First Symposium on Inverse Problems and its Applications, Ixtapa 2010

Casa abierta al tiempo

UNIVERSIDAD AUTÓNOMA METROPOLITANA

Dr. Enrique Fernández Fassnacht
Rector General

Dr. Javier Velázquez Moctezuma
Rector de la Unidad Iztapalapa

Dr. José Antonio de los Reyes Heredia
Director de la División de Ciencias Básicas e Ingeniería, UAM–Iztapalapa

Dr. Mario Pineda Ruelas
Jefe del Departamento de Matemáticas, UAM–Iztapalapa

Primera edición, 2011

Universidad Autónoma Metropolitana

Prolongación Canal de Miramontes No. 3855,
Col. Ex Hacienda San Juan de Dios.
Delegación Tlalpan C. P. 14387 México D. F.

Universidad Autónoma Metropolitana Unidad Iztapalapa
División de Ciencias Básicas e Ingeniería

ISBN 978-607-477-505-1

Impreso en México/*Printed in Mexico*

First Symposium on Inverse Problems and its Applications

Joaquín Delgado
L. Héctor Juárez
Patricia Saavedra
M. Luisa Sandoval

Editors

Contents

Preface

The theory of inverse problems has a long history that has been partly motivated by applications. Inverse problems arise in many disciplines and currently it is a major field of study in science and engineering. This subject has become very important on the discussion of strategic issues of national relevance, such as the optimization of water resources, seismology, oil recovery, energy and technological development in biology, medicine, physiology, industrial problems, and economics, among others. The study of inverse problems is a research field which has grown considerably in recent years, where various disciplines, like mathematics, physics, and computational science and engineering, all converge. The importance acquired by inverse problems, and related issues has been reflected in the growing number of Mexican researchers who have been involved in the study of theoretical and computational aspects of them, as well as in their applications.

This volume collects some of the papers presented at the First Symposium on Inverse Problems and Applications which was held from the 6th to the 8th of January 2010, in Ixtapa Guerrero, México. The purpose of this event was to bring together specialists from different disciplines that have contributed to the development of inverse problems, related topics and applications. This Symposium was organized by the research group of Numerical Analysis and Mathematical Modeling from the Universidad Autónoma Metropolitana-Iztapalapa (UAM-I), and by the research group of Differential Equations and Mathematical Modeling from the Benemérita Universidad Autónoma de Puebla (BUAP). These two academic groups are part of the Network of Direct and Inverse Problems in Biology and Engineering, and they participate in the faculty improvement program (PROMEP), supported by the Secretaría de Educación Pública of México (SEP).

The Symposium was structured in six one-hour plenary talks, given by national and international specialists in the discipline, and twenty-three regular half-hour talks, given by researchers who have contributed to the development of inverse problems, mainly in Mexico. Nineteen Mexican graduate students and five researchers participated with a presentation of their research in the special session of posters. The total number of attendees at the meeting was about sixty. All speakers were invited to submit a contribution for publication in these proceedings, and their draft papers went through a refereeing process, typical of a mathematical research journal. Those which were accepted appear in this volume.

In closing, we would like to acknowledge all people that contributed to the success of the meeting and to this special proceedings issue: mainly the organizers and conference speakers, the invited plenary speakers, participating students and researchers, contributing authors, and referees. We would like to acknowledge the financial support of SEP through the PROMEP program (awarded to the Network of Direct and Inverse Problems in Biology and Engineering) and the generous financial

support of the President of the Universidad Autónoma Metropolitana through the project 12/2008. We finally appreciate the enthusiastic help and support from our colleagues from BUAP University as well as the editing work from Daniel Espinosa-Pérez.

The Editors:

Joaquín Delgado
L. Héctor Juárez
Patricia Saavedra
María Luisa Sandoval

Departamento de Matemáticas, UAM-Iztapalapa
Mexico City, March 2011.

Chapter 1

Olfactory Cilia: A Case Study in Inverse Modeling

Donald A. French[1], C.W. Groetsch[2]

Abstract

Some recent work on the inverse problem of estimating spatial distributions of ion channels in frog olfactory cilia is surveyed. The exposition takes the form of a case study of successive refinement of mathematical models. Some approximation techniques for solutions of inverse problems originating from the models are suggested. Numerical examples of reconstructions of ion channel distributions using simulated and laboratory data are presented and a number of mathematical questions arising from the models are briefly considered.

My interest in smells made me wonder how we recognized and categorized odors, how the nose could instantly delineate esters from aldehydes, or recognize a category such as terpenes, as it were, at a glance. Poor as our sense of smell was compared to a dog's ... there nevertheless seemed in humans to be a chemical analyzer at work that was at least as sophisticated as the eye or the ear.
Oliver Sacks, *Uncle Tungsten*

[1]Department of Mathematical Sciences, University of Cincinnati, Cincinnati, OH 45221-0025, french@math.uc.edu
[2]School of Science and Mathematics, The Citadel, Charleston, SC 29409-6420 USA, charles.groetsch@citadel.edu

1.1 Introduction

The sense of smell in humans is relatively underdeveloped, but in many animal species this sense is both very keen and essential to individual life and survival of the species. Smell can be an indispensable factor in finding food and mates, and it has an important function in identifying friends and foes. Modeling and understanding *olfaction*, that is the physiological basis of the sense of smell, is therefore a significant aspect of the study of evolutionary and systems biology. Furthermore, design of inanimate olfactory sensors, *artificial noses*, is being actively investigated with the aim of thwarting chemical and explosive attacks by terrorists in airports and public places. A better understanding of olfaction is therefore important on a number of theoretical and practical fronts.

This essay is offered as a case study of successively refined models of an inverse problem occurring in olfaction science. We survey some recent work [4], [5], [6] on a very narrow problem in olfaction – the identification of a specific morphological feature of olfactory *cilia*, namely, the spatial distribution of sodium ion channels along the length of an olfactory cilium. Our presentation takes the form of a tutorial on successive models for this inverse problem. The initial model makes gross assumptions about the mechanism for activating sodium ion channels. This model results in an explicit closed form solution of the inverse problem of determining ion channel density from laboratory current data. A more refined model which takes into account details of the diffusion of an activating ligand during a laboratory procedure is then developed and a simple numerical procedure for approximating solutions of the resulting linear inverse problem is proposed. The final refinement of the model considers both diffusion of the ligand through the cilium interior and binding of the ligand to receptor sites on the ion channels. This leads to an inverse problem in the form of a nonlinear integral equation with a curious kernel whose values are determined by solving a PDE. In addition to surveying some recent work on ion channel distributions, the intent of this presentation is to provide students with a case study of successively refined models for this specific inverse problem in olfactory science.

Olfactory cilia are tiny hair-like structures extending from the dendritic bulb of an olfactory receptor neuron. These cilia reside in the mucus coating the surface of the nasal membranes. They are extremely thin and delicate; a bundle of about 300 olfactory cilia roughly matches the thickness of a human hair. The ciliary length can range from about 100 to 500 times the ciliary diameter. Aside from their role in olfaction, cilia in general have risen in prominence recently as it has been recognized that they are important in the study of developmental cellular biology. Furthermore, ciliary defects have been linked to Bardet-Biedl syndrome and other disorders [9].

The function of olfactory cilia is to transduce odor stimuli in the nasal mucus into electrical signals that are fed into the nervous system. This transduction is accomplished by a depolarizing influx of sodium and calcium ions from the nasal mucus in which the cilia float through ion channels distributed along the membrane forming the lateral surface of the cilia. The resulting electrical potential difference between the exterior and interior of a cilium gives rise to a transmembrane current thereby transducing an odor stimulus into an electrical signal. The spatial distribution of ion channels along the length of the cilium is a structural feature of interest. In particular, the question arises whether the ion channels are uniformly distributed or locally clustered.

1.2 A Laboratory Procedure

Basic structural parameters of frog (specifically, *rana pipiens*) olfactory cilia, including the diffusion coefficient, have been determined in the laboratory [1]. S.J. Kleene and his research group in the University of Cincinnati Medical College have developed a laboratory procedure to activate sodium ion channels in these cilia. In this procedure the full length of a single cilium is drawn into a recording pipette and the cilium is detached from the olfactory bulb at its base. The interior of the pipette contains a solution of sodium ions that are initially separated from the ciliary interior by the membrane of the cilium. After the cilium is excised from the base the tip of the pipette, which coincides with the open base of the cilium, is immersed in a bath of cyclic adenosine monophosphate (or cAMP), an ion channel activating ligand (from Latin: *ligare*, to bind), of known concentration K. The cAMP ligand then diffuses into the interior of the cilium starting from the base of the cilium and proceeding toward the distal end, opening sodium ion channels as it proceeds. All the while the pipette records the increasing transmembrane current. In what follows L denotes the length of the cilium, D the diffusion coefficient, and $I(t)$ the transmembrane current recorded by the pipette at time t after the pipette is immersed in the bath of cAMP. The long extremely thin tubular structure which is the cilium captured by the pipette will be identified with the interval $[0, L]$ of the x-axis, $x = 0$ corresponds to the open base of the cilium and $x = L$ corresponds to the distal end of the cilium. The inverse problem of reconstructing the spatial distribution of sodium ion channels will consist of approximating the channel density $\rho(x)$ for $x \in (0, L)$ on the basis of the recorded current signal $I(t)$, $t \in (0, T)$, where T is the termination time of the experiment (typically a second or two).

1.3 A Naïve Model

Good modeling practice begins with the simplest assumptions in order to "test the waters" and perhaps uncover some gross features of expected solutions of more refined models. We begin with a very simple model in which the ligand concentration within the cilium at position x and time t, $c(x, t)$, is assumed to satisfy the linear diffusion equation

$$\frac{\partial c}{\partial t} = D\frac{\partial^2 c}{\partial^2 x}, \qquad 0 < x < L, \quad 0 < t < T, \tag{1}$$

with the initial and boundary conditions

$$c(x, 0) = 0, x \in (0, L); \qquad c(0, t) = K; \qquad \frac{\partial c}{\partial x}(L, t) = 0. \tag{2}$$

Since the cilium is extremely long relative to its diameter, as a first approximation we take $L = \infty$. The diffusion equation then has the closed form solution [2]:

$$c(x, t) = K\text{erfc}(x/(2\sqrt{D\,t})) \tag{3}$$

where erfc is the complementary error function

$$\text{erfc}(z) = 1 - \frac{2}{\sqrt{\pi}}\int_0^z \exp\left(-\tau^2\right) d\tau.$$

The process by which the ligand opens the ion channels must now be modeled. To begin, we make the very naïve assumption that a channel at a given position x is not opened until the ligand concentration at x reaches half the bulk concentration of the bath, after which it remains open. In other words, the propensity of a channel at location x to open is measured by $H(c(x,t) - K/2)$, where $H(\cdot)$ is the Heaviside unit step function centered at the origin:

$$H(s) = \begin{cases} 0 & , \quad s < 0 \\ 1/2 & , \quad s = 0 \\ 1 & , \quad s > 0 \end{cases}$$

The local transmembrane current $i(x,t)$ at position x and time t is then assumed proportional to $H(c(x,t) - K/2)\rho(x)$, the constant of proportionality j_0 being a conductivity factor (with units of, say, pico amps per channel), i.e.,

$$i(x,t) = j_0 H(c(x,t) - K/2)\rho(x).$$

The total transmembrane current at time t is then

$$I(t) = j_0 \int_0^\infty H(c(x,t) - K/2)\rho(x)\, dx. \tag{4}$$

By (**3**) the level curve $\{(x,t) : c(x,t) = K/2\}$ has the form $\{(x, x^2/\beta^2) : x > 0\}$, where β is the solution of

$$1/2 = \text{erfc}\left(\frac{\beta}{2\sqrt{D}}\right).$$

By (**4**), we then have

$$I(t) = j_0 \int_0^\infty H(\beta^2 t - x^2)\rho(x)\, dx = j_0 \int_0^{\beta\sqrt{t}} \rho(x)\, dx,$$

and hence

$$\frac{dI}{dt} = \frac{j_0 \beta}{2\sqrt{t}} \rho(\beta\sqrt{t}),$$

or, in terms of the variable $y = \beta\sqrt{t}$, the density distribution of ion channels has the explicit representation:

$$\rho(y) = \frac{2I'((y/\beta)^2)y}{j_0\beta^2}. \tag{5}$$

This ultra-simple model suggests a couple of features of the computed channel density distribution $\rho(\cdot)$ that one might expect to find in more refined models. First, the derivative appearing in (**5**) suggests that the estimation of $\rho(\cdot)$ based on measurements of the current signal $I(\cdot)$ will be unstable ([3], [7]), that is, the problem of reconstructing $\rho(\cdot)$ from $I(\cdot)$ is ill-posed. Secondly, synthesized analytical models of $I(\cdot)$ fashioned to mimic the general features of observed laboratory data lead via (**5**) to unimodal profiles of the channel density that peak nearer to the base rather than the distal end of the cilium (see Figure 1). This foreshadows results appearing later in this essay.

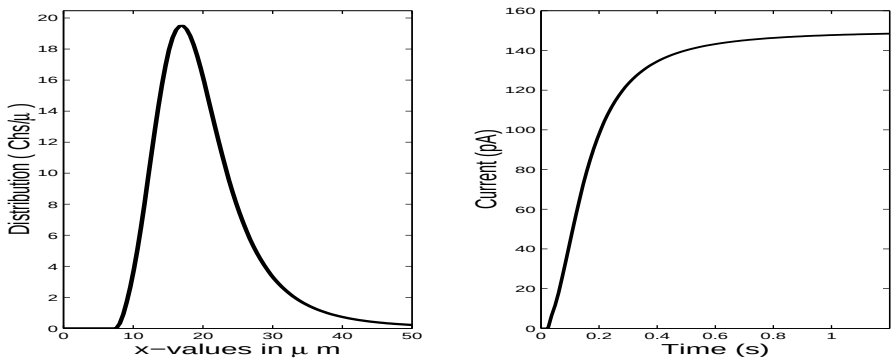

Figure 1.1: Analytical solution $\rho(x)$ from synthesized current $I(t)$

1.4 A More Realistic Linear Model

We now assume, more realistically, that the cilium has known finite length L. Also, we relax the assumption that led to the Heaviside kernel in (**4**) by adopting a model commonly used in the neuroscience community known as a *Hill function*:

$$F(c(x,t)) = \frac{c^n(x,t)}{c^n(x,t) + (K/2)^n} = \frac{1}{1 + (\frac{1}{2}K/c(x,t))^n},$$

instead of $H(c(x,t) - K/2)$. The positive exponent n is an experimentally determined parameter known as a Hill's exponent. Note that when $c(x,t) = K/2$, $F(c(x,t)) = 1/2$, while if n is large, $F(c(x,t)) \approx 1$ for $c(x,t) > K/2$, and $F(c(x,t)) \approx 0$ for $c(x,t) < K/2$. The model (**4**) then morphs into

$$I(t) = \int_0^L k(x,t)\rho(x)\,dx, \tag{**6**}$$

where $k(x,t) = j_0 F(c(x,t))$. One can see easily that $\frac{\partial k}{\partial x} < 0 < \frac{\partial k}{\partial t}$ and that the profiles $k(\cdot,t)$ "slide" from left to right as t increases as illustrated in Figure 2. Another notable feature of the spatial profiles of this kernel is that they "flatten out" for large x. For values of x in these flat tail regions it is clear that (**6**) can provide no detailed information on $\rho(x)$. These flat tails therefore contribute to the ill-posedness of the problem.

A direct approach to defeating the ill-posedness is to simply clip off the flat tails [6]. Given a small positive number ϵ, there is a unique positive number $T = T(\epsilon)$ satisfying

$$k(L,T) = \epsilon. \tag{**7**}$$

Furthermore, there is an increasing function $x_\epsilon : (0,T) \to (0,L)$ satisfying

$$k(x_\epsilon(t),t) = \epsilon.$$

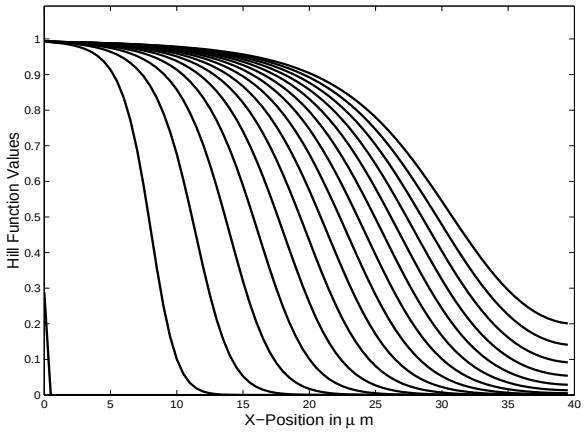

Figure 1.2: Propagation of the kernel as the ligand diffuses into a cilium.

We define the clipped kernel $k_\epsilon(\cdot,\cdot)$ on $[0,L]\times[0,T]$ by

$$k_\epsilon(x,t) = \begin{cases} k(x,t), & 0 \le x \le x_\epsilon(t) \\ 0, & x_\epsilon(t) \le x \le L \end{cases},$$

and replace (**6**) by the equation

$$I(t) = \int_0^L k_\epsilon(x,t)\rho_\epsilon(x)\,dx \tag{**8**}$$

or, equivalently:

$$I(t) = \int_0^{x_\epsilon(t)} k(x,t)\rho_\epsilon(x)\,dx$$

Under suitable smoothness conditions on $I(\cdot)$ it can be shown that (**8**) is equivalent to a Volterra integral equation of the second kind [6] and hence (**8**) can be expected to be better conditioned than (**6**).

Recall that the basic issue to be resolved is the form of a gross feature of the ion channel distribution: are the channels uniformly distributed or locally clustered? For this purpose a fairly simple approximation should suffice. We now describe a simple numerical procedure proposed in [5], [6]. Given a small positive number ϵ, determine $T = T(\epsilon)$ from (**7**). This may be accomplished by the bisection method applied to (**7**), but note that, owing to the nature of the kernel $k(\cdot,\cdot)$, each function evaluation requires a solution of the diffusion equation. For a relatively small positive integer n (n is intentionally kept small to mitigate ill-conditioning) form a time grid

$$0 < t_1 < t_2 < \cdots < t_n = T$$

and compute the spatial grid points

$$0 = x_0 < x_1 < \cdots < x_n = L$$

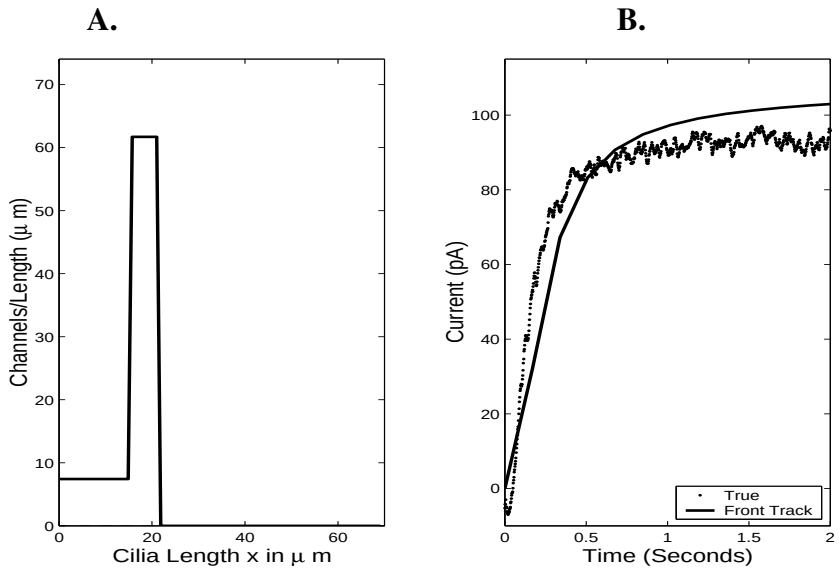

Figure 1.3: Linear Model – Laboratory Data (with $\epsilon = 0.35$, $N = 15$).

where $x_i = x_\epsilon(t_i), i = 1, \ldots, n-1$. Finally, form a piecewise constant approximation

$$\tilde{\rho_\epsilon} = \sum_{j=1}^{n} \rho_i \chi_{(x_{j-1}, x_j]}$$

where

$$\chi_{(x_{j-1},x_j]}(s) = \begin{cases} 1 & , \quad s \in (x_{j-1}, x_j] \\ 0 & , \quad s \notin (x_{j-1}, x_j] \end{cases}$$

is the characteristic function of the interval $(x_{j-1}, x_j]$. Collocation on the time grid is then applied to (**8**):

$$\begin{aligned} I(t_i) &= \textstyle\int_0^L k_\epsilon(x, t_i)\tilde{\rho}_\epsilon(x)\, dx \\ &= \textstyle\int_0^L k_\epsilon(x, t_i) \sum_{j=1}^{n} \rho_j \chi_{(x_{j-1},x_j]}(x)\, dx \\ &= \textstyle\sum_{j=1}^{i-1} \rho_j \int_0^{x_i} k(x, t_i)\chi_{(x_{j-1},x_j]}(x)dx + \rho_i \int_{x_{i-1}}^{x_i} k(x, t_i)\, dx, \end{aligned}$$

and therefore

$$\rho_1 = I(t_1) \left(\int_0^{x_1} k(x, t_1)\, dx \right)^{-1},$$

$$\rho_i = \left(I(t_i) - \sum_{j=1}^{i-1} \rho_j \int_0^{x_i} k(x, t_i)\chi_{(x_{j-1},x_j]}(x)\, dx \right) / \int_{x_{i-1}}^{x_i} k(x, t_i)\, dx$$

$$= \left(I(t_i) - \sum_{j=1}^{i-1} \rho_j \int_{x_{j-1}}^{x_j} k(x,t_i)\,dx \right) / \int_{x_{i-1}}^{x_i} k(x,t_i)\,dx.$$

Note that this provides a particularly simple explicit marching scheme for generating the coefficients of the approximation $\tilde{\rho}_\epsilon$. This scheme allows a stepwise imposition of a nonegativity constraint on the coefficients ρ_j, namely any computed coefficient that is negative may be replaced by zero.

1.5 A Nonlinear Refinement

An element of modeling that has been neglected so far is the binding of the ligand to receptor sites on the sodium ion channels. Diffusion and binding of cAMP as it progresses into the cilium is modeled by a nonlinear partial differential equation which depends on the channel distribution ρ. The local membrane potential satisfies a second-order boundary value problem which also depends on ρ and on the concentration of cAMP.

Binding of the ligand to receptors on the ion channels is modeled by a function $S(x,t)$ representing the total number, at time t, of bound cAMP molecules per unit length of the cilium in a thin cross-sectional ring at position x. This quantity is proportional to the channel density $\rho(x)$ and depends on the cAMP concentration $c(x,t)$. The dependence of binding on concentration is modeled by a Hill function, namely

$$S(x,t) = \beta\rho(x)\frac{c(x,t)^n}{c(x,t)^n + (K/2)^n}, \tag{9}$$

where β is a positive constant. This may be regarded as a conductance (the more binding, the more open channels, and hence the greater the ion migration), and hence the local transmembrane current $J(x,t)$ satisfies

$$J(x,t) \propto S(x,t)v(x,t),$$

where $v(x,t)$ is the transmembrane potential difference. The recorded current at time t is then

$$I(t) = \int_0^L J(x,t)\,dx = \gamma \int_0^L \rho(x)\frac{c(x,t)^n}{c(x,t)^n + (K/2)^n} v(x,t)\,dx \tag{10}$$

where the constant γ has units, say, of siemens per channel.

The binding has an effect on the concentration evolution as it takes cAMP out of the stream. Instead of (**3**), we now have

$$\frac{\partial c}{\partial t} = D\frac{\partial^2 c}{\partial x^2} - \alpha\frac{\partial S}{\partial t}$$

with a positive constant α. If we let

$$F(c) = \frac{c^n}{c^n + (K/2)^n},$$

then $S(x,t) = \beta\rho F(c(x,t))$ and hence, in view of (**9**), the diffusion equation (now nonlinear) may be written

$$\frac{\partial c}{\partial t} = \frac{D}{1 + \beta\alpha\rho F'(c)}\frac{\partial^2 c}{\partial x^2}. \tag{11}$$

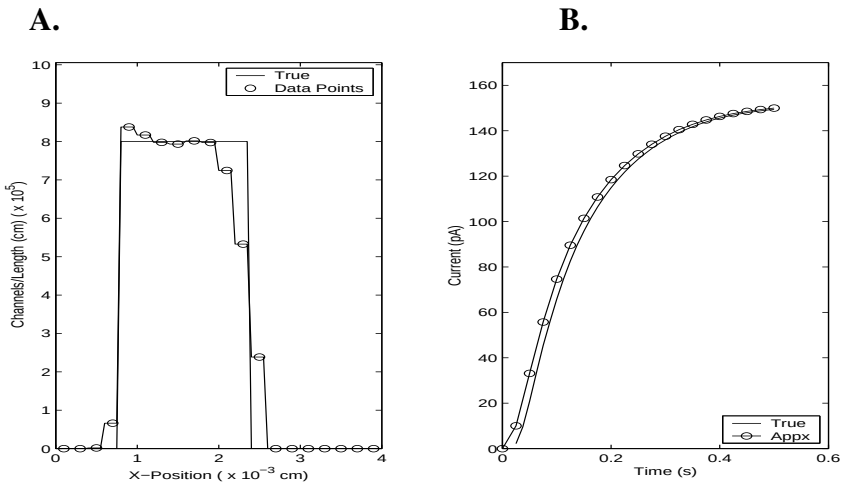

Figure 1.4: Nonlinear Model – Simulated Data **A.** True and approximate ρ functions. **B.** True and approximate current profiles.

Note that since $F'(c) \geq 0$, binding has the effect of slowing the diffusion (compare with (**1**)). To this we append the initial and boundary conditions

$$c(x,0) = 0, x \in (0,L); \qquad c(0,t) = K; \qquad \frac{\partial c}{\partial x}(L,t) = 0.$$

The other important thing to note about the nonlinear partial differential equation (**11**) is that the concentration $c(x,t)$, unlike in the case (**3**), now depends on the unknown density distribution $\rho(\cdot)$.

The equation for the local membrane potential $v(x,t)$, is a consequence of standard cable theory (see, e.g., [8]). Current between the two electrodes passes through two significant resistances in series: the resistance of the ciliary membrane, and a longitudinal resistance along the length of the solution filling the cilium. This second resistance varies with distance along the length of the cilium, as does the transmembrane voltage. We find that $v = v(x,t)$ satisfies

$$\frac{1}{r_a}\frac{\partial^2 v}{\partial x^2} = J \tag{12}$$

where $r_a = 1.49 \times 10^{11}(\text{S cm})^{-1}$ is the intracellular resistance to longitudinal current of the saline solution in the cilium (Here, r_a is formed by dividing the intracellular resistivity $R_i = 91.7\ \Omega-\text{cm}$ by the cross-sectional area A). We append the boundary conditions:

$$v(0,\cdot) = -50 \text{ mV} \text{ and } \frac{\partial v}{\partial x}(L,\cdot) = 0. \tag{13}$$

Our full model now consists of equations (**10**)–(**13**).

The dependence of the concentration and voltage on the channel density may be signified symbolically by $c(x,t,\rho(x))$ and $v(x,t,\rho(x))$, respectively. The integral equation (**10**) therefore has the

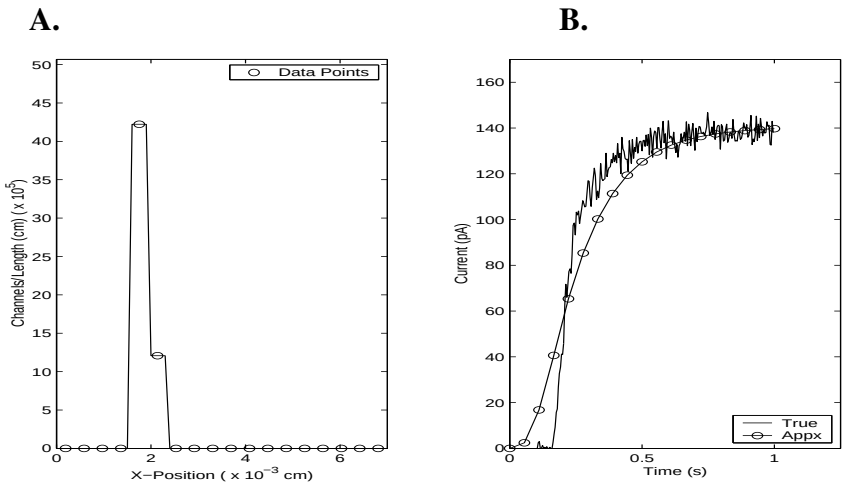

Figure 1.5: Nonlinear Model – Laboratory Data **A.** Approximate ρ function. **B.** True and approximate current profiles.

form

$$I(t) = \int_0^L \rho(x)k(x,t,\rho(x))\,dx, \tag{14}$$

where the kernel $k(\cdot,\cdot,\cdot)$ has the form

$$k(x,t,\rho(x)) = \gamma F(c(x,t,\rho(x)))v(x,t,\rho(x)), \qquad 0 < x < L, \qquad t > 0.$$

In [4] an iterative approach is taken to solving (**14**) in which a density distribution ρ_{Old} is assumed (initially, $\rho_{Old} = 0$), which is used to compute the kernel $k((\cdot,\cdot,\rho_{Old}(\cdot))$, and an updated distribution ρ_{New} is obtained by solving

$$I(t) = \int_0^L \rho_{New}(x)k(x,t,\rho_{Old}(x))\,dx. \tag{15}$$

Of course all computations are performed in a discrete setting. The calculations are carried out on three space-time meshes. A course mesh (10-20 subintervals) on the space $[0, L]$ is used to solve, for a given ρ_{Old}, the linear Fredholm integral equation (**15**) for the new approximation ρ_{New}. This low dimensionality is chosen to mitigate the ill-conditioned nature of the integral equation. A semi-implicit Crank-Nicolson method on a fine grid is used to solve (**11**) in order to approximate the concentration $c(x,t,\rho_{Old}(x))$ on $[0,L]\times[0,T]$, and (**12**) is solved by finite differences on the fine grid. A piecewise constant approximation for $\rho(\cdot)$ then leads, vis (**14**) to a linear system that is solved by Gauss-Seidel iteration, imposing a non-negativity constraint on the components at each step. Examples of typical computations, for synthesized and laboratory data, respectively, are shown in Figures 4 and 5. The naïve model, when applied to a synthetic current profile that mimics laboratory data, indicated a non-uniform channel distribution in which the the ion channels were clustered

nearer to the base than to the distal end of the cilium (see Figure 1). This was a hint of results to come from the refined models. Both the linear and the nonlinear models, when applied to laboratory data, consistently indicated a a clustering of the ion channels in the 30%-40% of the cilium length nearest to the base (see Figures 3 and 5). In any case, the mathematical models strongly suggest that sodium ion channels are not uniformly distributed along t entire length of the cilium, but tend instead to cluster nearer to the olfactory bulb.

