ELSEVIER

Chemical Engineering Journal 84 (2001) 23–29

Chemical Engineering Journal

www.elsevier.com/locate/cej

Applications of a moving finite element method

Maria do Carmo Coimbra [a,1], Carlos Sereno [b], Alírio Rodrigues [a,*]

[a] *Laboratory of Separation and Reaction Engineering, School of Engineering, University of Porto, Rua dos Bragas 4050-123 Porto, Portugal*
[b] *Departamento de Matemática e Informática, Universidade da Beira Interior, 6200 Covilhã, Portugal*

Received 17 January 2000; received in revised form 21 August 2000; accepted 23 August 2000

Abstract

The moving finite element method (MFEM) with polynomial approximations of any degree is applied to a variety of models described by partial differential equations (PDEs) of the type $\mathbf{G}\mathbf{u}_t = \mathbf{F}\mathbf{u}_{xx} + \mathbf{H}$, $a \le x \le b$, $t \ge 0$, $\mathbf{G}$ and $\mathbf{F}$ are full matrices. The objective of this work is to show that the proposed formulation of MFEM is a powerful tool to compute the numerical solution of time-dependent PDEs involving steep moving fronts. A strategy to choose the penalty constants was devised in relation with the ODE solver tolerances to improve the robustness of the method. Numerical results concerning combustion model, boundary layer problem, catalytic reactor and pressurization of adsorption beds illustrate the effectiveness of our scheme.

Keywords: Moving finite elements; Polynomial approximations; Partial differential equations; Catalytic reactor; Pressure swing adsorption

1. Introduction

In this paper, we propose extensions of the moving finite element method (MFEM) formulated by Sereno [1]. The present extension allows us to use the MFEM to solve a large class of time-dependent partial differential equations (PDEs). Chemical engineering models often lead to systems of PDEs whose solution contains sharp profiles in certain regions of the domain, moving with different velocities. When the classical finite element method is applied to problems of this type, a dense space grid is required to eliminate oscillations. The moving finite elements, originally developed by Miller and Miller [2,3], avoids this problem allowing the movement of the spatial grid. The positions of spatial nodes are unknowns and the minimization process places the nodes where they are needed. Our numerical code has been developed for one-dimensional spatial systems of time-dependent parabolic PDEs. Solutions are calculated using Galerkin's method with a piecewise polynomial basis in space. In each element, the solution is approached by a polynomial of arbitrary degree. Lagrange polynomials are used to define the basis functions. These basis functions are themselves time-dependent through the time dependence of the nodal position. We use a different space grid for each of the dependent variables. As it is known, MFEM has intrinsic singularities. To avoid them, we add penalty functions to the objective function. Our aim in this paper is to show that our formulation of MFEM is a powerful instrument to compute the numerical solution of a variety of difficult time-dependent PDEs involving fine scale phenomena such as steep moving fronts. In fact, one of the most significant features of our MFEM is that the use of polynomial approximations of arbitrary degree enables us to get numerical solutions of great precision with few nodes in each spatial grid. Numerical results are given to illustrate the effectiveness of our scheme. We present the results of some numerical experiments on reaction–diffusion equations, convection–diffusion equations and pseudo-homogeneous axial dispersion model of a non-isothermal tubular catalytic reactor. Finally, we apply the MFEM to the simulation of the pressurization of adsorption beds with a binary mixture of adsorbable gases.

2. The MFEM

Our formulation of MFEM has been designed to solve systems of PDE,

$$\mathbf{G}\mathbf{u}_t = \mathbf{F}\mathbf{u}_{xx} + \mathbf{H}, \qquad a \le x \le b, \quad t \ge 0, \tag{1}$$

where the solution $\mathbf{u}$ may be a vector, $\mathbf{u} = [u_1, \ldots, u_n]$ and the mth equation fits in the following master equation:

$$\sum_{j=1}^{n} g_{m,j}(x,t,\mathbf{u})\frac{\partial u_j}{\partial t} = \sum_{j=1}^{n} f_{m,j}\left(x,t,\mathbf{u},\frac{\partial \mathbf{u}}{\partial x}\right)\frac{\partial^2 u_j}{\partial x^2} + h_m\left(x,t,\mathbf{u},\frac{\partial \mathbf{u}}{\partial x}\right). \tag{2}$$

* Corresponding author. Tel.: +351-2204-1671; fax: +351-2204-1674.
E-mail address: arodrig@fe.up.pt (A. Rodrigues).
[1] CEDEC-FEUP.

