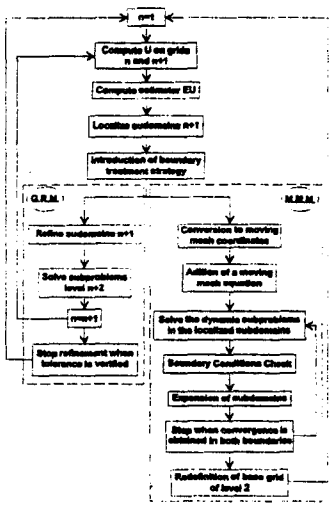

Figure 2.1: *Schematic resume of the adaptive numerical algorithms.*

3 Numerical Results

The two adaptive algorithms described in the previous section were applied to the solution of a parabolic/hyperbolic system of two P.D.E.'s. The quality of the results, defined by the profiles precision and the computational effort demanded, is established by comparison with the results obtained with a formulation of the Moving Finite Elements Method (MFEM) developed

by Duarte [3], based on Hermite polynomials approximations. The software designed for the application of every algorithm analised in this work, was executed in the same computer, a *Workstation* **SUN** *Sparcstation* of architecture **RISC** with 16 Mb of **RAM** memory.

3.1 *Example: Non-Isothermal Tubular Reactor*

This model [3] simulates the startup of an non-isothermal catalitic tubular reactor, subjected to a step disturbance in the reagent concentration of the feed stream. The reagent is consumed in a first order reaction A $\rightarrow$ P and it is assumed that the influence of axial diffusion can not be neglected. Thus, the problem is the following microscopic mass balance:

$$\frac{\delta u}{\delta t} = \frac{1}{Pe} \cdot \frac{\delta^2 u}{\delta z^2} - \frac{\delta u}{\delta z} - Da \cdot u \cdot v \cdot \epsilon^{-\gamma \cdot (\frac{1}{v} - 1)} \tag{3.1}$$

$$\frac{\delta v}{\delta t} = -\frac{\tau}{\tau_{hl}} \cdot \frac{\delta v}{\delta z} + \frac{\tau}{\tau_{hl}} \cdot \beta \cdot Da \cdot u \cdot v \cdot \epsilon^{-\gamma \cdot (\frac{1}{v} - 1)} - \frac{\tau}{\tau_{hl}} \cdot N_{wh} \cdot (v - v_w) \tag{3.2}$$

with the boundary conditions: $\frac{\delta u(0,t)}{\delta z} = Pe \cdot (u - 1)$, $\frac{\delta u(1,t)}{\delta z} = 0$ and $v(0,t) = 1$,
and the initial conditions: $u(z,0) = 0$ and $v(z,0) = 1$.

Here, u and v are the concentration of component A and the fluid's temperature, respectively, normalized in relation to the feeding stream conditions; v_w is the normalized temperature of the cooling fluid in relation to its initial temperature; $Pe = 10^4$ and $Da = 0.7$ are the Peclet and Damkhöler adimensional numbers, respectively; $\gamma = 21.8$ is the Arrhenius number; $\beta = 0.7$ is the adimensional adiabatic rising of the temperature; $N_{wh} = 33.7$ is the number of heat transfer units on the wall of the reactor; and $\frac{\tau}{\tau_{hl}} = 2.08 \times 10^{-4}$ is the relation between the propagation speeds of the massic and thermic waves, respectively. All these values were taken from [3].

3.1.1 Grid Refinement Method

The example was solved by the Grid Refinement Method on a temporal domain divided in three zones: Zone 1 - $t \in [0, 1.0[$, Zone 2 - $t \in [1.0, 100[$ and Zone 3 - $t \in [100, 1000]$; under the following conditions and parameters: biased upwind five points finite difference formulas for the spatial discretization on both variables (u and v) that ensure the correct reproduction of the positive movement of the fronts, specially the massic wave; tolerances - 0.005 in Zones 1 and 2 and 0.01 in Zone 3 for variable u and 0.001 in Zones 1 and 3 and 0.005 in Zone 2 for variable v; linear interpolations; and an uniform first level base grid with 31 nodes which implies that $N_{MAX} = 9$.