1.6 Prospects

A number of purely mathematical questions are suggested by our treatment of approximation techniques in the models discussed above. In the linear model of Section 4, usable criteria for existence and uniqueness of solutions of the integral equations (**6**) and (**8**) for the special class of kernels considered are desirable. Also, estimates for the error $\rho - \rho_\epsilon$ are needed. The important question of errors in the function $I(t)$ has not been treated, nor has a regularization-type theory [3] relating the cut-off level ϵ with data error level δ been developed. Specifically, if the measured current data $I_\delta(t)$ is determined within an error level δ, that is, $\|I - I_\delta\| \leq \delta$, then a relation between the error level δ and the cut-off level ϵ, satisfying $\epsilon = \epsilon(\delta) \to 0$, as $\delta \to 0$, is desired with the property that $\rho^\delta_{\epsilon(\delta)} \to \rho$, where $\rho^\delta_{\epsilon(\delta)}$ is the solution of (**8**) with I replaced by I_δ. Investigation of specific schemes for the choice of $\epsilon(\delta)$ with corresponding convergence rates might then be possible.

No existence, uniqueness, or convergence results are known for the numerical method associated with the nonlinear model in Section 5. Such questions suggest a more general investigation of bivariate operators $K(\cdot,\cdot)$ acting on linear spaces X and Y of the form $K(\cdot,\cdot) : X \times X \to Y$, where for each $\varphi \in X$, $K(\cdot,\varphi) : X \to Y$ is linear and $K(\varphi,\cdot) : X \to Y$ is nonlinear. Given $\psi_0 \in Y$ and $\varphi_0 \in X$, one needs existence and uniqueness results for the solution φ of the equation

$$\psi_0 = K(\varphi, \varphi_0),$$

and convergence results for the iteration method

$$\psi_0 = K(\varphi_{n+1}, \varphi_n),$$

which is an abstract form of (**15**).

Bibliography

[1] C. Chen, T. Nakamura, and Y. Koutalos, *Cyclic AMP diffusion coefficient in frog olfactory cilia*. Biophysical Journal **76** (1999), 2861-2867.

[2] J. Crank, *The Mathematics of Diffusion*, (2nd Ed.), Oxford University Press, Oxford, 2001.

[3] H. Engl, M. Hanke and A. Neubauer, *Regularization of Inverse Problems*, Kluwer, Dordrecht, 1996.

[4] D.A. French, R.J. Flannery, C.W. Groetsch, W.B. Krantz, and S.J. Kleene, *Numerical approximation of solutions of a nonlinear inverse problem arising in olfaction experimentation*, Mathematical and Computer Modelling **43** (2006), 945-956.

[5] D.A. French and C.W. Groetsch, *Integral equation models for the inverse problem of biological ion channel distributions*, Journal of Physics: Conference Series **73** (2007) 102006, 1-10. (doi:10.1088/1742-6596/73/1/012006)

[6] D.A. French and C.W. Groetsch, *Numerical solution of a class of integral equations arising in a biological laboratory procedure*, Integral Methods in Science and Engineering, Volume 2, Computational Methods, (C. Constanda and E. Perez, Eds.), pp. 161-171, Birkhäuser, Boston, 2010.

[7] C.W. Groetsch, *Differentiation of approximately specified functions*, American Mathematical Monthly **98** (1991), 847-850.

[8] J. Keener and J. Sneyd, *Mathematical Physiology*, Springer, New York, 1998.

[9] G. Vogel, Betting on cilia, *Science* **310**(2005) (Issue 5746), 216-218.

Chapter 2

A Virtual Control Approach to the Numerical Solution of Some Elliptic Boundary Value Problems

Roland Glowinski,[1] Qiaolin He[2]

Abstract

Virtual control is a generic name for a variety of computational methods where one takes advantage of the special structure of the problem under consideration (natural or after some modifications) to derive solution methods inspired from Control Theory. This approach will be illustrated with two examples associated with the numerical solution of two families of linear elliptic boundary value problems, namely: (i) The solution of second order linear elliptic problems by the Dirichlet to Neumann domain decomposition method. (ii) The fictitious domain solution of second order linear elliptic boundary value problems with a Neumann or Robin boundary condition on some internal obstacle. The results of numerical experiments concerning problems from (ii) will be reported; they will confirm the capabilities of the virtual control approach.

2.1 Introduction

The terminology virtual control was coined by J.L. Lions in [1]; it characterizes problems or methods where one takes advantage of a special structure of the problem under consideration to apply solution

[1]Department of Mathematics, University of Houston, Houston, TX 77204, USA, and Institute of Advanced Study, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong

[2]Department of Mathematics, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong

methods inspired from Control Theory, despite the fact that the original problem is neither a control nor a design problem. Actually, the virtual control approach was used before the terminology was coined; evidences of such a fact can be found in, e.g., [2]–[7] (see also the references therein). The main goal of this article is to discuss two applications well-suited to the virtual control approach, namely: (i) The solution of second order linear elliptic boundary value problems by the Dirichlet to Neumann domain decomposition method. (ii) The fictitious domain solution of second order linear elliptic equations with a Neumann or Robin boundary condition on some internal obstacle. In [8], the solution of basin modeling related elliptic problems has been addressed via a virtual control/domain decomposition approach of type (i). On the other hand the methodology associated with the problems of type (ii) being much more recent, it is the one we will mainly discuss in this article. Our discussion will be completed by the results of numerical experiments concerning the least-squares/fictitious domain solution of linear elliptic test problems with a Robin boundary condition on an internal obstacle.

2.2 Virtual control and domain decomposition

Let us consider the following elliptic boundary value problem

$$\alpha\psi - \nabla \cdot \mathbf{A}\nabla\psi = f \quad \text{in} \quad \Omega, \quad \psi = g \quad \text{on} \quad \Gamma, \tag{1}$$

where Ω is a bounded domain of $\mathbf{R}^d$ and $\Gamma = \partial\Omega$. Problem (**1**) is elliptic if we assume that $\alpha \geq 0$ and if $\mathbf{A}$ is uniformly positive definite over $\mathbf{\Omega}$. Suppose that $\mathbf{\Omega}$ is decomposed according to Figure 2.1, below; if one denotes, $\forall i = 1, 2, \psi|_{\Omega_i}$ by ψ_i, $\alpha|_{\Omega_i}$ by α_i, $\mathbf{A}|_{\Omega_i}$ by $\mathbf{A}|_i$ and $f|_{\Omega_i}$ by f_i, there is equivalence between (**1**) and

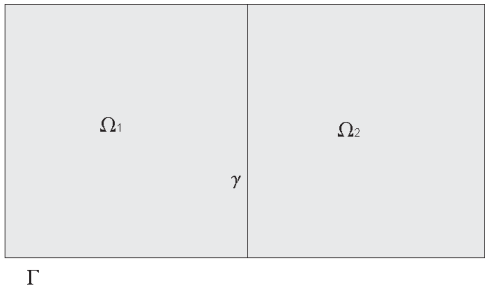

Figure 2.1: Domain decomposition of Ω

$$\begin{cases} \alpha_1\psi_1 - \nabla\cdot\mathbf{A}_1\nabla\psi_1 = f_1 \quad \text{in} \quad \Omega_1, \\ \psi_1 = g \quad \text{on} \quad \partial\Omega_1\cap\Gamma, \end{cases} \tag{2}$$

$$\begin{cases} \alpha_2\psi_2 - \nabla\cdot\mathbf{A}_2\nabla\psi_2 = f_2 \quad \text{in} \quad \Omega_2, \\ \psi_2 = g \quad \text{on} \quad \partial\Omega_2\cap\Gamma, \end{cases} \tag{3}$$

$$\psi_1 = \psi_2 \quad \text{on} \quad \gamma, \tag{4}$$

$$\mathbf{A}_1\nabla\psi_1\cdot\mathbf{n}_1 + \mathbf{A}_2\nabla\psi_2\cdot\mathbf{n}_2 = 0 \quad \text{on} \quad \gamma; \tag{5}$$

in (**5**), $\mathbf{n}_1$(resp., $\mathbf{n}_2$) denotes the unit normal vector at $\partial\Omega_1$(resp., $\partial\Omega_2$) outward to Ω_1(resp., Ω_2).

A virtual control formulation of (**2**)–(**5**) reads as follows:

$$\begin{cases} \alpha_1\psi_1 - \nabla\cdot\mathbf{A}_1\nabla\psi_1 = f_1 \quad \text{in} \quad \Omega_1, \\ \psi_1 = g \quad \text{on} \quad \partial\Omega_1\cap\Gamma, \\ \psi_1 = u \quad \text{on} \quad \gamma, \end{cases} \tag{6}$$

$$\begin{cases} \alpha_2\psi_2 - \nabla\cdot\mathbf{A}_2\nabla\psi_2 = f_2 \quad \text{in} \quad \Omega_2, \\ \psi_2 = g \quad \text{on} \quad \partial\Omega_2\cap\Gamma, \\ \mathbf{A}_2\nabla\psi_2\cdot\mathbf{n}_2 = -\mathbf{A}_1\nabla\psi_1\cdot\mathbf{n}_1 \quad \text{on} \quad \gamma, \end{cases} \tag{7}$$

$$\psi_2 - \psi_1 = 0 \quad \text{on} \quad \gamma. \tag{8}$$

The system (**6**)–(**8**) has the structure of an exact (virtual) controllability problem for the elliptic system (**6**), (**7**)(in the sense of , e.g., [9]). The least-squares/conjugate gradient solution of systems closely related to (**6**)–(**8**) has been discussed in [8], an article on the numerical solution of elliptic problems from basin modeling.

In the following sections, we are going to discuss the solution, by a similar approach, of linear elliptic problems with Neumann or Robin boundary conditions on an obstacle internal to Ω.

2.3 A family of linear elliptic problems with Neumann or Robin boundary conditions

We focus on the Robin problem only since it contains Neumann' s as a particular case. Let Ω and ω be two bounded domains of $\mathbf{R}^d$, such that $d \geq 1$ and $\omega \subset \Omega$. We denote by Γ and γ the boundaries of Ω and ω, respectively. A typical related configuration has been depicted in Figure 2.2.

The linear elliptic problem (of the Robin-Dirichlet type) that we consider reads as follows:

$$\alpha\psi - \mu\nabla^2\psi = f \quad \text{in} \quad \Omega\setminus\overline{\omega}, \tag{1}$$

$$\psi = g_0 \quad \text{on} \quad \Gamma, \tag{2}$$

$$\mu\left(\frac{\partial\psi}{\partial n} + \frac{\psi}{l}\right) = g_1 \quad \text{on} \quad \gamma, \tag{3}$$

where: α (resp., μ) is a non–negative (resp., a positive) constant, $f \in L^2(\Omega\setminus\overline{\omega}), g_0 \in H^{3/2}(\Gamma), g_1 \in H^{1/2}(\gamma)$, $\mathbf{n}$ is the unit normal vector at γ pointing outward of $\Omega\setminus\overline{\omega}$ and l is a characteristic distance.

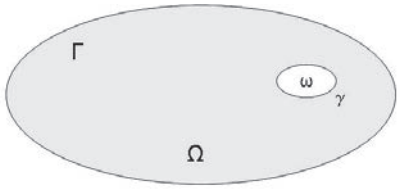

Figure 2.2: Problem geometry

We assume that Ω is convex and/or has a smooth boundary and γ is smooth. Problem (**1**)–(**3**) has a unique solution in $H^2(\Omega \setminus \overline{\omega})$ which is also the solution of the following linear variational problem:

$$\psi \in H^1(\Omega \setminus \overline{\omega}), \quad \psi = g_0 \quad \text{on} \quad \Gamma, \tag{4}$$

$$\alpha \int_{\Omega\setminus\overline{\omega}} \psi\varphi dx + \mu \int_{\Omega\setminus\overline{\omega}} \nabla\psi \cdot \nabla\varphi dx + \frac{\mu}{l} \int_{\gamma} \psi\varphi d\gamma$$
$$= \int_{\Omega\setminus\overline{\omega}} f\varphi dx + \int_{\gamma} g_1\varphi d\gamma, \quad \forall\varphi \in V_0, \tag{5}$$

where $dx = dx_1...dx_d$ and $V_0 = \{\varphi | \varphi \in H^1(\Omega \setminus \overline{\omega}), \quad \varphi = 0 \quad \text{on} \quad \Gamma\}$.

2.4 A virtual control/fictitious domain formulation of problem (1)–(3)

We proceed as follows to define a fictitious domain variant of problem (**1**)–(**3**):

(i) With $v \in L^2(\omega)$ we associate $\widetilde{f}(v)$ defined by

$$\widetilde{f}(v) \in L^2(\Omega), \quad \widetilde{f}(v)|_{\Omega\setminus\overline{\omega}} = f, \quad \widetilde{f}(v)|_{\omega} = v, \tag{1}$$

and then the solution $\{\psi_1, \psi_2\}$ of the following elliptic system

$$\alpha\psi_1 - \mu\nabla^2\psi_1 = \widetilde{f}(v) \quad \text{in} \quad \Omega, \quad \psi_1 = g_0 \quad \text{on} \quad \Gamma, \tag{2}$$

$$\alpha\psi_2 - \mu\nabla^2\psi_2 = v \quad \text{in} \quad \omega, \quad \mu\frac{\partial\psi_2}{\partial n} = \frac{\mu}{l}\psi_1 - g_1 \quad \text{on} \quad \gamma. \tag{3}$$

Both problems (**2**) and (**3**) have a unique solution in $H^1(\Omega)$ and $H^1(\omega)$, respectively (actually, ψ_1 and ψ_2 have both the H^2–regularity).

(ii) We define $\mathbf{A} : L^2(\omega) \to H^1(\omega)$ by

$$\mathbf{A}(v) = (\psi_2 - \psi_1)\,|_{\omega}\,. \tag{4}$$

Operator $\mathbf{A}$ is clearly affine and continuous.

(iii) We observe that if v verifies $\mathbf{A}(v) = 0$, we then have $\psi_2 = \psi_1$ on ω and it is easy to see that the H^2–regularity of ψ_1 and ψ_2 implies that $\psi_1|_{\Omega\setminus\overline{\omega}} = \psi$, where ψ is the solution of problem (**1**)–(**3**). We still have to show that indeed the functional equation

$$\mathbf{A}(u) = 0 \tag{5}$$

has a solution and to discuss its numerical solution.

In [10], the authors have proved the following

Theorem 2.4.1. *The functional equation* (**5**) *has infinity of solutions.*

Remark 2.4.2. *Problem* (**5**) *can be viewed as an exact controllability problem in the sense of [9]. From a practical point of view, problem* (**5**) *has infinitely many solutions. If a conjugate gradient algorithm operating in $L^2(\omega)$ is applied to a least–squares variant of* (**5**) *with* 0 *as initial guess, we can expect convergence to the unique solution of* (**5**) *of minimal norm in $L^2(\omega)$.*

Remark 2.4.3. *If there exists a 'natural' extension $\widetilde{f}$ of f over Ω (that is, $\widetilde{f} \in L^2(\Omega)$ and $\widetilde{f}\,|_{\Omega\setminus\overline{\omega}} = f$) we can replace $\widetilde{f}(v)$ in* (**1**) *by $\widetilde{f} + v\chi_\omega$. This will modify slightly the least–squares formulation and conjugate gradient algorithm to be discussed in the following parts of this article.*

2.5 A least–squares formulation of problem (5)

A 'reasonable' least–squares formulation of problem (**5**) reads as follows:

$$\begin{array}{llll} \text{Find} & u \in L^2(\omega) & \text{such that} \\ J(u) \leq J(v), & \forall & v \in L^2(\omega), \end{array} \tag{1}$$

with

$$J(v) = \frac{1}{2}\int_\omega \left[\alpha|\psi_2 - \psi_1|^2 + \mu|\nabla(\psi_2 - \psi_1)|^2\right] dx, \tag{2}$$

where in (**2**), ψ_1 and ψ_2 are the solutions of (**2**) and (**3**), respectively.

The functional $J(\cdot)$ defined by (**2**) and (**2**), (**3**) is clearly convex and C^∞ over $L^2(\omega)$. Moreover, if u is a solution of (**1**) it is characterized by

$$DJ(u) = 0, \tag{3}$$

where $DJ(\cdot)$ is the differential of $J(\cdot)$. In order to solve the least-squares problem (**1**), via (**3**), we advocate a conjugate gradient algorithm operating in the control space $L^2(\omega)$; such an algorithm requires the knowledge of $DJ(\cdot)$. Using standard techniques of Control Theory (see, e.g., [9] and [10]) we can show that to obtain the differential $DJ(v)$ of the least-squares functional J at $v \in L^2(\omega)$ we can proceed as follows:

(i) Solve problem (**2**) to obtain ψ_1.

(ii) Solve problem (**3**) to obtain ψ_2.

(iii) Solve the following (adjoint) Dirichlet problem

$$p_1 \in H_0^1(\Omega), \tag{4}$$

$$\int_\Omega [\alpha p_1\varphi + \mu\nabla p_1\cdot\nabla\varphi]\,dx = \int_\omega [\alpha(\psi_1-\psi_2)\varphi + \mu\nabla(\psi_1-\psi_2)\cdot\nabla\varphi]\,dx$$
$$+\frac{\mu}{l}\int_\gamma(\psi_2-\psi_1)\varphi d\gamma, \quad \forall \quad \varphi\in H_0^1(\Omega). \tag{5}$$

(iv) We have then

$$DJ(v) = (p_1-\psi_1)|_\omega + \psi_2. \tag{6}$$

2.6 On the conjugate gradient solution of the least–squares problem (1)

Taking advantage of relation (**6**), we can solve the least-squares problem (**1**) using a conjugate gradient algorithm operating in $L^2(\omega)$; such an algorithm reads as follows:

$$u^0 \quad \text{is given in} \quad L^2(\omega). \tag{1}$$

Solve the following three elliptic boundary value problems:

$$\alpha\psi_1^0 - \mu\nabla^2\psi_1^0 = \widetilde{f}(u^0) \quad \text{in} \quad \Omega, \quad \psi_1^0 = g_0 \quad \text{on} \quad \Gamma, \tag{2}$$

$$\alpha\psi_2^0 - \mu\nabla^2\psi_2^0 = u^0 \quad \text{in} \quad \omega, \quad \mu\frac{\partial\psi_2^0}{\partial n} = \frac{\mu}{l}\psi_1^0 - g_1 \quad \text{on} \quad \gamma, \tag{3}$$

$$\begin{cases} p_1^0 \in H_0^1(\Omega), \\ \int_\Omega \left[\alpha p_1^0\varphi + \mu\nabla p_1^0\cdot\nabla\varphi\right]dx = \int_\omega \left[\alpha(\psi_1^0-\psi_2^0)\varphi + \mu\nabla(\psi_1^0-\psi_2^0)\cdot\nabla\varphi\right]dx \\ +\frac{\mu}{l}\int_\gamma(\psi_2^0-\psi_1^0)\varphi d\gamma, \quad \forall \quad \varphi\in H_0^1(\Omega). \end{cases} \tag{4}$$

Set

$$g^0 = (p_1^0-\psi_1^0)|_\omega + \psi_2^0, \tag{5}$$

and

$$w^0 = g^0. \tag{6}$$

For $n \geq 0$, u^n, g^n and w^n being known, the last two different from 0, we compute u^{n+1}, g^{n+1} and, if necessary, w^{n+1} as follows:
Solve

$$\alpha\overline{\psi}_1^n - \mu\nabla^2\overline{\psi}_1^n = w^n\chi_\omega \quad \text{in} \quad \Omega, \quad \overline{\psi}_1^n = 0 \quad \text{on} \quad \Gamma, \tag{7}$$

$$\alpha\overline{\psi}_2^n - \mu\nabla^2\overline{\psi}_2^n = w^n \quad \text{in} \quad \omega, \quad \mu\frac{\partial\overline{\psi}_2^n}{\partial n} = \frac{\mu}{l}\overline{\psi}_1^n \quad \text{on} \quad \gamma, \tag{8}$$

and

$$\begin{cases} \overline{p}_1^n \in H_0^1(\Omega), \\ \int_\Omega \left[\alpha \overline{p}_1^n \varphi + \mu \nabla \overline{p}_1^n \cdot \nabla \varphi\right] dx = \int_\omega \left[\alpha(\overline{\psi}_1^n - \overline{\psi}_2^n)\varphi + \mu \nabla(\overline{\psi}_1^n - \overline{\psi}_2^n) \cdot \nabla \varphi\right] dx \\ + \frac{\mu}{l} \int_\gamma (\overline{\psi}_2^n - \overline{\psi}_1^n)\varphi d\gamma, \quad \forall \quad \varphi \in H_0^1(\Omega). \end{cases} \tag{9}$$

Set

$$\overline{g}^n = (\overline{p}_1^n - \overline{\psi}_1^n)|_\omega + \overline{\psi}_2^n, \tag{10}$$

and compute

$$\rho_n = \frac{\int_\omega |g^n|^2 dx}{\int_\omega \overline{g}^n w^n dx} \tag{11}$$

$$u^{n+1} = u^n - \rho_n w^n, \tag{12}$$

$$g^{n+1} = g^n - \rho_n \overline{g}^n. \tag{13}$$

If $\frac{\int_\omega |g^{n+1}|^2 dx}{\int_\omega |g^0|^2 dx} \leq tol$, take $u = u^{n+1}$ and $\psi = \psi_1^{n+1}|_{\Omega \setminus \overline{\omega}}$; else, compute

$$\gamma_n = \frac{\int_\omega |g^{n+1}|^2 dx}{\int_\omega |g^n|^2 dx} \tag{14}$$

and

$$w^{n+1} = g^{n+1} + \gamma_n w^n. \tag{15}$$

Do $n = n + 1$ and return to (**7**).

Other stopping criteria and functions $\tilde{f}$ can be used.

2.7 Finite element approximation of the least-squares problem (1)

From now on, we suppose that $\omega \subset \Omega \subset \mathbf{R}^2$ and that:

(i) Ω is convex and/or Γ is smooth.

(ii) $\gamma (= \partial\omega)$ is smooth.

Next, we introduce two finite element triangulations, namely, $\mathcal{T}_{h_1}$ for Ω and $\mathcal{T}_{h_2}$ for ω; we do not assume that $\mathcal{T}_{h_1}$ and $\mathcal{T}_{h_2}$ are nested. We still denote by Ω and ω the two polygonal domains associated with $\mathcal{T}_{h_1}$ and $\mathcal{T}_{h_2}$. From $\mathcal{T}_{h_1}$ and $\mathcal{T}_{h_2}$ we define the following spaces:

$$V_{h_1} = \{\varphi | \varphi \in C^0(\overline{\Omega}), \varphi|_T \in P_1, \forall T \in \mathcal{T}_{h_1}\}, \tag{1}$$

$$V_{h_2} = \{\varphi | \varphi \in C^0(\overline{\omega}), \varphi|_T \in P_1, \forall T \in \mathcal{T}_{h_2}\}, \tag{2}$$

$$V_{0h_1} = \{\varphi | \varphi \in V_{h_1}, \varphi = 0 \quad \text{on} \quad \Gamma\}. \tag{3}$$

The spaces V_{h_1}, V_{0h_1} and V_{h_2} are finite dimensional spaces approximating $H^1(\Omega)$, $H_0^1(\Omega)$ and $H^1(\omega)$, respectively. Actually, we also use V_{h_2} to approximate the control space $L^2(\omega)$. Concerning the approximation of the least-squares problem (**1**), we advocate (with $\mathbf{h} = \{h_1, h_2\}$):

$$\begin{array}{l} u_{\mathbf{h}} \in V_{h_2}, \\ J_{\mathbf{h}}(u_{\mathbf{h}}) \leq J_{\mathbf{h}}(v), \quad \forall \quad v \in V_{h_2}, \end{array} \tag{4}$$

where

$$J_{\mathbf{h}}(v) = \frac{1}{2} \int_{\omega} \left[\alpha |\psi_2 - \pi_2 \psi_1|^2 + \mu |\nabla(\psi_2 - \pi_2 \psi_1)|^2 \right] dx. \tag{5}$$

In (**5**),

- $\pi_2 : C^0(\overline{\Omega}) \rightarrow V_{h_2}$ is the standard linear interpolation operator associated with the vertices of $\mathcal{T}_{h_2}$.
- ψ_1 is the solution of the following fully discrete approximate Dirichlet problem:

$$\left\{ \begin{array}{l} \psi_1 \in V_{h_1}, \quad \psi_1 = g_{0\mathbf{h}} \quad \text{on} \quad \Gamma, \\ \int_{\Omega} \left[\alpha \psi_1 \varphi + \mu \nabla \psi_1 \cdot \nabla \varphi \right] dx = \int_{\Omega} f_{\mathbf{h}} \varphi dx + \int_{\omega} v \pi_2 \varphi dx, \\ \forall \quad \varphi \in V_{0h_1}; \end{array} \right. \tag{6}$$

  in (**6**), $g_{0\mathbf{h}}$ is an approximation of g_0 belonging to the space γV_{h_1} span by the traces on Γ of the functions of V_{h_1}, while $f_{\mathbf{h}} \in V_{h_1}$ vanishes at the vertices of $\mathcal{T}_{h_1}$ contained in ω and approximates f over $\Omega \setminus \overline{\omega}$.

- ψ_2 is the solution of the following fully discrete approximate Neumann problem:

$$\left\{ \begin{array}{l} \psi_2 \in V_{h_2}, \\ \int_{\omega} \left[\alpha \psi_2 \varphi + \mu \nabla \psi_2 \cdot \nabla \varphi \right] dx = \int_{\omega} v \varphi dx \\ + \frac{\mu}{l} \int_{\gamma} (\pi_2 \psi_1 - g_{1\mathbf{h}}) \varphi d\gamma, \quad \forall \quad \varphi \in V_{h_2}. \end{array} \right. \tag{7}$$

Computing $D(J_{\mathbf{h}}(v)), \forall v \in V_{h_2}$, and deriving a discrete variant of the conjugate gradient algorithm (**1**)–(**15**), are both (relatively) easy tasks, which have been discussed in [10].

2.8 Numerical experiments

The numerical experiments reported below have been performed at the Hong-Kong University of Science and Technology (HKUST). The test problem that we consider is defined as follows:

- $\Omega = (0,4) \times (0,4), \omega = \left\{ \{x_1, x_2\} \mid \left(\frac{x_1 - 2}{0.25}\right)^2 + \left(\frac{x_2 - 2}{0.125}\right)^2 < 1 \right\}$.
- $\alpha = 100, \mu = 0.1, l = 0.1$.
- The data f, g_0 and g_1 have been chosen so that the solution ψ of the corresponding Dirichlet-Robin problem (**1**)–(**3**) is given by

$$\psi(x_1, x_2) = x_1^3 - x_2^3.$$

In order to apply the fictitious domain methodology discussed in the preceding sections we have employed finite element triangulations $\mathcal{T}_{h_1}$ and $\mathcal{T}_{h_2}$ of the following types:

(i) $\mathcal{T}_{h_1}$ is uniform, as shown in Figure 2.3. Here h_1 denotes the length of the edges of the triangles of $\mathcal{T}_{h_1}$, adjacent to the right angles.

(ii) $\mathcal{T}_{h_2}$ is unstructured isotropic as shown in Figure 2.4. Here h_2 denotes the length of the largest edge(s) of $\mathcal{T}_{h_2}$.

When applying the discrete analogue of the conjugate gradient algorithm (**1**)–(**15**) to the solution of the discrete least-squares problems, we initialized with $u^0 = 0$ and took $tol = 10^{-10}$ in the stopping criterion. The various elliptic problems encountered at each iteration of the above conjugate gradient algorithm were solved by an algebraic multi-grid method.

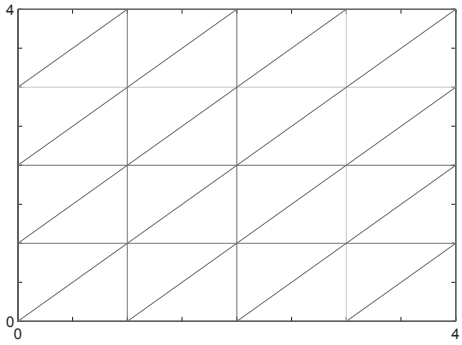

Figure 2.3: A typical triangulation of Ω.

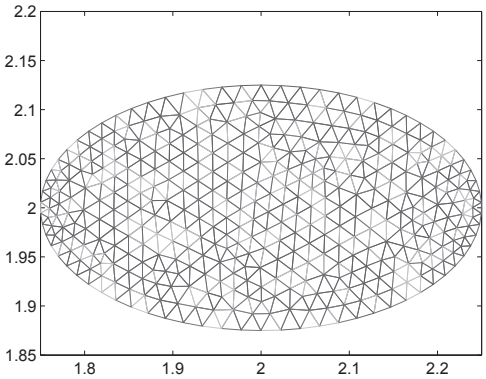

Figure 2.4: A typical triangulation of ω.

With h_2 fixed at $1/40$ we obtain the results reported in Table 2.1. In Table 1, nit denotes the number of conjugate gradient iterations necessary to achieve convergence, while $\| \psi_{\mathbf{h}} - \psi \|_{0,\infty}$ and $\| \psi_{\mathbf{h}} - \psi \|_{0,2}$ (resp., $|\psi_{\mathbf{h}} - \psi|_{1,2}$) denote the L^∞ and L^2 norms (resp., the L^2-norm of the gradient) of the approximation error $\psi_{\mathbf{h}} - \psi$.

In order to further investigate how these various errors behave as functions of the ratio $\frac{h_1}{h_2}$ additional numerical experiments have been performed (this time with $h_2 = 1/20$); their results have been reported in Table 2.2.