1385-8947/01/$ – see front matter
PII: S1385-8947(00)00266-7

Nomenclature

a, b	left and right end points of the space domain
c_i	penalty constants, $i = 1, 2, 3, 4$
$\ell_i^{m,\mathrm{L}}(x)$	ith Lagrange basis function
M_m	number of nodes in Δ_m
$N_{m,\mathrm{L}}$	number of interpolation points in $[X_{m,j}, X_{m,j+1}]$
R_m	PDE residual associated with the mth equation
$S_{m,j}$	internodal spring function
t	time independent variable
$\mathrm{tol}_1, \mathrm{tol}_2$	ODE solver tolerances
$\mathbf{u}(x,t)$	solution of the system of PDE, $\mathbf{u}(x,t) = [u_1(x,t), \ldots, u_n(x,t)]$
$\mathbf{U}$	numerical approximation of $\mathbf{u}$
$U_m(\Delta_{m,i})$	value of $U_m(t)$ at the ith node of Δ_m
$U_{m,j}(t)$	polynomial approximation to u_m in $[X_{m,j}, X_{m,j+1}]$
$U_{m,j}^i$	value of U_m in the ith interpolation point of jth finite element of the grid Π_m
$\dot{U}_{m,j}^i$	time derivatives of $U_{m,j}^i$
$v_i^{m,\mathrm{L}}$	relative positions of interpolation points
V_δ	a neighbourhood of a spacial node
x	space independent variable
$X_{m,j}$	left end point of the jth finite element of the grid Π_m
$\dot{X}_{m,j}$	time derivatives of $X_{m,j}$
$Z_{m,j}$	length of the jth element of grid Π_m

Greek symbols

$\varepsilon_{m,j}$	internodal viscosity function
$\varphi_{m,i}(x)$	global basis function
Δ_m	ordered set of all spatial and interpolation nodes
Π_m	space grid associated to u_m

We consider Dirichlet, Neumann or Robin boundary conditions and initial conditions satisfying

$$\mathbf{u}(x,0) = \mathbf{u}_0(x), \quad a \le x \le b. \tag{3}$$

In the work by Sereno [1], **G** and **F** in Eq. (1) are identity and diagonal matrices, respectively. In this work the MFEM with polynomial approximation of arbitrary degree is extended to allow full matrices for **G** and **F**. The complete discretization of (1) is obtained in two stages. We start with the discretization of space variables. Notice that every dependent variable has its own spatial grid. This semi-discrete procedure, in which we focus our attention, generate a system of ordinary differential equations (ODEs). In order to define the numerical approximation **U** to **u** and the new positions of the nodes, it is necessary to integrate the system of ODE. For that purpose, we used the package LSODI developed at the Lawrence Livermore National Laboratory [4]. Let

$$\Pi_m : a = X_{m,1} \le X_{m,2} \le \cdots \le X_{m,q_m} \le X_{m,q_m+1} = b \tag{4}$$

be the grid associated to u_m. In the jth finite element of the grid Π_m, we approximate u_m by an $N_{m,j} - 1$ polynomial obtained using Lagrange basis functions. The positions of the $N_{m,\mathrm{L}}$ points of interpolation in $[X_{m,j}, X_{m,j+1}]$ are optimized as in the orthogonal collocation method [1,5]. We define the polynomial approximation $U_{m,j}(t)$ as

$$U_{m,j}(t) = \sum_{i=1}^{N_{m,j}} \ell_i^{m,j}(x) U_{m,j}^i(t), \quad X_{m,j} \le x \le X_{m,j+1}, \tag{5}$$

where $\ell_i^{m,\mathrm{L}}(x)$ is the ith Lagrange basis function, $v_i^{m,\mathrm{L}}$ the relative positions of interpolations points and $U_{m,j}^i$ is the value of U_m in the ith interpolation point of jth finite element of the grid Π_m. The first and second spatial derivatives of $U_{m,j}(t)$ are also polynomials and they can be defined using the same interpolation points. The approximation $U_m(t)$ to $u_m(t)$ in $[a, b]$ is the continuous piecewise polynomial function $U_m(t)|_{[X_{m,j}, X_{m,j+1}]} = U_{m,j}(t)$. If we assume that Δ_m is the ordered set of all nodes, spatial nodes and points of interpolation, associated to Π_m, we can write

$$U_m(t) = \sum_{i=1}^{M_m} \varphi_{m,i}(x) U_m(\Delta_{m,i}), \quad a \le x \le b, \tag{6}$$

where $U_m(\Delta_{m,i})$ is the value of $U_m(t)$ at the ith node of Δ_m, M_m the number of nodes in Δ_m and $\varphi_{m,i}(x)$ is defined as

$$\varphi_{m,i}(\Delta_{m,i}) = \begin{cases} 0 & \text{if } i \ne j, \\ 1 & \text{if } i = j. \end{cases} \tag{7}$$

Therefore, the approximation $U_m(t)$ is a continuous piecewise polynomial function dependent on the nodal amplitudes $U_{m,j}^i$ and on the nodal position $X_{m,j}$. We obtain ODEs for the nodal amplitudes and the nodes positions by requiring that their time derivatives, $\dot{U}_{m,j}^i$ and $\dot{X}_{m,j}$, are chosen, at each instant so that

$$\sum_{m=1}^{n} \left[\int_a^b (R_m)^2 \right] \tag{8}$$

be minimized. The PDE residual associated with the mth equation is

$$R_m = \sum_{j=1}^{n} g_{m,j}(x,t,\mathbf{U}) \frac{\partial U_j}{\partial t} - \sum_{j=1}^{n} f_{m,j} \times \left(x, t, \mathbf{U}, \frac{\partial \mathbf{U}}{\partial x}\right) \frac{\partial^2 U_j}{\partial x^2} - h_m\left(x, t, \mathbf{U}, \frac{\partial \mathbf{U}}{\partial x}\right). \tag{9}$$

