Numerical parameters estimation in models of pollutant transport with chemical reaction

Fabiana Zama [⋆]
Roberta Ciavarelli, Dario Frascari, and Davide Pinelli [⋆⋆]

Bologna University

Abstract. In this work we present an iterative algorithm for solving a parameter identification problem relative to a system of diffusion, convection and reaction equations. The parameters to estimate are the retardation factors, diffusivity, reaction and transport coefficients relative to a model of pollutant transport with chemical reaction. The proposed method solves the nonlinear least squares problem by means of a sequence of constrained optimization problems. The algorithm does not depend on the type of discretization method used to solve the state equation. The results reported in the numerical tests show the efficiency of the algorithm in terms of performance and solution quality.

1 Introduction

Parameter estimation is a very important topic in applied sciences and chemical engineering: an overview of methods and applications can be found in [3]. The modeling of pollutant transport with (bio)chemical reaction gives raise to partial differential systems which are usually very complex. Therefore there is a need for efficient algorithms for solving parameter estimation problems.

In this work we consider the parameter estimation problem for a system of reaction, diffusion and transport equations:

$$\frac{\partial U}{\partial t} = \nabla \cdot (D \nabla U) - V \nabla U + R(U), \tag{1}$$

where $U \equiv (u_1, u_2, \ldots, u_{Nc})^t$ represents the concentration in the state variable (x, t), $x \in [a, b]$, $t \in [0, T]$. The coefficients D and V represent the diffusion coefficient and the fluid velocity, while the reaction term is represented by the function R which depends on the solution U. Free flow condition is assumed at outlet boundary and pulse functions are given at inlet boundary. Homogenous initial conditions are assumed. A typical model of pollutant transport and biodegradation is illustrated by Frascari et al. [8].

The parameter estimation problem can be formalized as a constrained optimization problem:

$$\min_q J(U, q) \quad s.t. \quad c(U, q) = 0$$

[⋆] Department of Mathematics
[⋆⋆] Department of Chemical, Mining and Environmental Engineering

where $c(U, q) = 0$ represents the governing PDEs system (1) or state equation, the objective function $J(U, q)$ is the distance between the measurements $y \in Y$ and the solution U of the state equation $c(U, q) = 0$, corresponding to the parameter $q \in Q$ in the measurement points. Introducing the reduced observation operator $F : Q \longrightarrow Y$, that maps the unique solution of the state equation $U(q)$ into the measurements space Y, we define the equivalent nonlinear least squares problem:

$$\min_q \frac{1}{2} \|F(q) - y\| \tag{2}$$

where $\| \cdot \|$ is the euclidean norm throughout the paper. The ill posedness of the problem is well known [6] and different methods are proposed in the literature to obtain stable solutions in the presence of noisy data.

In this paper we propose an iterative method that solves the nonlinear least squares problem (2) by computing a sequence of constrained optimization problems. The proposed algorithm computes the solution q and the proper smoothing parameters, suitable to overcome the instability problems that arise in the solution of nonlinear least squares problems. The necessary starting values and tolerance parameters are computed using information obtained by the given measurements.

The paper is organized as follows. In section 2 we formulate the parameter estimation problem as an optimization problem and describe the discrete optimization algorithm. In section 3 the described algorithm is tested to evaluate both efficiency and solution quality.

2   The optimization algorithm

Aim of this section is to describe the discrete optimization algorithm for parameter estimation in the contest of transport and chemical reaction.

Given a set of measurements $\mathbf{y} \in \mathbb{R}^{Nm}$ relative to the concentration of compound u_i ($i = 1, \ldots, N_c$) at points $(t_j, x_j) \in [0, T] \times [a, b]$. The problem consists in finding the parameters $\mathbf{q} \in \mathbb{R}^{Np}$ whose image $F(\mathbf{q}) \in \mathbb{R}^{Nm}$ is the least squares approximation of the data $\mathbf{y}$. The discrete nonlinear least squares problem is given by:

$$\min_{\mathbf{q}} J(\mathbf{q}), \quad J(\mathbf{q}) \equiv \frac{1}{2} \|F(\mathbf{q}) - \mathbf{y}\| \tag{3}$$