The application of the GRM to this example, with the above conditions, originated the results presented in Figures 3.1–3.4. Initially, during the propagation period of the massic wave ($t < 1.0$) the method reproduces reasonably (vd. Figure 3.1) the abrupt massic front introduced at the reactor's feeding section ($z = 0$) by the initial step disturbance in the concentration variable (u) defined on that boundary. The influence of convective mass displacement is dominant ($Pe = 10^4$), thus the moving fronts tend to be very steep. We can also notice the infuence of slight numerical dissipation on the solution profiles because the fronts thickness is larger than expected. During this period, the variation of the temperature profiles is very low, because the thermal wave propagates with a much less speed than the massic one (vd. Figure 3.2). After $t = 1.0$ the massic wave crashes with the right boundary of the spatial domain, and the massic profiles become very smooth (vd. Figure 3.3) and slowly decrease until they stabilize at the final steady-state ($t \approx 1000$). For larger values of time, the gradients on the temperature profiles become more noticeable (vd. Figure 3.4). The hotspot moves slowly through the reactor until it stabilizes

when the system reaches the final steady state ($t \approx 1000$).

The refinement procedure demanded by the problem is presented in Figures 3.5 and 3.6. Initially, the method simply reveals some difficulties in dealing with the disturbance introduced at the reactor's entrance (vd. Figure 3.5) by the left boundary condition ($z = 0$). After the propagation period of the massic front through the reactor (for $t > 1.0$), the method only shows some activity near the reactors exit (vd. Figure 3.6), where it has to deal with the Neumann boundary condition defined at that position ($z = 1$).