Observe that we have two significant problems to solve. The first one is due to the use of a continuous piecewise polynomial approximation of arbitrary degree. This kind

of approximation leads us to define a strategy to the treatment of second order terms. We interpreted second order terms in the sense of smoothing. In a neighbourhood V_δ of a node, where the second order terms are not defined, we smooth off our numerical approximation. By this process, the integrals involving second order terms exist and have consistent limits as δ tends to zero, independently of the smoothing process. We use a smoothing process based on Hermite polynomial approximation near a node. After that we use numerical quadrature to compute the integrals of the smooth approximation, in V_δ. This process avoids the use of a catalogue of integrals for standard forms of PDEs. The second one appears because there are intrinsic singularities, due to parallelism and element folding. To keep away from this problem, we introduce Miller's penalty functions into the minimization process. We add the expression

$$\sum_{m=1}^{n}\sum_{j=1}^{q_m}\varepsilon_{m,j}\frac{\partial Z_{m,j}}{\partial t} - S_{m,j} \tag{10}$$

to the objective function (8). The length of the jth element of grid Π_m is $Z_{m,j} = X_{m,j+1} - X_{m,j}$ and we use the internodal viscosity function and the internodal spring function

$$\varepsilon_{m,j} = \left(\frac{c_2}{Z_{m,j} - c_4} + c_3\right)\left(1 + \frac{c_4}{Z_{m,j} - c_4}\right)^2, \tag{11}$$

$$S_{m,j} = \frac{c_1}{Z_{m,j} - c_4}\left(1 + \frac{c_4}{Z_{m,j} - c_4}\right)^2, \tag{12}$$

where c_i are constants supplied by the user. The user may choose these constants for each grid and for each element. Penalty functions do not interfere on the solution, but exclusively on the movement of the nodes in order to prevent singularities. Their disadvantage is that it is not possible to set up a relation between them and the problem we are solving. So, if the nodes do not move in the "proper" way, the time of computation becomes rather large since the error control of the position of the spatial nodes forces a very frequent updating of the Jacobian. Some numerical experimentation is required in order to obtain an adequate set of constants. Note that the choice of large values for c_2 and c_3 will restrict the nodes movement and if we choose different c_1 values for each element of the grid the smaller one will correspond to the element that can reach the smaller length. c_4 is the minimum permissible cell width. For a summary of various MFEM, we address the reader to the book by Baines [6].

3. Numerical examples

We present a few numerical examples to illustrate the working and performance of our MFEM. All the numerical results presented here are obtained on a Pentium II processor at 266 MHz. Nodes are initially concentrated near $x = a$. The minimum permissible cell width is $c_4 = 10^{-5}$. The ODE solver tolerances, for nodal amplitudes tol_1 and for nodal position tol_2, are $\mathrm{tol}_1 = \mathrm{tol}_2 = \mathrm{tol} = 10^{-3}$. In order to improve the robustness of MFEM, we present a strategy to choose the penalty constants. In practice, we choose the values for constants c_1, c_2 and c_3 in relation with the ODE solver tolerances. For that standard choice of tol, we set $c_1 = 10^{-2} \times \mathrm{tol}$, $c_2 = \mathrm{tol}$ and $c_3 = 10 \times \mathrm{tol}$. We use Gaussian quadrature with six interior quadrature points to compute the integrals appearing in each one of the equation of the ODE systems.

3.1. A scalar combustion model

The first example is a scalar reaction–diffusion equation from combustion theory described in [7–9],

$$\frac{\partial T}{\partial t} = \frac{\partial^2 T}{\partial x^2} + D(1 + \alpha - T)\,\mathrm{e}^{-\delta/T},$$
$$0 \le x \le 1, \quad t \ge 0, \tag{13}$$

$$\frac{\partial T}{\partial x}(0, t) = 0, \qquad T(1, t) = 1, \quad t \ge 0,$$
$$T(x, 0) = 1, \quad 0 \le x \le 1, \tag{14}$$

where $D = R\,\mathrm{e}^{\delta}/\alpha\delta$ and R, α, δ are constants. It is well known that two phases can be distinguished in the solution: the formation of the front, ignition phase, and its propagation to the right end point, the propagation phase. The temperature of a reactant in a chemical system, T, initially at 1, increases gradually up to a maximum at $x = 0$. At a finite time ignition occurs and the temperature at $x = 0$ increases rapidly to $1 + \alpha$. A steep front then forms and propagates towards $x = 1$. The difficulty level of the problem is very much depending on the value of δ. We use this problem as a test example with a difficult parameter choice $\delta = 30$, $R = 5$ and $\alpha = 1$. We compute the solution with five finite elements and a polynomial approximation of degree 5 in each element, on a time interval from $t = 0.0$ to $t = 3.0$. For this problem CPU time is 2.9 s. Fig. 1 presents the temperature profiles.

