Identification problem in plug-flow chemical reactors using the adjoint method

A. Bermúdez[a], N. Esteban[b], J.L. Ferrín[a,*], J.F. Rodríguez-Calo[c], M.R. Sillero-Denamiel[d]

[a] *Departamento de Matemática Aplicada. Universidad de Santiago de Compostela. 15782 Santiago de Compostela. Spain*
[b] *Instituto Tecnológico de Matemática Industrial. 15782 Santiago de Compostela. Spain*
[c] *Centro de Tecnología Repsol. 28935 Móstoles (Madrid). Spain*
[d] *IMUS. Universidad de Sevilla. 41012 Sevilla. Spain*

Abstract

The aim of this work is to solve identification problems in plug-flow chemical reactors. For this purpose an adjoint-based algorithm for parameter identification problems in systems of partial differential equations is presented. The adjoint method allows us to calculate the gradient of the objective function and the constraint functions with respect to the unknown parameters significantly reducing the computer time. This leads to solve a minimization problem, in which an objective function is defined in order to quantify the mismatch between the observed data and the numerical solution of the parameterized chemical model. For solving the initial and boundary-value problem we use finite-difference schemes. More precisely, we propose a second-order BDF method initialized with a first-order one. The algorithm proposed was implemented in a computer program and some numerical results are shown. The efficiency of the adjoint method, compared with the classical formula of incremental quotients, is also presented.

Keywords: chemical kinetics, plug-flow reactor, identification, adjoint method, integral method

1. Introduction

Important models from different fields of engineering and applied sciences depend on unknown parameters which must be identified to define completely the system. Examples are common in biological systems, chemical reaction mechanisms, fluid mechanics, etc. Usually, the parameters are estimated by solving a minimization problem, where the objective function depends on the solution of the parameterized model. These models can be systems of numerical, ordinary (ODE) or partial differential equations (PDE). Many optimization algorithms consist of gradient-based iterative methods, which require, at each step, the computation of the derivatives of the objective function and possibly of the constraint functions with respect to the optimization variables. Parameter identification problems present two classical difficulties: first the number of variables and parameters of the model can be large and second the governing equations usually involve nonlinear functions of variables and parameters so, in practice, they have to be solved numerically. Therefore, a discrete minimization problem is considered instead of the continuous one, where the objective function depends on the numerical solution of the parameterized model.

[*] Corresponding author
Email addresses: alfredo.bermudez@usc.es (A. Bermúdez), noemi.esteban@usc.es (N. Esteban), joseluis.ferrin@usc.es (J.L. Ferrín), jfrodriguezc@repsol.com (J.F. Rodríguez-Calo), mremediossillero-ext@us.es (M.R. Sillero-Denamiel)

In this paper, we consider a quite general PDE-constrained optimization problem. The main goal is to propose an algorithm to compute the global solution of the discrete optimization problem obtained from the previous continuous one by solving the model of the system using some discretization procedure. More specifically, we concentrate our efforts in introducing an efficient strategy for computing the derivatives of the discrete objective function with respect to the optimization variables. Basically, three different alternatives could be used: classical finite-difference methods, automatic differentiation or adjoint method. The first one is very sensitive to the discretization step, leading to round-off errors when it is small and to truncation error when it is large. Additionally, the needed computer time is usually very high, mainly for problems with a large number of optimization variables.

The second one is a set of techniques to transform a program that calculates numerical values of a function, into a program that computes the values of the derivatives of this function with about the same accuracy and efficiency as the function values themselves. Therefore, its cost for our problem will be of the same order or, very likely, higher than solving the state equation. Some references about automatic differentiation are Bartholomew-Biggs et al. (2000) and Neidinger (2010). Furthermore, there are a lot of commercial and free packages as, for example, ADOL-C, CasADi or FABDAB++.

The third one comes from the mathematical theory of optimal control of systems governed by partial differential equations, as developed by J.L. Lions (see Lions (1971) and Lions (1972)). In optimal control theory, the gradient of the objective function is calculated indirectly by solving the so-called adjoint equation. Since the adjoint equation is linear, and thus of reduced complexity, the cost of obtaining the derivatives of the objective function with respect to each optimization variable from the solution of the adjoint equation is low. This procedure, which has been successfully used in different fields and applied to different problems (see, for example, Daescu et al. (2003) or Sandu et al. (2003) where it was applied in chemical kinetic systems, Jameson (1988) or Pironneau (1974) in design optimization in fluid dynamics, Maute et al. (2003) in aeroelastic systems, Meric (1985) in thermoelasticity or Vautard et al. (2000) in atmospheric chemistry), will be the one chosen in our work.

In this paper, we consider models of reaction systems where the physico-chemical magnitudes depend on the particular position in the reactor (distributed systems), but only on its axial coordinate, i.e., they are supposed to be constant in the cross-sections. They correspond to the so-called plug-flow reactors (PFR). We assume that stoichiometric matrix, empirical measurements for variables and functional expressions for the reaction velocities are known, but the latter depend on a set of unknown parameters. The proposed method to estimate the values of these unknown parameters consists in minimizing the deviation between the experimental measurements and the numerical solution of the parameterized model. This optimization problem is solved by iterative methods which require, at each step, the computation of both the objective function and its gradient. In order to obtain the value of the objective function, a finite difference approximation is used to solve the model which is an initial-boundary value problem. On the other hand, we employ the adjoint method to obtain the gradient of the objective function with respect to the unknown parameters. We notice that with this method these derivatives are exactly and efficiently calculated. In addition, since nonconvex nonlinear functions appear in the optimization problem, the standard local search tools may get easily trapped in local optima. Therefore, Global Optimization techniques must be used to obtain the global optimum of the problem. In this work, the well-known Variable Neighborhood Search (VNS) method is applied. For more details on the efficiency of this methodology, the reader is referred to Carrizosa et al. (2012), Mladenović and Hansen (1997) or Mladenović et al. (2008).

The identification problem presented in this paper is less treated in the bibliography (see, for example, Rodrigues et al. (2015a)) than the case where a stirred tank reactor is considered. In the latter, the mathematical model describing the temperature and species concentrations consists of ordinary differential equations. Different methods can be found in the bibliography identifying the kinetics of the reaction system from experimental data. On one side we have the so-called *simultaneous approach* where all reaction rates are identified simultaneously (see, for example, Bardow and Marquardt (2004) or Bhatt et al. (2012)). On the other side we have the so-called *incremental approach* where the identification problem can be decoupled dealing with each reaction in an individual manner. This is mainly done using the concept of extents as, for example, in Bhatt et al. (2011), Brendel et al. (2006), Rodrigues et al. (2015b) or Srinivasan et al. (2012). Let us emphasize that the adjoint method we consider in this paper can be applied to any of the two approaches.