Table 2.1: Numerical results obtained with ($h_2 = 1/40$)

h_1	nit	$\| \psi - \psi_h \|_{0,\infty}$	$\| \psi - \psi_h \|_{0,2}$	$\|\psi - \psi_h\|_{1,2}$
1/5	34	0.1046	7.8370E-03	0.2855
1/10	61	2.1845E-02	1.9028E-03	0.1423
1/20	59	4.5840E-03	4.7015E-04	7.1089E-02
1/40	68	1.1385E-03	1.1708E-04	3.5518E-02

Table 2.2: . Numerical results obtained with ($h_2 = 1/20$)

h_1	nit	$\| \psi - \psi_h \|_{0,\infty}$	$\| \psi - \psi_h \|_{0,2}$	$\|\psi - \psi_h\|_{1,2}$
1/10	33	2.1845E-02	1.9038E-03	0.1424
1/20	36	4.5840E-03	4.6807E-04	7.1063E-02
1/40	114	2.1385E-03	1.0163E-04	3.5434E-02
1/80	85	3.1514E-03	5.2854E-05	1.7532E-02

The above results suggest that:

(i) If $h_1 = h_2 = h$, the various approximation errors are of optimal order, that is $\| \psi - \psi_h \|_{0,\infty} = O(h^2)$, $\| \psi - \psi_h \|_{0,2} = O(h^2)$ and $|\psi - \psi_h|_{1,2} = O(h)$.

(ii) These orders are preserved if $h_1 \geq h2$.

(iii) To be on the safe side take $h_1 = h_2$.

(iv) If $h_1 = h_2 (= h)$ the number of iterations necessary to achieve convergence varies like $1/h$.

Some additional comments are in order; among them:

(a) In ref. [10] it has been shown that the above methodology can be applied to the solution of parabolic problems; using a more sophisticated stopping criterion taking into account the evolutionary character of the problem under consideration, the number of iterations necessary to achieve convergence, at each time step, reduces to a small value after few time steps.

(b) The ultimate goal of these investigations is the numerical simulation of particulate flow when a slip boundary condition takes place at the interface fluid–particles.

Acknowledgments: The authors acknowledge the support of the Institute for Advanced Study (IAS) at The Hong Kong University of Science and Technology. The work is partially supported by grants from RGC CA05/06.SC01 and RGC-CERG 603107.

Bibliography

[1] J.L. LIONS, *Virtual and effective control for distributed systems and decomposition of everything*, Journal d'Analyse Mathématique **80**(1), 257–297, 2000.

[2] BRISTEAU, M.O., R. GLOWINSKI, J. PÉRIAUX, P. PERRIER & O.PIRONNEAU, *On the numerical solution of nonlinear problems in fluid dynamics by least-squares and finite element methods. (I) Least-squares formulations and conjugate gradient solution of the continuous problems* , Comp. Meth. Appl. Mech. Eng., **17/18**, 619–657, 1979.

[3] GLOWINSKI, R., *Numerical Methods for Nonlinear Variational Problems*, Springer, New-York, NY, 1984 (2nd edition, Springer, Berlin, 2008).

[4] DINH, Q.V., R. GLOWINSKI, J. PÉRIAUX & G. TERRASSON, *On the coupling of viscous and inviscid models for incompressible fluid flows via domain decomposition.* In Domain Decomposition Methods for Partial Differential Equations, R. Glowinski, G. H. Golub, G. Meurant & J. Périaux, eds., SIAM, Philadelphia, 350–369, 1988.

[5] GLOWINSKI, R., J. PÉRIAUX & G. TERRASSON, *On the coupling of viscous and inviscid models for compressible fluid flows via domain decomposition.* In Domain Decomposition Methods for Partial Differential Equations, T. F. Chan, R. Glowinski, J. Périaux, O. Widlund, eds., SIAM, Philadelphia, 64-97, 1990.

[6] AUCHMUTY, G., E. J. DEAN, R. GLOWINSKI & S. C. ZHANG, *Control methods for the numerical computation of periodic solutions of autonomous differential equations.* In Control Problems for Systems Described by Partial Differential Equations and Applications, I. Lasiecka & R. Triggiani, eds., Lecture Notes in Control and Information Sciences, **Vol. 97**, Springer-Verlag, Berlin, 64-89, 1987.

[7] BRISTEAU, M.O., R. GLOWINSKI & J. PÉRIAUX, *Controllability methods for the computation of time-periodic solutions; applications to scattering*, J. Comp. Phys., **147**, 265-292, 1998.

[8] ZAKARIAN, E. & R. GLOWINSKI, *Domain decomposition methods applied to sedimentary basin modeling*, Mathematical & Computer Modelling, **30**(9-10), 153-178, 1999.

[9] GLOWINSKI, R. J.L.LIONS & J. HE, *Exact and Approximate Controllability for Distributed Parameter Systems: A Numerical Approach*, Cambridge University Press, Cambridge, UK, 2008.

[10] GLOWINSKI, R. & Q. HE, *A least-squares /fictitious domain method for linear elliptic problems with Robin boundary conditions*, Commun. Comput. Phys., 2010, to appear.

Chapter 3

Identificación de conductividad cuando depende de la presión

A. Fraguela[1], J. A. Infante[2], Á. M. Ramos[2], J. M. Rey[2]

Abstract

En la presente comunicación se estudia el problema inverso que consiste en determinar la conductividad térmica de un medio, cuando ésta depende del tiempo y se conoce la evolución de la temperatura en algunos puntos del medio convenientemente ubicados. Este tipo de situaciones se plantean en contextos de tecnología de alimentos, cuando se utilizan procesos térmicos a altas presiones y se requiere determinar cómo varía la conductividad del medio con la presión ejercida, con el objetivo de poder controlar el proceso.

Cuando se pueden despreciar ciertos fenómenos convectivos, el fenómeno se modeliza mediante la ecuación de transferencia de calor con un término fuente que depende de la temperatura y del incremento de la presión, ecuación que se completa con adecuadas condiciones inicial y de contorno. Para geometrías cilíndricas del medio, se puede efectuar una simplificación en la que se considera como dominio una bola de $\mathbb{R}^2$ de radio arbitrario, centrada en el origen.

Se presenta un resultado de unicidad de solución del problema inverso. Para ello, en primer lugar, se encuentra una representación adecuada de la solución, en función de sus valores en la frontera del dominio. A continuación, se demuestra que el coeficiente de conductividad térmico está unívocamente determinado si se conoce el valor de la temperatura en un punto de la frontera de la bola y en cualquier otro punto interior.

Por otra parte, se presenta una metodología que permite llevar a cabo la identificación

[1]Benemérita Universidad Autónoma de Puebla. Puebla, México
[2]Universidad Complutense de Madrid. Madrid, España

aproximada del coeficiente de conductividad térmica, así como resultados numéricos para datos sintéticos. Finalmente, se describe un algoritmo de regularización en el que se caracteriza la solución del problema inverso mediante una adecuada propiedad de minimización.

keywords: Modelling, Simulation, High pressure, Parameter identification, Thermal conductivity.

3.1 Introducción

Los tratamientos que combinan altas presiones con temperaturas moderadas se modelizan mediante ecuaciones en las que aparecen distintos parámetros físicos cuyo valor, si bien se suele conocer a presión atmosférica, está por determinar para otros valores de la presión. En este trabajo fijamos nuestra atención en las ecuaciones de transferencia de calor que pueden aparecer en los modelos matemáticos de estos procesos (véanse, por ejemplo, [6] y [7]). Supondremos que la conductividad térmica depende sólo de la presión: $k = k(P)$; esta hipótesis es adecuada, por ejemplo, en los procesos en los que el rango de temperaturas es moderado y no hay cambio de fase. Nos planteamos el problema de identificar esta función $k(P)$ a partir de ciertas mediciones experimentales de la temperatura. Si para determinar las mediciones se realiza un experimento en el que la curva de presión P considerada sea inyectiva (por ejemplo, si se trata de una función estrictamente creciente) el problema de identificar $k(P)$ es equivalente a identificar $k(t) = k(P(t))$.

Trabajaremos con un modelo simplificado en el que se está suponiendo que la muestra es de alimento sólido y se encuentra contenida en una cámara presurizada, de forma cilíndrica, con una razón de rellenado muy alta, por lo que no se tienen en cuenta fenómenos de convección (esta situación es una de las analizadas en [6] y en [7]). Además, se supone que la muestra de alimento intercambia calor con las paredes del equipo que, por simplicidad, se asume que están a temperatura conocida y dada por una función $T^e(t)$, siendo la única fuente de calor la correspondiente al aumento de presión. El coeficiente h de intercambio de calor se supone también conocido. Por último, suponemos que la conducción de calor en la dirección vertical es despreciable y que, por tanto, nuestro interés se centra en conocer lo que ocurre a una cierta altura de la muestra que es en la que estarán colocados los termopares para realizar las mediciones.

Teniendo en cuenta todas estas consideraciones, el modelo simplificado para el que abordamos el problema de identificación del coeficiente de conductividad térmica es el siguiente:

$$\begin{cases} \varrho C_p \dfrac{\partial T}{\partial t} - k(t)\Delta T = \alpha P'(t) T & \text{en } B_R \times (0, t_{\mathrm{f}}) \\ k(t)\dfrac{\partial T}{\partial \vec{n}} = h\left(T^e(t) - T\right) & \text{en } \partial B_R \times (0, t_{\mathrm{f}}) \\ T = T_0 & \text{en } B_R \times \{0\}, \end{cases} \tag{1}$$

donde $R > 0$ y $t_{\mathrm{f}} > 0$. Aquí, $B_R \subset \mathbb{R}^2$ denota la bola de centro 0 y radio R; $\alpha \geq 0$ es el coeficiente de dilatación de la muestra considerada; $\varrho, C_p \in \mathbb{R}$ son su densidad y calor específico, respectivamente; $P \in \mathcal{C}^1([0, t_{\mathrm{f}}])$ representa la presión que se ejerce con el equipo; $k \in \mathcal{C}([0, t_{\mathrm{f}}])$, con $k(t) \geq k_0 > 0$, es el coeficiente de conductividad desconocido; $T^e \in \mathcal{C}([0, t_{\mathrm{f}}])$ denota la temperatura a la que se encuentra la pared interior de la cámara presurizada; $\vec{n}$ es el vector unitario

normal exterior a la frontera de la bola B_R; $h > 0$ denota el coeficiente de intercambio de calor y T_0 es la temperatura inicial de la muestra de alimento sólido, que supondremos constante. El objetivo es identificar la función k conociendo mediciones de la temperatura T en ciertos puntos de la bola B_R. Puesto que, como ya se ha dicho, los valores a presión atmosférica de los coeficientes que aparecen en estas ecuaciones suelen ser conocidos, nosotros supondremos que la presión en el instante inicial es la atmosférica y que, por tanto, $k(0)$ es un dato del problema.

Para más detalles sobre este tipo de modelos y las unidades de cada uno de sus parámetros y funciones, remitimos a las referencias [5] y [6]. En [2], [3], [5] pueden encontrarse otras modelizaciones de este tipo de fenómenos, que también dan pie al estudio y resolución de problemas inversos.

3.2 Expresión de la solución en función de sus valores en el borde

Denotaremos

$$X = \left\{\varphi \in \mathcal{C}^{2,1}\big(B_R \times (0,t_{\mathrm{f}})\big) \cap \mathcal{C}^{1,0}\big(\overline{B_R} \times [0,t_{\mathrm{f}}]\big)\right\},$$

es decir, el conjunto de las funciones que son de clase dos en espacio y de clase uno en tiempo en el cilindro $B_R \times (0,t_{\mathrm{f}})$, y que tienen un orden de regularidad una unidad inferior en la adherencia de dicho cilindro.

Teorema 1: *Si la función k es lipschitziana en $[0,t_{\mathrm{f}}]$, la derivada de la función P es, en dicho intervalo, hölderiana de orden $\beta \in (0,1)$ y se verifica la condición de compatibilidad*

$$T^e(0) = T_0,$$

entonces el problema (**1**) *tiene una única solución (clásica) $T \in X$. Además, T es radial.* □

Dada una función radial $v \in X$ denotaremos por

$$\overline{v} : [0,R] \times [0,t_{\mathrm{f}}] \to \mathbb{R}$$

la función que toma los valores de v en cada circunferencia, es decir,

$$\overline{v}(r,t) = v(x,t) \text{ para todo } x \in B_R \text{ con } |x| = r \text{ y } t \in [0,t_{\mathrm{f}}].$$

Por otra parte, denotando

$$\mathcal{L}(\phi) = \varrho C_p \frac{\partial \phi}{\partial t} - k(t)\Delta\phi,$$

entonces el *adjunto formal* del operador $\mathcal{L}$ viene dado por

$$\mathcal{L}^*(\phi) = -\varrho C_p \frac{\partial \phi}{\partial t} - k(t)\Delta\phi.$$

Los operadores $\mathcal{L}$ y $\mathcal{L}^*$ están relacionados mediante la

Identidad de Lagrange: *Para cada par de funciones $\phi, \psi \in X$ se verifica que*

$$\int_0^{t_\mathrm{f}} \int_{B_R} \big(\mathcal{L}(\phi)\psi - \phi\mathcal{L}^*(\psi)\big)dxdt = \varrho C_p \int_{B_R} \big(\phi(x,t_\mathrm{f})\psi(x,t_\mathrm{f}) - \phi(x,0)\psi(x,0)\big)\,dx$$

$$- \int_0^{t_\mathrm{f}} k(t) \int_{\partial B_R} \left(\frac{\partial \phi}{\partial \vec{n}}\psi - \phi\frac{\partial \psi}{\partial \vec{n}}\right) dxdt. \quad \square$$

El siguiente resultado proporciona un principio de comparación para el problema asociado al operador $\mathcal{L}$:

Principio de comparación: *Sean $v_1, v_2 \in X$ verificando*

$$\begin{cases} \mathcal{L}(v_1) \le \mathcal{L}(v_2) & \text{en } B_R \times (0,t_\mathrm{f}) \\ k(t)\dfrac{\partial v_1}{\partial \vec{n}} + hv_1 \le k(t)\dfrac{\partial v_2}{\partial \vec{n}} + hv_2 & \text{en } \partial B_R \times (0,t_\mathrm{f}) \\ v_1 \le v_2 & \text{en } B_R \times \{0\}. \end{cases}$$

Entonces

$$v_1(x,t) \le v_2(x,t),\ (x,t) \in \overline{B_R} \times [0,t_\mathrm{f}]. \quad \square$$

A continuación vamos a encontrar una representación integral de la solución del problema (**1**) en función de sus valores en el borde (que posteriormente supondremos obtenidos por medio de mediciones experimentales).

Teorema 2: *Utilizando la notación*

$$m(t) = \overline{T}(R,t),\ \gamma(r,\theta) = R^2 - 2Rr\cos\theta + r^2,\ K(s) = \int_s^{t_\mathrm{f}} k(z)dz,$$

$$g(t,\tau) = \frac{1}{K(\tau) - K(t)}\ \ y\ \ Q(t,\tau) = e^{\frac{\alpha}{\varrho C_p}(P(t)-P(\tau))},$$

la solución del problema (**1**) *puede expresarse como*

$$\overline{T}(r,t) = T_0 Q(t,0) + \frac{Rh}{4\pi}\int_0^t \big(T^e(\tau) - m(\tau)\big)\,Q(t,\tau)g(t,\tau)\int_0^{2\pi} e^{-\frac{\varrho C_p}{4}\gamma(r,\theta)g(t,\tau)}\,d\theta d\tau$$

$$+ \frac{R}{2\pi}\int_0^t \left(m'(\tau) - \frac{\alpha}{\varrho C_p}\,m(\tau)P'(\tau)\right) Q(t,\tau)\int_0^{2\pi} \frac{R - r\cos\theta}{\gamma(r,\theta)}\,e^{-\frac{\varrho C_p}{4}\gamma(r,\theta)g(t,\tau)}\,d\theta d\tau$$

para $r \in [0,R)$ y $t \in [0,t_\mathrm{f}]$. $\square$

3.3 Unicidad de solución del problema inverso

En esta sección abordamos la unicidad de solución del problema inverso, entendiendo por ello que la función $k(t)$ esté unívocamente determinada en un intervalo $[0, t_{\mathrm{f}}]$ por los valores que tome la función $\overline{T}(r,t)$ en $r = R$ y en otro punto $r_0 \in [0, R)$, para todo $t \in [0, t_{\mathrm{f}}]$.

Es de reseñar que esto no siempre ocurre. Por ejemplo, si la temperatura exterior evoluciona en la forma

$$T^e(t) = T_0 e^{\frac{\alpha}{\varrho C_p}(P(t)-P(0))}$$

la propia función $T(t) = T^e(t)$ es solución del problema directo (**1**), independientemente de cuál sea la función k.

Para asegurar la unicidad de solución del problema de identificar el coeficiente de conductividad, restringiremos el contexto en que se plantea el problema y trabajaremos bajo las siguientes hipótesis:

(H1) $T^e(t) \equiv T_0$ para todo $t \in [0, t_{\mathrm{f}}]$.

(H2) La presión P es lineal y estrictamente creciente. Por tanto, $P' \equiv \beta > 0$.

El Principio de Comparación permite obtener los siguientes resultados técnicos, que ayudarán a demostrar la unicidad del coeficiente k.

Lema 1: *Bajo las hipótesis* **(H1)**, **(H2)** *y suponiendo que k es lipschitziana en $[0, t_{\mathrm{f}}]$ con $k \geq k_0 > 0$, se verifica que*

$$T_0 \leq T(x,t) \leq T_0 e^{\frac{\alpha}{\varrho C_p}(P(t)-P(0))} \tag{2}$$

para todo $(x,t) \in \overline{B_R} \times [0, t_{\mathrm{f}}]$. Además, para cada $t^* \in (0, t_{\mathrm{f}})$ existe $\tau^* \in (0, t^*)$ tal que

$$T_0 < m(\tau^*). \quad \square$$

Lema 2: *Bajo las hipótesis* **(H1)**, **(H2)**, *supongamos que $k \in \mathcal{C}^1([0, t_{\mathrm{f}}])$, $k \geq k_0 > 0$ y que se verifica la condición*

$$k'(t) \leq k(t)\,\frac{f'(t)}{f(t) - T_0} = k(t)\,\frac{\alpha\beta}{\varrho C_p}\,\frac{1}{e^{\frac{\alpha\beta}{\varrho C_p}t} - 1}. \tag{3}$$

cuando $t \in [0, t_{\mathrm{f}}]$. Entonces,

$$\frac{\partial T}{\partial t}(x,t) - \frac{\alpha}{\varrho C_p}\,T(x,t)P'(t) \leq 0 \tag{4}$$

para todo $(x,t) \in \overline{B_R} \times [0, t_{\mathrm{f}}]$. En particular, para $t \in (0, t_{\mathrm{f}})$ se verifica

$$m'(t) - \frac{\alpha}{\varrho C_p}\,m(t)P'(t) = m'(t) - \frac{\alpha\beta}{\varrho C_p}\,m(t) \leq 0. \quad \square$$

Observación: La condición (**3**) sobre el crecimiento de k sólo supone una restricción en los intervalos de tiempo en que la función k crezca (de hecho, la verifican de forma automática las funciones constantes y las decrecientes). Además, puesto que

$$\lim_{t\to 0^+} \frac{1}{e^{\frac{\alpha\beta}{\varrho C_p}t}-1} = \infty,$$

dicha condición no supone ninguna restrición sobre k para tiempos cortos.

Las funciones elegidas en los ejemplos numéricos verifican esta acotación, la cual hay que interpretar como parte de la información a priori necesaria para poder identificar el coeficiente de conductividad. Esta información viene a decir que el coeficiente k no puede tener cambios bruscos, típicos de los procesos en que se produce cambio de fase, lo cual no ocurre en los casos que estamos analizando, como ya se comentó en la Introducción. □

En el supuesto de que existan funciones k_1 y k_2 distintas que proporcionen la misma medición $m(t)$ en el extremo derecho R y, además, una misma medición en algún otro punto $r_0 \in [0, R)$, los siguientes resultados prueban que ambas funciones deben coincidir. Trataremos en primer lugar el caso en que la conductividad térmica sea constante (en el que las hipótesis requeridas son más generales) y, posteriormente, se presentará el resultado para el caso general.

Teorema 3: *Sean T_1 y T_2 las respectivas soluciones de los problemas*

$$\begin{cases} \varrho C_p \dfrac{\partial T}{\partial t} - k_1 \Delta T = \alpha P'(t) T & \text{en } B_R \times (0, t_{\mathrm{f}}) \\ k_1 \dfrac{\partial T}{\partial \vec{n}} = h\left(T^e(t) - T\right) & \text{en } \partial B_R \times (0, t_{\mathrm{f}}) \\ T = T_0 & \text{en } B_R \times \{0\} \end{cases}$$

y

$$\begin{cases} \varrho C_p \dfrac{\partial T}{\partial t} - k_2 \Delta T = \alpha P'(t) T & \text{en } B_R \times (0, t_{\mathrm{f}}) \\ k_2 \dfrac{\partial T}{\partial \vec{n}} = h\left(T^e(t) - T\right) & \text{en } \partial B_R \times (0, t_{\mathrm{f}}) \\ T = T_0 & \text{en } B_R \times \{0\}, \end{cases}$$

siendo k_1 y k_2 dos constantes positivas. Supongamos **(H1)**, **(H2)** *y que para todo $t \in [0, t_{\mathrm{f}}]$ se verifica*

$$\overline{T}_1(R,t) = \overline{T}_2(R,t) \ \text{ y } \ \overline{T}_1(r_0,t) = \overline{T}_2(r_0,t) \ \text{ para algún } r_0 \in [0,R).$$

Entonces $k_1 = k_2$. □

Teorema 4: *Sean T_1 y T_2 las respectivas soluciones de los problemas*

$$\begin{cases} \varrho C_p \dfrac{\partial T}{\partial t} - k_1(t)\Delta T = \alpha P'(t)T & \text{en } B_R \times (0, t_{\rm f}) \\ k_1(t)\dfrac{\partial T}{\partial \vec{n}} = h\left(T^e(t) - T\right) & \text{en } \partial B_R \times (0, t_{\rm f}) \\ T = T_0 & \text{en } B_R \times \{0\} \end{cases}$$

y

$$\begin{cases} \varrho C_p \dfrac{\partial T}{\partial t} - k_2(t)\Delta T = \alpha P'(t)T & \text{en } B_R \times (0, t_{\rm f}) \\ k_2(t)\dfrac{\partial T}{\partial \vec{n}} = h\left(T^e(t) - T\right) & \text{en } \partial B_R \times (0, t_{\rm f}) \\ T = T_0 & \text{en } B_R \times \{0\}, \end{cases}$$

siendo $k_i \in \mathcal{C}^1([0, t_{\rm f}])$, con $k_i \geq k_0 > 0, i = 1, 2$. Supongamos **(H1)**, **(H2)** *y que para todo $t \in [0, t_{\rm f}]$ se verifica*

$$\overline{T}_1(R, t) = \overline{T}_2(R, t) \ \text{ y } \ \overline{T}_1(r_0, t) = \overline{T}_2(r_0, t) \ \text{ para algún } r_0 \in [0, R).$$

Si las funciones k_i son localmente analíticas por la derecha en $[0, t_{\rm f})$ y verifican

$$\int_0^{t_{\rm f}} k_i(s)ds \leq \frac{\varrho C_p (R - r_0)^2}{4}, \ i = 1, 2 \tag{5}$$

y

$$k_i'(t) \leq k_i(t)\, \frac{\alpha\beta}{\varrho C_p} \frac{1}{e^{\frac{\alpha\beta}{\varrho C_p} t} - 1}, \ t \in [0, t_{\rm f}], \ i = 1, 2,$$

entonces $k_1 = k_2$. □

Los resultados anteriores indican que si suponemos que las mediciones de que disponemos en r_0 y en R son las correspondientes a una temperatura que se modeliza mediante las ecuaciones (**1**), con k verificando las hipótesis requeridas, entonces la función k queda unívocamente determinada a partir de dichas mediciones.

Por otra parte, es de destacar que cuanto más separadas se hayan hecho las dos mediciones (i. e., cuanto más cercano a cero esté r_0), menos restrictiva será la condición (**5**) relativa a la información a priori sobre k y, por tanto, podremos garantizar la unicidad de solución del problema inverso para un conjunto más amplio de funciones.

3.4 Identificación numérica sin regularización. Ejemplos numéricos

Comenzamos esta sección describiendo la metodología utilizada para resolver el problema inverso, la cual se basa en un método de colocación con funciones continuas lineales a trozos en una partición temporal. Buscaremos una aproximación de este tipo para la función k (es decir, las incógnitas serán los valores de k en cada uno de los puntos de dicha partición).

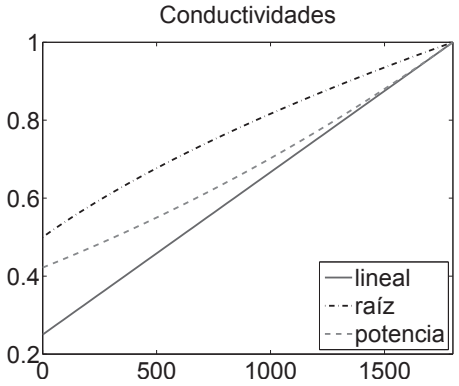

Figura 3.1: Conductividades consideradas en las pruebas numéricas realizadas.

Suponemos que las mediciones experimentales se han realizado en el centro y la frontera de la bola, en cada uno de los instantes $t_i, i = 1, 2, \ldots, n$, de dicha partición. A continuación, se escribe la igualdad del Teorema 2 con $r = 0$ y $t = t_i$, se sustituye el valor $\overline{T}(0, t_i)$ por la medición en el centro de la muestra en ese instante y se sustituye la función m por la interpolación lineal a trozos de las mediciones en el borde. La derivada de m se aproxima mediante la fórmula progresiva de primer orden. La aproximación de las integrales en $(0, t_i)$ correspondientes se realiza mediante la regla de los trapecios, tomando como valor de los integrandos en t_i cero (límite por la derecha del integrando). Estas aproximaciones dan lugar a un sistema de n ecuaciones no lineales en el que aparecen n incógnitas, los n valores de k en los instantes $t_i, i = 1, 2, \ldots, n$. Como ya se ha dicho, el valor de k en el instante inicial lo supondremos conocido, pues se corresponde con el valor a presión atmosférica.

Presentamos los resultados obtenidos en la identificación del coeficiente de conductividad para tres ejemplos de prueba en los que se ha utilizado el mismo incremento lineal de presión (véase (**6**)) y la misma perturbación de los datos (tal y como se explica más adelante). Se ha partido de tres tipos distintos de dependencias funcionales para la función k: lineal, en forma de radical y en forma de potencia. Los datos del dominio, así como para los parámetros del problema físico, se han tomado como en el caso del alimento de tipo sólido considerado en [6] (es decir, los correspondientes a la tilosa). Se supone un tratamiento como el P2 de dicho trabajo, con la salvedad de que el aumento de presión se hace de una forma mucho más lenta (en 1800 segundos en lugar de 183) para que sea más notorio el efecto de la conductividad.

En concreto, la presión se ha tomado como

$$P(t) = 0'2t + 0'1 \qquad \textbf{(6)}$$

y las funciones k objeto de identificación se han elegido (véase la Figura 3.1):

1) $k(t) = \dfrac{0'75t + 450}{1800}$

2) $k(t) = \sqrt{\dfrac{0'75t + 450}{1800}}$

3) $k(t) = \left(\dfrac{0'25t + 1350}{1800}\right)^3$.

Se han considerado funciones k crecientes, pues físicamente es razonable esperar que la conductividad crezca cuando se aumenta la presión (véase, por ejemplo, los datos para distintos materiales en [8]). El tamaño de estas funciones está elegido para que tengan el mismo orden que la conductividad media de la tilosa en [6], cuyo valor es $0'559$. Todas ellas satisfacen las desigualdades (**3**) y (**5**), relativas a la información a priori sobre k.

Los resultados obtenidos se muestran en la Figura 3.2. Como se observa, el error en la identificación de k es mayor al final del intervalo temporal. Esto lo explica el hecho de que los valores de k en los instantes finales intervienen tan sólo en las últimas ecuaciones y, al ser su valor menos determinante en el sistema no lineal, la solución numérica con el grado de precisión requerido admite mayores errores al final de dicho intervalo. Esto implica que sea más difícil que, en dichos instantes, los valores que se obtengan para k cambien respecto al valor inicial que se propone para la iteración. Los mayores errores en la temperatura calculada tras la identificación se obtienen, coherentemente, cuando se produce el mayor error en k, manteniéndose ligados ambos errores a lo largo del tiempo. En cualquier caso, los errores cometidos en la aproximación de T están por debajo del orden del error en las mediciones, i. e., del orden de la perturbación, lo cual hace que los resultados sean muy satisfactorios.

También se indica en la Figura 3.2 (identificado como "% máx. de error"), para cada uno de los tres experimentos numéricos, el mayor valor del error relativo porcentual, es decir, si denotamos por $\widetilde{T}$ la solución del problema correspondiente a la identificación aproximada de k, la cantidad

$$\max_{k=1,2,\ldots,n} \left(\frac{||\widetilde{T}(\cdot,t_k) - T(\cdot,t_k)||_{\mathcal{C}(B_R)}}{||T(\cdot,t_k)||_{\mathcal{C}(B_R)}} \times 100 \right).$$

Calculamos la que tomaremos como solución "exacta" del problema directo mediante la herramienta `pdetool` de Matlab, en una partición equiespaciada de 61 instantes del intervalo temporal $[0, 1800]$. Seguidamente, se extrae su valor en el centro de la bola y en un punto de la frontera, para dichos instantes. Las *mediciones con error* se generan perturbando los valores de la temperatura en estos dos puntos, de la misma forma en los tres casos, y mediante una perturbación de orden del 1% del rango de las temperaturas.