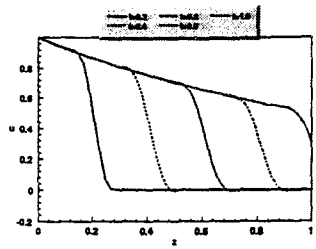

Figure 3.1: *Concentration profiles (Tfinal = 1).*

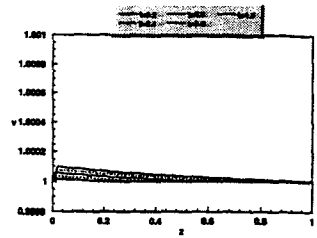

Figure 3.2: *Temperature profiles (Tfinal = 1).*

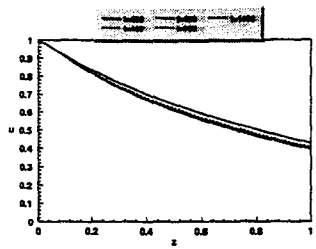

Figure 3.3: *Concentration profiles (Tfinal = 1000).*

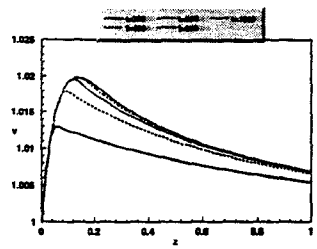

Figure 3.4: *Temperature profiles (Tfinal = 1000).*

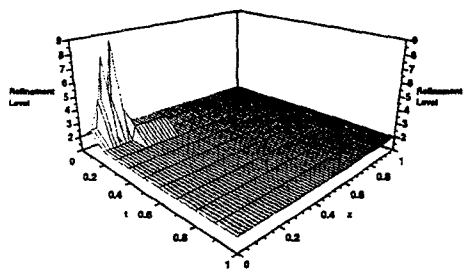

Figure 3.5: *Refinement distribution (Tfinal = 1).*

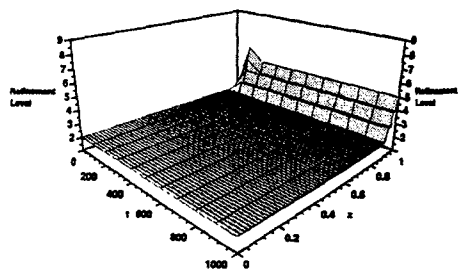

Figure 3.6: *Refinement distribution (Tfinal = 1000).*

3.1.2 Moving Mesh Method

The results obtained by the application of the MMM on a temporal domain partitioned in four subintervals defined by: Zone 1 - $t \in [0, 1.0[$, Zone 2 - $t \in [1.0, 10[$, Zone 3 - $t \in [10, 100[$ and Zone 4 - $t \in [100, 1000]$; with the following conditions: five points biased upwind discretizations for both variables; absolute tolerances - 1×10^{-4} in zones 1–3 and 5×10^{-4} in Zone 4, for variable u and 5×10^{-4} in Zone 1, 1×10^{-4} in Zone 2, 5×10^{-5} in Zone 3 and 1×10^{-5} in Zone 4, for variable v; initial base grid nonuniform with 20 nodes, mainly concentrated near the left boundary; linear type interpolations; $\Delta z_{MIN} = 1 \times 10^{-5}$ and $\Delta z_{MAX} = 2.3 \times 10^{-2}$; and internodal viscosity factor $\lambda = 0.75$, are depicted in Figures 3.7–3.12.

The algorithm MMM reproduces correctly the steep massic fronts and their movement during the massic wave propagation period (vd. Figure 3.7). We notice some slight oscillations for earlier times that clearly disappear afterwards. The temperature profiles evolution is satisfatory and does not represent any additional difficulties in the advance of the temporal integration (vd. Figure 3.8). After the massic propagation period ($t > 1.0$), the results obtained for both variables are very similar to the numerical profiles computed by the GRM algorithm (vd. Figures 3.9 and 3.10). During the initial subinterval ($t < 1.0$), the grid associated with the concentration variable (u) shows significant activity (vd. Figure 3.11) and the nodal movement follows the massic wave path through the reactor. On the other hand, the grid coupled with the temperature variable (v) remains static (vd. Figure 3.12), because the temporal gradients of this variable are very small. After the initial period (for $t > 1.0$) both grids hardly show any activity, due to the smoothness of the profiles.