The present paper is organized as follows. A general PDE-constrained optimization problem arising from the parameter identification problem of a reaction system in a PFR is presented in Section 2. For the sake of completeness, the model is deduced for the unsteady and non-isothermal case so the temperature is also calculated by including the energy conservation equation. The numerical methods for solving this optimization problem are introduced in Section 3. They include a BDF second-order finite-difference scheme for solving the system of partial differential equations, a so-called simultaneous integral method for reaching the global optimum and the discrete adjoint method for computing the gradient of the objective function with respect to the optimization variables. In Section 4, we apply the proposed algorithm to a plug-flow reactor and show the advantage of using the adjoint method. We finish with some conclusions.

2. Identification problem

In this section we develop the identification problem for isothermal or non-isothermal PFRs when the kinetics of the reactions follow the generalized functional form:

$$\delta_l(\theta, \mathbf{y}, \mathbf{z}) = B_l e^{-E_l/\mathcal{R}\theta} \prod_{k=1}^{M_l} \left[\sum_{r=1}^{N} G^l_{kr} y_r + \sum_{r=N+1}^{N+N_c} G^l_{kr} z_{r-N} + b_{kl} \right]^{\alpha_{kl}}, \ l = 1, \ldots, L \tag{1}$$

where θ is the temperature, $\mathbf{y}$ are the species concentrations vector, $\mathbf{z}$ are the catalysts concentrations vector, $\mathcal{R}$ is the universal gas constant, N is the number of species, N_c is the number of catalysts, L is the number of reactions and B_l (Arrhenius pre-factors), E_l (activation energies), G^l_{kr}, b_{kl} and α_{kl} are the unknown parameters to be found. In what follows, all these parameters to be identified will be denoted by vector $\mathbf{\Theta}$. We also introduce notation $\mathbf{x}$ for the vector containing species concentrations and temperature, i.e., $\mathbf{x} = (\mathbf{y}^t, \theta)^t \in \mathbb{R}^{N+1}$. Kinetics given by equation (1) were provided by the engineers of Repsol (an integrated oil and gas Company), taking into account that it can represent all the situations observed in the experimental setups they are trying to simulate.

Then, being provided experimental data for species concentrations and temperature at the reactor exit, $(\widehat{\mathbf{x}^{e,n}})$, at several times, (S^e), and for different experiments $(\mathcal{E})$, the identification problem is formulated as

$$\min_{\mathbf{\Theta}} \widehat{\mathcal{J}}(\mathbf{\Theta}), \tag{2}$$

being

$$\widehat{\mathcal{J}}(\mathbf{\Theta}) = \mathcal{J}(\mathbf{x}(\mathbf{\Theta}), \mathbf{\Theta}) := \sum_{e\in\mathcal{E}} \sum_{i=1}^{N+1} \sum_{t^e_n \in S^e} \omega_{ien} (x^{e,n}_i(\mathbf{\Theta}) - \widehat{x^{e,n}_i})^2, \tag{3}$$

where $x^{e,n}_i(\mathbf{\Theta})$ is the i-th component of the solution of the mathematical model describing the PFR at time $t^e_n \in S^e$ and at the reactor exit.

Methodology to solve the optimization problem (2)-(3) is the goal of this paper. This will be addressed in the next section.

2.1. PFR reactor model

Plug flow reactors, also called continuous tubular reactors or piston flow reactors, are reactors with cylindrical geometry. In principle, due to viscosity, the velocity of the flow in a pipe is null on the wall. Moreover, in the laminar regime, the velocity profile is parabolic with the maximum at the central axis of the pipe. However, the plug flow is a simple model where the velocity is assumed to be constant on any cross-section of the pipe (but it may depend on time).

Furthermore, it will be assumed that the physico-chemical magnitudes may vary along the reactor but they are constant in any cross-section. This means that they depend on time and on one space variable.

Let z be the axial coordinate of the reactor of length L. Then $z \in [0, \mathrm{L}]$. By assuming that (i) all thermodynamic magnitudes depend only on z and t, (ii) the diffusive, conductive and viscous dissipation terms can be neglected, (iii) there is no external volumetric heat source, and (iv) the density of the mixture in the reactor is constant and equal to the density of the input mixture, the full model for a non-adiabatic PFR is

$$\frac{\partial \mathbf{y}}{\partial t} + v\frac{\partial \mathbf{y}}{\partial z} = A\boldsymbol{\delta}(\theta, \mathbf{y}, \mathbf{z}), \tag{4}$$

$$\frac{\partial \theta}{\partial t} + v\frac{\partial \theta}{\partial z} = \frac{1}{\hat{\mathbf{w}}'(\theta)\cdot\mathbf{y}}\Big(- \Delta\hat{\mathbf{H}}(\theta)\cdot\boldsymbol{\delta}(\theta, \mathbf{y}, \mathbf{z}) + \frac{2g}{R}\,(\theta_{ext} - \theta)\Big), \tag{5}$$

$$\mathbf{y}(0,t) \;\text{ and }\; \theta(0,t) \text{ are given, } \;\forall t \in [0,T], \tag{6}$$

$$\mathbf{y}(z,0) = \mathbf{y}^0(z), \;\; \theta(z,0) = \theta^0(z), \tag{7}$$

being $v(t)$ the axial velocity of the flow at time t, A the stoichiometric matrix, $\boldsymbol{\delta}$ the reaction velocity vector with components given by (1), θ_{ext} the outside temperature, g the convective heat transfer coefficient and R the radius of the reactor. The components of vector $\hat{\mathbf{w}}(\theta) \in \mathbb{R}^N$ are defined by

$$\hat{w}_i(\theta) = \mathcal{M}_i\hat{e}_i(\theta), \tag{8}$$

where $\mathcal{M}_i$ is the molecular mass of species i, while the components of the L-dimensional vector

$$\Delta\hat{\mathbf{H}}(\theta) := A^t\hat{\mathbf{w}}(\theta) \tag{9}$$

are the heat of reactions at temperature θ. Moreover, the function of temperature $\hat{e}_i(\theta)$ is the specific (i.e., per unit mass) internal energy of the i-th species. It is given by

$$\hat{e}_i(\theta) := e_i^* + \int_{\theta^*}^{\theta} \hat{c}_i(s)\,ds, \tag{10}$$

where e_i^* is the internal energy of formation of the i-th species at temperature θ^* and $c_i = \hat{c}_i(\theta)$ is the specific heat of the i-th species at temperature θ.

It must be taken into account that $v(t)$ is supposed to be given (in fact, it is the velocity of the input current at the inlet of the reactor which can be obtained as its volumetric flow rate, $u_{in}(t)$, divided by the area of the reactor cross-section).