3.5 Algoritmo de regularización

En esta sección se presenta un algoritmo de regularización para el problema en estudio. Consideramos los espacios

$$\mathcal{Z} = L_2(0, t_\mathrm{f}) \text{ y } \mathcal{U} = L_2(0, t_\mathrm{f}) \times L_2(0, t_\mathrm{f})$$

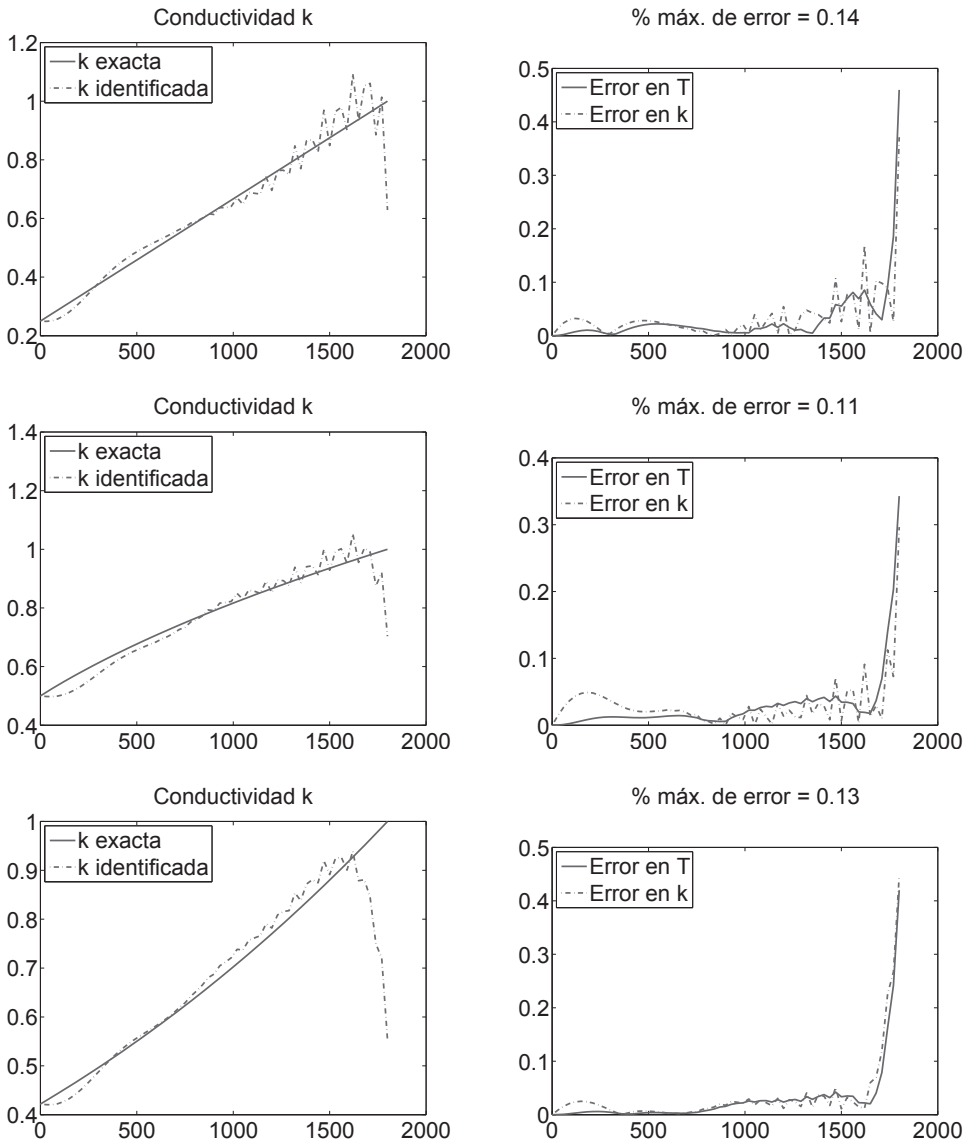

Figura 3.2: Conductividad térmica (**Izquierda**) y error en la temperatura (**Derecha**) correspondientes al caso en que k es una recta (**Arriba**), una raíz (**Centro**) y una potencia (**Abajo**).

y, para cada $\varepsilon > 0$, el subconjunto compacto y convexo $\mathcal{K}_\varepsilon \subset \mathcal{Z}$ definido por

$$\mathcal{K}_\varepsilon = \Bigg\{ k \in H^1(0,t_{\rm f}) : k(t) \geq k_0,\ t \in (0,t_{\rm f}),$$
$$k(t) \equiv k_0,\ t \in [0,\varepsilon],$$
$$\int_0^{t_{\rm f}} k(s)ds \leq \frac{\varrho C_p (R-r_0)^2}{4},$$
$$k'(t) \leq k(t)\,\frac{\alpha\beta}{\varrho C_p}\,\frac{1}{e^{\frac{\alpha\beta}{\varrho C_p}t}-1} \ \text{ c.d. en } (0,t_{\rm f}) \Bigg\}.$$

Sean $m(t) = \overline{T}(R,t), m^0(t) = \overline{T}(r_0,t)$, mediciones exactas de temperatura, las cuales corresponden a la solución del modelo para un único $k \in \mathcal{K}_\varepsilon$. Denotando por $m_\delta(t)$ y $m^0_\delta(t)$ las respectivas mediciones perturbadas con un error de orden δ, es decir,

$$\mathrm{d}\left(\begin{pmatrix} m_\delta \\ m^0_\delta \end{pmatrix}, \begin{pmatrix} m \\ m^0 \end{pmatrix}\right) = \sqrt{||m_\delta - m||^2 + ||m^0_\delta - m^0||^2} \leq \delta$$

Definiendo el operador

$$\begin{array}{rccc} A: & \mathcal{K}_\varepsilon & \longrightarrow & \mathcal{U} \\ & k & \mapsto & \left(m(t), m^0(t)\right), \end{array}$$

nos planteamos el problema de resolver, de forma estable, la ecuación

$$Ak = \left(m_\delta(t), m^0_\delta(t)\right),$$

a través de la solución del problema de optimización

$$\min_{k \in \mathcal{K}_\varepsilon} \mathrm{d}^2\left(Ak, \begin{pmatrix} m_\delta \\ m^0_\delta \end{pmatrix}\right).$$

La metodología que diseñamos para aproximar dicha solución se basa en la discretización del conjunto $\mathcal{K}_\varepsilon$ y el funcional A mediante funciones continuas y lineales a trozos en los intervalos $[t_{j-1}, t_j]$, donde t_j son los instantes en que se toman las n mediciones:

$$t_j = \frac{j t_\mathrm{f}}{n}.$$

El parámetro ε debe tomarse con la restricción de que sea menor que el intervalo entre dos instantes de medición, es decir,

$$0 < \varepsilon < \frac{t_\mathrm{f}}{n}.$$

Según muestra el método de soluciones aproximadas (véase [1]), el proceso iterativo correspondiente, debe detenerse en la primera iteración i en la que se verifique

$$\mathrm{d}^2\left(Ak_i, \begin{pmatrix} m_\delta \\ m^0_\delta \end{pmatrix}\right) \leq \delta^2,$$

en la cual se obtiene una solución regularizada.

La dificultad de este algoritmo radica en la necesidad de proyectar la dirección de descenso en cada paso iterativo del método de optimización utilizado, sobre el compacto $\mathcal{K}_\varepsilon$, cuyo interior es vacío en $\mathcal{Z}$ (véase [4])

3.6 Conclusiones

Planteado el problema de identificar el coeficiente de conductividad térmica cuando depende de la presión:

- Hemos encontrado una representación de la solución del problema directo a través de sus valores en la frontera.
- Esta representación de la solución nos ha permitido asegurar la unicidad del coeficiente de conductividad térmica cuando se conoce la temperatura en dos puntos (uno en el interior de la muestra y otro en el borde).
- Suponiendo conocidas mediciones experimentales sintéticas con error de la temperatura en esos dos puntos y en una serie de instantes de tiempo, hemos conseguido identificar, de forma aproximada, el coeficiente de conductividad, sin utilizar un método explícito de regularización y sin considerar las restricciones de unicidad.
- Hemos mostrado que el error cometido en las temperaturas que se obtienen usando la identificación calculada es del orden del error en las mediciones, al menos para los problemas sintéticos considerados.
- Hemos mostrado la posibilidad de identificar $k(t)$ en el marco del modelo, para datos sintéticos con error, considerando las restricciones de unicidad y un algoritmo explícito de regularización.
- Falta investigar la idoneidad del modelo cuando se trabaje con datos reales.

Bibliografía

[1] A. M. Denisov, *Elements of the Theory of Inverse Problems*. Inverse and Ill–Posed Problems Series. VSP. 1999.

[2] A. Fraguela, J. A. Infante, B. Ivorra, Á. M. Ramos, J. M. Rey y N. Smith. *Inverse problems in High Pressure Processes and Food Engineering*. Proceedings of First Symposium on Inverse Problems and Applications, Ixtapa, México.

[3] A. Fraguela, J. A. Infante, Á. M. Ramos y J. M. Rey. *Identification of a heat transfer coefficient when it is a function depending on temperature*. WSEAS Trans. Math. **7(4)** (2008) 160–172. ISSN: 1109-2769.

[4] S. F. Gilyazov, *Regularization of ill posed problems by Iteration Methods*, Mathematics and its Applications, Volume 499, Springer, 1999.

[5] J. A. Infante, *Análisis numérico de modelos matemáticos y problemas inversos en tecnología de alimentos*. Tesis doctoral. Universidad Complutense de Madrid. 2009.

[6] J. A. Infante, B. Ivorra, Á. M. Ramos and J. M. Rey, *On the Modelling and Simulation of High Pressure Processes and Inactivation of Enzymes in Food Engineering*. Mathematical Models and Methods in Applied Sciences (M3AS) **19 (12)** (2009) 2203–2229. ISSN: 0218-2025.

[7] L. Otero, Á.M. Ramos, C.de Elvira and P.D. Sanz, *A Model to Design High-Pressure Processes Towards an Uniform Temperature Distribution*. Journal of Food Engineering **78** (2007) 1463–1470. ISSN: 0260-8774.

[8] E. W. Lemmon, M. O. McLinden y D. G. Friend. *Thermophysical properties of fluid systems*. En NIST Chemistry Web Book. NIST Standard Reference Database **69**, P.J. Linstron y W.G. Mallard eds. National Institute of Standards and Technology. Recurso electrónico `http://webbook.nist.gov/chemistry/`.

Chapter 4

Inverse problems in High Pressure Processes and Food Engineering

A. Fraguela[1], J. A. Infante[2], B. Ivorra[2], A. M. Ramos[2],
J. M. Rey[2], N. Smith[2]

Abstract

Nowadays, in industrialized countries, food products that are frequently consumed are processed in order to prolong their shelf life, to avoid as much as possible their decomposition, and to maintain or even improve their natural qualities such as flavor and color. Decomposition of food is mainly due to microorganisms and enzymes, since they are involved in the physical and chemical processes of transformation of food substances. At present, consumers look for minimally processed, additive-free food products that maintain their organoleptic properties. This has promoted the development of new technologies for food processing. One of these new emerging technologies is high hydrostatic pressure, as it has turned out to be very effective in prolonging the shelf life of foods without losing its properties.

This work deals with the modeling and simulation of the effect of the combination of Thermal and High Pressure Processes, focusing on the inactivation that occurs during the process of certain enzymes and microorganisms that are harmful to food. We propose various mathematical models that study the behavior of these enzymes and microorganisms during and after the process, and study some related inverse problems.

[1]Benemérita Universidad Autónoma de Puebla. Puebla, México
[2]Universidad Complutense de Madrid. Madrid, España

Notation

$A(t;T,P)$	Enzymatic activity at time t, for a process at constant pressure P and temperature T	
A_0	Enzymatic activity at time 0	
D	Decimal reduction time	[min]
$D_{T_{\rm ref}},\ D_{P_{\rm ref}}$	Decimal reduction time at $T_{\rm ref}$/$P_{\rm ref}$	[min]
E_a	Activation energy	[kJ/mol]
k	Inactivation rate	[min^{-1}]
$k_{T_{\rm ref},P_{\rm ref}}$	Inactivation rate at $T_{\rm ref}$ and $P_{\rm ref}$	[min^{-1}]
$N(t;T,P)$	Microbial population at time t, for a process at constant pressure P and temperature T	[cfu/g]
N_0	Initial microbial population	[cfu/g]
$P,\ P_{\rm ref}$	Pressure / Reference pressure	[MPa]
t	Time	[min]
$T,\ T_{\rm ref}$	Temperature / Reference temperature	[K]
ΔV^*	Volume of activation	[cm^3/mol]
z_T	Temperature resistant coefficient	[K]
z_P	Pressure resistant coefficient	[MPa]
Ω^*	Whole domain of the device	
$\Omega^*_{\rm F}$	Sample food domain	
$\Omega^*_{\rm P}$	Pressurizing medium domain	
H	Heat transfer coefficient	[W m^{-2} K^1]
α	Thermal expansion coefficient	[K^1]
η	Dynamic viscosity	[Pa s]
ρ	Density	[Kg m^{-3}]
$\mathbf{g}$	Gravity vector	[m s^{-2}]
$t_{\rm f}$	Final time	[s]
S	Surface area of the heat being transferred	[m^2]
V	Heated volume	[m^3]
C_p	Heat capacity	[J Kg^{-1}K]
p	Mass transfer pressure	[Pa]

4.1 Introduction

Food Engineering has been studied in the past decades, specially from mid-twentieth century to now on. Obviously, humans have been interested in food conservation since ancient times, using traditional techniques such as desiccation, conservation in oil, salting, smoking, cooling, etc. Due to the massive movement of the population to the city, a great supply of food in adequate conditions was necessary. Therefore, the food industry was developed in order to guarantee large-scale food techniques, to prolong its shelf life, and to make logistic aspects such as transport, distribution and

storage, easier.

Classical industrial processes are based on thermal treatments. For example, pasteurization, sterilization and freezing. The disadvantages of freezing are non-homogeneous crystallization, that produces big crystals that may damage the food. For classical heat application processes, temperature is in a range of 60 to 120°C, and the processing time can vary from a few seconds to several minutes. The main aim of these processes is to inactivate microorganisms and enzymes that are harmful to food, in order to prolong its shelf life, to maintain or even to improve its natural qualities, and mainly to provide consumers with products in good conditions. The problem of processing food via thermal treatments is that it may loose a significant part of its nutritional and organoleptic properties. At present, consumers look for minimally processed, additive-free food products that maintain such properties. Therefore the development of new technologies with lower processing temperatures has increased notoriously in the past years.

One of the new emerging technologies in this field is the combination of thermal treatments (at moderate temperatures) with high hydrostatic pressure, thereby reducing the problems described above. Many companies are using this technology and it is being increasingly used in countries such as Japan, USA and UK. Recent studies [3, 13] have proven that high pressure causes inactivation of enzymes and microorganisms in food, while leaving small molecules (such as flavor and vitamins) intact, and therefore it does not modify significantly the organoleptic properties of the food. High pressure can also be used for freezing, resulting in uniform nucleation and crystallization. Our aim is to model mathematically these high pressure processes, in order to simulate and optimize them.

Two principles underlie the effect of high pressure. Firstly, Le Chatelier Principle, according to which any phenomenon (phase transition, chemical reaction, chemical reactivity, change in molecular configuration) accompanied by a decrease in volume will be enhanced by pressure. Secondly, pressure is instantaneously and uniformly transmitted independently of the size and the geometry of the food (isostatic pressure).

4.2 Mathematical modelling of microbial and enzymatic inactivation

Kinetic parameters and models are used for the development of food preservation processes to ensure safety. They also provide tools to compare the impact of different process technologies on the reduction of microbial populations or enzymatic activity. In this section we present mathematical models and the parameters that describe Microbial and Enzymatical Inactivation[3] due to the combination of thermal and high pressure treatments.

In order to describe changes in microbial populations as a function of time, when the food sample is processed at temperature T and pressure P we can use the first-order kinetic model[4]:

[3] **Inactivation** may be defined as the reduction of undesired biological activity, such as enzymatic catalysis and microbial contamination.

[4] Higher-order models that describe changes in microbial populations as a function of time can also be found in the literature [15].

$$\begin{cases} \dfrac{\mathrm{d}N(t;T,P)}{\mathrm{d}t} = -k(T,P)N(t;T,P), & t \geq 0, \\ N(0;T,P) = N_0, \\ [\textbf{Solution} : N(t;T,P) = N_0 \ \exp(-k(T,P)\ t), & \text{for isobaric/isothermal processes} \\ N(t;T,P) = N_0 \ \exp(-\displaystyle\int_0^t k(T(s),P(s))\ \mathrm{d}s, & \text{for dynamic processes }] \end{cases} \tag{1}$$

where $N(t;T,P)$ is the microbial population at time t, when the food sample is processed at temperature T and pressure P, N_0 is the initial microbial population and $k(T,P)$ is the *inactivation rate constant* [min^{-1}], also called *death velocity constant* in the case of microorganisms. Therefore, we have encountered the first inverse problem: To identify $k(T,P)$ for adequate ranges of temperature and pressure. The same model can be used to estimate the changes in the enzymatic activity as a function of time by changing $N(t;T,P)$ for $A(t;T,P)$, and N_0 for A_0.

Another equation used very often (e.g. [14]) to calculate changes of microbial population as a function of time is the following:

$$\begin{cases} \log\left(\dfrac{N(t;T,P)}{N_0}\right) = \dfrac{-t}{D(T,P)}, \\ N(0;T,P) = N_0, \\ \textbf{Solution} : N(t;T,P) = N_0\ 10^{-\frac{t}{D(T,P)}}, & \text{for isobaric/isothermal processes} \\ N(t;T,P) = N_0\ 10^{-\int_0^t \frac{1}{D(T(s),P(s))}\ \mathrm{d}s}, & \text{for dynamic processes} \end{cases} \tag{2}$$

where $D(T,P)$ is the *decimal reduction time* [min], or time required for a 1-log-cycle[5] reduction in the microbial population. We have encountered another inverse problem: to identify $D(T,P)$ for adequate ranges of temperature and pressure.

4.2.1 Identification of kinetic parameters

For isostatic processes, $k(T)$ can be given by Arrhenius' equation:

$$k(T) = k_{T_{\mathrm{ref}}} \exp\left(\left(\frac{-E_a}{R}\right)\left(\frac{1}{T} - \frac{1}{T_{\mathrm{ref}}}\right)\right), \tag{3}$$

where $k(T)$ [min^{-1}] is the inactivation rate for an arbitrary temperature T [K], T_{ref} [K] is a reference temperature, $k_{T_{\mathrm{ref}}}$ [min^{-1}] is the inactivation rate at reference temperature, E_a [J/mol] is the activation energy[6] and $R = 8314$ [J/(mol K)] is the universal gas constant. And for isothermal

[5] A 1-log-cycle reduction is equivalent to reducing the population dividing it by ten. In the same way, a n log-cycle is equivalent to reducing the population dividing it by 10^n.

[6] Activation energy (chemistry): the minimum amount of energy that is required to activate atoms or molecules to a condition in which they can undergo chemical transformation or physical transport.

processes, $k(P)$ is given by the following equation (based on Eyring's equation):

$$k(P) = k_{P_{\text{ref}}} \exp\left(\frac{-\Delta V^*(P - P_{\text{ref}})}{RT}\right), \tag{4}$$

where $k(P)$ [min^{-1}] is the inactivation rate for an arbitrary pressure P [MPa], P_{ref} [MPa] is a reference pressure, $k_{P_{\text{ref}}}$ [min^{-1}] is the inactivation rate at reference pressure and ΔV^* [cm^3/mol] is the volume of activation[7].

For general temperature and pressure dependent processes, $k(T, P)$ may be calculated by a combination of Arrehnius' and Eyring's equations (other possibilities may be found in the literature):

$$k(T, P) = k_{T_{\text{ref}}, P_{\text{ref}}} \exp\left(-B\left(\frac{1}{T} - \frac{1}{T_{\text{ref}}}\right)\right) \exp(-C(P - P_{\text{ref}})), \tag{5}$$

or by a more complex choice given by:

$$k(P,T) = \quad k_{\text{ref}} \exp\left(\tfrac{-\Delta V_{\text{ref}}}{RT}(P - P_{\text{ref}})\right) \exp\left(\tfrac{\Delta S_{\text{ref}}}{RT}(T - T_{\text{ref}})\right) \tag{6}$$

$$\exp\left(\tfrac{-\Delta\kappa}{2RT}(P - P_{\text{ref}})^2\right) \exp\left(\tfrac{-2\Delta\zeta}{RT}(P - P_{\text{ref}})(T - T_{\text{ref}})\right) \tag{7}$$

$$\exp\left(\tfrac{\Delta C_p}{RT}\left(T(\ln\tfrac{T}{T_{\text{ref}}} - 1) + T_{\text{ref}}\right)\right) + \text{high order terms}, \tag{8}$$

where $k(T, P)$ [min^{-1}] is the inactivation rate for temperature T [K] and pressure P [MPa], and $k_{T_{\text{ref}}, P_{\text{ref}}} = k(T_{\text{ref}}, P_{\text{ref}})$[min^{-1}], B [K] and C [MPa] are kinetic constants that express the dependence of $k(T, P)$ on temperature and pressure.

By construction $k(T, P)$ and $D(T, P)$ are related by $k = \frac{\ln(10)}{D}$, thereby it is possible to move from one model to the other. However, we may also calculate $D(T, P)$ directly by using suitable equations. For $D(T)$ and $D(P)$, we have, resp. [14]:

$$\log\left(\frac{D(T)}{D_{T_{\text{ref}}}}\right) = -\frac{T - T_{\text{ref}}}{z_T} \tag{9}$$

$$\log\left(\frac{D(P)}{D_{P_{\text{ref}}}}\right) = -\frac{P - P_{\text{ref}}}{z_P} \tag{10}$$

where z_T [K] (resp., z_P [MPa]) is the thermal (resp., pressure) resistance constant that can be defined as the temperature (resp., pressure) increase needed to accomplish a 1-log-cycle reduction in the decimal reduction time value D [min]; $D_{T_{\text{ref}}}$ (resp., $D_{P_{\text{ref}}}$) [min] is the reference decimal reduction time at reference temperature T_{ref} [K] (resp., reference pressure P_{ref} [MPa]) within the range of temperatures (resp., pressures) used to generate experimental data.

Therefore, the inverse problems consisting of identifying functions $k(T, P)$ and/or $D(T, P)$ are converted into parameter estimation problems (we have to identify E_a, ΔV^*, z_T, z_P, etc...).

This parameter identification may be done using linear regression. For example, if we have experimental data of the concentration of a certain microorganism in food after being processed for

[7]The volume of activation is interpreted, according to transition state theory, as the difference between the partial molar volumes of the transition state (V) and the sums of the partial volumes of the reactants at the same temperature and pressure.

different times and at different pressures and constant temperatures, we could proceed as follows: Firstly we consider the measurements done at the same pressure, therefore we would follow model (**1**) and model (**2**). Using linear regression we identify the kinetic parameters k and D. Secondly, as we have data measured at different pressure values, we follow equations (**4**) and (**10**) in order to find a formula to express the pressure dependence of $k(P)$ and $D(P)$. The parameters we identify are ΔV^*, $P_{\rm ref}$ and $k_{P_{\rm ref}}$ for $k(P)$; z_P, $P_{\rm ref}$ and $D_{P_{\rm ref}}$ for $D(P)$. We do this again using linear regression. For general processes we could use, for instance, equation (**5**) with multiple linear regression or equation (**6**) with non–linear regression techniques.

4.3 Modelling the temperature profiles

As we can see in Section 4.2, kinetic equation (**1**) describing the enzymatic activity evolution, in function of time t, needs to know the time evolution of the pressure $P(t)$ and temperature $T(t)$.

In practice, the pressure evolution, $P(t)$, is known as it is imposed by the user and the limits of the equipment. In the case of the temperature evolution $T(t)$, it is necessary to consider the adiabatic heating effects due to the work of compression/expansion in the considered High Pressure device. The temperature of the processed food may change with time and with space, therefore we need a heat transfer model capable of predicting the temperature for the processed food.

In order to determine $T(t)$, we may consider various kinds of models based on ODEs, for the simplest ones, or PDEs, for the more complex ones.

4.3.1 ODEs based model

A first model, for studying the temperature evolution, can be obtained by combining Newton's law of cooling

$$\rho V C_p \frac{\mathrm{d}T}{\mathrm{d}t} = HS(T^{\rm e} - T) \tag{11}$$

with the adiabatic heating effect due to the change of pressure

$$\frac{\Delta T}{\Delta P} = \frac{\alpha T}{\rho C_p}, \tag{12}$$

where α is the thermal expansion coefficient [K^{-1}], $\rho = \rho(T,P)$ the density [$\mathrm{Kg\,m}^{-3}$], S the surface area where the heat is being transferred [m^2], V the heated volume [m^3], H the heat transfer coefficient [W m^{-2} K^1] and $C_p = C_p(T,P)$ the heat capacity [J Kg^{-1}K].

In this case, $T(t)$ is governed by

$$\frac{\mathrm{d}T}{\mathrm{d}t} = H(T^{\rm e} - T) + \alpha T \frac{\mathrm{d}P}{\mathrm{d}t}. \tag{13}$$

In equations (**11**)–(**13**), the coefficients can be estimated by solving suitable inverse problems. For example, in the next Section we will study an inverse problem to obtain an estimation of H.

However, the model described by (**13**) does not take into account the spatial distribution of the temperature. This is important, as in practice, this distribution, and thus the enzymatic activity one, is not homogeneous in space. To bypass this problem, we can consider models based on PDEs.

4.3.2 PDEs based model

Here, we first present a heat transfer model taking into account only conduction effects and then a second model also including the convection effect. As these models are time and spatial dependent, we also introduce a brief description of the domain describing the High Pressure device considered in our simulations.

Spatial domain description

High Pressure experiments are often carried out in a cylindrical pressure vessel (typically a hollow steel cylinder, as the one presented in Figure 4.1) previously filled with the food and the pressurizing medium [9,12]. The sample is located in the inner chamber at a temperature that can be the same or different to the one in the pressurizing medium and/or the solid domain surrounding it, which may cool or warm the food following user criteria.

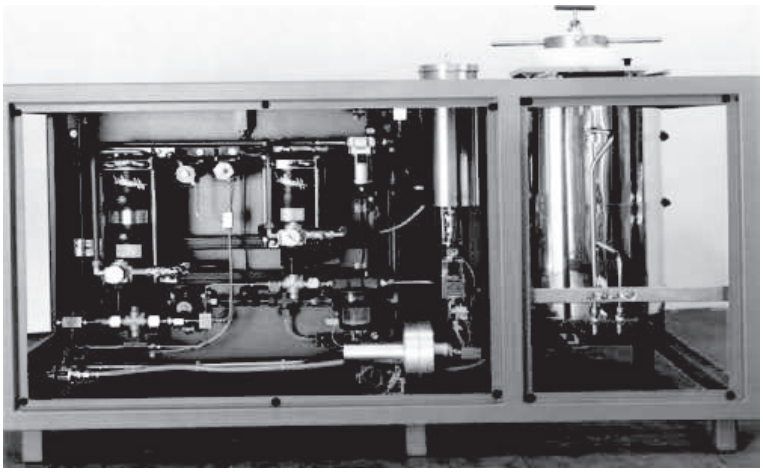

Figure 4.1: Example of a High Pressure device (ACB GEC Alsthom, Nantes, France). More details can be found in [12].

Let us consider three domains: the whole domain Ω^* of the High Pressure device; the domain Ω_{F}^* where the sample of food is located; and the domain Ω_{P}^* occupied by the pressurizing medium. Due to the characteristics of this kind of processes, we assume that thermally induced flow instabilities are negligible. Therefore, axial symmetry allows us to use cylindrical coordinates and the corresponding domain is given by half of a cross section (intersection of the cylinder with a plane containing the axis) and are denoted by Ω, Ω_{F} and Ω_{P}, respectively (see Figure 4.2).

Heat transfer by conduction

We consider the heat conduction equation

$$\rho C_p \frac{\partial T}{\partial t} - \nabla \cdot (k \nabla T) = \alpha \frac{dP}{dt} T \text{ in } \Omega^* \times (0, t_{\mathrm{f}}), \tag{14}$$

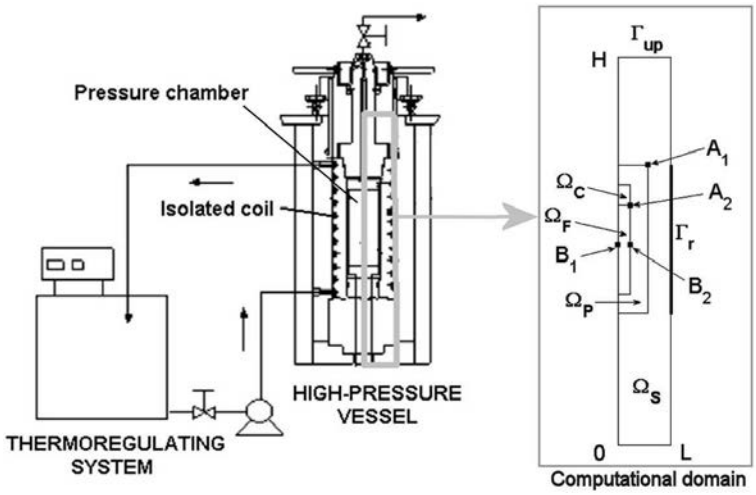

Figure 4.2: Scheme of the High Pressure device presented in Figure 4.1 (**Left**) and its corresponding 2D computational domain (**Right**). More details about the notations can be found in [9].

where T [K] is the temperature, $k = k(T, P)$ [$\mathrm{W\,m^{-1}K^{-1}}$] is the thermal conductivity, t_{f} [s] is the final time and $\alpha = \alpha(T, P)$ is given by

$$\alpha = \begin{cases} \text{thermal expansion coefficient } [\mathrm{K^{-1}}] \text{ of the food in } \Omega_{\mathrm{F}}^*, \\ \text{thermal expansion coefficient } [\mathrm{K^{-1}}] \text{ of the pressurizing fluid in } \Omega_{\mathrm{P}}^*, \\ 0, \text{ elsewhere.} \end{cases}$$

The conductive heat transfer equation (**14**) is completed with appropriate initial and boundary conditions depending on the +High Pressure machine, in order to determine the solution that we are looking for (see [9]).

As previously, the coefficients in (**14**) can be evaluated considering inverse problems. For instance, the thermal conductivity k can be estimated in function of the pressure P [5, 6, 8].

This model is suitable when the filling ratio of the food sample inside the vessel is much higher than the one of the pressurizing medium. When the filling ratio of the food inside the vessel is not much higher than the one of the pressurizing medium, the solution of this model is very different from the experimental measurements [12]. One way to solve this problem is to include the convection phenomenon that takes place in the pressurizing medium. The resulting model is more expensive from a computational point of view but the results are more accurate.

Heat transfer by conduction and convection

The non–homogeneous temperature distribution induces a non–homogeneous density distribution in the pressurizing medium and consequently a buoyancy fluid motion (i.e., free convection).