By applying the first order conditions we obtain the nonlinear system:

$$\mathcal{J}_F^t(\mathbf{q})(F(\mathbf{q}) - \mathbf{y}) = 0, \quad \mathcal{J}_F(\mathbf{q}) \in \mathbb{R}^{Nm \times Np}, \quad (\mathcal{J}_F(\mathbf{q}))_{i,j} = \frac{\partial F_i}{\partial q_j}(\mathbf{q})$$

This problem is solved iteratively by setting an initial guess $\mathbf{q}^{(0)}$ and defining a direction $\mathbf{s}^{(k)}$ s.t. $\mathbf{q}^{(k+1)} = \mathbf{q}^{(k)} + \mathbf{s}^{(k)}$, $k \geq 0$, where $\mathbf{s}^{(k)}$ is obtained as solution of the linear system $H_k \mathbf{s}^{(k)} = -G_k^t \mathbf{r}_k$ where

$$G_k \equiv \mathcal{J}_F(\mathbf{q}^{(k)}), \quad \mathbf{r}_k \equiv F(\mathbf{q}^{(k)}) - \mathbf{y}$$

and H_k is the Hessian of the objective function $J(\mathbf{q}^{(k)})$ in (3). Although the dimensions of the linear system are small ($N_p \times N_p$), the computation of second order information is very expensive. In this work we use first order approximation given by the Gauss Newton method which is equivalent to defining $\mathbf{s}^{(k)}$ as solution of the linearized problem:

$$\min_{\mathbf{s}} \frac{1}{2}\|G_k\mathbf{s} + \mathbf{r}_k\|, \quad k \geq 0. \tag{4}$$

It is well known that instabilities often occur in the solution of this unconstrained linear least squares problem and it is necessary to introduce some smoothing technique to obtain stable solutions in the presence of data noise. A possible strategy is to add a constraint to the problem and compute the direction $\mathbf{s}^{(k)}$ as solution of the following constrained optimization problem:

$$\min_{\mathbf{s}} \frac{1}{2}\|G_k\mathbf{s} + \mathbf{r}_k\|, \quad s.t. \ \|\mathbf{s}\| \leq \Delta_k \tag{5}$$

where Δ_k represents the smoothness level required in the solution $\mathbf{s}$. The algorithm that we propose here allows us to solve problem (5) iteratively by computing the approximate solution of the equivalent dual lagrangian problem. The smoothed solution $\mathbf{s}^{(k)}$ is obtained by applying a few steps of Constrained Least Squares Regularization `CLSRit` algorithm [1] . Furthermore we define a suitable size of the initial trust region Δ_0, by using the problem data and we update it to compute $\Delta_k > 0$ by means of the trust region update method [4].

This algorithm can be viewed as an implementation of the Levemberg Marquardt Trust Region method, widely used both in constrained optimization and in the contest of parameter estimation [5], [4]. The Trust Region Constrained Least Squares Regularization `TRCLSR`, reported in table 1, can be split in the following steps:

- Computation of the initial trust region size Δ_0 (paragraph 2.1).
- Computation of the direction $\mathbf{s}^{(k)}$ (paragraph 2.2).
- Update of the trust region size Δ_k (paragraph 2.1).
- Solution update and stopping rules (paragraph 2.3).

The following input parameters are required: the starting value for the unknown parameters $\mathbf{q}^{(0)}$, the relative tolerance τ_J of the objective function J, the absolute tolerance τ_s of the step size $\|\mathbf{s}^{(k)}\|$, the problem data $\mathbf{y}$ and the function F that maps the parameters into the data space.

2.1   Update of Δ_k

An initial estimate of the size of Trust Region parameter Δ_0 can be obtained by computing a Tikhonov [7] regularized solution of problem (4) with regularization parameter $\alpha = 10^{-6}$ i.e.:

$$\Delta_0 = \|\bar{\mathbf{s}}^{(0)}\|, \quad \bar{\mathbf{s}}^{(0)} \ \ s.t. \ \ (G_0^t G_0 + \alpha I)\bar{\mathbf{s}}^{(0)} = G_0^t(F(\mathbf{q}^{(0)}) - \mathbf{y})$$

Algorithm 1 ($\mathtt{TRCLSR}(F, \mathbf{y}, \mathbf{q}^{(0)}, \tau_J, \tau_s)$)