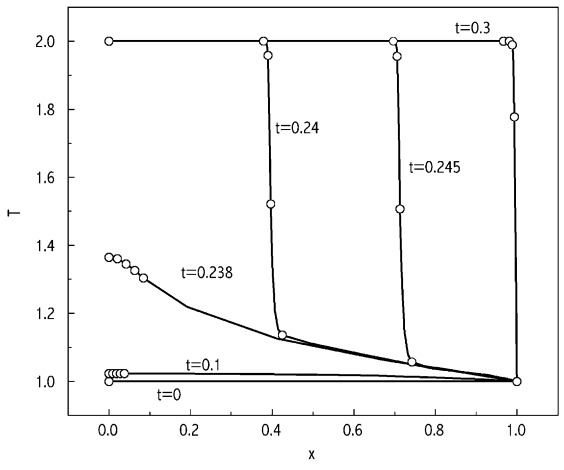

Fig. 1. Scalar combustion, temperature profiles for $\delta = 30$.

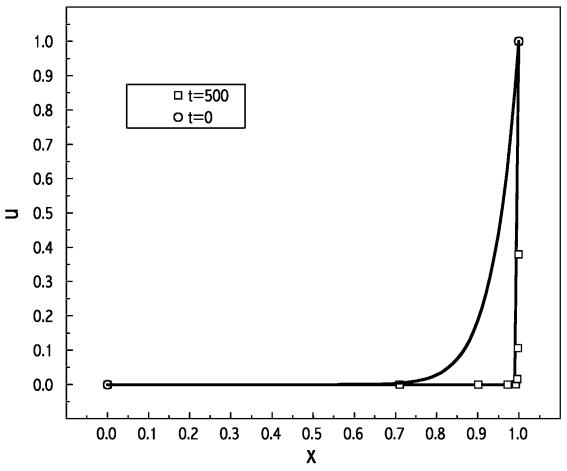

Fig. 2. The boundary layer problem.

The solution is shown for $t = 0, 0.1, 0.238, 0.24, 0.245$ and 0.3. The numerical solution is accurate and when compared with other methods, MFEM is quite advantageous, allowing the use of few nodes, without losing the quality of the solution.

3.2. A boundary layer problem

The second problem that we studied is a boundary layer problem. It is a convection–diffusion equation described in [10],

$$\frac{\partial u}{\partial t} = -\frac{\partial u}{\partial x} + \nu \frac{\partial^2 u}{\partial x^2}, \qquad 0 \le x \le 1, \quad t \ge 0, \tag{15}$$

$$u(0,t) = 0, \qquad u(1,t) = 1, \quad t \ge 0, \tag{16}$$

$$u(x,0) = x^{16}, \quad 0 \le x \le 1 \tag{17}$$

with diffusion coefficient $\nu = 10^{-3}$, $\nu = 10^{-4}$. We solve this problem for $0 \le t \le 500$ with eight finite elements and approximation polynomials of degree 3 in each element. The CPU time, in seconds, is 3.5 for $\nu = 10^{-3}$ and 3.7 for $\nu = 10^{-4}$. Once more the MFEM gives a solution of great quality with few nodes as it is shown in Fig. 2. We presents curves solution for $t = 0$ and 500 and nodes position at these instants. Notice that for $t = 0$ nodes are concentrated near $x = 0$.

3.3. A non-isothermal tubular catalytic reactor

Our third example arising from chemical engineering is described in [11,12]. This model simulates a tubular catalytic reactor through the pseudo-homogeneous model with axial dispersion in the fluid concentration. The model equations, a mass balance equation and a energy balance equation, are

$$\frac{\partial c}{\partial t} = \frac{1}{Pe}\frac{\partial^2 c}{\partial x^2} - \frac{\partial c}{\partial x} - Da\, cT \exp(-\gamma(T^{-1} - 1)), \tag{18}$$

$$\frac{\partial T}{\partial t} = \frac{\tau}{\tau_{hl}}\left[-\frac{\partial T}{\partial x} + \beta\, Da\, cT \times \exp(-\gamma(T^{-1} - 1)) - N(T - 1)\right], \tag{19}$$

$0 \le x \le 1$ and $t \ge 0$. c and T are the fluid concentration and the fluid temperature normalized by feed concentration and temperature, respectively. x is the space variable normalized by the length of the bed and t is the time variable normalized by space–time τ. Pe is the Peclet number, Da the Damköhler number, γ the Arrhenius number, β the adiabatic temperature rise, τ_{hl} the time constant for thermal wave propagation and N the number of transfer units for the heat transfer at the wall. Initial and boundary conditions are

$$u(x,0) = 0, \qquad v(x,0) = 1, \quad 0 \le x \le 1, \tag{20}$$

$$\frac{\partial u}{\partial x}(0,t) = Pe(u-1), \qquad \frac{\partial u}{\partial x}(1,t) = 0, \qquad v(0,t) = 1, \quad t \ge 0. \tag{21}$$

The integration time interval is $0 \le t \le 1500$ and for spatial discretization, we use eight finite elements for each grid and polynomial approximation of degree 6 in each element. We use the same penalty constants except for the grid associated to the temperature, where the value of c_1 is 10^{-4} for the four last elements. Fig. 3(a) presents the concentration and Fig. 3(b) shows the temperature profiles. MFEM gives solutions of great accuracy. At short time, concentration profiles contain steep fronts that moves to the right. In Fig. 4(a) and (b) the concentration and temperature histories, respectively, are represented. The CPU time needed to complete the integration is 39.6 s. In this problem, we consider $\tau/\tau_{hl} = 2.08 \times 10^{-4}$, $\gamma = 21.8$, $N = 33.7$, $Pe = 10^4$ and $D = \beta = 0.7$.