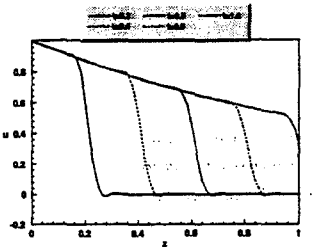

Figure 3.7: *Concentration profiles (Tfinal = 1).*

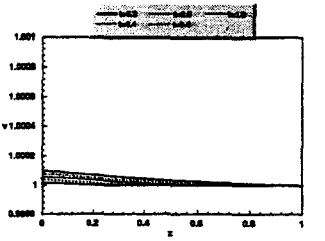

Figure 3.8: *Temperature profiles (Tfinal = 1).*

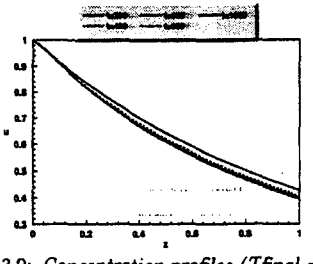

Figure 3.9: *Concentration profiles (Tfinal = 1000).*

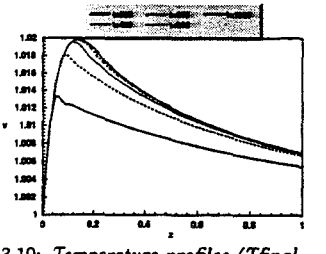

Figure 3.10: *Temperature profiles (Tfinal = 1000).*

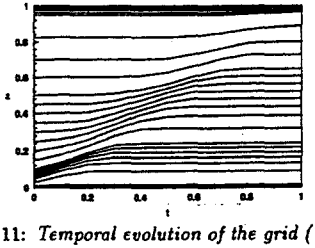

Figure 3.11: *Temporal evolution of the grid (Tf = 1 - u).*

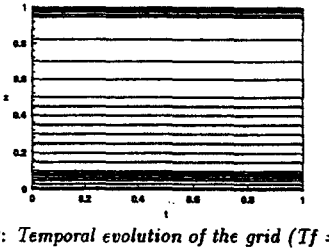

Figure 3.12: *Temporal evolution of the grid (Tf = 1 - v).*

3.1.3 Comparison of the Numerical Performance Obtained by Each Algorithm

In Table 3.1, we compare the perfomance of the developed algorthms with the numerical behaviour of the MFEM formulation presented by [3].

The GRM demands a reasonably large computational effort in dealing with the presented example, and still introduces some numerical instability on the numerical results. The solution

profiles obtained by the MFEM and MMM algorithms are very similar. However, MMM reveals to be more effective in this case, because it exhibits lower computational time.

Table 3.1: *Computational performances for the example.*

Method	$Tcpu(s)$
G.R.M.	19180.2
M.M.M.	8711.9
M.F.E.M. [3]	10436.2

4 Conclusions

From the solution of the example presented in this work, we can conclude that the GRM algorithm reveals some difficulties in describing high gradient profiles and it may develop numerical dissipation, mainly due to the simplicity and imprecision of the Dirichlet boundary treatment strategy for the refinement subproblems. On the other hand, it can be shown that GRM is a very efficient method for models that involve relatively smooth profiles [1].

The MMM algorithm is suitable to reproduce moving abrupt fronts or waves. The results obtained are very exact with hardly any numerical instability. The boundary condition procedure coupled with the MMM algorithm, based on linear interpolations on time for the nodes near the subdomain boundaries proved to be very effective and reliable.

As it was expected, linear interpolations are the most adequate to deal with abrupt fronts characterized by large spatial variations, on the solution profiles.

Acknowledgment: This work was supported by a grant from JNICT (Junta Nacional de Investigação Científica e Tecnológica).

References

[1] P. M. P. Brito, "Aplicação de Métodos Numéricos Adaptativos na Integração de Sistemas Algébrico-Diferenciais Caracterizados por Frentes Abruptas". *MSc Thesis*, Universidade de Coimbra, F. C. T., Coimbra, 1998.

[2] P. M. P. Brito, A. A. T. G. Portugal, "Application of Adaptive Methods Based on Finite Difference Discretizations to Systems of PDAE's", *Proc. Numerical Methods for Differential Equations - 1998*, Coimbra - in the press.

[3] B. P. M. Duarte, "Método dos Elementos Finitos Móveis Aplicado à Resolução de Modelos de Frente de Reacção", *PhD Thesis*, Universidade de Coimbra, F. C. T., Coimbra, 1994.

[4] B. Fornberg, "Generation of Finite Difference Formulas on Arbitrarily Spaced Grids", *Math. of Comput.* **51** (1988) 699–706.

[5] R. P. F. Guiné, "Resolução de Sistemas de Equações Diferenciais de Derivadas Parciais — Algoritmos de Refinamento Espacial", *MSc Thesis*, Universidade de Coimbra, F. C. T., Coimbra, 1996.

[6] L. R. Petzold, "A Description of DASSL: A Differential/Algebraic System Solver", Sandia Tech. Rep. 82-8637, 1982.

[7] L. R. Petzold, "Observations on an Adaptive Moving Grid Method for One-Dimensional Systems of Partial Differential Equations", *Appl. Numer. Math.* **3** (1987) 347–360.

[8] W. E. Schiesser. *The Numerical Method of Lines: Integration of Partial Differential Equations*, Academic Press Inc., San Diego, 1991.