For the sake of simplicity, we will write system (4)-(7) in the following more compact form:

$$\frac{\partial \mathbf{x}}{\partial t} + v\frac{\partial \mathbf{x}}{\partial z} = \mathbf{F}(t, \mathbf{x}, \mathbf{z}), \tag{11}$$

$$\mathbf{x}(0,t) \text{ are given,} \tag{12}$$

$$\mathbf{x}(z,0) = \begin{pmatrix} \mathbf{y}^0(z) \\ \theta^0(z) \end{pmatrix} \tag{13}$$

where

$$\mathbf{F}(t,\mathbf{x},\mathbf{z}) = \begin{pmatrix} A\boldsymbol{\delta}(\theta,\mathbf{y},\mathbf{z}) \\ \frac{1}{\hat{\mathbf{w}}'(\theta)\cdot\mathbf{y}}\Big(-\Delta\hat{\mathbf{H}}(\theta)\cdot\boldsymbol{\delta}(\theta,\mathbf{y},\mathbf{z}) + \frac{2g}{R}\left(\theta_{ext}-\theta\right)\Big) \end{pmatrix}. \tag{14}$$

3. Numerical methodology

3.1. Solver

As we have said at the beginning of Section 2, the initial-boundary value problem (11)-(14) must be solved to obtain $\mathbf{x}^{e,n}(\boldsymbol{\Theta}) = (\mathbf{y}^{e,n}(\boldsymbol{\Theta})^t, \theta^{e,n}(\boldsymbol{\Theta}))^t$. This can be done by using, for instance, a finite difference method both in time (t) and in space (z). For the sake of stability, the partial derivatives with respect to z will be approximated by using an upwind differentiation formula, whereas they are taken implicit in time.

For time discretization, we have chosen a first-order backward differentiation formula (BDF1) for the first time step and a second-order backward differentiation formula (BDF2) for the subsequent ones.

Let us denote by Δt and h the steps for the time and spatial discretizations, respectively, and K and J the number of time and spatial mesh points, named $t_i = i\Delta t$, $i = 0,\ldots,K$ and $z_j = jh$, $j = 1,\ldots,J$. Thus, $\Delta t = T/K$ and $h = L/J$. By denoting $\mathbf{x}_j^i$ and $\mathbf{z}_j^i$ approximations of $\mathbf{x}(t_i, z_j)$ and $\mathbf{z}(t_i, z_j)$, respectively, and $v_i = v(t_i)$, the resulting discretized equations, for $j = 1,\ldots,J$, can be written as (for the sake of simplicity superscript e will not be included):

- For $t = t_1$:

$$\frac{\mathbf{x}_j^1 - \mathbf{x}_j^0}{\Delta t} + v_1 \frac{\mathbf{x}_j^1 - \mathbf{x}_{j-1}^1}{h} = \mathbf{F}(t_1, \mathbf{x}_j^1, \mathbf{z}_j^1). \tag{15}$$

- For $n = 2,\ldots,K$:

$$\frac{3\mathbf{x}_j^{n+1} - 4\mathbf{x}_j^n + \mathbf{x}_j^{n-1}}{2\Delta t} + v_{n+1}\frac{\mathbf{x}_j^{n+1} - \mathbf{x}_{j-1}^{n+1}}{h} = \mathbf{F}(t_{n+1}, \mathbf{x}_j^{n+1}, \mathbf{z}_j^{n+1}). \tag{16}$$

Let us notice that each time step amounts to solve J nonlinear systems of $N+1$ equations (one system for each spatial mesh-point except for the leftmost one) and $N+1$ unknowns (N species concentrations and temperature). They can be written in a more compact form as follows:

$$\widetilde{B}\mathbf{x}_h = \widetilde{\mathbf{F}} \tag{17}$$

where

$$\widetilde{B} = \begin{pmatrix} B_1 & & & & 0 \\ C_1 & B_2 & & & \\ C_2 & C_1 & B_3 & & \\ & \ddots & \ddots & \ddots & \\ 0 & & C_2 & C_1 & B_K \end{pmatrix} \tag{18}$$

is a $K \times K$ block-matrix and

$$\widetilde{\mathbf{F}} = (\mathbf{b_1}^t, \ldots, \mathbf{b_K}^t)^t. \tag{19}$$

B_i, $i = 1, \ldots, K$, C_1 and C_2 are $J \times J$ block matrices, each of them of size $(N+1) \times (N+1)$, and $\mathbf{b_i} \in \mathbb{R}^{J \times (N+1)}$. Their expressions are

$$B_1 = \begin{pmatrix} \left(1 + \frac{\Delta t}{h} v_1\right) I & & & & O \\ -\frac{\Delta t}{h} v_1 I & \left(1 + \frac{\Delta t}{h} v_1\right) I & & & \\ & -\frac{\Delta t}{h} v_1 I & \left(1 + \frac{\Delta t}{h} v_1\right) I & & \\ & & \ddots & \ddots & \\ O & & & -\frac{\Delta t}{h} v_1 I & \left(1 + \frac{\Delta t}{h} v_1\right) I \end{pmatrix}, \tag{20}$$

$$B_r = \begin{pmatrix} \left(1 + \frac{2\Delta t}{3h} v_r\right) I & & & & O \\ -\frac{2\Delta t}{3h} v_r I & \left(1 + \frac{2\Delta t}{3h} v_r\right) I & & & \\ & -\frac{2\Delta t}{3h} v_r I & \left(1 + \frac{2\Delta t}{3h} v_r\right) I & & \\ & & \ddots & \ddots & \\ O & & & -\frac{2\Delta t}{3h} v_r I & \left(1 + \frac{2\Delta t}{3h} v_r\right) I \end{pmatrix}, \quad r = 2, \ldots, K, \tag{21}$$

$$C_1 = \begin{pmatrix} -\frac{4}{3} I & & O \\ & \ddots & \\ O & & -\frac{4}{3} I \end{pmatrix}, \tag{22}$$

$$C_2 = \begin{pmatrix} \frac{1}{3} I & & O \\ & \ddots & \\ O & & \frac{1}{3} I \end{pmatrix}, \tag{23}$$

$$\mathbf{b_1} = \begin{pmatrix} \mathbf{x}_1^0 + \frac{\Delta t}{h} v_1 \mathbf{x}_0^1 + \Delta t \mathbf{F}(t_1, \mathbf{x}_1^1, \mathbf{z}_1^1) \\ \mathbf{x}_2^0 + \Delta t \mathbf{F}(t_1, \mathbf{x}_2^1, \mathbf{z}_2^1) \\ \vdots \\ \mathbf{x}_J^0 + \Delta t \mathbf{F}(t_1, \mathbf{x}_J^1, \mathbf{z}_J^1) \end{pmatrix}, \tag{24}$$

$$\mathbf{b_2} = \begin{pmatrix} \frac{1}{3} \mathbf{x}_1^0 + \frac{2\Delta t}{3h} v_2 \mathbf{x}_0^2 + \frac{2}{3} \Delta t \mathbf{F}(t_2, \mathbf{x}_1^2, \mathbf{z}_1^2) \\ \frac{1}{3} \mathbf{x}_2^0 + \frac{2}{3} \Delta t \mathbf{F}(t_2, \mathbf{x}_2^2, \mathbf{z}_2^2) \\ \vdots \\ \frac{1}{3} \mathbf{x}_J^0 + \frac{2}{3} \Delta t \mathbf{F}(t_2, \mathbf{x}_J^2, \mathbf{z}_J^2) \end{pmatrix}, \tag{25}$$

$$\mathbf{b_r} = \begin{pmatrix} \frac{2\Delta t}{3h} v_r \mathbf{x}_0^r + \frac{2}{3} \Delta t \mathbf{F}(t_r, \mathbf{x}_1^r, \mathbf{z}_1^r) \\ \frac{2}{3} \Delta t \mathbf{F}(t_r, \mathbf{x}_2^r, \mathbf{z}_2^r) \\ \vdots \\ \frac{2}{3} \Delta t \mathbf{F}(t_r, \mathbf{x}_J^r, \mathbf{z}_J^r) \end{pmatrix}, \quad r = 3, \ldots, K, \tag{26}$$

where $I = I_{(N+1)\times(N+1)}$.