3.4. Pressurization of adsorption beds

Finally, we present an example arising from pressurization swing adsorption (PSA) process. In [13], we presented a simplified model for pressurization in a PSA process and simulated the pressurization of adsorption beds with a mixture of an inert and a active species, helium and methane. Here, we are going to apply that model to the pressurization of adsorption beds with a binary mixtures of adsorbable species. Modelling of pressurization involves mass and mechanical energy balance equations. Langmuir equilibrium between adsorbed phase concentration and gas phase concentration inside pores for active species is assumed. The system considered here consists of an adsorption bed of length L and cross-section area S. At the top and the bottom there are two dead volumes, with the same cross-section area, S, and length L_1 and L_2, respectively. The bed porosity is ε and total porosity is ε_t. The bed is packed with adsorbent particles of diameter D_p and the apparent density of the adsorbent is ρ_{ap}.

The model is based on the following assumptions: equilibrium model, isothermal operation and ideal gas behaviour.

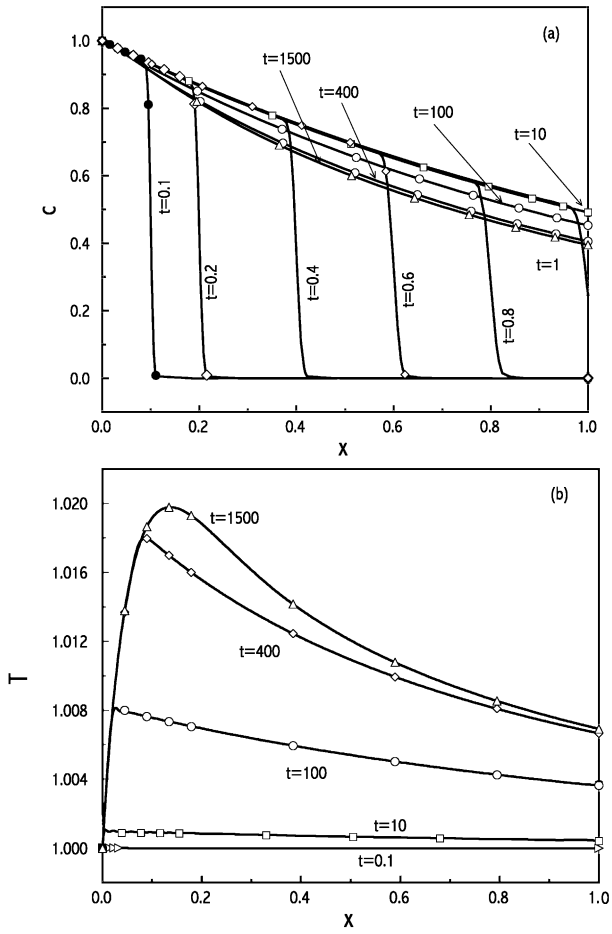

Fig. 3. (a) Non-isothermal tubular catalytic reactor, axial profiles for concentration. (b) Non-isothermal tubular catalytic reactor, axial profiles for temperature.

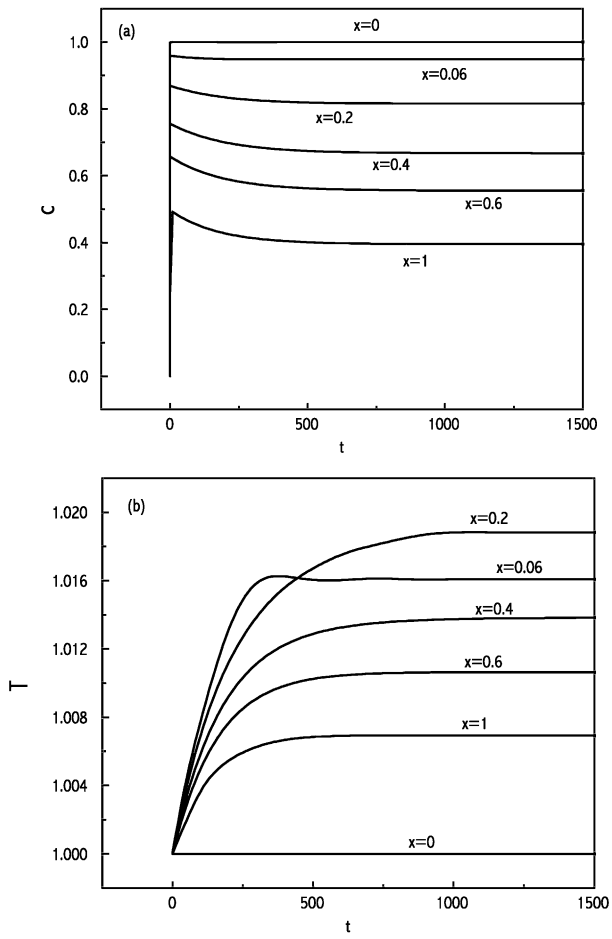

Fig. 4. (a) Non-isothermal tubular catalytic reactor, histories for concentration. (b) Non-isothermal tubular catalytic reactor, histories for temperature.