Compute $U^{(0)}$ solving the PDE state equation $c(U^{(0)}, \mathbf{q}^{(0)}) = 0$;
Compute Jacobian G_0 as in subsection 2.4
Compute $\bar{\mathbf{s}}^{(0)}$ s.t. $(G_0^t G_0 + 1.e - 6I)\bar{\mathbf{s}}^{(0)} = G_0^t(F(\mathbf{q}^{(0)}) - \mathbf{y})$;
Set $\Delta_0 = \|\bar{\mathbf{s}}^{(0)}\|$;
$k = 0$
repeat
     Compute direction $\mathbf{s}^{(k)}$ as in subsection 2.2
     Compute Δ_{k+1} as in subsection 2.1
     Compute $\mathbf{q}^{(k+1)}$ as in subsection 2.3;
     Solve PDE state equation $c(U^{(k+1)}, \mathbf{q}^{(k+1)}) = 0$;
     Compute G_{k+1} (subsection 2.4)
     $k = k + 1$
until $\left(|J(\mathbf{q}^{(k+1))}) - J(\mathbf{q}^{(k)})| < \tau_J |J(\mathbf{q}^{(k)})| \;\; or \;\; \|\mathbf{s}^{(k)}\| < \tau_s \right)$

Table 1. Algorithm $\mathtt{TRCLSR}$

the value of the parameter α should be small enough to avoid the instability of the linear system without smoothing too much the solution. At each step k the update Δ_{k+1} is performed following the Trust Region algorithm (see algorithm 4.1 in [4]).

2.2   Computation of the direction $\mathbf{s}^{(k)}$

The direction $\mathbf{s}^{(k)}$ is computed by solving problem (5) in its equivalent lagrangian dual form [2]:

$$\max_\lambda \Phi(\lambda), \quad \Phi(\lambda) \equiv \min_\mathbf{q} \mathcal{L}(\mathbf{s}, \lambda). \tag{6}$$

where $\mathcal{L}$ is the lagrangian function: $\mathcal{L}(\mathbf{s}, \lambda) \equiv \frac{1}{2}\|G_k \mathbf{s} + \mathbf{r}_k\| + \lambda(\|\mathbf{s}\| - \Delta_k)$. Solving the dual problem (6) requires to find $\hat{\lambda}$ s.t. $\|\mathbf{s}(\hat{\lambda})\| = \Delta_k$ where $\mathbf{s}(\hat{\lambda})$ is the solution of the following linear system

$$(G_k^t G_k + \hat{\lambda}I)\mathbf{s}(\hat{\lambda}) = -G_k^t \mathbf{r}_k$$

The nonlinear equation $\|\mathbf{s}(\lambda)\| - \Delta_k = 0$ is solved by the hybrid method proposed in [1]. Given a starting value $\lambda_0 > 0$ s.t. $\mathbf{s}(\lambda_0) \le \Delta_k$ and a value $k_s > 2$ s. t. $0 < \lambda_0 < \lambda_{k_s} < \hat{\lambda}$, compute $(\mathbf{s}_\ell, \lambda_\ell)$ where $\lambda_0 = \|\mathbf{r}_k\|$ and

$$\lambda_\ell = \lambda_{\ell-1} + \mathcal{S}_{\ell-1}, \quad \ell \ge 1$$

where

$$\mathcal{S}_{\ell-1} = \begin{cases} sign(\|\mathbf{s}(\lambda_{\ell-1})\| - \Delta_k)\dfrac{\lambda_0}{2^{\ell-1}} & \ell \le k_s + 1 \\[2ex] \dfrac{\|\mathbf{s}(\lambda_{\ell-1})\| - \Delta_k}{\|\mathbf{s}(\lambda_{\ell-1})\| - \|\mathbf{s}(\lambda_{\ell-2})\|}(\lambda_{\ell-2} - \lambda_{\ell-1}) & \ell > k_s + 1 \end{cases} \tag{7}$$

where $\mathbf{s}(\lambda_\ell)$ satisfies: $(G_k^t G_k + \lambda_\ell I)\mathbf{s}(\lambda_\ell) = -G_k^t \mathbf{r}_k$ Under the given hypotheses it is proven that λ_ℓ converges to the solution $\hat{\lambda}$ of the dual problem (6) and the sequence $\{\mathbf{s}_\ell\}$ converges to $\hat{\mathbf{s}} \equiv \mathbf{s}(\hat{\lambda})$ which is the solution of the problem (5) [1].