This fluid motion may strongly influence the temperature distribution. In order to take into account this fact, a non–isothermal flow model is considered. Therefore, we suppose that the fluid

velocity field, $\mathbf{u}$ [$\mathrm{m\,s^{-1}}$], satisfies Navier–Stokes' equations for compressible Newtonian fluid under Stoke's assumption (see, for instance, [1]). The resulting system of equations is:

$$\begin{cases} \rho C_p \dfrac{\partial T}{\partial t} - \nabla \cdot (k \nabla T) + \rho C_p \mathbf{u} \cdot \nabla T = \alpha \dfrac{dP}{dt} T & \text{in} \quad \Omega^* \times (0, t_{\mathrm{f}}), \\ \rho \dfrac{\partial \mathbf{u}}{\partial t} - \nabla \cdot \eta (\nabla \mathbf{u} + \nabla \mathbf{u}^{\mathbf{t}}) + \rho (\mathbf{u} \cdot \nabla) \mathbf{u} & \\ \qquad = -\nabla p - \dfrac{2}{3} \nabla (\eta \nabla \cdot \mathbf{u}) + \rho \mathbf{g} & \text{in} \quad \Omega_{\mathrm{P}}^* \times (0, t_{\mathrm{f}}), \\ \dfrac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 & \text{in} \quad \Omega_{\mathrm{P}}^* \times (0, t_{\mathrm{f}}), \end{cases} \tag{15}$$

where $\mathbf{g}$ is the gravity vector [$\mathrm{m\,s^{-2}}$], $\eta = \eta(T, P)$ is the dynamic viscosity [$\mathrm{Pa\,s}$], $p = p(x, t)$ is the pressure generated by the mass transfer inside the fluid [Pa], and $P + p$ is the total pressure [Pa] in the pressurizing medium Ω_{P}^*. System (**15**) is completed with appropriate point, boundary and initial conditions. If the food sample is liquid two more equations for its velocity and density should be added (see [8, 9]).

The coefficients in (**15**) can be determined by considering various inverse problems (see [5, 8]).

4.3.3 Coupling of Inactivation and Heat–Mass Transfer Models

An example of final temperature and food sample enzymatic activity distributions obtained using a numerical version of models (**1**) and (**15**) (see [9] for more details) is given by Figure 4.3. We have considered the following treatment:

The initial temperature is

$$T_0 = \begin{cases} 40\,^{\mathrm{o}}\mathrm{C} \text{ in } \Omega_{\mathrm{S}}, \\ 22\,^{\mathrm{o}}\mathrm{C} \text{ in } \Omega \backslash \Omega_{\mathrm{S}} \end{cases}$$

and the pressure is linearly increased during the first 305 seconds until it reaches 600 MPa. Thus, the pressure generated by the equipment satisfies $P(0) = 0$ and

$$\frac{dP}{dt} = \begin{cases} \dfrac{120}{61}\, 10^6 \ \ \mathrm{Pa\,s^{-1}}, & 0 < t \leq 305, \\ 0 \ \ \mathrm{Pa\,s^{-1}}, & t > 305. \end{cases}$$

The considered enzyme is Lipoxygenase (LOX): This enzyme is present in various plants and vegetables such as green beans and green peas. It is responsible for the appearance of undesirable aromas in those products.

Equation (**6**) is used to describe κ with $P_{\mathrm{ref}} = 500$ MPa, $T_{\mathrm{ref}} = 298$ K, $\kappa_{\mathrm{ref}} = 1.34 \times 10^{-2}$ $\mathrm{min^{-1}}$, $\Delta V_{\mathrm{ref}} = -308.14\,\mathrm{cm^3 mol^{-1}}$, $\Delta S_{\mathrm{ref}} = 90.63\,\mathrm{J\,mol^{-1} K^{-1}}$, $\Delta C_p = 2466.71\,\mathrm{J\,mol^{-1} K^{-1}}$, $\Delta\zeta = 2.22\,\mathrm{cm^3 mol^{-1} K^{-1}}$, $\Delta\nu = -0.54\,\mathrm{cm^6 J^{-1} mol^{-1}}$ (see Ref. [10] for more details).

4.4 Identification of a heat transfer coefficient

In this section, we focus our attention on an inverse problem concerning the identification of the heat exchange coefficient H (assuming it depends on pressure or/and temperature) between a certain

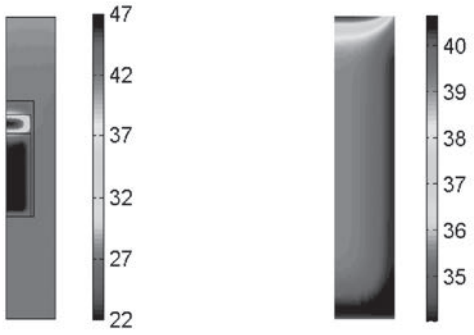

Figure 4.3: Temperature distributions in Ω (**Left**) and LOX enzymatic distributions in Ω_{S} (**Right**) at time $t = 15$ min after the considered process.

material with the external environment. The goal is to identify H in order to get a solution for the corresponding model, approximating some given temperature measurements. For simplicity, let us suppose that the sample is small enough to be able to assume that the temperature gradient inside it is negligible. Equation (**13**) provides a simple mathematical model describing this phenomenon through the following initial value problem (*direct problem*):

$$\begin{cases} \dfrac{\mathrm{d}T}{\mathrm{d}t}(t) = H(T(t), P(t))(T^e - T(t)) + \alpha \dfrac{\mathrm{d}P}{\mathrm{d}t}(t)T(t),\ t \geq t_0 \\ T(t_0) = T_0. \end{cases} \tag{16}$$

where H is the pressure/temperature dependent heat exchange coefficient. In order to solve problem (**16**), constants $T_0,\ T^e \in \mathbb{R}$, pressure curve P and function
$H : [T_a, T_b] \times [P_{\min}, P_{\max}] \to \mathbb{R}$ are needed ($[T_a, T_b]$ and $[P_{\min}, P_{\max}]$ are suitable ranges of temperature and pressure, respectively).

The values of T_0 and T^e can be obtained by measuring devices (*thermocouples*), the coefficient α is assumed to be known and the pressure is provided by the equipment. However, function H cannot be obtained easily. Our goal is to identify function H (*inverse problem*) from experimental measurements; by doing so, we are able to approximate the solution of model (**16**) for other data T_0, T^e and P (provided they are kept in the initial ranges of temperature and pressure $[T_a, T_b]$ and $[P_{\min}, P_{\max}]$, respectively) without requiring new measurements. The main difficulties are:

- Function H may depend on the solution of the state equation T.
- Temperature and pressure measurements can be given with errors.

In some contexts, one can assume that H belongs to a particular type of functions (for example, H is constant or a polynomial) which only needs to identify one or more real parameters. However, in this work we try to identify function H only knowing that it is a continuous and positive function.

We note that the value of H is not relevant when T is close to T^e. So, we set a *threshold* μ to separate it from T^{e} ($H(T, P)$ is not identified for values of T too close to T^{e}).

We consider two cases: In the first and simplest one, coefficient H depends only on the (known) pressure. We design an *ad hoc* experiment for the determination of this coefficient. If this methodology cannot be performed, we propose a numerical algorithm that allows to approximate function H (sometimes with a greater accuracy than the previous method).

In the second case, H only depends on the temperature. Then, we can assume that the pressure remains constant, and so the second term on the right hand side of the equation of problem (**16**) vanishes.

We denote by $T_k = T(t_k)$ the temperature values at measure instants t_k and by $\widetilde{T}_k$ its approximation with un error less or equal to $\delta > 0$.

4.4.1 First case: $H = H(P)$

Problem (**16**) becomes

$$\begin{cases} \dfrac{\mathrm{d}T}{\mathrm{d}t}(t) = H(P(t))(T^{\mathrm{e}} - T(t)) + \alpha \dfrac{\mathrm{d}P}{\mathrm{d}t}(t)T(t),\ t \in [t_0, t_{\mathrm{f}}] \\ T(t_0) = T_0. \end{cases} \tag{17}$$

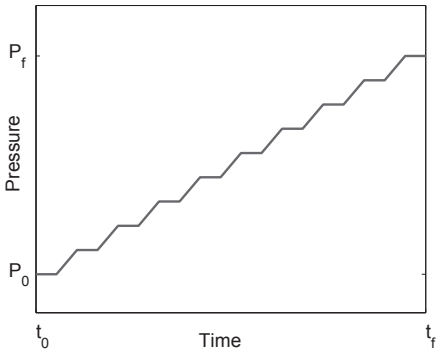

Figure 4.4: *Ad hoc* experiment.

Ad hoc experiment

We consider a pressure profile as in Figure 4.4.

When the pressure is constant, explicit solution of system (**17**) can be obtained. This leads to the approximations:

$$\widetilde{H}_k = \frac{1}{h} \ln \left(\frac{\widetilde{T}_{2k} - T^e}{\widetilde{T}_{2k+1} - T^e} \right) \quad (\approx H(P(t_{2k}))) \,,$$

where $h = t_{k+1} - t_k$. Denoting

$$\sigma_k = \frac{\widetilde{T}_k - T_k}{T_k - T^e},$$

the error in this approximation can be expressed as

$$\widetilde{H}_k - H(P(t_{2k})) = \frac{1}{h} \ln\left(\frac{1+\sigma_{2k}}{1+\sigma_{2k+1}}\right).$$

Iterative algorithm

Since

$$H(P(t)) = \frac{T'(t) - \alpha P'(t) T(t)}{T^e - T(t)}$$

our goal is to approximate

$$\frac{T'(t_k) - \alpha P'(t_k) T_k}{T^e - T_k}.$$

With this aim, we define the discrete derivative operator $R_h : \mathcal{C}([t_0, t_{\rm f}]) \to \mathcal{C}([t_0, t_{\rm f}])$ given by

$$R_h(v)(t) = \begin{cases} \dfrac{-3v(t) + 4v(t+h) - v(t+2h)}{2h} + \Psi_h(t_0),\ t \in [t_0, t_0 + h] \\ \dfrac{v(t+h) - v(t-h)}{2h},\ t \in [t_0 + h, t_{\rm f} - h] \\ \dfrac{3v(t) - 4v(t-h) + v(t-2h)}{2h} + \Psi_h(t_f - 3h),\ t \in [t_{\rm f} - h, t_{\rm f}] \end{cases}$$

where

$$\Psi_h(v)(t) = \frac{v(t+3h) - 3v(t+2h) + 3v(t+h) - v(t)}{2h}.$$

Thus, by taking

$$\widetilde{H}_k = \frac{R_h(\widetilde{T})(t_k) - \alpha P'(t_k) \widetilde{T}_k}{T^e - \widetilde{T}_k} \quad (\approx H(P(t_k))$$

we are able to prove (see [8]) the following error estimate

$$\left| H(P(t_k)) - \widetilde{H}_k \right| \leq \frac{1}{\mu - \delta} \left(\frac{29 M_3}{6} h^2 + \frac{4\delta}{\mu h} \left(\widetilde{M} - \widetilde{m} + 2\mu \right) \right) + \frac{\alpha P'_M T^e \delta}{\mu(\mu - \delta)}, \qquad \textbf{(18)}$$

where

$$\widetilde{m} = \min_{t \in [t_0, t_{\rm f}]} \widetilde{T}(t),\ \widetilde{M} = \max_{t \in [t_0, t_{\rm f}]} \widetilde{T}(t),\ M_3 = \max_{s \in [t_0, t_{\rm f}]} |T'''(s)| \text{ y } P'_M = \max_{s \in [t_0, t_{\rm f}]} P'(s).$$

In order to minimize the bound in (**18**), we choose

$$h^* = \left(\frac{12(\widetilde{M} - \widetilde{m} + 2\mu)}{29 \mu M_3} \delta \right)^{\frac{1}{3}}$$

which provides the estimate

$$\left| H(P(t_k) - \widetilde{H}_k \right| \leq \frac{1}{\mu - \delta} \left(522 M_3 \frac{(\widetilde{M} - \widetilde{m} + 2\mu)^2}{\mu^2} \delta^2 \right)^{\frac{1}{3}} + \frac{\alpha P'_M T^e}{\mu(\mu - \delta)} \delta.$$

The iterative algorithm that we have developed can be described as:

DATA:
- $\{\widehat{T}_k\}_{k=0}^p$: Temperature measurements at times $\{\tau_k\}_{k=0}^p$.
- $\widehat{\delta} > 0$: bound on the error in the measurements.
- $\mu > 0$: threshold.
- $\varepsilon > 0$: stopping test parameter.
- $h > 0$: initial (tentative) value for the time step.

Step 1: Determine $\widetilde{T}$ (interpolation function of the values $\{\widehat{T}_k\}_{k=0}^p$) and δ according to $\widehat{\delta}$.

Step 2: While relative error of h is greater than ε:

- Determine the new discrete times $\{t_k\}$ and compute $\{\widetilde{T}_k\}$.
- Compute a value Λ_3 approximating M_3.
- Consider $h = \left(\dfrac{12(\widetilde{M} - \widetilde{m} + 2\mu)}{29\mu\Lambda_3}\, \delta \right)^{\frac{1}{3}}$.

Step 3: Obtain the final discrete times $\{t_k\}$ and the values $\{\widetilde{T}_k\}$.

Step 4: Compute the approximations $\widetilde{H}_k = \dfrac{R_h(\widetilde{T})(t_k) - \alpha P'(t_k)\widetilde{T}_k}{T^e - \widetilde{T}_k}$.

Figures 4.5 and 4.6 show the results for a numerical test implemented with this algorithm.

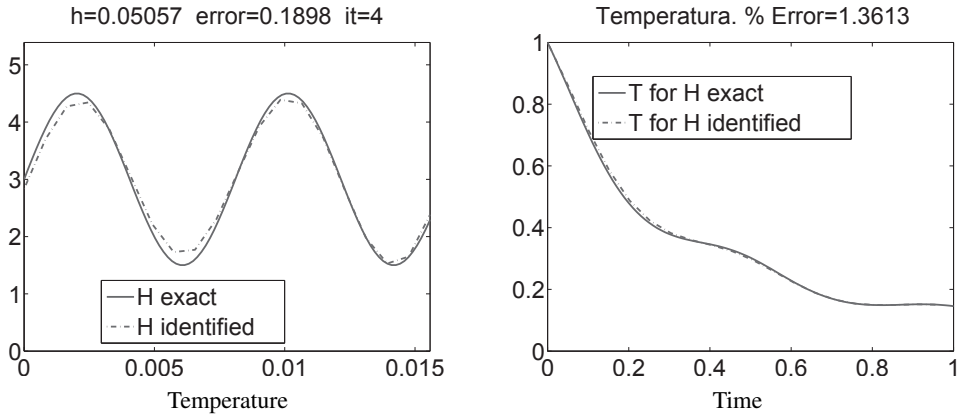

Figure 4.5: Iterative algorithm (Left: coefficient H. Right: temperature T)

Nondimensionalization of the problem

By considering the new variables

$$t^* = \frac{t - t_0}{t_{\mathrm{f}} - t_0}, \quad T^*(t^*) = \frac{T(t) - T^e}{T_0 - T^e} \text{ and } P^*(t^*) = \big(P(t) - P_0\big)\alpha,$$

problem (**17**) can be rewritten as

$$\begin{cases} \dfrac{dT^*}{dt^*}(t^*) = -H^*(P^*(t^*))T^*(t^*) + \dfrac{dP^*}{dt^*}(t^*)\left(T^*(t^*) + T^{ea}\right),\ t^* \in (0,1) \\ T^*(0) = 1, \end{cases}$$

where

$$\begin{cases} H^*(s) = (t_{\mathrm{f}} - t_0)H\left(\dfrac{s}{\alpha} + P_0\right) \quad \left(\Rightarrow\ H^*(P^*(t^*)) = (t_{\mathrm{f}} - t_0)H(P(t))\right) \\ T^{ea} = \dfrac{T^e}{T_0 - T^e}. \end{cases}$$

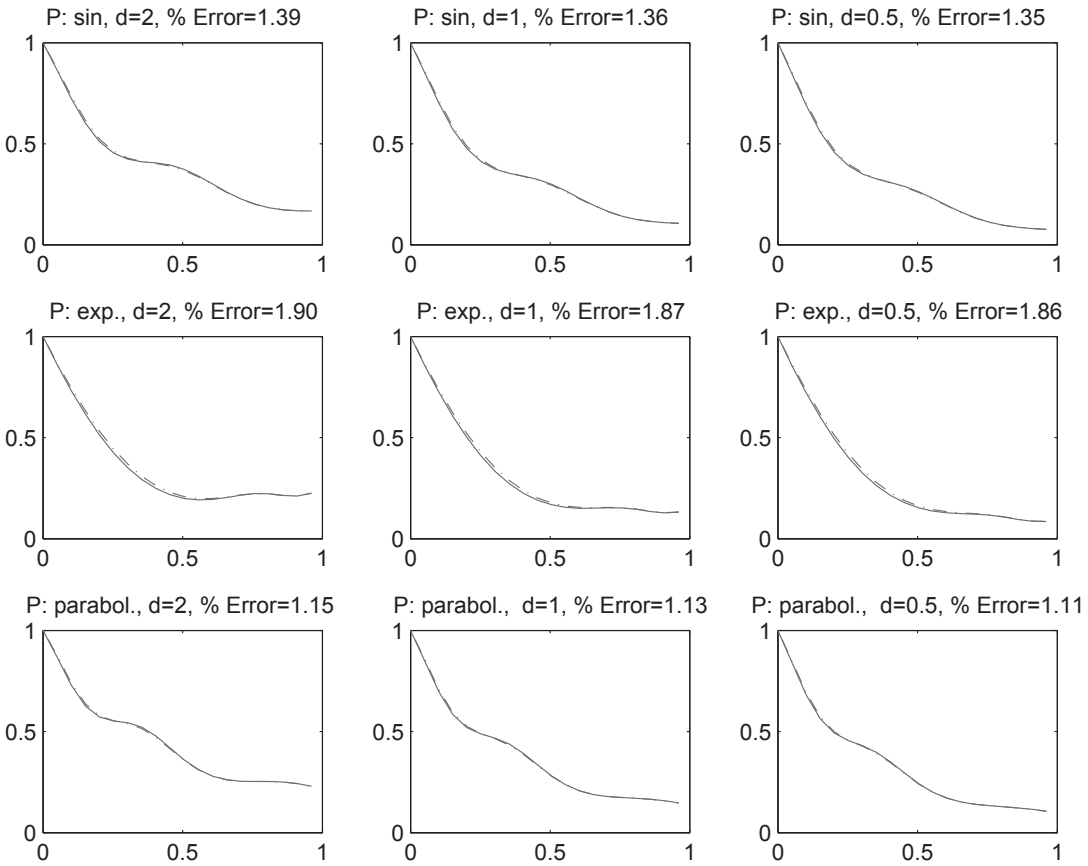

Figure 4.6: Iterative algorithm. Temperature for several T^{ea} and P.

4.4.2 Second case: $H = H(T)$

Now, problem (**16**) becomes

$$\begin{cases} \dfrac{\mathrm{d}T}{\mathrm{d}t}(t) = H(T(t))(T^{\mathrm{e}} - T(t)),\ t \in [t_0, t_{\mathrm{f}}] \\ T(t_0) = T_0. \end{cases} \tag{19}$$

Classical regularization methods

From (**19**) we have

$$\int_{t_0}^{t} u(s)\mathrm{d}s = \int_{t_0}^{t} H(T(s))\mathrm{d}s = \int_{t_0}^{t} \frac{T'(s)}{T^{\mathrm{e}} - T(s)}\mathrm{d}s = -\ln\left(\frac{T^{\mathrm{e}} - T(t)}{T^{\mathrm{e}} - T_0}\right),$$

where

$$u(t) = H(T(t)).$$

Thus, by defining $K : L^2(t_0, t_{\mathrm{f}}) \to L^2(t_0, t_{\mathrm{f}})$ as

$$Ku(t) = \int_{t_0}^{t_{\mathrm{f}}} u(s)\mathrm{d}s,$$

our problem is equivalent to solving

$$Ku = y,$$

where

$$y(t) = -\ln\left(\frac{T^e - T(t)}{T^e - T_0}\right).$$

Due to measurement errors, available data are

$$y_\delta(t) = -\ln\left(\frac{T^e - \widetilde{T}_\delta(t)}{T^e - T_0}\right),$$

which leads us to solve the approximate problem

$$Ku_\delta = y_\delta.$$

We have considered the following classical methods, based on regularization strategies:

- **Tikhonov's regularization**: We minimize the functional

  $$J_\alpha(x) = \|Kx - y_\delta\|^2_{L^2(0,t_{\mathrm{f}})} + \alpha\|x\|^2_{L^2(0,t_{\mathrm{f}})}$$

  with a suitable choice of α based on the Morozov's discrepancy principle.

- **Landweber iterative method**: This method is defined by the sequence

  $$\begin{cases} x_0 = 0 \\ x_m = (I - aK^*K)x_{m-1} + aK^*y,\ m = 1, 2, \ldots \end{cases}$$

  with $0 < a < \dfrac{1}{||K||}$.

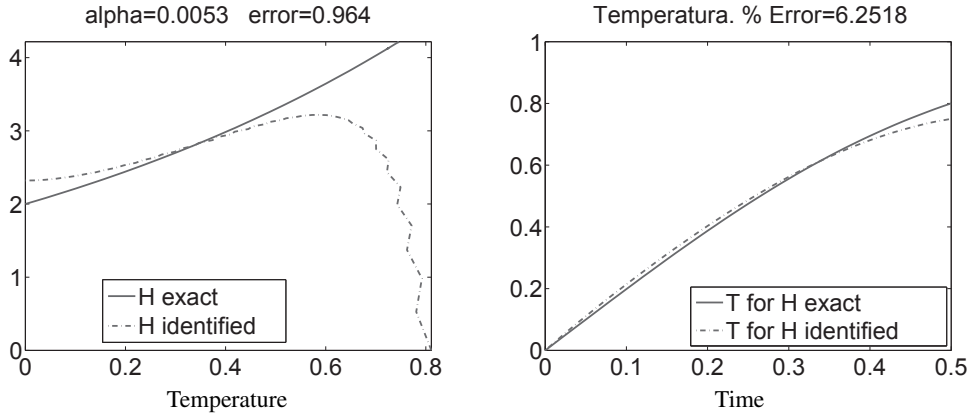

Figure 4.7: Morozov's discrepancy principle.

Numerical tests

Figures 4.7, 4.8 and 4.9 show the results obtained when applying the classical methods described above and a suitable version of the iterative algorithm of Section 4.4.1 (it is easy to prove that, in fact, this algorithm is also a regularization strategy with $R_h : \mathcal{C}[t_0, t_{\rm f}] \to \mathcal{C}[t_0, t_{\rm f}]$).

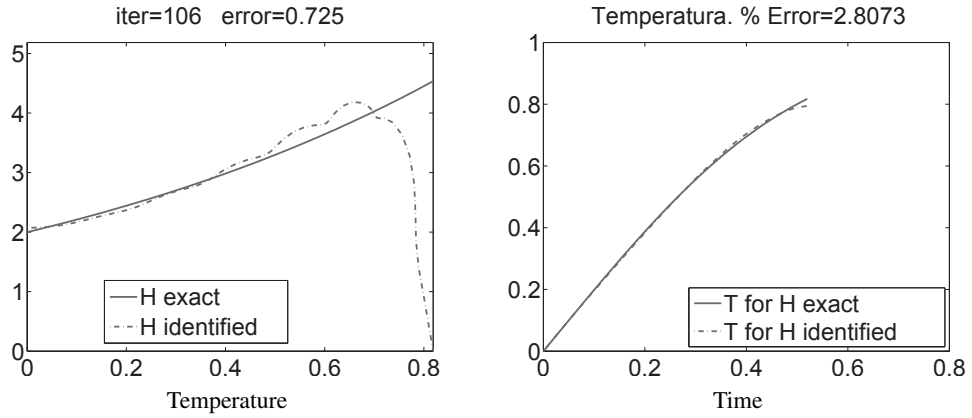

Figure 4.8: Landweber iterative method.

We point out that our iterative algorithm provides, in general, the best results. Moreover, the identification of H with this algorithm provides, for other sets of data, temperatures closer to the exact ones than those obtained after identifying H with the other methods. Finally, we observe that considering a higher order discrete derivative operator R_h the results improve.

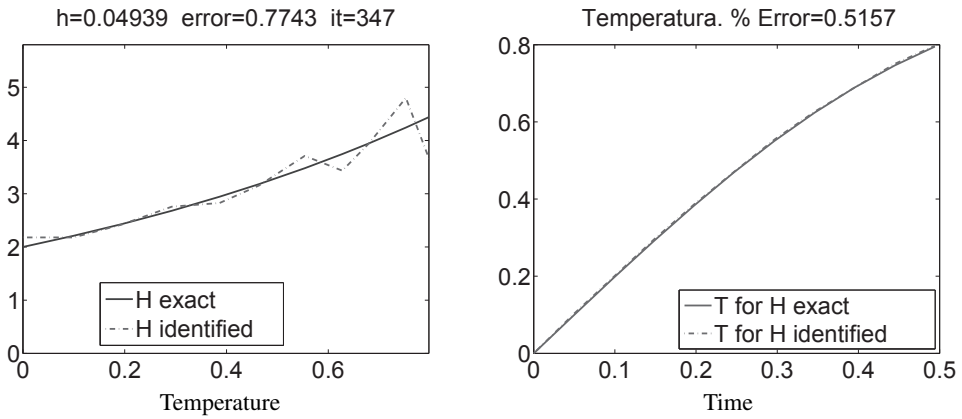

Figure 4.9: Iterative algorithm.

Bibliography

[1] R. Aris, *Vectors, Tensors, and the Basic Equations of Fluid Mechanics*. (Dover Publications, Inc. New York, 1989).

[2] G. Arroyo, P.D. Sanz, G. Préstamo, *Effect of high pressure on the reduction of microbial population in vegetables*. Journal of Applied Microbiology 82. 1997. pp. 735-742.

[3] J.C. Cheftel, *Review: high-pressure, microbial inactivation and food preservation*. Food Science Technology International 1. 1995 pp. 75-90.

[4] Dr. Secondo Gola, *Resultados microbilógicos encargados por Esteban Espuña, S.A*, a la Stazione Sperimentale per l'industria delle conserve alimentari in Parma. Parma, 5 Marzo 2004.

[5] A. Fraguela, J.A. Infante, Á.M. Ramos and J.M. Rey, *Identificación de la conductividad de un material cuando depende de la presión a la que está sometido*. Proceedings of First Symposium on Inverse Problems and Applications, Ixtapa, México.

[6] A. Fraguela, J. A. Infante, Á. M. Ramos and J. M. Rey, *Identification of a Heat Transfer Coefficient when it is a Function Depending on Temperature*. WSEAS Transactions on Mathematics, ISSN: 1109-2769, Issue 4, Volume 7, April 2008, pp. 160-172.

[7] M. Hendrickx, L. Ludikhuyze, I. Van den Broeck and C. Weemaes, *Effects of high pressure on enzymes related to food quality*. Trends in Food Science and Technology 9, 5. Elsevier, 1998 pp. 197-203.

[8] J. A. Infante, *Análisis numérico de modelos matemáticos y problemas inversos en tecnología de alimentos. Ph. D. Thesis*. Universidad Complutense de Madrid, November 24, 2009.

[9] J.A. Infante, B. Ivorra, Á.M. Ramos and J.M. Rey, *On the Modelling and Simulation of High Pressure Processes and Inactivation of Enzymes in Food Engineering*. Mathematical Models

and Methods in Applied Sciences (M3AS), 19 (**12**), $2203 - 2229$. ISSN: 0218-2025 (paper), ISSN: 1793-6314 (online).

[10] I. Indrawati, L.R. Ludikhuyze, A.M. van Loey and M.E. Hendrickx, *Lipoxygenase Inactivation in Green Beans (Phaseolus vulgaris L.) Due to High Pressure Treatment at Subzero and Elevated Temperatures*, J. Agric. Food Chem. **48** (2000) 1850–1859.

[11] L. Ludikhuyze, I. Van den Broeck, C. Weemaes, C. Herremans, J.F. Van Impe, M. Hendrickx and P. Tobback, *Kinetics for Isobaric-Isothermal Ianctivation od Bacillus subtilis α-Amylase*. Biotechnol. Prog. , 1997, 13, pp. 523-538.

[12] L. Otero, Á.M. Ramos, C.de Elvira and P.D. Sanz, *A Model to Design High-Pressure Processes Towards an Uniform Temperature Distribution*. Journal of Food Engineering, ISSN: $0260 - 8774$, Vol 78 (2007), $1463 - 1470$.

[13] J.P.P.M. Smeltx, *Recent advances in the microbiology of high pressure processing*. Trends in Food Science and Technology 9, Elsevier, 1998 pp. 152-158.

[14] U.S. Food and Drug Administration. Center for Food Safety and Apllied Nutrition, Kinetics of Microbial Inactivation for Alternative Food Processing Technologies. 2000.
(Web: http://www.cfsan.fda.gov/$\sim$comm/ift-over-html)

[15] R. Xiong, G. Xie, A.E. Edmondson, M.A. Sheard, *A mathematical model for bacterial inactivation*. International Journal of Food Microbiology 46 (1999) pp. 45-55. Elsevier.

Chapter 5

Estimation of parameters for an Influenza A(H1N1) $SIRC$ Model with Delay

Gerardo Emilio García Almeida[1], Eric José Avila Vales[1],
Daniel Israel Cauich Pacheco[1], Luis Blanco Cocom[1]

Abstract

We propose a $SIRC$ model with delay to describe the recent influenza A(H1N1) outbreak in the state of Yucatán, Mexico. The incubation period is introduced as a delay in some of the equations in the model. Our interests regarding this model are to perform some stability analysis and to estimate the reproduction number R_0 and the parameters involved in the equations.

5.1 Introduction

Influenza A (H1N1) is a recent well-known public health issue that emerged in Mexico in February 2009 caused by a new strain of the Influenza A virus [4]. Therefore it is important to find the behavior of this new type of virus. One way to achieve this is to use mathematical models like SIR, $SIRS$, $SIRC$, etc. The aim of the present work is to show some advances regarding the implementation of a $SIRC$ model with delay for influenza A (H1N1). We are interested in the stability analysis of the model and the estimation of the basic reproduction number and some of the parameters involved in the equations. Data provided by the health authorities (Secretaría de Salud de Yucatán (SSY))

[1]Facultad de Matemáticas de la Universidad Autónoma de Yucatán, galmeida@uady.mx, avila@uady.mx, danielisr_4@hotmail.com, luisd.blanco@hotmail.com

in the Mexican state of Yucatán from April 26th to September 5th 2009 was used to estimate those parameters. The data available is the report of new cases of infected people per week reported by SSY.