We also suppose that the subsonic flow and pressure drop versus flow condition relationships are instantaneously established locally in the bed. Equilibrium relations are represented by Langmuir isotherms,

$$q_i^* = \frac{Q_i k_i c_i^*}{1 + k_A c_A^* + k_B c_B^*}, \quad i = A, B, \tag{22}$$

where c_i^* is the gas phase concentration of species i, q_i^* is the adsorbed phase concentration of species i. k_i is the equilibrium constant for component i and Q_i the corresponding capacity. Initially the total pressure of the system, p^* is $p^* = p_l^*$ and the mole fraction of species A, y_A, is $y_A = y_{A_0}$. At $t = 0$ the bed is fed with a mixture of oxygen and nitrogen at a pressure $p^* = p_h^*$ through a valve with section S_1 and flow coefficient C_v. The mathematical model involving a mass balance to the species A, an overall mass balance and a mechanical energy balance is described by

$$\varepsilon_t \frac{\partial c_A^*}{\partial t} + \rho_{ap}(1-\varepsilon)\frac{\partial q_A^*}{\partial t} = -\varepsilon \frac{\partial}{\partial z}(c_A^* v^*) + \varepsilon D_{ax} \frac{\partial}{\partial z}\left(c^* \frac{M_B}{M} \frac{\partial y_A}{\partial z}\right), \tag{23}$$

$$\varepsilon_t \frac{\partial \rho^*}{\partial t} + M_A \rho_{ap}(1-\varepsilon)\frac{\partial q_A^*}{\partial t} + M_B \rho_{ap}(1-\varepsilon)\frac{\partial q_B^*}{\partial t} = -\varepsilon \frac{\partial}{\partial z}(\rho^* v^*), \tag{24}$$

$$\frac{\partial}{\partial t}(\rho^* v^*) = -\frac{\partial}{\partial z}(\rho^*(v^*)^2) - \frac{\partial p^*}{\partial z} - 150\frac{\mu^* v^*}{D_p^2}\left(\frac{1-\varepsilon}{\varepsilon}\right)^2 + \text{sign}\left(\frac{\partial p^*}{\partial z}\right) 1.75 \frac{\rho^*(v^*)^2}{D_p}\frac{1-\varepsilon}{\varepsilon}, \tag{25}$$

where the independent variables are time t and axial distance z. v^* is the gas velocity, D_{ax} the axial dispersion coefficient, M the molecular mass, M_i the molecular mass of species i, ρ^* the fluid density, μ^* the fluid viscosity of the binary mixture defined by Curtiss and Hirschfelder [14] relation. In the simplified model, we neglect the kinetic energy change and the momentum accumulation terms from Eq. (25) and this enables us to define the mass velocity, $G^* = \rho^* v^*$, in terms of partial pressure of species A, p_A^* and p^*. Let us consider the dimensionless model, where independent variables are x and θ, respectively. Space is normalized by L and time

normalized by $\tau_c = L/v_{\text{ref}}$, where $v_{\text{ref}} = \sqrt{RT/M_{\text{B}}}$, R the ideal gas constant and T the absolute temperature of the system. The dependent variables are p_{A} the normalized partial pressure and normalized total pressure p, both normalized by $p_{\text{ref}} = p_h^* - p_l^*$. The mass velocity, G, is normalized by $G_{\text{ref}} = p_{\text{ref}}/v_{\text{ref}}$. Most of the parameters used in calculations are those used by Mendes [15] and all the variables are conveniently normalized.

Boundary conditions are determined by the condition on dead volumes, so we introduced four extra dependent variables $p_{\text{A}}|_{x=0^+}$, $p|_{x=0^+}$, $p_{\text{A}}|_{x=1^-}$ and $p|_{x=1^-}$. The mass balances in the regions around the top and bottom of the bed define four new ODEs that will be integrated with the system arising from space discretization,

$$\left.\frac{\partial p_{\text{A}}}{\partial\theta}\right|_{x=0^+} = \frac{S_1}{SL_1}\Gamma G_{\text{in}} - \left.\frac{\varepsilon}{L_1}\,\frac{p_{\text{A}}}{\rho}\right|_{x=0^+} G|_{x=0^+} - \frac{\varepsilon}{L_1}\frac{1}{Pe}\left.\frac{\lambda}{\rho}\right|_{x=0^+}, \tag{26}$$

$$\omega\left.\frac{\partial p_{\text{A}}}{\partial\theta}\right|_{x=0^+} + \left.\frac{\partial p}{\partial\theta}\right|_{x=0^+} = \frac{S_1}{SL_1}G_{\text{in}} - \frac{\varepsilon}{L_1}G|_{x=0^+}, \tag{27}$$