2.3   Solution update $\mathbf{q}^{(k+1)}$ and stopping conditions

After the computation of each direction $\mathbf{s}^{(k)}$, the solution $\mathbf{q}^{(k+1)}$ is updated as follows:

$$\mathbf{q}^{(k+1)} = \begin{cases} \mathbf{q}^{(k)} + \mathbf{s}^{(k)} & \rho_k > \eta, \quad 0 \le \eta \le 0.25 \\ \mathbf{q}^{(k)} & otherwise \end{cases}$$

where the parameter ρ_k is given by

$$\rho_k = \frac{J(\mathbf{q}^{(k)}) - J(\mathbf{q}^{(k)} + \mathbf{s}^{(k)})}{m_k(0) - m_k(\mathbf{s}^{(k)})}$$

and it represents the ratio between the actual reduction $J(\mathbf{q}^{(k)}) - J(\mathbf{q}^{(k)} + \mathbf{s}^{(k)})$ and the reduction predicted in J by the model function m_k:

$$m_k(\mathbf{s}) \equiv J(\mathbf{q}^{(k)}) + \mathbf{s}^t G_k^t (F(\mathbf{q}^{(k)}) - \mathbf{y}) + \mathbf{s}^t G_k^t G_k \mathbf{s}.$$

The iterations are stopped when the relative reduction of the objective function J is below a given tolerance τ_J or when the increase of the step size $\|\mathbf{s}^{(k)}\|$ is less than a given threshold τ_s.

2.4   Computation of the Jacobian Matrix G_k

In our tests we used central finite difference approximation (FD). The i-th row of the Jacobian matrix $(G_k)_i$ is obtained as:

$$(G_k)_i = \frac{F(U(\mathbf{q}^{(k)} + \varepsilon \mathbf{e}_i)) - F(U(\mathbf{q}^{(k)} - \varepsilon \mathbf{e}_i))}{2\varepsilon}, \quad i = 1, \ldots, N_p$$

where $\mathbf{e}_i$ is the $i - th$ canonical basis vector, and $\varepsilon = 1.e - 4$. Each row $(G_k)_i$ requires the solution of two state equations to compute $U(\mathbf{q}^{(k)} + \varepsilon \mathbf{e}_i)$ and $U(\mathbf{q}^{(k)} - \varepsilon \mathbf{e}_i)$. Therefore the number of PDE solutions for each iteration k is $2 \cdot N_p + 1$.

3   Numerical results

In this section we test the proposed algorithm for the estimation of selected parameters in the time evolution model of Butane (C_B), Oxygen (C_O) and Chloroform (C_{CF}) concentrations in a column bioreactor. Taking advantage of symmetry, the problem is solved along one section of the spatial domain. The model

is given by the following system of diffusion transport and reaction equations representing the concentrations in time $t \in [0, T]$ and space variable $x \in [a, b]$:

$$
\begin{cases}
\delta_B \dfrac{\partial C_B}{\partial t} = -V \dfrac{\partial C_B}{\partial x} + (D_B + \alpha_L V) \dfrac{\partial^2 C_B}{\partial x^2} \\[2mm]
\delta_O \dfrac{\partial C_O}{\partial t} = -V \dfrac{\partial C_O}{\partial x} + (D_O + \alpha_L V) \dfrac{\partial^2 C_O}{\partial x^2} + K_O C_O \cdot \\[2mm]
\delta_{CF} \dfrac{\partial C_{CF}}{\partial t} = -V \dfrac{\partial C_{CF}}{\partial x} + (D_{CF} + \alpha_L V) \dfrac{\partial^2 C_{CF}}{\partial x^2}
\end{cases}
\tag{8}
$$

The parameter α_L represents the longitudinal dispersivity, V is the water velocity, δ_B and δ_{CF} are the Butane and Chloroform retardation factors and K_O is the abiotic oxygen consumption rate. The parameters D_B, D_O and D_{CF} represent the molecular diffusivities in water ($D_B = 1.03e - 9\ m^2 s^{-1}$, $D_O = 2.5e - 9\ m^2 s^{-1}$, $D_{CF} = 1.e - 9\ m^2 s^{-1}$). Inlet boundary conditions are given by $C_B(t, a) = B_1 p(t, 0.625, 1.875), C_O(t, a) = C_1 p(t, 4, 5.625), C_{CF}(t, a) = CF_1 p(t, 7.125, 8.125)$ where $p(t, \tau_1, \tau_2)$ represents the unit smoothed pulse function:

$$
p(t, \tau_1, \tau_2) = \begin{cases}
1/(1 + e^{-(t-\tau_1)/\tau}) & t \in [\tau_1 - \Delta_\tau, \tau_1 + \Delta_\tau], \quad \Delta_\tau = 0.321 \\
1 & t \in (\tau_1 + \Delta_\tau, \tau_2 - \Delta_\tau) \\
1 - 1/(1 + e^{-(t-\tau_2)/\tau}) & t \in [\tau_2 - \Delta_\tau, \tau_2 + \Delta_\tau] \\
0 & \text{otherwise}
\end{cases}
$$