5.2 Description of the model

In the $SIRC$ model, the population is divided into four parts: The susceptible population S, the infected population I, the recovered population R and the cross-immune population C. The cross-immune compartment accounts for those individuals that have a partial immunity due to previous exposure to other strains of the influenza virus and provides a simplification to multi-strain models. In our case it can model the loss of immunity of the recovered individuals after a certain amount of time due to the emergence of new strains of the virus. A more detailed discussion on this issue can be found in [1].

One assumption usually done with influenza models that we also will make is that the total population is constant. This makes sense provided that the duration of the disease is short, usually around a week, compared to the expected lifetime of a human being. Using this assumption we can normalize the model and make S, I, R and C denote the corresponding fractions of the total population in the corresponding compartments rather than the number of individuals in them.

We introduce the incubation period as a delay τ in some of the equations of the model. A typical value of 2 days will be used for this delay in the numerical implementation of the model used to estimate some of the parameters involved in the equations.

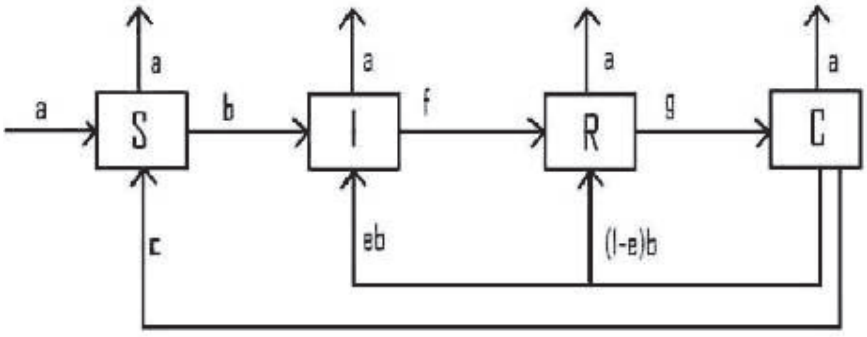

Figure 5.1: Flow diagram of the $SIRC$ model.

The description of the parameters of the preceding figure is the following:

$a =$ Birth (and death) rate of human population. Total population is assumed constant.
$b =$ Contact rate (or force of infection) of virus A (H1N1).
$c =$ Rate at which cross-immune population become susceptible again.
$e =$ Rate at which cross-immune population exposed to virus becomes infected, i.e., develops the disease again.
$f =$ Rate of recovery of infected population.
$g =$ Rate at which recovered population becomes cross-immune population, i.e., moves from total to partial immunity.

5.3 The $SIRC$ model with delay

The equations of the $SIRC$ model with delay are given by:

$$\begin{aligned}
\frac{dS(t)}{dt} &= a(1 - S(t)) - bS(t)I(t - \tau) + cC(t),\\
\frac{dI(t)}{dt} &= bS(t)I(t - \tau) + ebC(t)I(t) - (a + f)I(t),\\
\frac{dR(t)}{dt} &= (1 - e)bC(t)I(t) + fI(t) - (a + g)R(t),\\
\frac{dC(t)}{dt} &= gR(t) - bC(t)I(t) - (a + c)C(t),
\end{aligned}$$

where the parameters have the same meaning as in the previous section.

This system has two points of equilibrium:

- Disease-free equilibrium $P_0 = (1, 0, 0, 0)$
- Endemic equilibrium $P^* = (S^*, I^*, R^*, C^*)$, where

$$\begin{aligned}
S^* &= \frac{a + f}{b} - e\left[\frac{gfI^*}{[(a + g) - (1 - e)\, g]\, bI^* + (a + c)\,(a + g)}\right],\\
R^* &= \frac{fI^*\,(bI^* + a + c)}{[(a + g) - (1 - e)\, g]\, bI^* + (a + c)\,(a + g)},\\
C^* &= \frac{gfI^*}{[(a + g) - (1 - e)\, g]\, bI^* + (a + c)\,(a + g)},
\end{aligned}$$

and I^* is a root of the quadratic equation $pI^2 + qI + r = 0$, where

$$\begin{aligned}
p &= ba^2 + fab + aegb\,,\\
q &= ba\,\{f\,(2a + g + c) + (a + c)\,(a + g) + a\,(a + eg) - b\,(a + eg)\}\,,\\
r &= a\,(a + c)\,(a + g)\,(a + f)\,(1 - R_0)\,.
\end{aligned}$$

Here $R_0 = \dfrac{b}{a + f}$ is usually known as the *basic reproduction number.*

The linearization around the disease-free equilibrium is given by:

$$\begin{aligned}
\frac{dS(t)}{dt} &= -aS(t) + cC(t) - bI(t-\tau),\\
\frac{dI(t)}{dt} &= -(a+f)I(t) + bI(t-\tau),\\
\frac{dR(t)}{dt} &= fI(t) - (a+g)R(t),\\
\frac{dC(t)}{dt} &= gR(t) - (a+c)C(t),
\end{aligned}$$

and the corresponding characteristic equation for P_0 is

$$\det\left[\begin{pmatrix} \lambda+a & be^{-\lambda\tau} & 0 & c\\ 0 & \lambda+(a+f)-be^{-\lambda\tau} & 0 & 0\\ 0 & -f & \lambda+(a+g) & 0\\ 0 & 0 & -g & \lambda+(a+c)\end{pmatrix}\right]$$

$$= (\lambda+a)[\lambda+(a+f)-be^{-\lambda\tau}][\lambda+(a+g)][\lambda+(a+c)] = F(\lambda,\tau) = 0$$

5.4 Stability of the points of equilibrium

Regarding the equilibrium P_0 we have the following result:

- Case $\tau = 0$
  When the delay is zero, the second factor of the characteristic equation reduces to $[\lambda + (a + f) - b]$ and therefore the corresponding characteristic root is negative if and only if $a + f > b$, i.e., $R_0 < 1$. Being the remaining three characteristic roots negative, this implies that when $R_0 < 1$, P_0 is locally asymptotically stable.

- Case $\tau > 0$
  When the delay is zero, the second factor of the characteristic equation, which is $[\lambda + (a + f) - be^{-\lambda\tau}]$ has no pure imaginary roots for any value of the delay τ if $R_0 < 1$. Hence all the roots of the characteristic equation have negative real parts and we get that P_0 is locally asymptotically stable regardless of the value of the delay.

Regarding the equilibrium P^*, proceeding in a similar way as we did with the disease-free equilibrium we linearize the system around P^* and obtain the characteristic equation for the endemic equilibrium P^*. We can verify that P^* is positive if and only if $R_0 > 1$. For $\tau = 0$ the characteristic

equation for P^* can be written as $\lambda^4 + a_1\lambda^3 + a_2\lambda^2 + a_3\lambda + a_4 = 0$, where

$$\begin{aligned}
a_1 &= 3a + f + c + 2bI^* \,, \\
a_2 &= a\,(a + c + bI^*) + f\,(a + c) + bI^*\,[f - g\,(1 - e)] \\
&\quad + (a + bI^*)\,(2a + f + c + bI^*) + b^2S^*I^* + eb^2I^*C^* \,, \\
a_3 &= (a + bI^*)\,[(a + f)\,(a + c) + bI^*\,(a + f - g\,(1 - e))] \\
&\quad + b^2S^*I^*\,(2a + c + bI^*) + b^2I^*C^*\,(c + e\,(a + bI^*)) \\
&\quad + eb^2I^*C^*\,[a + f - g\,(1 - e)] + fbI^*\,(bS^* - ge) \,, \\
a_4 &= b^2I^*\,[a + f - g\,(1 - e)]\,[bS^*I^* + C^*\,(c + e\,(a + bI^*))] \\
&\quad + bI^*\,[bS^*\,(a + f)\,(a + c) - (c + e\,(a + bI^*)\,gf)] \,.
\end{aligned}$$

Using the Routh–Hurwitz stability criteria, P^* is locally asymptotically stable if $a_i > 0$ for $i = 1,\ 2,\ 3,\ 4$; $a_1a_2 - a_3 > 0$ and $a_1a_2a_3 - a_1^2a_4 - a_3^2 > 0$. If for the values of the parameters given the previous conditions hold and P^* starts locally asymptotically stable for $\tau = 0$, being the characteristic equation a continuous function of τ and an analytic function of λ we have that for small values of τ all the roots of this equation will have negative real part as a consequence of Rouché's Theorem. Therefore P^* will not change its stability for small values of τ if it starts locally asymptotically stable for $\tau = 0$. There can be a change in the stability of P^* (Hopf bifurcation) if some pair of its characteristic roots cross the imaginary axis. In order to find out if this happens, taking into account the assumption that the total population is constant we can express R in terms of S, I and C, reducing the model to a system of three equations. We noted that the characteristic equation of the original system of four equations for P^* can be expressed as

$$(\lambda + a)\left(\lambda^3 + b_1\lambda^2 + b_2\lambda + b_3 - \left(b_4\lambda^2 + b_5\lambda + b_6\right)e^{-\lambda\tau}\right) = 0,$$

where the second factor of the left hand side is precisely the characteristic equation of the system of three equations for P^*. When $\tau = 0$, the Routh–Hurwitz stability criteria can be applied to the cubic equation that this second factor is reduced to, obtaining a simpler set of conditions for local stability:

(H1) $\quad b_1 - b_4 > 0,$

(H2) $\quad b_3 - b_6 > 0,$

(H3) $\quad (b_1 - b_4)\,(b_2 - b_5) - (b_3 - b_6) > 0,$

where

$$\begin{aligned}
b_1 - b_4 &= 2a + c + g + 2bI^*, \\
b_2 - b_5 &= a\,(a + g + c + 3bI^*) + bI^*\,(bI^* + g + c + f + eg) + cg, \\
b_3 - b_6 &= bI^*((fg + bcC^* - begC^*)\,(1 - e) + (f + eg)\,bI^* + c\,(f + g) \\
&\qquad + a\,(a + c + g + f + bI^*)).
\end{aligned}$$

Defining

$$c_1 = b_1^2 - b_4^2 - 2b_2 \quad , \quad c_2 = b_2^2 + 2b_4b_6 - 2b_1b_3 - b_5^2.$$

and observing that $b_1 - b_4$ is obviously positive, we obtain the following result regarding P^*:

If (H2), (H3) hold and c_1, c_2 are both positive, then P^* is locally asymptotically stable for all $\tau \geq 0$. Here the coefficients of the characteristic equation are given by

$$
\begin{aligned}
b_1 &= 3a + c + f + g + 2bI^* - beC^*, \\
b_2 &= cf + 2a\,(g + c) - b^2eC^*I^* + 3a^2 + bgI^* + cg + 4abI^* - bceC^* \\
&\quad + beg\,(I^* - C^*) + bcI^* - 2abeC^* + 2fbI^* + (bI^*)^2 + 2af + fg, \\
b_3 &= a^3 + a^2\,(2bI^* + c + g + f - beC^*) + a((g + f)\,c + fg + b^2eC^*I^* \\
&\quad - becC^* + beg\,(I^* - C^*) + (2f + c + g + bI^*)\,bI^*) + cfg - begcC^* \\
&\quad - b^2egC^*I^* + (cg + egbI^* + cf + fg + bcC^* - bceC^* + fbI^*)\,bI^*, \\
b_4 &= bS^*, \\
b_5 &= S^*\left(2ab + bc + bg + b^2I^*\right), \\
b_6 &= S^*\left[a\left(ab + bc + bg + b^2I^*\right) + b\,(begI^* + cg)\right].
\end{aligned}
$$

5.5 Estimation of some of the parameters of the model

According to [1], we provide the following table with known ranges of the values of the parameters involved in the equations of the $SIRC$ model. The table includes the proposed (fixed) value to be used for the parameter a, related with the expected lifetime of a human being (considered independent of the disease) and the value of two days as typical incubation period for Influenza. For the remaining parameters an initial average value is used to generate some graphs showing the typical behavior of the model.

Parameter	Min	Max	Proposed value or initial value
a	$\frac{1}{80}$ year^{-1}	$\frac{1}{40}$ year^{-1}	$\frac{1}{60}$ year^{-1}
b	52year^{-1}	1825 year^{-1}	450 year^{-1}
c	$\frac{1}{5}$ year^{-1}	$\frac{1}{2}$ year^{-1}	$\frac{7}{20}$ year^{-1}
e	0.05	0.2	0.125
f	$\frac{365}{7}$year^{-1}	$\frac{365}{2}$year^{-1}	$\frac{365}{4.5}$year^{-1}
g	$\frac{1}{2}$ year^{-1}	1 year^{-1}	$\frac{3}{4}$ year^{-1}
τ	$\frac{1}{365}$ year	$\frac{5}{365}$ year	$\frac{2}{365}$ year

5.5.1 Estimation procedure

We consider $a = \frac{1}{60}$ year^{-1} and $\tau = \frac{2}{365}$ year fixed and we estimate the remaining parameters using the known range of values for them given in the previous table as input for a genetic algorithm. We start it generating in a random way a initial population θ of size N, where each individual of that population is a vector $\widehat{\theta}_i$. Following the scheme introduced by Goldberg [9]

and Ison et al [8], we define the *genetic coding* or *chromosome* of the individual $\widehat{\theta}_i$ as an array of m consecutive genes (parameters), one for each of the entries of $\widehat{\theta}_i$. This array is build normalizing each entry according to the interval of admissible values of the gene and storing the first s decimal places. Each of these $\widehat{\theta}_i$ is taken as initial value for a local optimization routine using the lsqcurvefit tool of the optimization toolbox of MATLAB choosing as the input function the numerical solution of the system obtained by *dde23*. The lsqcurvefit function was used with options=optimset('MaxTime',300,'Display','iter','Diagnostics','on') and a lower bound for the parameters of lb=[0;0;0;0;0]. This optimization procedure yields a new individual $\widehat{\theta}_i$. Fitness of the individuals is given by the function

$$f\left(\widehat{\theta}_i\right) = \frac{1}{r_i},$$

where r_i is the norm of the residual for the new individual $\widehat{\theta}_i$ generated by the local optimization procedure just mentioned before. The fitness of the whole population is defined as the sum of all the fitnesses of the individuals of the population.

The individuals of the population mutate with probability P_m and crossover with probability P_c. Selections are made by the roulette method. The process continues until the number of allowed generations is exceeded. The last step is to locally optimize the final population and extract from the population the estimated solution θ^* with the smallest residual.

Next, we give some define some variables to be used in the pseudocode of the algorithm:

1. $Ngen$ is the number of allowed generations,
2. $Nind$ is the number of individuals in the generation θ,
3. $Ngenes$ is the number of genes of each chromosome,
4. L is the length of the genes,
5. $Rank$ is a matrix of 2xN genes that contains the search intervals of the parameters to be estimated,
6. P_c and P_m are the crossover and mutation probabilities, respectively, and θ^* is the best approximation (solution) in the final population.

Algorithm 1 Pseudocode

Randomly initialize the population θ and define the values of [$Nind$, $Ngen$, $Ngenes$, L, P_c, P_m, $Rank$]
Read system of delay differential equations (DDEt's)$\dot{y} = f(t, \tau, y, \theta)$
Read table of experimental data T
for $iter = 1$ to $Ngen$ **do**
  for $i = 1$ to $Nind$ **do**
    Take θ_i as initial value to optimize the system of DDEt's
    Let $[\theta_i, r_i] \leftarrow$ optimization(DDEt's,T,θ_i)
    Let $Fitness(i) = 1/(r_i + 1)$ (The added 1 is to avoid a division by zero error when $r_i = 0$)
  end for
  $Fitness(iter) = sum(Fitness(:))$
  Let $n \leftarrow$ genotype(θ)
  Select and Mutate population n with probability P_c and P_m, respectively
  Let $\theta \leftarrow$ phenotype(n)
end for
Take θ_i as initial valuet to optimize the system of DDEt's
Let $[\theta_i, r_i] \leftarrow$ optimization(DDEt's,T,θ_i)
Choose the θ^* with least r_i in the population θ
Plot the result compared with T
End

5.5.2 Numerical results

First of all, in figures 5.2, 5.3 and 5.4, we show three graphs of the solution using *dde23* of the model with the proposed initial values of the parameters. Time is measured in days in all of the following graphs.

In the figure 5.5 we show the result of the estimation of parameters with lsqcurvefit using as data the numerical solution of the model by *dde23* with the proposed initial values and using as initial values for the parameter estimation the vector $x0 = [b; c; e; f; g] = [0.5; 0.001; 0.125; 0.33; 0.002]$ with corresponding $R_0 = 1.5149\text{e}{+000}$. The exact solution is
$x0 = [1.2329\text{e}{+000};\ 9.5890\text{e}{-004}; 1.2500\text{e}{-001}; 2.2222\text{e}{-001}; 2.0548\text{e}{-003}]$ with corresponding $R_0 = 5.5468\text{e}{+000}$ and the obtained approximate solution is $x = [b; c; e; f; g]$
$= [1.1547\text{e}{+000}; 2.5323\text{e}{+000}; 9.6598\text{e}{-002}; 1.8065\text{e}{-001}; 1.5817\text{e}{-002}]$ with corresponding $R_0 = 6.3901\text{e}{+000}$. The norm of the residual was $2.4817\text{e}{-006}$.

We observe that we obtain a solution that is different that the original one by noting that some of the parameters are very different compared with the original ones, although the norm of the residual is very small. This can be interpreted as the problem having more than one solution, or the residual having many local minimums that have very similar values. Making similar experiments changing the initial value of the parameters we can obtain other different solutions with residuals very close to zero. One possible cause of the multiple numerical solutions is that we are only using the susceptible population data to fit the parameters due to the limitations of the available data given by the Yucatán

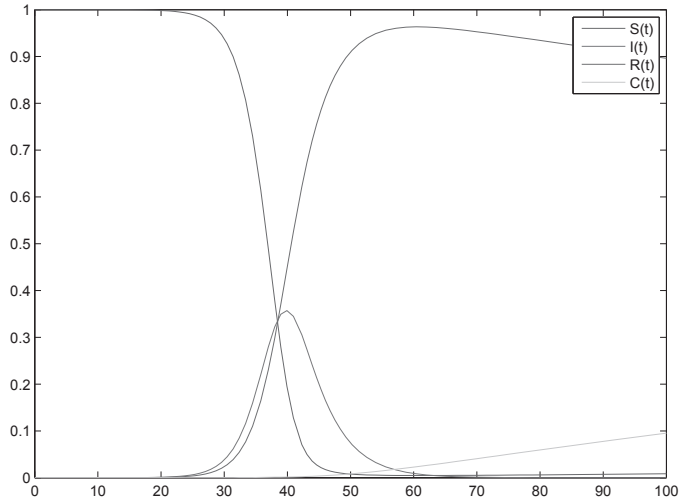

Figure 5.2: Solution of the model with proposed initial values of the parameters for time from 0 to 100 days.

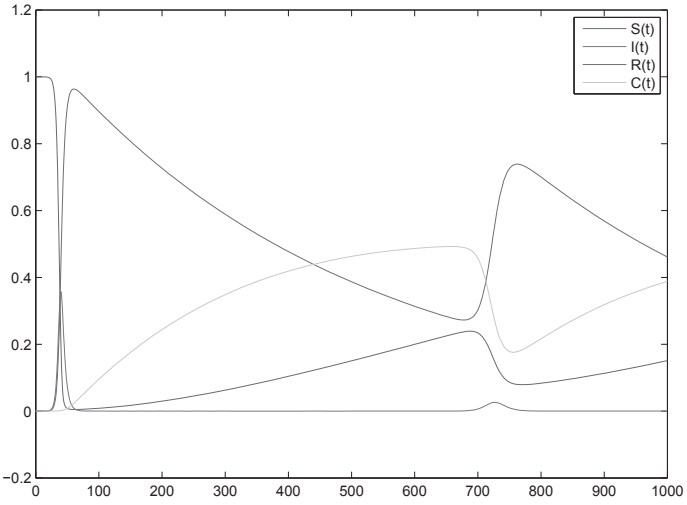

Figure 5.3: Solution of the model with proposed initial values of the parameters for time from 0 to 1000 days.

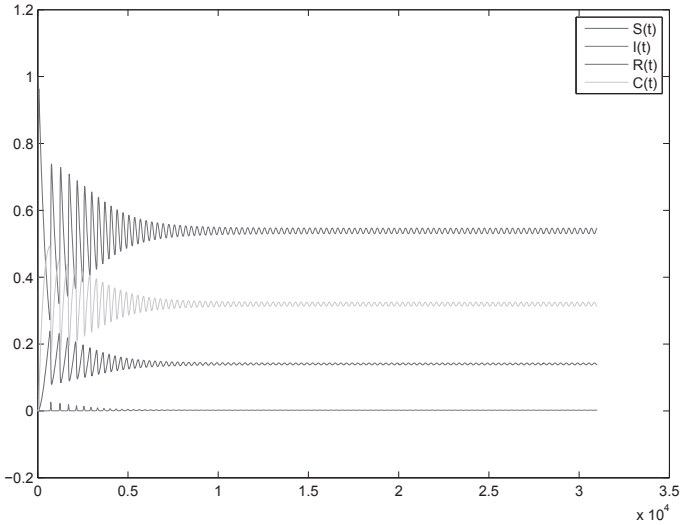

Figure 5.4: Solution of the model with proposed initial values of the parameters for time from 0 to 31000 days (84.9 years).

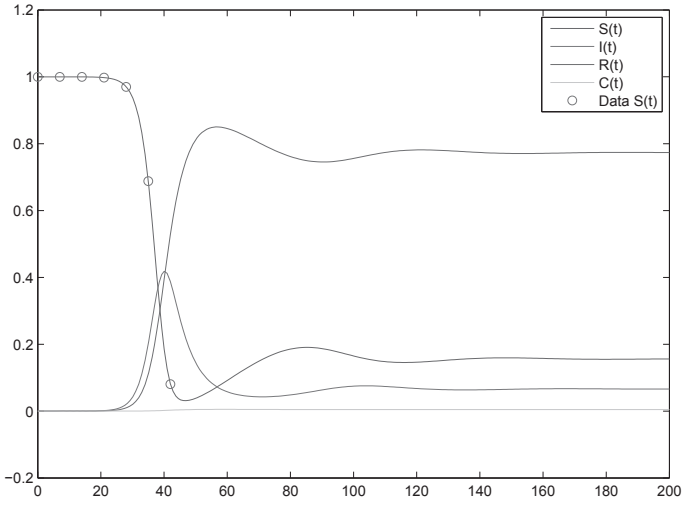

Figure 5.5: Solution of the model with the parameters obtained by the estimation procedure using lsqcurvefit for time from 0 to 200 days.

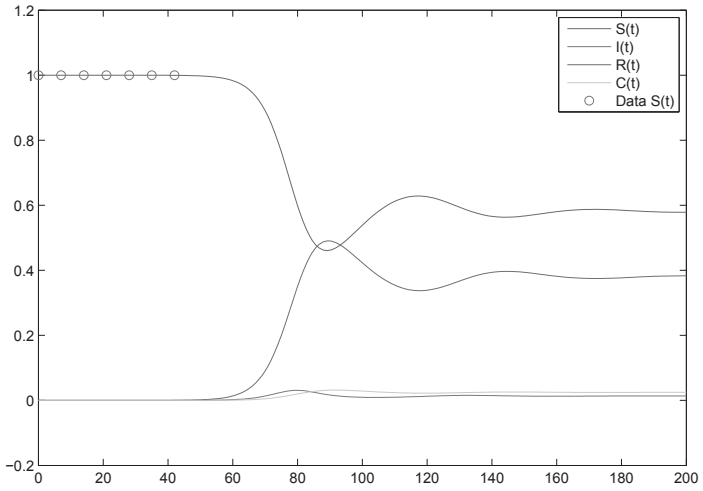

Figure 5.6: Graph corresponding to an individual obtained in an intermediate step of the algorithm for time from 0 to 200 days.

health authorities. This fact requires the use of a procedure to avoid taking a local minimum as the solution without comparing other possible local minima. We chose a genetic algorithm as such a procedure.

Figures 5.6, 5.7 and 5.8 show the results obtained by using the genetic algorithm described before with an initial population of 20 individuals and 5 generations using real data provided by the Yucatán health authorities. The first one corresponds to one individual generated in an intermediate step of the algorithm with $x = [b; c; e; f; g] = [2.6219\text{e} + 000; 8.2559\text{e} - 001; 3.8692$
$\text{e} - 001; 1.5482\text{e} + 000; 5.6256\text{e} - 002]$ with corresponding $R_0 = 1.6935\text{e}{+}000$ and norm of the residual equal to 1.0904e-007.

The last two graphs correspond to the solution obtained by the genetic algorithm. The estimated parameters are $x = [b; c; e; f; g] = [1.9069\text{e}{+}000; 2.3558$
$\text{e}{+}001; 3.3828\text{e}{-}001; 1.2885\text{e}{+}000; 1.4392\text{e}{+}001]$ with corresponding $R_0 = 1.4799\text{e}{+}000$. The norm of the residual was $3.7315\text{e}{-}008$.

We only used data from the first six epidemiological weeks because the cross-immune population is zero or very close to zero during this period (according to the model with the proposed initial values for the parameters), allowing us to compute the susceptible population with high precision. Available data covers eighteen epidemiological weeks. The data provided by the Yucatán health authorities consists of the new reported cases by epidemiological week. Therefore we cannot know the total number of infected people after some time from the beginning of the outbreak because recovered people is not reported. If the cross-immune population is zero or very small, the data allows us to only know the susceptible population with precision.

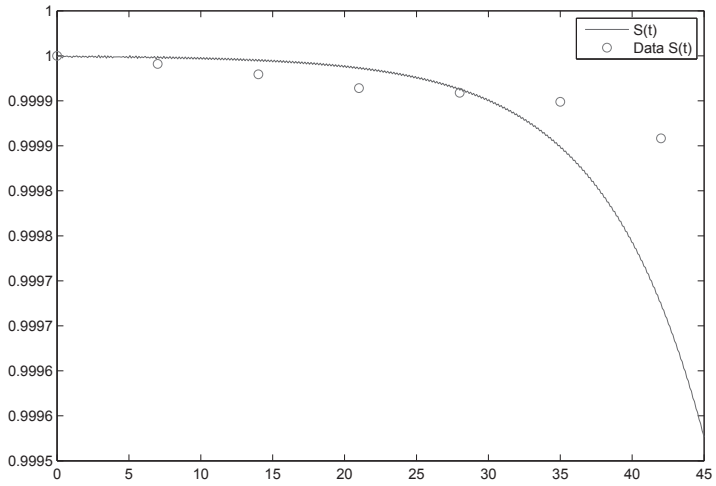

Figure 5.7: Fine detail of the susceptible fraction of the population predicted by the model with the estimated parameters compared with the given data for the estimation. Time varies from 0 to 45 days.

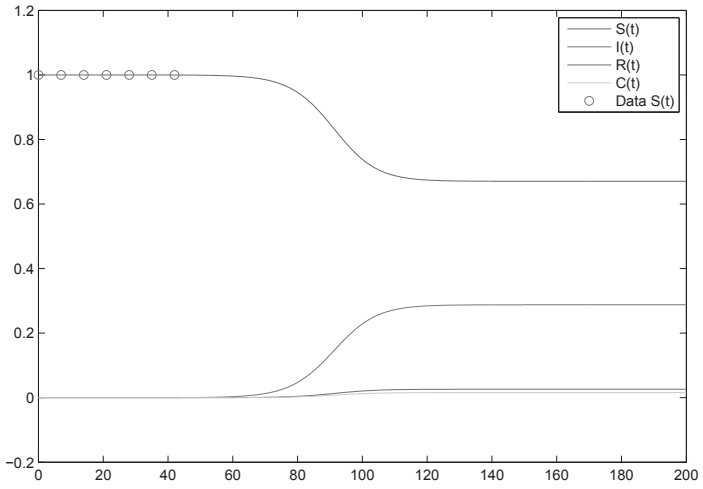

Figure 5.8: Model obtained with the estimated parameters based on real data for time from 0 to 200 days.

5.6 Conclusions

The limitations of the provided data allows us to use only the susceptible population as available information to estimate the parameters during the first six weeks of the outbreak. This allows for the existence of many possible solutions as it was evident with the synthetical data used to test the procedure. Nevertheless, the obtained values for the basic reproduction number R_0 had little variations compared to those of some of the parameters. Another observed fact is that the real data shows very little variation of the susceptible population that causes the quick convergence to the endemic equilibrium with no oscillations that are typical of Influenza outbreaks, as it is shown by the synthetical data in the first three graphs provided in the numerical results section of this paper. This suggests that the real data is not very reliable, due probably to a great quantity of underreported cases. We also note that our estimated basic reproduction number R_0 is close to the values of it obtained in [4].

Acknowledgements

We wish to thank Mexican CONACYT (Sistema Nacional de Investigadores grants 33365 and 15284), Universidad Autónoma de Yucatán and the organizers of the First Symposium on Inverse Problems and Applications for the support that made this work possible.

Bibliography

[1] R. Casagrandi, R. Bolzoni, S.A. Levin and V. Andreasen, *The SIRC model and influenza A*, Mathematical biosciences 200 (2006) 152-169.

[2] W. Chinviriyasit, *Numerical modeling of the transmission dynamics of influenza*, The first International symposium on Optimization and Systems Biology (OSB'07) 52-59.

[3] L. Jódar, R.J. Villanueva, A.J. Arenas and G.C. González, *Nonstandard numerical methods for a mathematical model for influenza disease*, Mathematics and Computers in simulation 79 (2008) 622-633.

[4] G. Cruz-Pacheco, L. Durán, L. Esteva, A.A: Minzoni, M. López-Cervantes, P. Panayotaros, A.Ajued Ortega and I. Villaseñor Ruiz, *Modelling of the influenza A(H1N1)v outbreak in Mexico City, April-May 2009, with control sanitary measures*, Eurosurveillance Vol. 14 Issue 26, 2 July 2009, http://www.eurosurveillance.org

[5] Secretaría de Salud de Yucatán, *Data from Secretaría de Salud de Yucatán (SSY),* http://www.salud.yucatan.gob.mx/principal/neumoniaH1N1.pdf

[6] World Health Organization, *Data from World Health Organization,* http://new.paho.org/hq/images/atlas/en/atlas.html

[7] Lawrence F. Shampine, I. Gladwell and S. Thompson, *Solving ODEs with MATLAB*, Cambridge University Press, 2003.

[8] M. Ison, J. Sitt and M. Trevisan, *Curso de Sistemas Complejos: Algoritmos Genéticos, Aplicación en MATLAB*, http://www.df.uba.ar/users/mison/genetico.tar.gz , 2005.