$$\left.\frac{\partial p_{\text{A}}}{\partial\theta}\right|_{x=1^-} = \frac{\varepsilon}{L_2}\left.\frac{p_{\text{A}}}{\rho}\right|_{x=1^-} G|_{x=1^-} + \frac{\varepsilon}{L_2}\frac{1}{Pe}\left.\frac{\lambda}{\rho}\right|_{x=1^-}, \tag{28}$$

$$\omega\left.\frac{\partial p_{\text{A}}}{\partial\theta}\right|_{x=1^-} + \left.\frac{\partial p}{\partial\theta}\right|_{x=1^-} = \frac{\varepsilon}{L_2}G|_{x=1^-}, \tag{29}$$

where $\Gamma = y_{\text{A}_{\text{feed}}}/(y_{\text{A}_{\text{feed}}}\omega+1)$ and G_{in}, is the mass velocity at the left end of dead volume on the top. Initially, $\theta < 0$, we have

$$p_{\text{A}} = y_{\text{A}_0}p_l, \qquad p = p_l, \qquad G = 0. \tag{30}$$

At $t = 0$, the feed conditions are

$$p_{\text{A}_{\text{feed}}} = y_{\text{A}_{\text{feed}}}p_{\text{feed}}, \quad p_{\text{feed}} = p_h, \tag{31}$$

so, at the left end of dead volume on the top,

$$p_{\text{in}} = \begin{cases} 0.53p_h & \text{if } p|_{x=0^+} \le 0.53p_h, \\ p|_{x=0^+} & \text{if } p|_{x=0^+} \ge 0.53p_h, \end{cases} \tag{32}$$

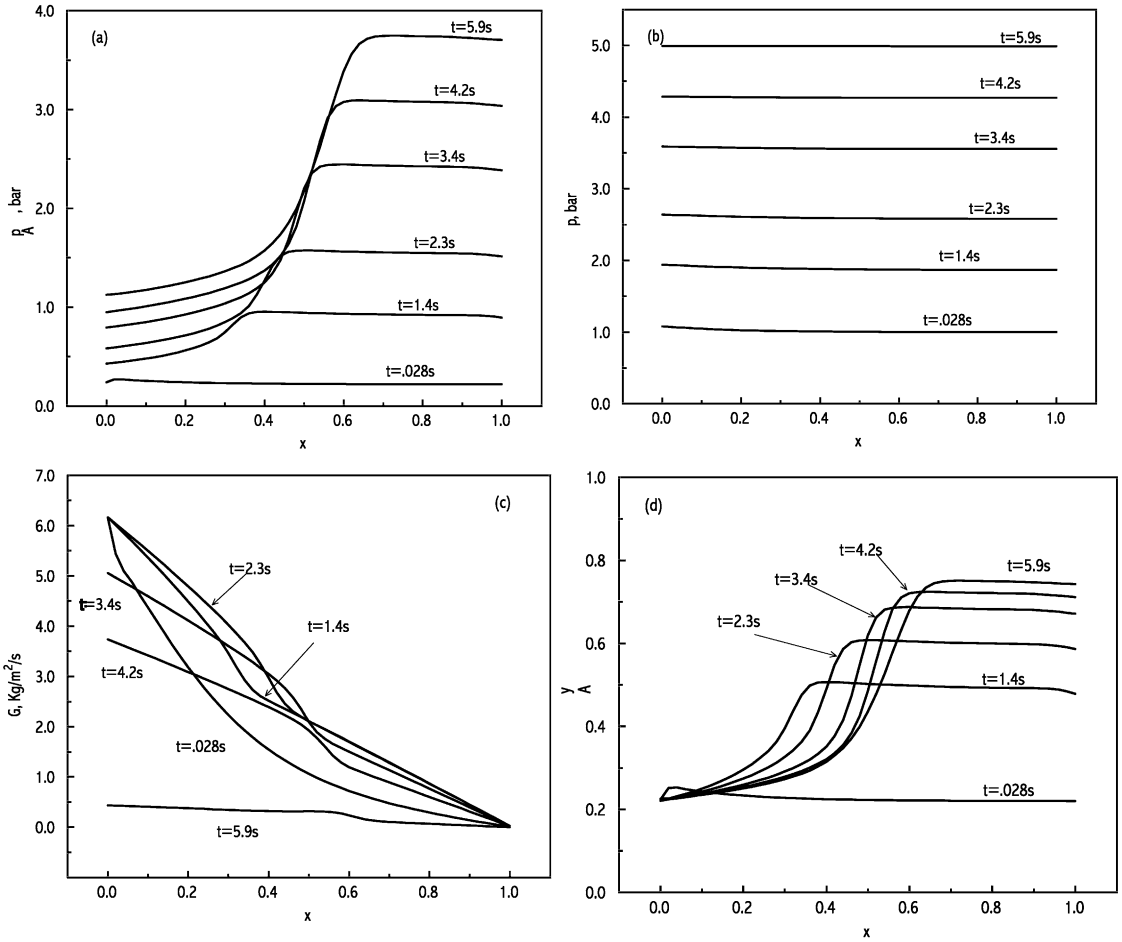

Fig. 5. (a) PSA, axial profiles for partial pressure of oxygen. (b) PSA, axial profiles for total pressure. (c) PSA, axial profiles for mass velocity. (d) PSA, axial profiles for mole fraction of oxygen.