Free flow boundary conditions are assumed at the outlet:

$$
\frac{\partial}{\partial x} C_B(t, b) = 0, \frac{\partial}{\partial x} C_O(t, b) = 0, \frac{\partial}{\partial x} C_{CF}(t, b) = 0
$$

and homogeneous initial conditions are assumed ($C_B(0, x) = 0, C_O(0, x) = 0, C_{CF}(0, x) = 0$).

The test problem is obtained by solving the state equation (8) in the domain $[a, b] \times [0, T]$ with $a = 0$, $b = 2$ and $T = 15$, using the Crank Nicolson method on a mesh of $M = X_s \times T_s$ uniformly spaced points. The measurements $\mathbf{y}$ are obtained by sampling on a uniform grid, with $N = T_m \times N_m$ points, each component (C_B, C_O, C_{CF}) of the solution of (8), computed with the parameter vector $\mathbf{q} = [V, \alpha_L, \delta_B, \delta_{CF}, K_O, B_1, O_1, CF_1]$, reported in table 2. In the following para-

Parameter	Units	Value	Parameter	Units	Value
V	md^{-1}	0.75	K_O	d^{-1}	0.035
α_L	m	0.12	B_1	$molm^{-3}$	0.26
δ_B	–	1.14	O_1	$molm^{-3}$	0.47
δ_{CF}	–	1.01	CF_1	$molm^{-3}$	3.4e-3

Table 2. Value of the parameters used to obtain measurement data $\mathbf{y}$.

graphs we report the results of experiments to test the algorithm with respect

δ_0	k	ℓ	$\|\mathbf{r}^k\|$	$\|\mathbf{q}^* - \mathbf{q}^k\|/\|\mathbf{q}^*\|$
8.e-2	5	95	5.607516e-3	4.647818e-4
1.e-1	5	92	5.607516e-3	4.647818e-4
5.e-1	7	107	5.607516e-3	4.647814e-4
6.e-1	8	123	5.607516e-3	4.647789e-4
7.e-1	12	182	5.607516e-3	4.647818e-4
8.e-1	3	71	4.603204e-1	8.000000e-1
9.e-1	3	71	5.018664e-1	9.000000e-1
1.2	4	89	3.544192e-1	1.304530

Table 3. Results obtained with 40×12 measurements by changing the starting guess $\mathbf{q}^0$ as in (9). The state equation is solved using a mesh size $M = 257 \times 128$.

to initial value $\mathbf{q}^0$, mesh size M and data noise. All the experiments were performed using MATLAB (R2010a) on a workstation with 6 Intel(R) Core(TM) i7 processors and 24 GByte ram. In all the tests reported in the following paragraphs, algorithm TRCLSR in table 1 has the following tolerance parameters: $\eta = 0$, $\tau_J = 10^{-7}$ and $\tau_s = 10^{-8}$.

3.1 Starting guess $\mathbf{q}^0$

In this experiment we apply the algorithm TRCLSR to estimate the parameters $\mathbf{q}^* = [V, \alpha_L, \delta_B, \delta_{CF}, K_O]$ by changing the starting guess $\mathbf{q}^0$ in order to get an assigned relative error δ_0, i.e. $\|\mathbf{q}^0 - \mathbf{q}^*\|/\|\mathbf{q}^*\| = \delta_0$.

Table 3 shows the relative error (fifth column) and the residual norm (fourth column) obtained with $8\% \leq \delta_0 \leq 120\%$ and mesh size $M = 32896$. The measurements are obtained by uniformly sampling C_B, C_O, C_{CF} on $N = T_m \times N_m$ points with $T_m = 40$ and $N_m = 4$. Computing the starting vector as follows

$$\mathbf{q}^0 = \mathbf{q}^* + \delta_0 \|\mathbf{q}^*\| \eta, \tag{9}$$

where η is a uniform random vector s.t. $\|\eta\| = 1$, we observe that, in order to have an accurate solution, the maximum allowed δ_0 is 70% and this value does not depend on the mesh size used to solve the state equation (8). Figure 1(a) shows the results of the same experiment carried out using meshes of increasing size $8256 \leq M \leq 424571$: it is clear a significant error increase when δ_0 is beyond the percentage allowed. When $\delta_0 \leq 70\%$ the algorithm converges to the optimal solution with residual norm and relative error independent on $\mathbf{q}^0$.