[9] D. Goldberg, *Genetic Algorithms in Search Optimization and Machine Learning*, Addison-Wesley Longman Inc., 1989.

[10] A. Pinninghof, E. Matthews and H. Díaz, *Diseño de Redes Viales Urbanas Usando Algoritmos Genéticos*, Rev. Ing. Infor. Ed 10, 2004.

Chapter 6

Estimación de Parámetros de un Modelo Matemático de la Influenza A(H1N1)

Justino Alavez Ramírez[1], Guillermo Gómez Alcaraz[2],
Luis Manuel Hernández Gallardo[2], Jesús López Estrada[2],
Cruz Vargas-de-León[3]

Resumen

Se propone un modelo matemático de cuatro poblaciones (susceptibles, asintomáticos, sintomáticos y recuperados) con el propósito de entender y predecir la evolución de la epidemia de la influenza A(H1N1). Usando los datos de los casos confirmados de la población del país de abril y mayo de 2009 para la estimación numérica de los parámetros en el modelo propuesto, se prueba que la población de individuos asintomáticos no resulta ser potencialmente infecciosa como se pensó en un principio.

6.1 Introducción

La gripe A(H1N1) que surgió en 2009 fue una pandemia causada por una variante del virus de la influenza tipo A de origen porcino (subtipo H1N1), que la Organización Mundial de la Salud (OMS) reconoció oficialmente como Virus H1N1/09 Pandémico. Esta nueva cepa viral fue conocida

[1]División Académica de Ciencias Básicas, Universidad Juárez Autónoma de Tabasco, Cunduacán 86690 Tabasco, México, justino.alavez@basicas.ujat.mx

[2]Depto. de Matemáticas, Fac. de Ciencias, UNAM, 04510 México, D.F.

[3]Unidad Académica de Matemáticas, Universidad Autónoma de Guerrero, México. Facultad de Estudios Superiores Zaragoza, UNAM, México.

como gripe porcina (nombre que se le dió inicialmente), gripe norteamericana (por la Organización Mundial de la Salud Animal) y nueva gripe (por la Unión Europea). El 30 de abril de 2009, la OMS decidió denominarla gripe A(H1N1) [1]. El origen de la infección es una variante de la cepa H1N1 con material genético proveniente de una cepa aviaria, dos cepas porcinas y una humana, que sufrió una mutación y dió un salto entre especies (o heterocontagio) de los cerdos a los humanos, permitiendo el contagio de persona a persona [1]. El 11 de junio de 2009, la OMS la clasificó como de nivel de alerta seis; es decir, pandemia actualmente en curso que involucra la aparición de brotes comunitarios (ocasionados localmente sin la presencia de una persona infectada proveniente de la región del brote inicial). Ese nivel de alerta no define la gravedad de la enfermedad producida por el virus, sino su extensión geográfica [2]. La tasa de letalidad de la enfermedad que inicialmente fue alta, pasó a ser baja al iniciar los tratamientos con antivirales a los que es sensible. Sin embargo, la futura evolución del virus es impredecible, como constató la directora general de la OMS Margaret Chan el 4 de mayo de 2009, ya que "puede que en un mes este virus desaparezca, puede que se quede como está o puede que se agrave" [1]. Se presume que el nuevo virus de la influenza A(H1N1) pandémica pudo haber surgido en México y en el sur de California [3].

La influenza A(H1N1) es contagiosa pero tratable y controlable, si se diagnostica a tiempo. Los síntomas son parecidos a los del catarro común, pero más severos, y su inicio es generalmente abrupto. Por ejemplo, en las personas infectadas por la influenza A(H1N1) un intenso dolor de cabeza se presenta súbitamente, y en las personas infectadas por el catarro común es raro que manifiesten dolor de cabeza [4]. La transmisión de la influenza A(H1N1) es de persona a persona, similar a la de la influenza estacional; es decir, principalmente cuando las personas infectadas por el virus de la influenza hablan, tosen o estornudan hasta un metro de distancia de otras aparentemente sanas. Las personas también pueden infectarse al tocar enseres domésticos o algo que contenga el virus de la influenza y después llevarse las manos a la boca o nariz.

Los primeros reportes médicos indicaban que las personas infectadas podrían infectar a otras, a partir del primer día antes de desarrollar síntomas y hasta siete o más días después de enfermarse. Lo anterior significaba que una persona era capaz de transmitir la influenza a otra persona antes de que supiera que estaba enferma. Esta consideración nos llevó a la propuesta (sección 6.2) de un un modelo matemático de cuatro poblaciones (susceptibles, asintomáticos, sintomáticos y recuperados), con el objetivo de entender y predecir la evolución de la epidemia de la influenza A(H1N1), mediante la estimación numérica de los parámetros del modelo (sección 6.4) usando como datos los casos confirmados de la población del país, del 10 de abril al 22 de mayo de 2009. Como consecuencia de esta investigación, se muestra que la población de individuos asintomáticos no es potencialmente infecciosa como se pensó en un principio. En la sección 6.3, se introduce un número de reproductividad básico del virus, y en la sección 6.5, se discuten los resultados y sus posibles implicaciones.

Un primer trabajo orientado a estudiar la evolución de la epidemia con datos de la ciudad de México y con un modelo matemático tipo compartamental es el de Cruz-Pacheco *et al.* [5]. El estudio matemático basado en ecuaciones diferenciales de la evolución de las epidemias data desde el siglo XVIII [6]. Sin embargo, los fundamentos matemático del estudio de la epidemiología basados en modelos campartamentales se desarrollaron hasta el siglo XX [7, 8, 9, 10, 11].

6.2 Modelo matemático

Se considera un modelo tipo poblacional bajo el supuesto de que los individuos de la población nacen y mueren a la misma tasa debido al tiempo relativamente corto de la epidemia. Como se dijo en la Introduccion, los reporte médicos indicaban en un principio que los individuos asintomáticos también debían ser considerados infecciosos al igual como aquellos individuos con síntomas, razón por lo que se incluyen en el modelo a desarrollar, ambas poblaciones. La población total N se dividió en cuatro subpoblaciones: la población S de los individuos susceptibles al tiempo t, la población A de individuos asintomáticos infecciosos al tiempo t, la población I de los individuos sintomáticos infecciosos al tiempo t y la población R de los individuos recuperados al tiempo t. En la figura 6.1, se muestra el diagrama de transferencia de una población a otra.

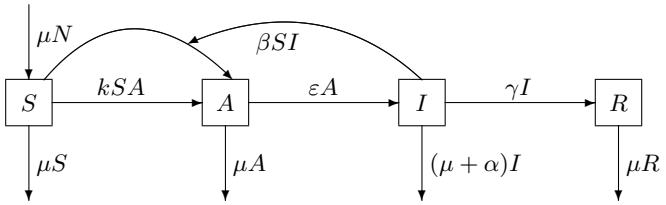

Figura 6.1: Diagrama de transferencia entre las poblaciones.

Del diagrama de transferencia de la figura 6.1, resulta el siguiente sistema de ecuaciones diferenciales ordinarias:

$$\begin{aligned} \dot{S} &= \mu N - \beta SI - \kappa SA - \mu S \\ \dot{A} &= \beta SI + \kappa SA - (\varepsilon + \mu) A \\ \dot{I} &= \varepsilon A - (\gamma + \mu + \alpha) I \\ \dot{R} &= \gamma I - \mu R \end{aligned} \tag{1}$$

donde μ es la tasa de mortalidad natural de la población, β es la tasa de encuentros probables entre individuos sintomáticos infecciosos I con individuos susceptibles S, k es la tasa de encuentros probables entre individuos asintomáticos infecciosos A con individuos susceptibles S, ε es la tasa de transferencia de los individuos asintomáticos A a individuos sintomáticos infecciosos I, γ es la tasa de recuperación de los individuos sintomáticos infecciosos I y α es la tasa de mortalidad inducida por la enfermedad. Todos los parámetros en el sistema (**1**) son positivos. Modelos similares se pueden encontrar en [5, 7, 8, 9, 10, 11].

6.3 Estados de equilibrio: número de reproductividad básico R_0

Los puntos de equilibrio (S^*, A^*, I^*, R^*) del sistema (**1**) se obtienen resolviendo el sistema de ecuaciones algebraicas:

$$\begin{aligned} \mu N - \beta SI - \kappa SA - \mu S &= 0 \\ \beta SI + \kappa SA - (\varepsilon + \mu) A &= 0 \\ \varepsilon A - (\gamma + \mu + \alpha) I &= 0 \\ \gamma I - \mu R &= 0 \end{aligned} \tag{2}$$

Si $I^* = 0$, entonces $R^* = 0$, $A^* = 0$ y $S = N$, por lo que $E_0 = (N, 0, 0, 0)$ es el punto de equilibrio libre de epidemia, usualmente se le llama punto de equilibrio trivial.

Si $I^* \neq 0$, se sigue de la cuarta ecuación de (**2**) que

$$R^* = \frac{\gamma}{\mu} I^* , \tag{3}$$

y de la tercera ecuación de (**2**) se obtiene

$$A^* = \frac{\gamma + \mu + \alpha}{\varepsilon} I^* , \tag{4}$$

y de la segunda ecuación de (**2**) resulta

$$S^* = \frac{(\varepsilon + \mu)A^*}{\beta I^* + kA^*} = \frac{(\varepsilon + \mu)(\gamma + \mu + \alpha)}{\varepsilon\beta + k(\gamma + \mu + \alpha)} . \tag{5}$$

Ahora, sustituyendo (**4**) en la primera ecuación de (**2**) se obtiene

$$[\varepsilon\beta S^* + k(\gamma + \mu + \alpha)S^*]I^* = \varepsilon\mu N - \varepsilon\mu S^* ,$$

de donde

$$I^* = \frac{\varepsilon\mu N}{[\varepsilon\beta + k(\gamma + \mu + \alpha)]S^*} \left(1 - \frac{S^*}{N}\right) .$$

Sustituyendo (**5**) en la ecuación anterior, resulta

$$I^* = \frac{\varepsilon\mu N}{(\varepsilon + \mu)(\gamma + \mu + \alpha)} \left(1 - \frac{1}{R_0}\right) , \tag{6}$$

donde

$$R_0 = \frac{N[\varepsilon\beta + k(\gamma + \mu + \alpha)]}{(\varepsilon + \mu)(\gamma + \mu + \alpha)} ,$$

es un número adimensional y que se define como el número de reproductividad básico del virus.

Introduciendo R_0 en (**5**) e I^* en (**3**) y (**4**), se sigue que

$$\begin{aligned} S^* &= \frac{N}{R_0} , \\ R^* &= \frac{\gamma\varepsilon N}{(\varepsilon + \mu)(\gamma + \mu + \alpha)} \left(1 - \frac{1}{R_0}\right) , \\ A^* &= \frac{\mu N}{\varepsilon + \mu} \left(1 - \frac{1}{R_0}\right) . \end{aligned} \tag{7}$$

Así que el punto de equilibrio no trivial o epidémico resulta ser $E_1 = (S^*, A^*, I^*, R^*)$, donde las componentes están dadas por (**6**) y (**7**). Dicho punto de equilibrio es admisible si y sólo si $R_0 > 1$.

Theorem 6.3.1. *Bajo el supuesto de que todos los parámetros del sistema* (**1**) *son positivos:*

i) Si $R_0 \leq 1$, entonces E_0 es el único estado de equilibrio admisible.

ii) *Si* $R_0 > 1$*, entonces el sistema* (**1**) *tiene dos puntos de equilibrio admisibles. El trivial* E_0 *y el no trivial* E_1.

Demostración. Obviamente si $R_0 = 1$, entonces E_1 se reduce a E_0; y si $R_0 < 1$, entonces E_1 tiene tres componentes negativos, por consiguiente no es admisible. Así que E_0 es el único estado de equilibrio admisible. □

De aquí la importancia epidemiológica de R_0, permite pronosticar si una epidemia progresará o desaparecerá en una población.

6.4 Estimación numérica de los parámetros

Para la estimación numérica de los parámetros del modelo (**1**), se consideraron los datos publicados del número de individuos sintomáticos infecciosos I en el país, el 18 de junio de 2009 en el portal de la Secretaría de Salud [12]. Se tomaron los datos que corresponden del 10 de abril al 22 de mayo, mismos que se reproducen en la tabla 6.1.

Fecha	10/04	11/04	12/04	13/04	14/04	15/04	16/04	17/04
I	3	4	11	17	25	17	12	26
Fecha	18/04	19/04	20/04	21/04	22/04	23/04	24/04	25/04
I	32	44	106	112	148	218	274	309
Fecha	26/04	27/04	28/04	29/04	30/04	01/05	02/05	03/05
I	385	404	292	270	221	201	182	206
Fecha	04/05	05/05	06/05	07/05	08/05	09/05	10/05	11/05
I	224	223	231	172	162	132	136	156
Fecha	12/05	13/05	14/05	15/05	16/05	17/05	18/05	19/05
I	129	112	91	76	72	85	94	70
Fecha	20/05	21/05	22/05					
I	67	78	62					

Tabla 6.1: Casos confirmados de los individuos sintomáticos en México [12].

Los funcionarios médicos de los Estados Unidos advirtieron que el total de infectados en todo su país debía de considerarse igual, por factor de 10, al total de casos confirmados por la infección del virus A(H1N1), pues muchos de los casos –por ser leves– no acudieron al médico, ni al hospital, y por ende no quedaron registrados. Los casos confirmados por laboratorio representan tan solo una fracción del número probable de casos reales de enfermos, debido al mismo subregistro. Para México, los médicos consideraron mayor el subregistro y las estimaciones más conservadoras estimaban que los casos confirmados debían multiplicarse por un factor de 100, otros por uno de 700, y otros hasta por uno de 1000. Ante este hecho, se decidió estimar los parámetros multiplicando los datos de la tabla 6.1 primero por un factor de 1, después por un factor de 10, 25, 50, 100, 700 y 1000, respectivamente.

Considerando que la esperanza de vida media de la población en México es de 74.6 años [13],

se deduce ([14, 15]) que la tasa de nacimientos y mortalidad natural de la población es de

$$\mu \cong \frac{\ln 2}{74.6 \times 365\ días} = 2.5456 \times 10^{-5}/día\,,$$

y tomando $N = 107\,550\,697$, como la población total estimada del país en mayo de 2009 [16], se obtuvieron las estimaciones de los parámetros que se muestran en las tablas 6.2 y 6.3. Los resultados gráficos se muestran en las figuras 6.2 al 6.9. Los experimentos numéricos se hicieron con el software *DIFFPAR* [17].

Parám./Factor	1	10	25	50
β	2.39×10^{-3}	1.98×10^{-4}	7.37×10^{-5}	3.50×10^{-5}
k	8.74×10^{-20}	8.74×10^{-20}	5.34×10^{-25}	8.74×10^{-22}
ε	6.69×10^{-2}	6.92×10^{-2}	7.03×10^{-2}	7.11×10^{-2}
γ	1.65×10^{4}	1.65×10^{3}	6.59×10^{2}	3.30×10^{2}
α	1.62×10^{-8}	1.62×10^{-2}	1.62×10^{-1}	1.62×10^{-4}
R_0	15.58	12.94	12.01	11.41

Tabla 6.2: Resultados de la estimación de parámetros al multiplicar los casos confirmados por factores de 1, 10, 25 y 50, respectivamente.

Parám./Factor	100	700	1000
β	1.69×10^{-5}	2.38×10^{-6}	1.68×10^{-6}
k	8.74×10^{-22}	8.74×10^{-22}	8.74×10^{-22}
ε	7.17×10^{-2}	7.20×10^{-2}	7.19×10^{-2}
γ	164.83	23.56	16.48
α	1.62×10^{-1}	1.62×10^{-6}	1.62×10^{-8}
R_0	11.02	10.87	10.95

Tabla 6.3: Resultados de la estimación de parámetros al multiplicar los casos confirmados por factores de 100, 700 y 1000, respectivamente.

6.5 Discusión

Una primera conclusión, la curva de ajuste a la población de individuos sintomáticos infecciosos I se ajusta muy bien a los casos confirmados (figura 6.3), como también ocurre con las curvas de ajuste a los datos multiplicados por factores de 10, 25, 50, 100, 700 y 1000, respectivamente (véanse las figuras 6.4–6.9). Nótese que todas las figura muestran la misma tendencia de la figura 6.3. Otro resultado interesante, los valores que se obtienen de la tasa de recuperación γ de la población de los individuos sintomáticos infecciosos I, que van desde 16.48 hasta 1.65×10^4 (tablas 6.2 y 6.3), plantea la pregunta ¿Es cierto que los individuos sintomáticos se recuperan tan relativamente rápido?

También debe hacerse notar que los valores de la tasa de mortalidad α inducida por la enfermedad, cuyos valores van desde 1.62×10^{-8} hasta 1.62×10^{-1} (tablas 6.2 y 6.3), implican que la

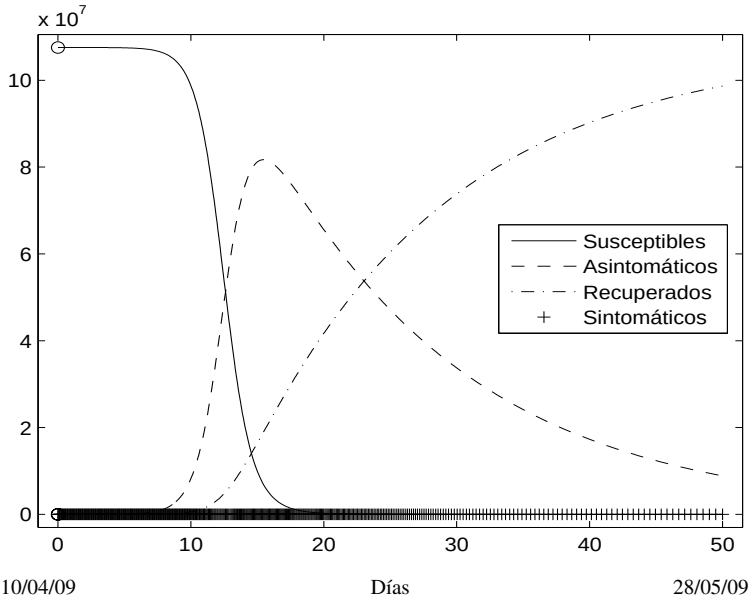

Figura 6.2: Dinámica de la influenza A(H1N1) obtenida con los datos confirmados (tabla 6.1). Se obtienen dinámicas similares con datos amplificados por factores de 10, 25, 50, 100, 700 y 1000, respectivamente.

tasa de letalidad de la enfermedad no es muy alta. Lo que confirma lo dicho en la Introducción: *que la tasa de letalidad pasa a ser baja al iniciar los tratamientos con antivirales.*

Comparemos los valores de la tasa de infección β de los individuos sintomáticos I con la tasa de infección k de los individuos asintomáticos A. Los valores de β van desde 1.68×10^{-6} hasta 2.39×10^{-3} (tablas 6.2 y 6.3), mientras que los valores de k van desde 5.34×10^{-25} hasta 8.74×10^{-20} (tablas 6.2 y 6.3). Esto significa que la letalidad de la población de los individuos asintomáticos A es prácticamente nula comparada con la población de los individuos sintomáticos I. Esta última conclusión está muy en consonancia con *la comunicación personal de médicos del INER consistente en que es poco significativa la aportación de los asintomáticos por las causas clínicas por ellos citadas*.

Agradecimientos

Los autores agradecen a los árbitros anónimos por sus valiosos comentarios y sugerencias que contribuyeron para mejorar el manuscrito.

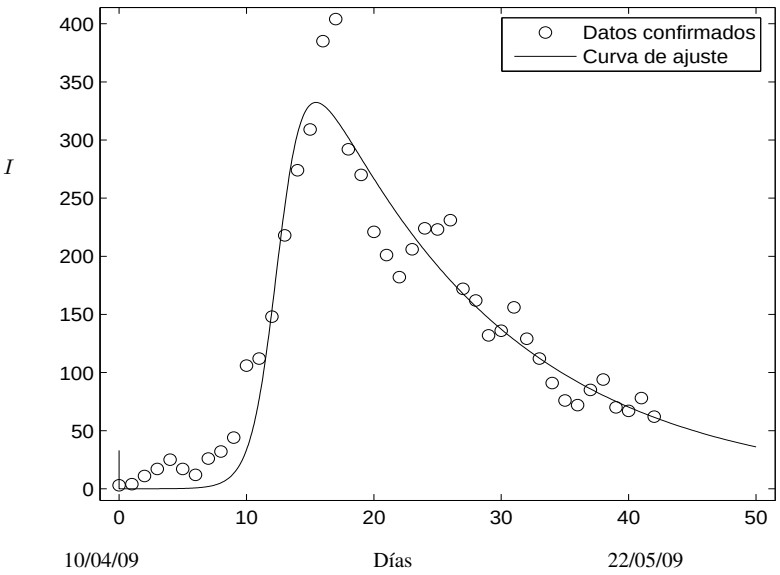

Figura 6.3: Evolución de la población de individuos sintomáticos I.

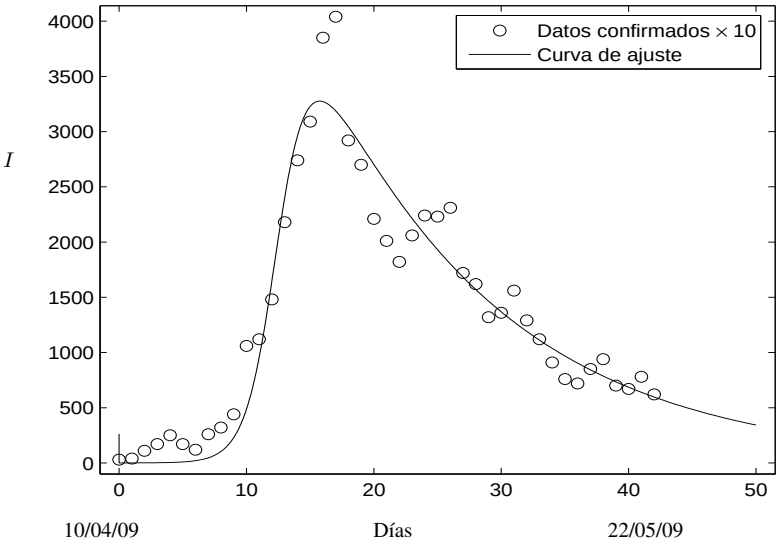

Figura 6.4: Evolución de la población de individuos sintomáticos I con datos multiplicados por un factor de 10.

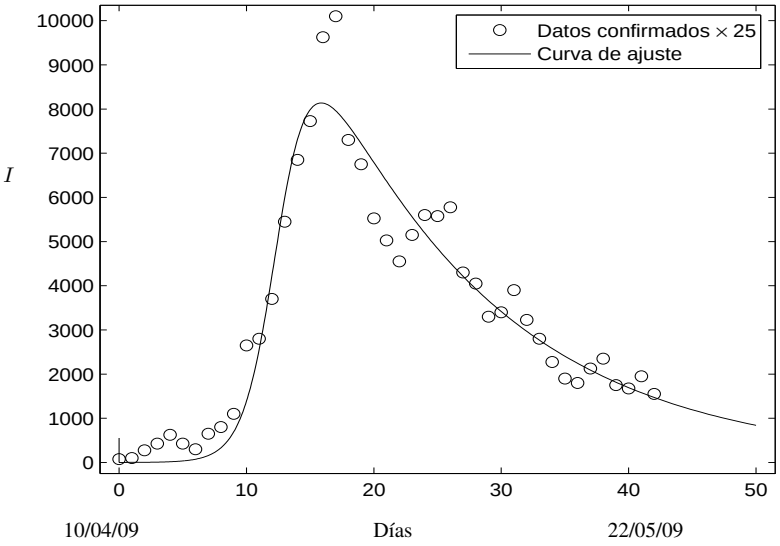

Figura 6.5: Evolución de la población de individuos sintomáticos I con datos multiplicados por un factor de 25.

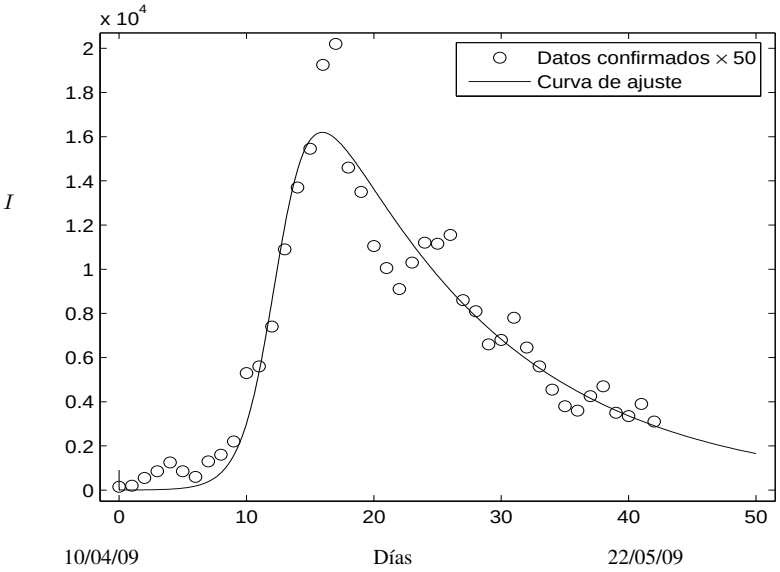

Figura 6.6: Evolución de la población de individuos sintomáticos I con datos multiplicados por un factor de 50.

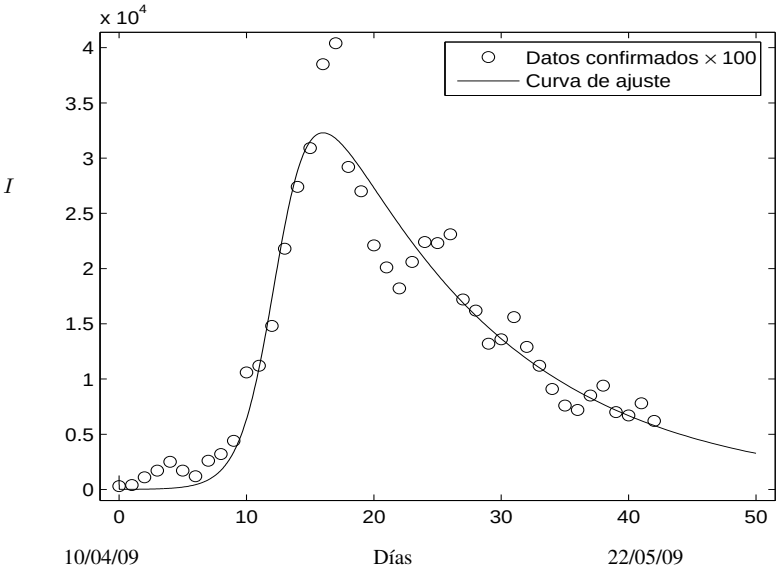

Figura 6.7: Evolución de la población de individuos sintomáticos I con datos multiplicados por un factor de 100.

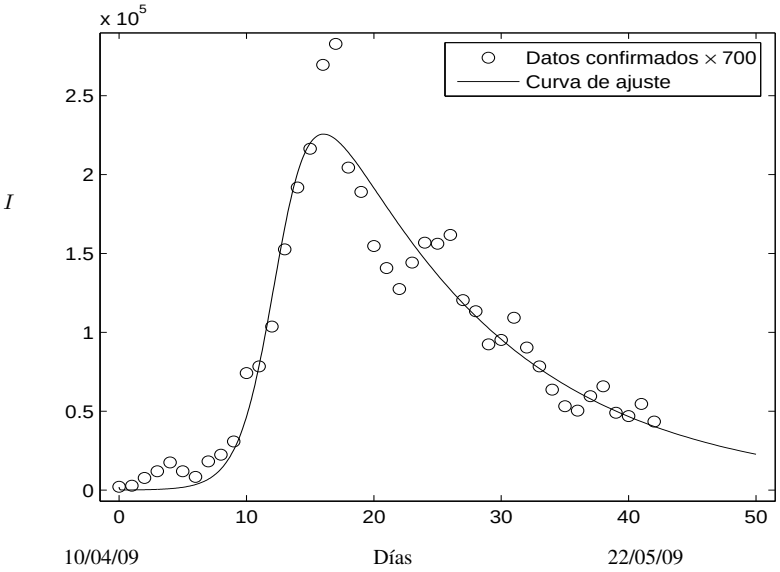

Figura 6.8: Evolución de la población de individuos sintomáticos I con datos multiplicados por un factor de 700.

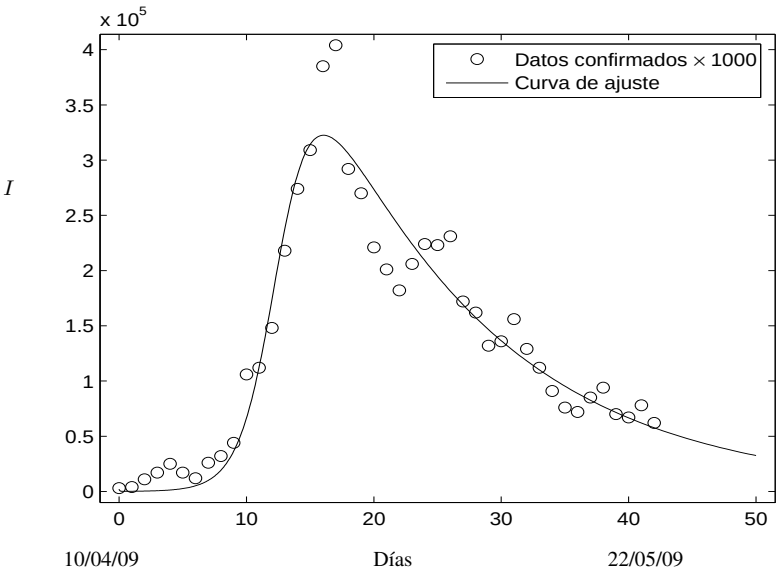

Figura 6.9: Evolución de la población de individuos sintomáticos I con datos multiplicados por un factor de 1000.

Bibliografía

[1] *Pandemia de Gripe A(H1N1) de 2009*. http://es.wikipedia.org/wiki/Pandemia_de_gripe_A_(H1N1)_de_2009

[2] D. Balcan, H. Hu, B. Goncalves, P. Bajardi, Ch. Poletto, J.J. Ramasco, D. Paolotti, N. Perra, M. Tizzoni, W. Van den Broeck, V. Colizza, and A. Vespignani, *Seasonal transmission potential and activity peaks of the new influenza A(H1N1): A Monte Carlo likelihood analysis based on human mobility*. BMC Medicine, **7**(45), 2009.