3.2. Integral method

In this subsection we are going to explain how the optimization problem (2)-(3) is solved. It must be kept in mind that there are real variables in $\mathbf{\Theta}$ and integer variables too, according to the user's choice. Due to this, we will make a process in several steps:

1. As initial solution, the best of several solutions chosen at random is selected, thus providing a first value for the objective function. The values of the involved parameters are within user-specified ranges.
   (a) If all the parameters are real numbers, we start with the improvement phase, in which the continuous problem is optimized. The solution is further improved by using the VNS.
   (b) If there are integer parameters, a multi-start is planned for these parameters, whereas optimization is only done for the real ones. Again, the solution is further improved by using the VNS.
2. For the vector of unknowns, coming from the second step, we have to make a perturbation of radius r of each component (starting with $r = 1$), taking into account whether the components must take real or integer values.
   (a) Firstly, we consider the case where the chosen unknown takes a real value. We define an interval where the perturbation will take place. In the neighborhood of radius r, this interval has the following form:
   $$\left[c - \frac{c - L_c}{r_{max}} * r, c + \frac{U_c - c}{r_{max}} * r\right], \tag{27}$$
   where c is the best value found up to now for the parameter that we are considering, L_c and U_c are the lower and the upper bounds of c, respectively, and r_{max} (maximum value that r can take) is the maximum number of allowed neighborhoods. In general, this interval is asymmetric. Once this interval is defined, a random number is selected in it.
   (b) Secondly, let us see the case where the parameter can only take an integer value. We define the same previous interval for the integer unknown, and later, we modify this interval as follows:
   $$\left[\lceil c - \frac{c - L_c}{r_{max}} * r\rceil, \lfloor c + \frac{U_c - c}{r_{max}} * r\rfloor\right]. \tag{28}$$
   An integer number γ between $\lceil c - \frac{c-L_c}{r_{max}} * r\rceil$ and $\lfloor c + \frac{U_c-c}{r_{max}} * r\rfloor$ is chosen. Here, $\lfloor . \rfloor$ and $\lceil . \rceil$ denote the floor (it returns the integer part of the number) and the ceiling (it returns the integer part of the number plus one) functions respectively.

   Then we proceed with a multi-start with random assignment of parameters within the intervals previously explained. If an improvement in the value of the objective function is achieved, value of r is reset to 1. Otherwise $r = r + 1$. This process is repeated until either the maximum number of neighborhoods r_{max} or the computation time provided by the user to complete this step is reached.
3. The vector of parameters obtained in the previous step is introduced as a new initial solution for optimization, thereby providing a value of the objective function to be compared with that obtained in step 1.

The process is repeated until the maximum run-time is reached or until the value of the objective function is less than a user-defined tolerance.

In this optimization process, the derivatives of the objective function are needed, and for this we are going to use the adjoint method which will be explained below.

3.3. Adjoint method

In order to obtain the gradient of the objective function involved in the optimization problem (2)-(3), we could proceed by an approximation of the partial derivatives through numerical derivation as follows

$$\frac{\partial \widehat{\mathcal{J}}}{\partial \Theta_i}(\boldsymbol{\Theta}) \approx \frac{\widehat{\mathcal{J}}(\boldsymbol{\Theta} + \varepsilon_i \mathbf{e}_i) - \widehat{\mathcal{J}}(\boldsymbol{\Theta})}{\varepsilon_i}, \tag{29}$$

for $i \in \{1, \dots, N_p\}$, where N_p is the number of parameters to identify, ε_i is the discretization step for the i-th variable and $\mathbf{e}_i$ is the i-th vector of the canonical basis of $\mathbb{R}^{N_p}$.

However, it is well-known that the computing time for this methodology is very high, due to the fact that the initial-boundary value problem (4)-(7) must be solved once for each partial derivative. Moreover, the choice of steps ε_i is not an easy task. We notice that, if these steps are not properly chosen, then the accuracy is very poor leading to a bad convergence of the optimization algorithm. As a less expensive and more accurate (in fact, exact) alternative, we will use the adjoint method described below.

The first step is to write the continuous optimization problem (2)-(3) in a discrete form:

$$\min_{\boldsymbol{\Theta}} \widehat{\mathcal{J}}_d(\boldsymbol{\Theta}), \tag{30}$$

where

$$\widehat{\mathcal{J}}_d(\boldsymbol{\Theta}) = \mathcal{J}_{h,\Delta t}(\mathbf{x}_h(\boldsymbol{\Theta}), \boldsymbol{\Theta}) := \sum_{e\in\mathcal{E}} \sum_{i=1}^{N+1} \sum_{t_n^e \in S^e} \omega_{ien} (x_{d,i}^{e,n^*(n)}(\boldsymbol{\Theta}) - \widehat{x_i^{e,n}})^2, \tag{31}$$

$\mathbf{x}_h$ is the solution of (17) and $x_{d,i}^{e,n^*(n)}$ is the i-th component of $\mathbf{x}_h$ at time $t^*_{n^*(n)} = t^e_n$. For the sake of simplicity, we assume that the set of observation instants is a subset of the time mesh, i.e., $S^e \subset \{t^*_n = n\Delta t,\ n = 0, \dots, K\}$.

Differentiation of the discrete objective function, $\widehat{\mathcal{J}}_{\Delta t}$, with respect to $\boldsymbol{\Theta}$ leads to

$$\nabla_{\boldsymbol{\Theta}} \widehat{\mathcal{J}}_d(\boldsymbol{\Theta}) = \nabla_{\boldsymbol{\Theta}} \mathbf{x}_h \cdot \nabla_{\mathbf{x}} \mathcal{J}_{h,\Delta t}(\mathbf{x}_h(\boldsymbol{\Theta}), \boldsymbol{\Theta}) + \nabla_{\boldsymbol{\Theta}} \mathcal{J}_{h,\Delta t}(\mathbf{x}_h(\boldsymbol{\Theta}), \boldsymbol{\Theta}). \tag{32}$$

The gradient of the discrete variables $\mathbf{x}_h$ with respect to the optimization variables, $\nabla_{\boldsymbol{\Theta}} \mathbf{x}_h$, could be obtained from the differentiation of the solution of system (15)-(16) by using the implicit function theorem. However, it is well-known that the computational cost of this method is high when the number of optimization variables becomes large, because one should solve N_p linearized state equations, one for each independent variable. For this reason, we are going to use the adjoint approach, that we recall below.