3.2 Estimate accuracy

The quality of parameters estimate improves by increasing the accuracy of the solution of the state equation (8). In table 4 are reported the residual norm (fifth column) and relative error (sixth column) obtained by increasing the size of the mesh $T_s \times X_s$ (first and second columns) used to solve (8). We can not get the same conclusion by increasing the number of measurements N. As it can be observed in figure 1(b), more than $N = 100$ measurement points do not lead to a sharp decrease of the relative error.

T_s	X_s	k	ℓ	$\|\mathbf{r}^k\|$	$\|\mathbf{q}^* - \mathbf{q}^k\|/\|\mathbf{q}^*\|$
257	128	7	107	5.607516e-3	4.647814e-4
513	256	7	100	9.653799e-4	9.616222e-5
917	463	8	104	1.030716e-4	1.055537e-5
1013	617	7	93	1.394869e-4	8.879562e-6

Table 4. Results obtained with 40×12 measurements by changing the mesh size $M = T_s \times X_s$ in the state equation solution.

3.3   Noisy Data

In this paragraph we analyze the solution in the presence of noise on the measured data. The noisy data $\mathbf{y}^\delta$ are computed so as to achieve a predetermined level of noise δ: $\mathbf{y}^\delta = \mathbf{y} + \delta\|\mathbf{y}\|\eta$, where η is a random vector with $\|\eta\| = 1$. In table 5 we report the results obtained by solving the state equation with mesh size $M = 424571$ ($X_s = 917, T_s = 463$) and measurements obtained by sampling C_B, C_O, C_{CF} on a uniform grid with $T_m = 40$ and $N_m = 4$ points. The noise δ, reported in column 1, is increased from 0.01% to 10% and we observe the same behavior in the residual norm (column fourth). The quality of the result is still good, as can be observed from the relative error (fourth column table 5) and the graph in figure 3. The plots in figure 2 show the relative error and residual

δ	k	ℓ	$\|\mathbf{r}^k\|$	$\|\mathbf{q}^* - \mathbf{q}^k\|/\|\mathbf{q}^*\|$
1.e-4	7	98	4.984954e-4	3.761874e-5
1.e-3	7	103	2.404441e-3	1.147139e-4
1.e-2	7	114	2.159983e-2	1.841462e-3
1.e-1	6	111	2.363836e-1	5.460248e-3

Table 5. Results obtained with noise added to 40×12 measurements and solving (8) with $M = 424571$.

convergence history relative to the case $M = 424571$ with noise $\delta = 1.e - 3$.