$$p_{A_{in}} = y_{A_{feed}} p_{in}, \tag{33}$$

$$G_{in} = \begin{cases} 1.410 \times 10^{-5} C & \text{if } p|_{x=0^+} \leq 0.53 p_h, \\ 1.663 \times 10^{-5} C \sqrt{1 - \left(\dfrac{p|_{x=0^+}}{p_h} \right)^2} & \text{if } p|_{x=0^+} \geq 0.53 p_h, \end{cases} \tag{34}$$

where

$$C = p_h \frac{C_v}{S_1} y_{A_{feed}} \omega + 1. \tag{35}$$

We consider the integration time interval from $\theta = 0$ to the instant at which $p(1, \theta)$ is very close to p_h. Fig. 5(a)–(d) presents axial profiles of total pressure, partial pressure of species A, mass velocity and mole fraction of species A, respectively, when $y_{A_{feed}} = y_{A_0} = 0.22$, $C_v = 0.05$ and the dispersion parameter $Pe = Lv_{ref}/D_{ax} = 10^4$. We consider eight finite elements in each grid and a polynomial approximation of degree 6 in each element. For this example tolerance of ODE solver must be smaller, $tol_1 = tol_2 = 10^{-5}$. In order to define the penalty functions, we set $c_1 = 10^{-4}$, except for the first element of the grid associated with p_A, where $c_1 = 10^{-6}$, and $c_2 = c_3 = 10^{-3}$. The CPU time to achieve the pressurization is 197.0 s.

4. Conclusions

In this paper, we described MFEM based on polynomial approximations of any degree. It has been shown that MFEM is potentially very efficient to solve difficult problems, specially if we are interested in obtaining solutions that involve sharp moving fronts. As we pointed out, a disadvantage of MFEM is that it may produce a singular matrix. A simple way to avoid singularities is the use of penalty functions. Penalty functions do not interfere on the solution, but experience made clear that, in general, the choice of penalty constants may interact with the amount of computation involved. We present a strategy to choose the empirical penalty constants, establishing a relationship with them and the value of ODE solvers tolerance. This choice seems to be well suited for many problems. Numerical examples show the excellent behaviour of MFEM with polynomial approximations of arbitrary degree. It is important to stress that solutions of high precision are obtained efficiently using a space discretization with few nodes. MFEM code is a powerful instrument to compute the numerical solutions of implicit systems of time-dependent PDEs involving fine scale phenomena such as steep moving fronts.

References

[1] C. Sereno, Método dos elementos finitos móveis aplicações em engenharia química, Ph.D. Thesis, University of Porto, Porto, 1989.
[2] K. Miller, Moving finite elements, Part I, SIAM J. Numer. Anal. 18 (1981) 1019–1032.
[3] K. Miller, R.N. Miller, Moving finite elements, Part II, SIAM J. Numer. Anal. 18 (1981) 1033–1057.
[4] A.C. Hindmarsh, LSODE and LSODI, two new initial value ordinary differential equation solvers, ACM-SIGNUM Newslett. 15 (1980) 10–11.
[5] C. Sereno, A. Rodrigues, J. Villadsen, The moving finite element method with polynomial approximation of any degree, Comput. Chem. Eng. 15 (1991) 25–33.
[6] M.J. Baines, Moving Finite Elements, Oxford University Press, Oxford, 1994.
[7] S. Adjerid, J.E. Flaherty, A moving finite element method with error estimation and refinement for one-dimension time-dependent partial differential equations, SIAM J. Numer. Anal. 23 (1986) 778–796.
[8] L.R. Petzold, Observation on an adaptive moving grid method for one-dimensional systems of PDE, Appl. Numer. Math. 3 (1987) 347–360.
[9] J.C. Verwer, J.G. Blom, J.M. Sanz-Serna, An adaptive moving grid method for one-dimensional systems of partial differential equations, J. Comput. Phys. 82 (1989) 454–486.
[10] N. Carlson, K. Miller, Design and application of a gradient-weighted moving finite element code I: in one dimension, SIAM J. Sci. Comput. 19 (3) (1998) 728–765.
[11] R.M. Ferreira, Contribuição para o estudo de reactores catalíticos em leito fixo: efeito da convecção em catalisadores de poros largos e casos de catalisadores bidispersos, Ph.D. Thesis, University of Porto, Porto, 1988.
[12] C. Sereno, A. Rodrigues, J. Villadsen, Solution of partial differential equations systems by the moving finite element method, Comput. Chem. Eng. 16 (1992) 583–592.
[13] M.C. Coimbra, C. Sereno, A. Rodrigues, Moving finite elements: simulation of a pressurization of adsorption beds with a mixture of helium–methane, in: Proceedings of the First Meeting on Numerical Methods for Differential Equation, University of Coimbra, Coimbra, 1995, pp. 9–17.
[14] R. Bird, W. Stewart, E.N. Lightfoot, Transport Phenomena, Wiley, New York, 1960.
[15] A. Mendes, Adsorção com modulação da pressão, Aplicação à separação do oxigénio do ar, Ph.D. Thesis, University of Porto, Porto, 1993.