First, by using (17) we notice that $\nabla_{\boldsymbol{\Theta}} \mathbf{x}_h$ satisfies

$$\nabla_{\boldsymbol{\Theta}} \mathbf{x}_h \cdot \widetilde{B}^t = \nabla_{\boldsymbol{\Theta}} \widetilde{\mathbf{F}}. \tag{33}$$

Now, we introduce the adjoint state $\mathbf{p}$ as the solution of the following linear system:

$$\widetilde{B}^t \mathbf{p} = \nabla_{\mathbf{x}} \mathcal{J}_{h,\Delta t}(\mathbf{x}_h(\boldsymbol{\Theta}), \boldsymbol{\Theta}). \tag{34}$$

Then, by using (33) and (34) in (32), we obtain $\nabla_{\boldsymbol{\Theta}} \widehat{\mathcal{J}}_d(\boldsymbol{\Theta})$ in terms of the adjoint state, namely

$$\nabla_{\boldsymbol{\Theta}} \widehat{\mathcal{J}}_d(\boldsymbol{\Theta}) = \nabla_{\boldsymbol{\Theta}} \widetilde{\mathbf{F}} \mathbf{p} + \nabla_{\boldsymbol{\Theta}} \mathcal{J}_{h,\Delta t}(\mathbf{x}_h, \boldsymbol{\Theta}). \tag{35}$$

Notice that this method only needs to solve one *linear* system, the so-called adjoint-state equation (33). The computation of the derivatives appearing in the above equations, namely, $\nabla_{\mathbf{x}} \mathcal{J}_{\Delta t}$, $\nabla_{\boldsymbol{\Theta}} \mathcal{J}_{\Delta t}$ and $\nabla_{\boldsymbol{\Theta}} \widetilde{\mathbf{F}}$ can be done analytically.

Remark 3.1. Because of the triangular block structure of matrix $\widetilde{B}$, the solution of the linear system in (34) can be obtained in a recursive manner, starting from the final time, by solving $J \times K$ linear systems of size $N+1$. Moreover, the derivative of the discrete objective function can be obtained cumulatively from the adjoint state. In practice, these two steps are done simultaneously.

4. Numerical experiments

In this section we will show the results of some numerical experiments obtained with the developed code. First, we build a synthetic experiment where the expressions of the kinetics of the reactions are provided, which allow us to solve the model of the reactor (equations (4)-(7)) with different initial values to generate the "experimental" data. Second, we use the integral method introduced in the previous section to solve the raised identification problem and show the obtained results, comparing them with the exact ones. Finally, we remark on the difference in computing time required for calculating the gradient of the objective function using the conventional method of incremental quotients and the adjoint one.

4.1. Describing the test problem

The performance of the integral method for solving the identification problem will be shown by constructing a test case consisting of a set of six reactions (L=6) involving twelve species (N=12) and one catalyst (N_c=1), and using certain given expressions for the kinetics of these reactions. More precisely, the reactions, catalyzed by C_1, are

$$E_1 + E_2 \longrightarrow E_3 + E_4, \tag{36}$$

$$E_2 \longrightarrow E_3 + \frac{1}{2}E_5, \tag{37}$$

$$E_2 \longrightarrow E_6 + E_7, \tag{38}$$

$$E_2 \longrightarrow E_8 + E_9, \tag{39}$$

$$E_2 \longrightarrow E_{10} + E_{11}, \tag{40}$$

$$E_4 + E_7 \longrightarrow E_{12}, \tag{41}$$

whereas expressions of the components of the reaction velocity vector are

$$\delta_1(\theta, \mathbf{y}) = 5 \cdot 10^{10} e^{-62737/\mathcal{R}\theta} y_1 y_2 z_1, \tag{42}$$

$$\delta_2(\theta, \mathbf{y}) = 2 \cdot 10^{10} e^{-71097/\mathcal{R}\theta} y_2^2 z_1^{0.5}, \tag{43}$$

$$\delta_3(\theta, \mathbf{y}) = 10^{10} e^{-66914/\mathcal{R}\theta} y_2^2 z_1, \tag{44}$$

$$\delta_4(\theta, \mathbf{y}) = 5 \cdot 10^{9} e^{-85734/\mathcal{R}\theta} y_2 z_1^{1.1}, \tag{45}$$

$$\delta_5(\theta, \mathbf{y}) = 4 \cdot 10^{9} e^{-133829/\mathcal{R}\theta} y_2 z_1^{0.5}, \tag{46}$$

$$\delta_6(\theta, \mathbf{y}) = 2 \cdot 10^{9} e^{-63987/\mathcal{R}\theta} y_4 y_7 z_1^{1.2}. \tag{47}$$

The reactor considered in this experimental setup has one meter long ($L = 1$) and a volume of 0.7854 m^3, and the outside temperature is $\theta_{ext} = 300$ K. Specific heat of all the N species is equal to 2000 J/kg K, whereas their molecular weights are

$$\boldsymbol{\mathcal{M}} = (0.05, 0.14, 0.12, 0.07, 0.04, 0.122, 0.018, 0.07, 0.07, 0.09, 0.05, 0.088)^t \text{ kg/kmol}. \tag{48}$$

Heat of the reactions at the reference temperature of 298.15 K are $\Delta\hat{\mathbf{H}}(\theta) = (1.5, 1, 0.8, 0.7, 0.6, 0.5)^t$ J/kmol. Finally, the overall heat transfer coefficient, g, is equal to 6 W/m^2 K (see (5)).

With these data, 10 synthetic experiments (all of them with a catalyst concentration of 0.001 mol/l at any time) were built considering different initial and boundary values of species concentrations and temperature as shown in Table 1. Each data set corresponds to 11 equally spaced times ($K = 11$), until reaching the end time $T = 20$ seconds. The values given in this way by the mathematical model governing the reactor will be used as experimental data in the identification process, without adding perturbations. When a row in Table 1 has two lines, it means that the

	$\mathcal{E}_1$	$\mathcal{E}_2$	$\mathcal{E}_3$	$\mathcal{E}_4$	$\mathcal{E}_5$	$\mathcal{E}_6$	$\mathcal{E}_7$	$\mathcal{E}_8$	$\mathcal{E}_9$	$\mathcal{E}_{10}$
$y_1(0,t)$ [mol/l]	1 1	1 1	2 2	2 2	1.5 1.5	1 1.5	1.1 1.25	1.3 2	2 1	1 1.5
$y_2(0,t)$ [mol/l]	1 1	2 2	1 1	2 2	1 1	1 1	1 1	1 1	2 2	2 2.5
$y_3(0,t)$ [mol/l]	0	0	0	0	0	0	0	0	0	0
$y_4(0,t)$ [mol/l]	0	0	0	0	0	0	0	0	0	0
$y_5(0,t)$ [mol/l]	0	0	0	0	0	0	0	0	0	0
$y_6(0,t)$ [mol/l]	0	0	0	0	0	0	0	0	0	0
$y_7(0,t)$ [mol/l]	0	0	0	0	0	0	0	0	0	0
$y_8(0,t)$ [mol/l]	0	0	0	0	0	0	0	0	0	0
$y_9(0,t)$ [mol/l]	0	0	0	0	0	0	0	0	0	0
$y_{10}(0,t)$ [mol/l]	0	0	0	0	0	0	0	0	0	0
$y_{11}(0,t)$ [mol/l]	0	0	0	0	0	0	0	0	0	0
$y_{12}(0,t)$ [mol/l]	0	0	0	0	0	0	0	0	0	0
$\theta(0,t)$ [K]	383	373	383	373	383	373	383	373	383	373
$u_{in}(t)$ [l/s]	0.01 0.001	0.01 0.0001	0.01 0.1	0.001 0.01	0.1 0.01	0.01 0.01	0.01 0.01	0.01 0.01	0.01 0.01	0.01 0.01

Table 1: Inlet conditions for the experiments

first one contains the values until the time is equal to 10 seconds and the second one from that time on.