3.4   Algorithm efficiency

The efficiency of the algorithm can be measured by outer iteration numbers (k) and by the inner iterations ℓ, reported in tables 3, 4 and 5 (columns k and ℓ). We observe a small number of outer iterations (k) with respect to the inner iterations ℓ. The outer iterations (k) are computationally expensive since each step requires $2 \cdot N_p + 1$ solutions of the state equation (8), as shown in paragraph 2.4. Although the number of internal iterations is quite large it is relative to the the solution of a small size linear system $N_p \times N_p$ ($N_p = 5$), so it's generally inexpensive.
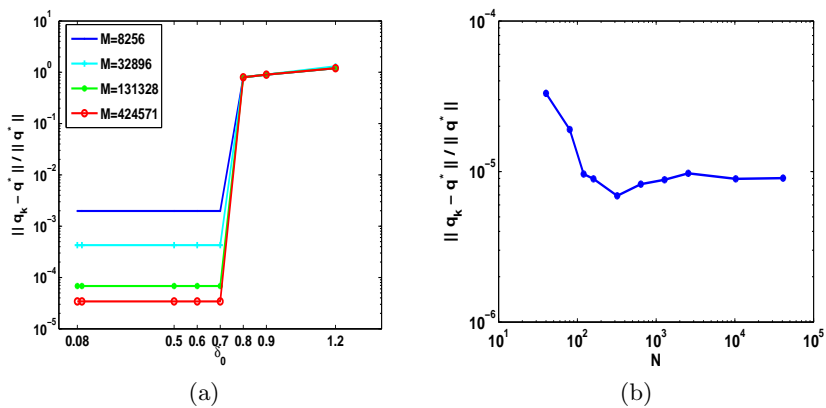
Fig. 1. (a) Relative errors obtained by changing δ_0 and using meshes of incresing size M. (b) Relative Errors obtained by increasing the measurements N (state equation solved with $M = 474571$).
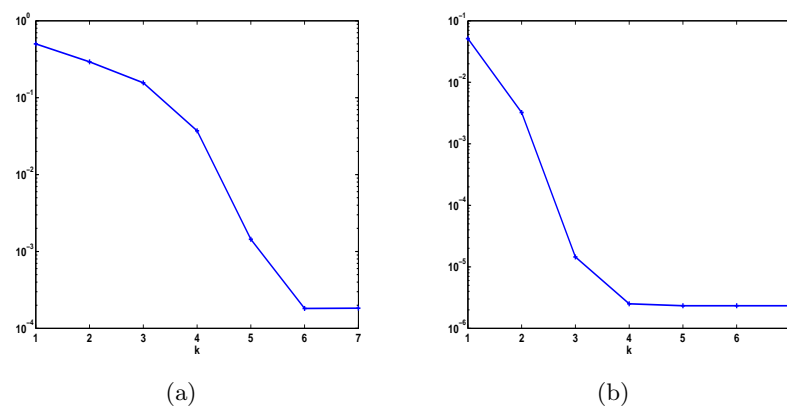

Fig. 2. Algorithm convergence in the case $M = 424571$ with noise $\delta = 1.e - 3$: (a) Relative Error: $\|\mathbf{q}^* - \mathbf{q}^k\|/\|\mathbf{q}^*\|$ (b) Residual norm: $\|\mathbf{r}_k\|$
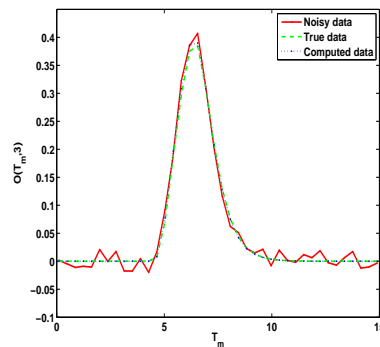
Fig. 3. Results for oxygen concentration at x = 1.1313. The `computed data` are relative to $\mathbf{q}_k$ parameters, the `noisy Data` are obtained by $\mathbf{y}^\delta$ with $\delta = 10\%$ and `true data` are computed using $\mathbf{q}^*$.

4   Conclusions

We can conclude that the algorithm `TRCLSR` computes accurate estimates of the parameters of the model (8), with up to 70% relative error on the initial guess. Furthermore, studies with data affected by noise show that the algorithm can determine accurate solutions with residual norm related to the level of added noise.

Future work will focus on the use of experimental data and more complex nonlinear models. The proposed optimization algorithm is independent of the type of discretization used to solve the state equation, therefore different PDE solutors will be tested.

References

1. E. Loli Piccolomini and F. Zama. An Iterative algorithm for large size Least-Squares constrained regularization problems. *Applied Mathematics and Computation*, (217), 2011.
2. J. Nash and A. Sopher. *Linear and nonlinear programming*. McGraw Hill, 1996.
3. P. Englezos and N. Kalogerakis. *Applied Parameter Estimation for Chemical Engineers*. Marcel Dekker, New York, Basel, 2001.
4. J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, 2006.
5. R. Beckrt and M. Braack and B. Vexler. *Compust. Theory and Modelling* (8) 2004.
6. G. Chavent, *Nonlinear Least Squares for Inverse Problems*, Springer, 2009.
7. A. N. Tikhonov, V. Y. Arsenin, *Solutions of Ill–Posed Problems*, *John Wiley & Sons*,1977
8. D. Frascari and M. Cappelletti and S. Fedi and A. Verboschi and R. Ciavarelli and M. Nocentini and D. Pinelli, Application of the growth substrate pulsed feeding technique to a process of chloroform aerobic cometabolism in a continuous-flow sand-filled reactor. *Process Biochemistry*, 2011, doi:10.1016/j.procbio.2011.08.019.