Thus, since the solution of the problem is known, the accuracy of the methodology proposed in this paper can be analyzed by looking at the functional expressions of the reaction rates. We can also compare the values of species concentrations and temperature obtained by solving the mathematical model of the reactor (equations (4)-(7)) for the identified coefficients. This will be done in Section 4.2.

4.2. Numerical results

For the initial multi-start of the integral method, random values of the parameters are chosen in the intervals detailed in Table 2.

Parameter	Value
B_i	[0, 10^{14}]
E_i	[0, 200000]
G^l_{kr}	1
b_{kr}	0
α^l_{kr}	[0, 2]

Table 2: Ranges where parameters belong

Once the methodology developed in Section 3 has been used to solve the identification problem stated in Section 4.1, the solution obtained leads to the following reaction velocities

$$\delta_1^c(\theta, \mathbf{y}) = 3.39 \cdot 10^{13} e^{-61481.2/\mathcal{R}\theta} y_1 y_2 z_1^2, \tag{49}$$

$$\delta_2^c(\theta, \mathbf{y}) = 2.02 \cdot 10^{13} e^{-102817.9/\mathcal{R}\theta} y_2^{1.81}, \tag{50}$$

$$\delta_3^c(\theta, \mathbf{y}) = 3.66 \cdot 10^{13} e^{-73142.7/\mathcal{R}\theta} y_2^2 z_1^{1.91}, \tag{51}$$

$$\delta_4^c(\theta, \mathbf{y}) = 4.68 \cdot 10^{13} e^{-186096.1/\mathcal{R}\theta} y_2^{0.21} z_1^{0.02}, \tag{52}$$

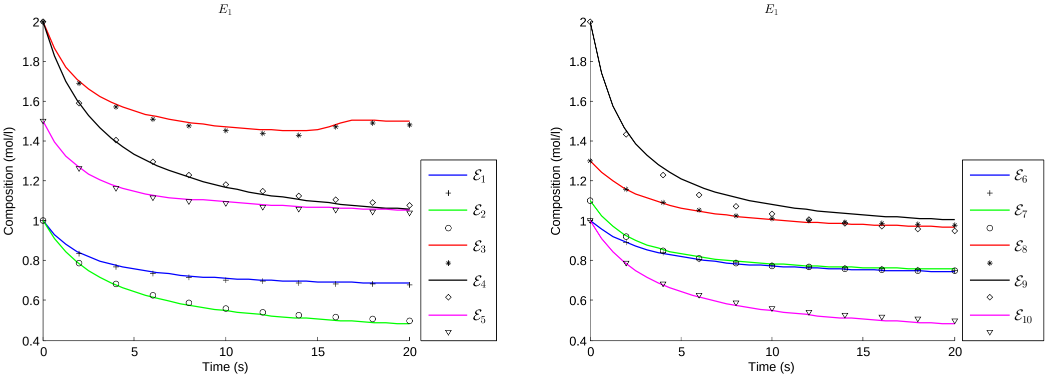
$$\delta_5^c(\theta, \mathbf{y}) = 4.15 \cdot 10^{13} e^{-105823.5/\mathcal{R}\theta} z_1^{1.05}, \tag{53}$$

$$\delta_6^c(\theta, \mathbf{y}) = 3.84 \cdot 10^{12} e^{-170237.8/\mathcal{R}\theta} y_7^{1.24}, \tag{54}$$

and to a value of 0.3475 for the objective function. It must be taken into account that the identification problem involves $N_p = 68$ parameters to identify with only 100 data. A better coincidence between expressions of the reactions rates provided by the identification methodology and the exact ones could be obtained if more experimental data were used. However, we have tried to solve the problem under realistic conditions where experimental data are limited, to show the good performance of the method.

Moreover, in order to analyze the goodness of the obtained approximation, we will also compare graphically the values of the species concentrations and temperature obtained by solving the model governing the reactor for the identified reaction rates (49)-(54), with those computed from the "true" model (42)-(47), i.e., with the "experimental values". This is done in Figures 1 to 6 where only the more representative species (those having values above 0.05 mol/l) and the temperature are plotted.

More precisely, Figures 1 to 5 represent the concentrations of species E_1 to E_5, showing the first five experiments on the left side and the last five ones on the right one (where $\mathcal{E}_i$ denotes the i-th experiment). Similarly, Figure 6 shows the temperature. In these plots, curves correspond to numerical solution and symbols to "experimental" values.

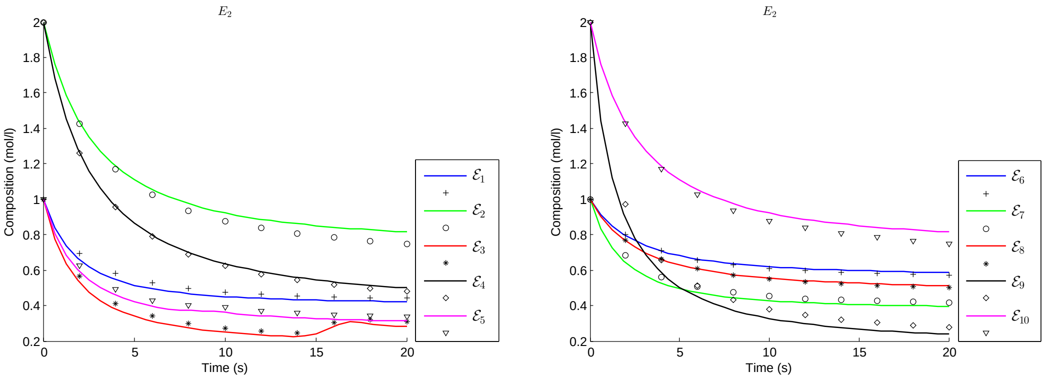


(a) Experiments 1 to 5

(b) Experiments 6 to 10

Figure 1: Observed and computed concentration of species E_1 versus time at reactor exit

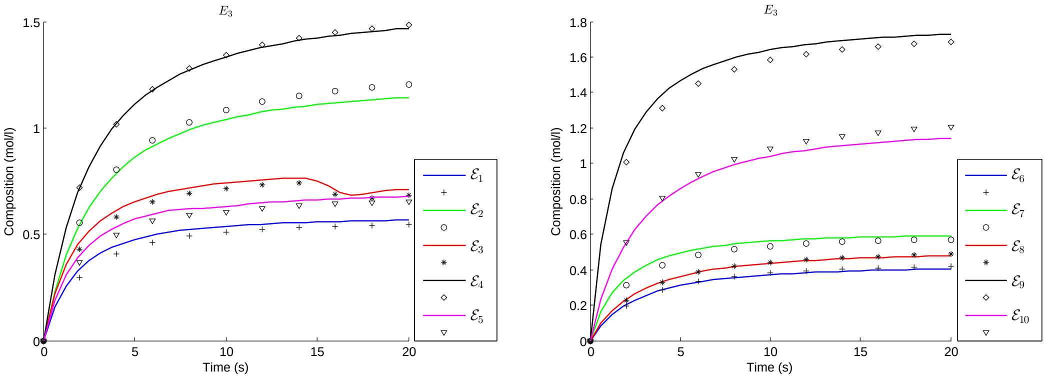
Since the identification was performed taking into account the absolute errors, which corresponds to consider the weights in the objective function equal to 1 (ω_{ien} in equation (3)), a better fit of the variables with bigger values is achieved. In particular, the temperature is very well adjusted (see Figure 6) as well as the concentrations of species E_i, $i = 1, \ldots, 4$. On the other hand, species E_5 seems to be worse approximated than the previous ones. However it is observed that the errors are very small in absolute value and trends are correctly captured.



(a) Experiments 1-5

(b) Experiments 6-10

Figure 2: Observed and computed concentration of species E_2 versus time at reactor exit



(a) Experiments 1-5

(b) Experiments 6-10

Figure 3: Observed and computed concentration of species E_3 versus time at reactor exit

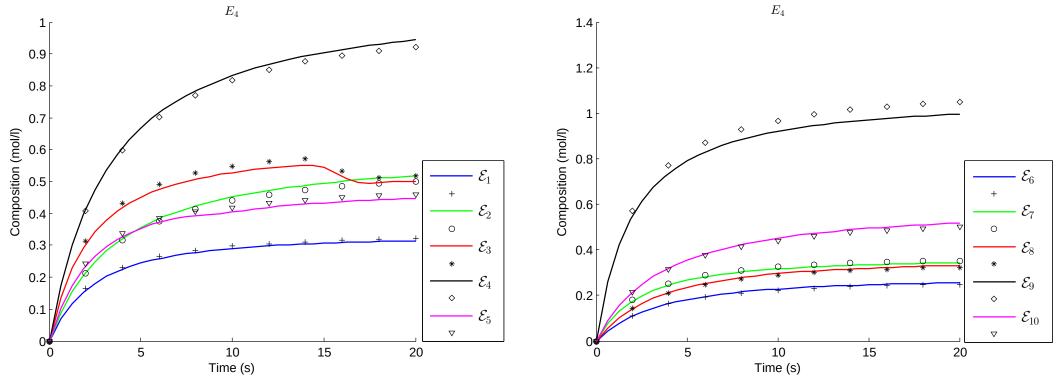
4.3. Performance of adjoint method compared with incremental quotients

In order to show the gain of using adjoint method versus incremental quotients for computing the gradient of the objective function, $\nabla_{\boldsymbol{\Theta}} \widehat{\mathcal{J}}_d(\boldsymbol{\Theta})$, we proceed with one of the experiments detailed in Section 4.1 where $K = 256$, $J = 64$ and $N_p = 68$. The computer time using a Scilab implementation and running in an Intel i5-4670 at 3.4 GHz, is shown in Table 3.

Method	Computer time
Adjoint method	5 min 38.65 s
Incremental quotients	223 min 56.85 s

Table 3: Computer time for evaluating the gradient of the objective function

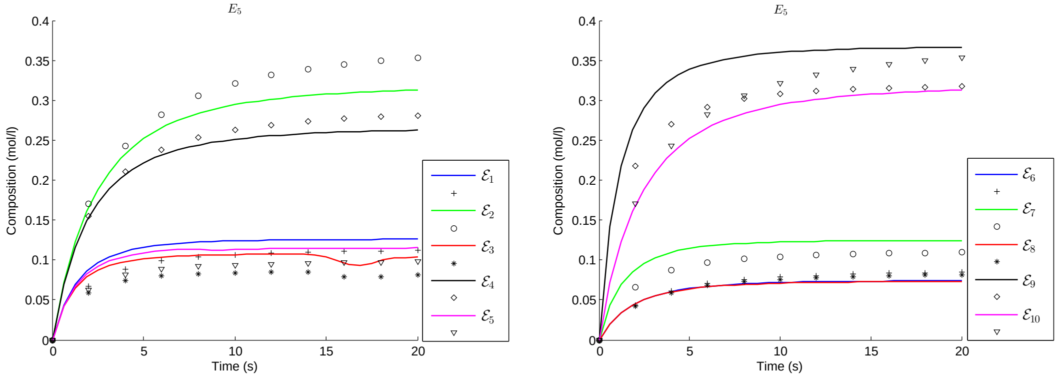
According to times shown in Table 3, with the data used in this demo, the adjoint method is almost 40 times faster than the incremental quotients for calculating only once the gradient of the objective function. It must be taken into account that the adjoint method needs one evaluation of the state equation ((4)-(7)), whereas the method using incremental quotients needs $N_p + 1$. For the sake of completeness of information, evaluating the state equation takes 3 min 33 s.



(a) Experiments 1-5

(b) Experiments 6-10

Figure 4: Observed and computed concentration of species E_4 versus time at reactor exit



(a) Experiments 1-5

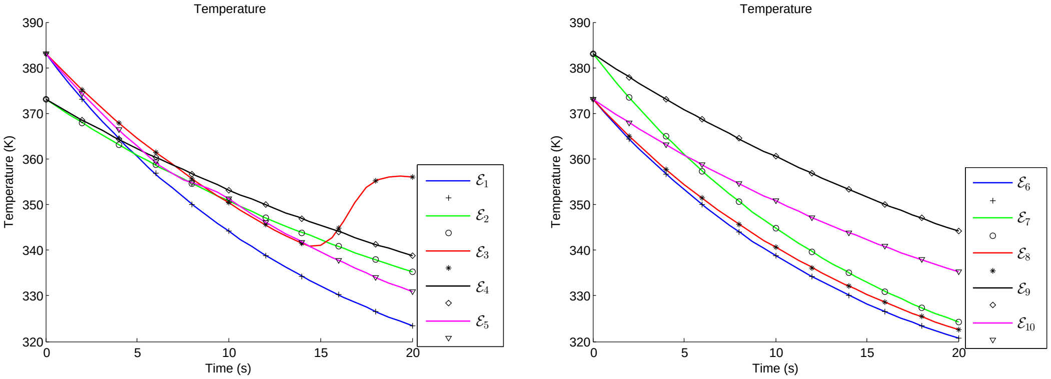
(b) Experiments 6-10

Figure 5: Observed and computed concentration of species E_5 versus time at reactor exit

ε_i	Absolute error	Relative error
10^{-5}	14.357735	358.66526
10^{-6}	1.4444965	36.084431
10^{-7}	0.1445373	3.6106331
10^{-8}	0.0144545	0.3610829
10^{-9}	0.0014458	0.0361160
10^{-10}	0.0001597	0.0039883
10^{-11}	0.0007239	0.0180835

Table 4: Differences between adjoint method and incremental quotients according to discretization step

Discretization steps used to approximate the partial derivatives (ε_i in equation (29)) were taken in the range $[10^{-5}, 10^{-11}]$. The values of the gradient of the objective function so obtained are compared in Table 4 with those provided by the adjoint method. The relative error is done with respect to the norm of the gradient calculated with the adjoint method. As it can be seen in this example, the error produced by difference quotients, which was reduced when the discretization



(a) Experiments 1-5

(b) Experiments 6-10

Figure 6: Observed and computed temperature versus time at reactor exit

step was chosen smaller, increases again when this is less or equal than 10^{-11}.

5. Conclusions

An integral method for solving a parameter identification problem in systems of partial differential equations, using the adjoint state for the calculation of the gradient of the objective function, has been proposed. The adjoint method provide better results, in terms of accuracy and computer time, the higher the number of parameters to be identified. This is because it only requires one solution of the state equation and the construction and solution of a linear system while, if the gradient of the objective function were calculated with a method of incremental quotients, this would require the solution of the state equation $N_p + 1$ times, being N_p the number of parameters to be identified. Moreover, the adjoint method is not only faster but avoids the problem of accuracy associated with the choice of the discretization step in the approximation of the partial derivatives using incremental quotients; in fact, it is exact.

This methodology has been applied to the identification problem in an unsteady plug-flow reactor where temperature is also calculated. Numerical solution obtained for a synthetic case was compared with the exact solution used to provide the experimental data for the optimization problem, with good agreement. Better fit could be obtained if more experimental data were considered or if appropriate weights were used in the objective function.

Actually, it is being successfully used in reactors with real data by the Spanish energy company Repsol in its Technology Center. Unfortunately, confidentiality issues prevent such experiments can be shown in this paper.

Acknowledgments

Part of this research was developed as an activity in the Joint Research Unit Repsol-ITMATI (code file: IN853A 2014/03) which is funded by FEDER, the Galician Agency for Innovation (GAIN) and the Ministry of Economy and Competitiveness in the framework of the Spanish Strategy for Innovation in Galicia. It was also financed by FEDER and the Spanish Ministry of Science and Innovation under research projects ENE2013-47867-C2-1-R and MTM2013-43745-R, and by FEDER and Xunta de Galicia under research project GRC13-014.

Bardow, A., Marquardt, W., 2004. Incremental and simultaneous identification of reaction kinetics: Methods and comparison. Chem. Eng. Sci. 59, 2673–2684. doi:10.1016/j.ces.2004.03.023.

Bartholomew-Biggs, M., Brown, S., Christianson, B., Dixon, L., 2000. Automatic differentiation of algorithms. J. Comput. Appl. Math. 124, 171–190. doi:10.1016/S0377-0427(00)00422-2.

Bhatt, N., Amrhein, M., Bonvin, D., 2011. Incremental Identification of Reaction and Mass - Transfer Kinetics Using the Concept of Extents. Ind. Eng. Chem. Res. 50, 12960–12974. doi:10.1021/ie2007196.

Bhatt, N., Kerimoglu, N., Amrhein, M., Marquardt, W., Bonvin, D., 2012. Incremental identification of reaction systems-A comparison between rate-based and extent-based approaches. Chem. Eng. Sci. 83, 24–38. doi:10.1016/j.ces.2012.05.040.

Brendel, M., Bonvin, D., Marquardt, W., 2006. Incremental identification of kinetic models for homogeneous reaction systems. Chem. Eng. Sci. 61, 5404–5420. doi:10.1016/j.ces.2006.04.028.

Carrizosa, E., Dražić, M., Dražić, Z., Mladenović, N., 2012. Gaussian variable neighborhood search for continuous optimization. Comput. Oper. Res. 39, 2206–2213. doi:10.1016/j.cor.2011.11.003.

Daescu, D.N., Sandu, A., Carmichael, G.R., 2003. Direct and adjoint sensitivity analysis of chemical kinetic systems with KPP: II - Numerical validation and applications. Atmos. Environ. 37, 5097–5114. doi:10.1016/j.atmosenv.2003.08.020.

Jameson, A., 1988. Aerodynamic design via control theory. J. Sci. Comput. 3, 233–260. doi:10.1007/BF01061285.

Lions, J., 1971. Optimal Control of Systems Governed by Partial Differential Equations. Springer Verlag, Berlin .

Lions, J., 1972. Nonhomogeneous boundary value problems and applications. Springer Verlag, Berlin .

Maute, K., Nikbay, M., Farhat, C., 2003. Sensitivity analysis and design optimization of three-dimensional non-linear aeroelastic systems by the adjoint method. Int. J. Numer. Meth. Engng. 56, 911–933. doi:10.1002/nme.599.

Meric, R., 1985. Coupled optimization in steady-state thermoelasticity. J. Thermal Stresses 8, 333–347. doi:10.1080/01495738508942240.

Mladenović, N., Dražić, M., Kovačevic-Vujčić, V., Čangalović, M., 2008. General variable neighborhood search for the continuous optimization. Eur. J. Oper. Res. 191, 753–770. doi:10.1016/j.ejor.2006.12.064.

Mladenović, N., Hansen, P., 1997. Variable neighborhood search. Comput. Oper. Res. 24, 1097–1100. doi:10.1016/S0305-0548(97)00031-2.

Neidinger, R., 2010. Introduction to Automatic Differentiation and MATLAB Object-Oriented Programming. SIAM Rev. 52, 545–563. doi:10.1137/080743627.

Pironneau, O., 1974. On Optimum Design in Fluid Mechanics. J. Fluid Mech. 64, 97–110. doi:10.1017/S0022112074002023.

Rodrigues, D., Billeter, J., Bonvin, D., 2015a. Incremental Model Identification of Distributed Two-phase Reaction Systems. IFAC-PapersOnLine 48, 266–271. doi:10.1016/j.ifacol.2015.08.192.

Rodrigues, D., Srinivasan, S., Billeter, J., Bonvin, D., 2015b. Variant and invariant states for chemical reaction systems. Comput. Chem. Eng. 73, 23–33. doi:10.1016/j.compchemeng.2014.10.009.

Sandu, A., Daescu, D.N., Carmichael, G.R., 2003. Direct and adjoint sensitivity analysis of chemical kinetic systems with KPP: Part I - Theory and software tools. Atmos. Environ. 37, 5083–5096. doi:10.1016/j.atmosenv.2003.08.019.

Srinivasan, S., Billeter, J., Bonvin, D., 2012. Extent-based incremental identification of reaction systems using concentration and calorimetric measurements. Chem. Eng. J. 207-208, 785–793. doi:10.1016/j.cej.2012.07.063.

Vautard, R., Beekmann, M., Menut, L., 2000. Applications of adjoint modelling in atmospheric chemistry: Sensitivity and inverse modelling. Environ. Model. Softw. 15, 703–709. doi:10.1016/S1364-8152(00)00058-X.