[3] L. García-García, J.L. Valdespino-Gómez, E. Lazcano-Ponce, A. Jiménez-Corona, A. Higuera-Iglesias, P. Cruz-Hervert, B. Cano-Arellano, A. García-Anaya, E. Ferreira-Guerrero, R. Baez-Saldaña, L. Ferreyra-Reyes, S. Ponce-de-León-Rosales, C. Alpuche-Aranda, M.H. Rodríguez-López, R. Pérez-Padilla, M. Hernández-Avil, *Partial protection of seasonal trivalent inactivated vaccine against novel pandemic influenza A/H1N1 2009: Case-control study in Mexico city*. BMJ 339:b3928 (2009).

[4] *Influenza: Signos y Síntomas de la Influenza*. Instituto nacional de Salud Pública y Secretaría de Salud, 2010. http://bvs.insp.mx/docs/signos_sintomas.pdf

G. Cruz-Pacheco, L. Duran, L. Esteva, A.A. Minzoni, M. López-Cervantes, P. Panayotaros, A. Ahued Ortega, I. Villaseñor Ruíz, *Modelling of the influenza A(H1N1)v outbreak in Mexico city, april-may 2009, with control sanitary measures*. Eurosurveillance, **14**(26), 2009.

[5] K. Dietz and J.A.P. Heesterbeek, *Daniel Bernoulli's epidemiological model revisited*. Mathematical Biosciences, **180**, pp. 1–21 (2002).

[6] W.O. Kermack and A.G. McKendrick, *A contributions to the mathematical theory of epidemics*. Proceedings of the Royal Society of London, Series A-**115**, pp. 700–721 (1927).

[7] F. Brauer, P. van den Driessche, and J. Wu (Eds.), *Mathematical Epidemiology*. Springer-Verlag Berlin Heidelberg (2008).

[8] M.Y. Li and J.S. Muldowney, *Global stability for the SEIR model in epidemiology*. Mathematical Biosciences, **125**, pp. 155–164 (1995).

[9] H.W. Hethcote, *The basic epidemiology models I & II: Models, expressions for R_0, parameter estimation, and applications*. Master Review, 2005.
http://www.worldscibooks.com/etextbook/7020/7020_chap01.pdf

[10] M.E. Alexander and S.M. Moghadas, *Bifurcation analysis of an SIRS epidemic model with generalized incidence*. SIAM J. Appl. Math., **65**(5), pp. 1794–1816 (2005).

[11] *Situación Actual de la Epidemia, Reportes de la Secretaría de Salud de México*, 2009.
http://portal.salud.gob.mx

[12] V. Partida Bush, *Proyecciones de la Población de México 2005-2050*. CONAPO, 2006.

[13] L. Min, Y. Su and Y. Kuang, *Mathematical analysis of a basic virus infection model with application to HBV infection*. Rocky Mountain Journal of Mathematics, **38**(5), pp. 1573–1585 (2008).

[14] A.S. Perelson and P.W. Nelson, *Mathematical analysis of HIV-1 dynamics in vivo*. SIAM Review, **41**(1), pp. 3–44 (1999).

[15] *Brote de Influenza A H1N1 México*, Boletín Diario No. 22 de la Dirección General Adjunta de Epidemiología, Gobierno Federal, 18 de mayo de 2009.

[16] L. Edsberg and P.–Å. Wedin, *Numerical tools for parameter estimation in ODE-systems*. Optimization Methods and Software, **6**(3), pp. 193–217 (1995).

Chapter 7

Deconvolution, parameter estimation and image recovering

Mario Medina[1], Eymard Hernández[1]

Abstract

We present an approach to blind deconvolution by using the Lucy-Richardson algorithm EM. For this purpose we use a statistical tool algorithm, the so called *expectation-maximization*. The Lucy-Richardson algorithm is compared with classical techniques as the *inverse filter* and *Wiener filter*. These techniques are widely used to restore degraded images where the degradation is due to blurring or a combination of blurring and noise. The problem of restoration of images is approached from a more general setting, the field of inverse problems, these problems are present in many ways in science.

If a image only presents blurring, the Lucy-Richardson algorithm allows to reconstruct the image with a quality comparable to that of *inverse filter* and *Wiener filter*. If the image also presents noise, the *inverse filter* is not efficient since the noise in amplified. At the end of this paper we compare results obtained with the *inverse filter*, *Wiener filter* and *Lucy-Richardson* for images with degradation due only to blurring or to noise or due to a combination of them.

7.1 Introduction

Images are essential in science and in everyday life. For us it is natural to see landscapes, people, pictures in real life. Images reach many areas; from photography to astronomy, going through other areas such as medical imaging, microscopy. In each case these images are the basis of what we

[1]Department of Mathematics, UAM-Iztapalapa, México D.F., mvmg@xanum.uam.mx, eymardh7@gmail.com

observe, so we want to see the representation which is closest to the reality of the scenes in question, which we call the original, real or true image.

The observation process is not perfect, this because when capturing an image a degradation phenomena appears, due to the capture device or the environment. In image restoration the aim is to recover an estimate of the original image from the degraded observations. The deconvolution problem can be cast in the form of a Fredholm integral equation of the first kind:

$$g(x) = \int H(x,y)f(y)dy, \tag{1}$$

in which $f(y)$ is the function of interest (image to be recovered), $g(x)$ is the function accessible to measurement (observed image), and $H(x,y)$ is the kernel of the integral equation (point spread function). The equation (1) is the imaging equation of the deconvolution (restoration) problem. We shall concentrate on the matricial version of the convolution equation (**1**), given by

$$g = Hf + \eta, \tag{2}$$

where g, f and η (noise) are MN-dimensional vectors, and H is a $MN \times MN$ matrix which represents the degradation process embedded in the image formation process.

Since the data are obtained by measurements and therefore subject to accidental errors, the direct inversion of the ill-posed problem (in the sense of Hadamard) represented by (1) magnifies the accidental errors giving unacceptable result. This fact forces to recognize that the image deconvolution problem should be treated by regularization principles or statistical estimation methods.

The most used statistical estimation methods are the maximum entropy (ME), Bayesian methods and the maximum likelihood method (MLE). The philosophy behind the statistical inversion methods is to recast the inverse problem in the form of a statistical quest for information. We have directly observable quantities and others that cannot be observed. In inverse problems, some of the unobservable quantities are of primary interest.
The statistical inversion approach is based on the following principles:

- All variables included in the model are modelled as random variables.
- The degree of information concerning these values is coded in the probability distributions.
- The solution of the inverse problem is the posterior probability distribution.

In statistical inverse problems, all parameters are viewed as random variables. We denote random variables by capital letters, for example X, Y and Z. Their realizations by lowercase letters, x, y and z are realization of the random variables X, Y and Z respectively. In the Bayesian framework, the inverse problem is expressed in the following way: given the data y of random variable Y, find the conditional probability distribution

$$p(x|y) = \frac{p(x,y)}{p(y)},$$

if $p(y) = \int_{\mathbb{R}^n} p(x,y)dx \neq 0$, is called the posterior distribution of random variable X. This distribution expresses what we know about X after the realized observation y. The nonobservable random variable X that is of primary interest is called the unknown.

7.2 Methods and comparison

The objective of the blind deconvolution problem is to estimate both f and H from g in equation (**2**). The deconvolution is a mathematical operation that is used in image restoration to recover data that has been degraded by any physical process. The direct problem can be described as a convolution equation. The blind deconvolution is an ill-posed problem in the sense Hadamard [2]. The sensitivity of the solution with respect to noise remains a serious concern.

7.2.1 Inverse filter

This may be obtained by the method of least-squares. In order to restore an image, we minimize an operator J that measures the difference between the observed image and the re-degraded estimated image

$$J(\hat{f}) = \|g - H\hat{f}\|^2 \tag{3}$$

This equation provides a least-squares problem, we seek $\hat{f}$ such that $J(\hat{f})$ is minimal

$$\begin{aligned} \|g - H\hat{f}\|^2 &= \left(g - H\hat{f}\right)^T \left(g - H\hat{f}\right) \\ &= g^T g - g^T H\hat{f} - \hat{f}H^T g \\ &+ \hat{f}H^T H\hat{f}. \end{aligned}$$

To find the minimum of J, we derive and equate to zero

$$\begin{aligned} \frac{\partial J(\hat{f})}{\partial \hat{f}} &= \frac{\partial \left[\left(g - H\hat{f}\right)^T \left(g - H\hat{f}\right)\right]}{\partial \hat{f}} \\ &= \frac{\partial(g^T g - g^T H\hat{f} - \hat{f}^T H^T g + \hat{f}^T H^T H\hat{f})}{\partial \hat{f}} \\ &= \frac{\partial(g^T g)}{\partial \hat{f}} - \frac{\partial(g^T H\hat{f})}{\partial \hat{f}} - \frac{\partial(\hat{f}^T H^T g)}{\partial \hat{f}} \\ &+ \frac{\partial(\hat{f}^T H^T H\hat{f})}{\partial \hat{f}} \\ &= -2H^T g + 2H^T H\hat{f} \end{aligned}$$

that means

$$H^T g = H^T H\hat{f}$$

In the spatial-domain (not Fourier domain), we obtain

$$\hat{f} = \left(H^T H\right)^{-1} H^T g = H^{-1} g, \tag{4}$$

when the involved inverses are defined. In the frequence domain we get

$$\hat{F}(u,v) = \frac{G(u,v)}{H(u,v)} \tag{5}$$

From equation (**2**), by using the Fourier transform, we obtain, in the frecuency domain, $G(u,v) = H(u,v)F(u,v) + N(u,v)$. In consequence,

$$\hat{F}(u,v) = F(u,v) + \frac{N(u,v)}{H(u,v)}, \tag{6}$$

for $u = 0, 1, 2, M-1$, and $v = 0, 1, 2, N-1$. The main problem of this method arises when $H(u,v)$ is close to zero or becomes zero for some (u,v) in the frecuency-domain.

As mentioned before we are dealing with an ill-posed problem. Small changes in the input data give rise to non controlled results in the solution. Regularization theory can be used to solve this problem. A technique that uses stochastic regularization is the *Wiener filter*.

7.2.2 Wiener filter

Is one of earliest and best known approaches to linear image restoration. A Wiener filter seeks an estimate for the degradation of the image that minimizes the statistical error function

$$e_i = \mathbf{f}_i - \hat{\mathbf{f}}_i \tag{7}$$

Where the elements of the error vector e_i may be positive or negative. This filter is obtained from the observed ones optimally in a minimum mean-square error (MMSE) sense for a large number of images. Thus the optimization problem is given by

$$E\left[\left(e^T e\right)\right] = E\left[Tr\left(e^T e\right)\right], \tag{8}$$

where E and Tr represent the expectation and trace operators, respectively.

If we substitute the term $\hat{\mathbf{f}}$ by **Yg**, where **Y** is the desired filter and g is the observed image for the equation **g=Hf + n**, where **n** is the noise, we have

$$\begin{aligned}
J(\mathbf{Y}) &= E\left[Tr\{(\mathbf{f}-\hat{\mathbf{f}})(\mathbf{f}-\hat{\mathbf{f}})^T\}\right] \\
&= E\left[Tr\left\{(\mathbf{f}-(\mathbf{Y}\,\mathbf{g}))(\mathbf{f}-(\mathbf{Y}\,\mathbf{g}))^T\right\}\right] \\
&= E\left[Tr\left\{(\mathbf{f}-(\mathbf{Y}\,\mathbf{(Hf+n)}))(\mathbf{f}-(\mathbf{Y}\,\mathbf{(Hf+n)}))^T\right\}\right] \\
&= E\left[Tr\left\{(\mathbf{f}-(\mathbf{Y}\,\mathbf{Hf+Yn}))(\mathbf{f}-(\mathbf{YHf+Yn}))^T\right\}\right] \\
&= E\left[Tr\left\{(\mathbf{f}-\mathbf{Y}\,\mathbf{Hf\text{-}Yn})(\mathbf{f}-\mathbf{YHf\text{-}Yn})^T\right\}\right] \\
&= E\left[Tr\left\{(\mathbf{f}-\mathbf{Y}\,\mathbf{Hf\text{-}Yn})(\mathbf{f}^T-(\mathbf{YHf})^T-(\mathbf{Yn})^T)\right\}\right] \\
&= E\left[Tr\left\{\mathbf{f}^T(\mathbf{f}-\mathbf{Y}\,\mathbf{Hf\text{-}Yn})-(\mathbf{YHf})^T(\mathbf{f}-\mathbf{Y}\,\mathbf{Hf\text{-}Yn})-(\mathbf{Yn})^T(\mathbf{f}-\mathbf{Y}\,\mathbf{Hf\text{-}Yn})\right\}\right]
\end{aligned}$$

Since the operators E and Tr are linear, can be exchanged. Observe that $Tr(A) = Tr(A^T)$ and since **f** and **n** are assumed to be independent, then $E(\mathbf{nf}^T) = E(\mathbf{fn}^T) = 0$.

By expanding $J(\mathbf{Y})$ we obtain

$$J(Y) = Tr\left(\mathbf{R_f} - 2\mathbf{YHR_f} + \mathbf{YHR_f}\mathbf{H}^T\mathbf{Y}^T + \mathbf{YR_n}\mathbf{Y}^T\right),$$

where $\mathbf{R}_f$ and $\mathbf{R}_n$ are arrays of autocorrelation, see [8].

Differentiating $J(\mathbf{Y})$ for $\mathbf{Y}$ and equating to zero we obtain the following equation

$$-2\mathbf{R_f}\mathbf{H}^T + 2\mathbf{YHR_f}\mathbf{H}^T + 2\mathbf{YR_n} = \mathbf{0}.$$

Then the matrix $\mathbf{Y}$ is given by

$$\mathbf{Y} = \mathbf{R_f}\mathbf{H}^T\left(\mathbf{HR_f}\mathbf{H}^T + \mathbf{R_n}\right)^{-1}.$$

So,

$$\begin{aligned}\hat{\mathbf{f}} &= \mathbf{Yg} \\ &= \mathbf{R_f}\mathbf{H}^T\left(\mathbf{HR_f}\mathbf{H}^T + \mathbf{R_n}\right)^{-1}\mathbf{g}.\end{aligned}$$

In the Fourier domain we have

$$\hat{F}(u,v) = \frac{H^*(u,v)}{|H^2(u,v)|^2 + \frac{R_\eta(u,v)}{R_f(u,v)}}, \tag{9}$$

where $\mathbf{R_f}(u,v)$ and $\mathbf{R_n}(u,v)$ are auto-correlation matrices and $\mathbf{H}^*(u,v)$ is the complex conjugate of $\mathbf{H}(u,v)$.

7.2.3 Lucy-Richardson Algorithm

In its first versions this method was proposed by Richardson [10] y Lucy [11] from Bayes's Theorem.

The development of this algorithm uses the *Expectation-Maximization Algorithm*.

Expectation-Maximization Algorithm.

Given θ_0, for $\nu = 1, ..., N-1$.

step one (expectation). To this step calculates the conditional expectation $Q(\theta|y;\theta_\nu)$ for the vector observed y and the approximation of Maximum Likelihood Estimator θ_ν. In the discrete case has the form

$$Q(\theta \mid \mathbf{y};\theta_\nu) = \sum_{\mathbf{x}} l(\mathbf{x},\mathbf{y})(\theta;\mathbf{x},\mathbf{y})p_{\mathbf{x}|\mathbf{y}}(\mathbf{x} \mid \mathbf{y};\theta_\nu),$$

step two (maximization). To this step calculates a maximum $\theta_{\nu+1}$ of $Q(\theta \mid \mathbf{y};\theta_\nu)$.

end

This algorithm is useful to estimate lost or hidden data on a given problem.

Lucy-Richardson Algorithm

For our problem, we consider a linear system

$$g = Hf \tag{10}$$

where the $m \times n$ coefficient matrix H and the $n \times 1$ vector g have nonnegative components. Here we minimize the *Kullback-Leibler* information divergence subject to positive constraints. The problem is hence

min

$$\rho_{HL}(\mathbf{g}, H\mathbf{f}) = \langle \mathbf{g}, \log(g/H\mathbf{f}) \rangle = \sum_{i=1}^{m} g_i(\log g_i - \log[H\mathbf{f}]_i), \tag{11}$$

subject to

$$[H\mathbf{f}]_i, g_i, h_{ij}, f_j \quad \geq \quad 0, \qquad j = 1, ..., n \tag{12}$$

$$\sum_{i=1}^{m}[H\mathbf{f}]_i, \sum_{i=1}^{m} g_i, \sum_{i=1}^{m} h_{ij}, \sum_{j=1}^{n} f_j \quad = \quad 1, \qquad i = 1, ..., m, \tag{13}$$

that is $\sum_{i=1}^{m}[H\mathbf{f}]_i = 1$, $\sum_{i=1}^{m} g_i = 1$, $\sum_{i=1}^{m} h_{ij} = 1$ y $\sum_{j=1}^{n} f_j = 1$.

Note that the problem of minimizing (11) with conditions (12) and (13) is equivalent to the following maximization problem (for properties logarithms)

max

$$J(\mathbf{f}) = \sum_{i=1}^{m} g_i \log[H\mathbf{f}]_i \tag{14}$$

subject to

$$[H\mathbf{f}]_i, g_i, h_{ij}, f_j \quad \geq \quad 0, \qquad j = 1, ..., n$$

$$\sum_{i=1}^{m}[H\mathbf{f}]_i, \sum_{i=1}^{m} g_i, \sum_{i=1}^{m} h_{ij}, \sum_{j=1}^{n} f_j \quad = \quad 1, \qquad i = 1, ..., m$$

The EM algorithm uses two discrete random variables, one that represents the incomplete observed data (partial knowledge of these) and a second random variable representing complete missing data. Also the algorithm uses a function or joint distribution, see [3]. We consider X and Y are random variables representing complete and incomplete data with indices $j = 1, ..., n$, $i = 1, ..., m$, respectively. Considering the conditions (12), (13) for $[Hf]_i$

$$P\{X = j, Y = i\} = p_{X,Y}(j, i; \mathbf{f}) = h_{ij} f_j,$$

Hence

$$\begin{aligned} p_Y(i,\mathbf{f}) &= \sum_{j=1}^{n} p_{X,Y}(j,i;\mathbf{f}) \\ &= \sum_{j=1}^{n} h_{ij} f_j \\ &= [H\mathbf{f}]_i, \end{aligned}$$

this is

$$p_Y(i;\mathbf{f}) = [H\mathbf{f}]_i.$$

Under these assumptions and following a similar reasoning to [9], we reach the following expression.

$$Q(\mathbf{f}|y;\mathbf{f}_\nu) = \sum_{i=1}^{m}\sum_{j=1}^{n} rg_i[\log h_{ij} f_j^\nu]\hat{p}_{ij}^\nu.$$

In the expectation-maximization algorithm, last equation represents the *first step*. For the next step, we use optimization on $Q(\mathbf{f}|y;\mathbf{f}_\nu)$, this is the *step two* in the expectation-maximization method.

Deriving the expression for $Q(\mathbf{f}|y;\mathbf{f}_\nu)$ and equaling to zero, we get

$$\frac{\partial}{\partial f_l} Q(\mathbf{f}|y;\mathbf{f}_\nu) = \frac{\partial}{\partial f_l} \sum_{i=1}^{m}\sum_{j=1}^{n} rg_i[\log h_{ij} + \log f_j^\nu]\hat{p}_{ij}^\nu = 0. \tag{**15**}$$

Whereas the restrictions (12), (13) for f_j; $\sum_{j=1}^{n} f_j = 1$, the equation (**15**) is written as

$$\frac{\partial}{\partial f_l}\left[Q(\mathbf{f}|y;\mathbf{f}_\nu) - \lambda\left(\sum_{j=1}^{n} f_j - 1\right)\right] = 0, \tag{**16**}$$

where $\lambda \neq 0$. In this way

$$\begin{aligned} \frac{\partial}{\partial f_l} Q(\mathbf{f}|y;\mathbf{f}_\nu) &= \frac{\partial}{\partial f_l} \sum_{i=1}^{m}\sum_{j=1}^{n} rg_i[\log h_{ij} + \log f_j^\nu]\hat{p}_{ij}^\nu \\ &= \sum_{i=1}^{m} rg_i \left[\frac{1}{f_l^\nu}\right]\hat{p}_{ij}^\nu, \end{aligned}$$

hence

$$\begin{aligned} \frac{\partial}{\partial f_l}\left[Q(\mathbf{f}|y;\mathbf{f}_\nu) - \lambda\left(\sum_{j=1}^{n} f_j - 1\right)\right] &= \left[\frac{\partial}{\partial f_l} Q(\mathbf{f}|y;\mathbf{f}_\nu) - \lambda\left(\sum_{j=1}^{n}\frac{\partial}{\partial f_l} f_j - \frac{\partial}{\partial f_l} 1\right)\right] \\ &= \sum_{i=1}^{m} rg_i\left[\frac{1}{f_l^\nu}\right]\hat{p}_{il}^\nu - \lambda \\ &= 0. \end{aligned}$$

Therefore

$$f_l = \frac{r}{\lambda}\sum_{i=1}^{m} g_i \hat{p}_{il}^{\nu}. \tag{17}$$

From equations (12) and (13) for f_j, we have

$$\begin{aligned} 1 &= \sum_{j=1}^{n} f_j \\ &= \sum_{j=1}^{n} \frac{r}{\lambda}\sum_{i=1}^{m} g_i \hat{p}_{ij}^{\nu}, \end{aligned}$$

in the second equality we use equation $(\mathbf{17})$. Rearranging

$$\begin{aligned} \sum_{j=1}^{n} \frac{r}{\lambda}\sum_{i=1}^{m} g_i \hat{p}_{ij}^{\nu} &= \frac{r}{\lambda}\sum_{j=1}^{n}\sum_{i=1}^{m} g_i \hat{p}_{ij}^{\nu} \\ &= \frac{r}{\lambda}\sum_{i=1}^{m}\sum_{j=1}^{n} g_i \hat{p}_{ij}^{\nu} \\ &= \frac{r}{\lambda}\sum_{i=1}^{m} g_i \left(\sum_{j=1}^{n} \hat{p}_{ij}^{\nu}\right), \end{aligned}$$

and using

$$p_{X|Y}(j|i;\mathbf{f}^{\nu}) = \frac{h_{ij} f_j^{\nu}}{\sum_{l=1}^{n} h_{il} f_l^{\nu}} := \hat{p}_{ij}^{\nu}, \tag{18}$$

see [9] page 58, it follows that

$$\begin{aligned} \frac{r}{\lambda}\sum_{i=1}^{m} g_i \left(\sum_{j=1}^{n} \hat{p}_{ij}^{\nu}\right) &= \frac{r}{\lambda}\sum_{i=1}^{m} g_i \left(\sum_{j=1}^{n} \frac{h_{ij} f_j^{\nu}}{\sum_{l=1}^{n} h_{il} f_l^{\nu}}\right) \\ &= \frac{r}{\lambda}\sum_{i=1}^{m} g_i \left(\frac{\sum_{j=1}^{n} h_{ij} f_j^{\nu}}{\sum_{l=1}^{n} h_{il} f_l^{\nu}}\right) \\ &= \frac{r}{\lambda}\sum_{i=1}^{m} g_i \\ &= \frac{r}{\lambda} \end{aligned}$$

by transitivity

$$1 = \frac{r}{\lambda}, \qquad \lambda \neq 0.$$

Therefore $\lambda = r$, hence $\lambda \in \mathbb{R}$ and for the equations (**17**) and (**18**)

$$\begin{aligned} f_j &= \sum_{i=1}^{m} g_i \hat{p}_{ij}^{\nu} \\ &= \sum_{i=1}^{m} g_i \frac{h_{ij} f_j^{\nu}}{\sum_{l=1}^{n} h_{il} f_l^{\nu}} \\ &= f_j^{\nu} \sum_{i=1}^{m} h_{ij} \left(\frac{g_i}{\sum_{l=1}^{n} h_{il} f_l^{\nu}} \right). \end{aligned}$$

Thus obtaining the estimate $\hat{\mathbf{f}}$ of $\mathbf{f}$ for the EM algorithm, this produce

$$f_j^{\nu+1} = f_j^{\nu} \sum_{i=1}^{m} h_{ij} \left(\frac{g_i}{\sum_{l=1}^{n} h_{il} f_l^{\nu}} \right), \qquad j = 1, ..., n. \tag{**19**}$$

to the equation (**19**) is called *the Lucy-Richardson algorithm EM*.

7.2.4 Results

In Figure 1 we show several images to compare results obtained by using Wiener filter, inverse filter and Lucy-Richardson EM methods to an degraded image, where the degradation is due only to blurring. In this case the image obtained by the inverse filter has a very good quality, and so those obtained by the Wiener filter and by Lucy-Richardson EM. When the degraded image also presents noise, the results obtained by the inverse filter are not encouraging since the noise is amplified. On the other side the results obtained with the Wiener filter and Lucy-Richardson EM are more stable under the presence of noise.

In Figure 2, it is shown a table of the comparison of errors for the methods used in this study. The first and second columns of the table display the length and angle of a lineal blurring; the third, fourth and fifth columns display the noise types: salt and pepper noise and noise of gaussian type; the sixth column displays the numerical error of the degraded image using the norm of Frobenius $e = \|f - \hat{f}\|_F$; the seventh column shows the error for the inverse filter; the eighth, the error for the Wiener filter and the last column it is shown the error for the Lucy-Richardson algorithm.

7.2.5 Conclusions

Equation (**11**) guarantees that the solution is always positive on each pixel. So, this method has this advantage over methods of linear deconvolution which may have negative values. These techniques (maximum likelihood, EM algorithm) suffers the problem of amplification on of noise when iterate many times. After many iterations the algorithm does not yield significant improvement. No criterion to stop iterations for this method. Some interesting questions to consider in a future work are related to compute an estimation of the number of iterations to obtain an acceptable quality in the image or the use of regularization methods to improve the results obtained by Lucy-Richardson. Another problem is to obtain a better Hf in (**11**) or better estimators for H from g. In this way we would get an improved algorithm.

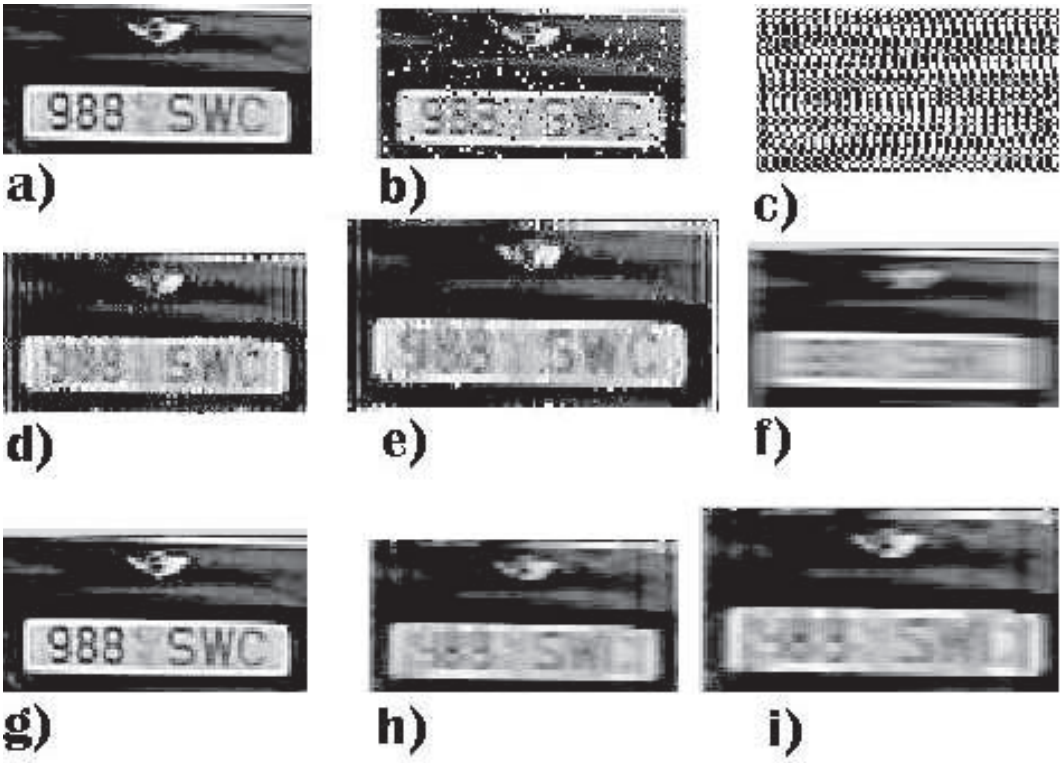

Figure 7.1: Comparison images a. original, b. degraded (angle = 0, length=7, noise: salt and pepper = 0.1), c. Inverse filter, d. Wiener filter, e. Lucy-Richardson 30 iterations, f. degraded image (only blurring angle=0,length=12), g. Inverse filter, h. Wiener filter and i. Lucy-Richardson 30 iterations.

Blurring		Noise			Error(Frobenius)			
Length	Angle	Salt and pepper	Gaussian		Degraded image	Inverse filter	Wiener filter	Lucy-Richardson
			variance	Mean				
3	0	0.1	X	X	7.572	160.606	10.633	10.115
3	45	0.1	X	X	9.539	1820.7	12.181	9.794
7	45	0.12	X	X	17.526	1.946.650	16.164	14.852
7	90	0.05	X	X	15.286	2.109xE+16	11.307	11.134
7	180	0.05	X	X	12.005	2.260xE+06	11.005	10.609
1	0	X	0.01	0	6.324	6.300	6.569	6.503
7	0	X	0.01	0	10.362	1.603xE+06	12.834	12.198
7	0	X	0..01	0.18	14.834	1.598xE+06	16.860	16.792
7	0	X	0.10	0.18	21.306	4246xE+06	29.949	25.756
1	0	0.11	X	X	12.764	12.562	6.798	NaN
1	0	0.17	X	X	16.105	15.728	7.766	NaN
4	0	0.11	X	X	15.677	5.772.930	12.168	7.964
4	0	0.17	X	X	16.513	7.439.650	16.741	9.109
5	0	0.17	X	X	16.905	285.567	17.019	9.992
6	0	0.17	X	X	17.530	7.403.730	17.715	10.32
14	0	0.17	X	X	17.995	1.350xE+17	17.972	17.021
14	45	0.17	X	X	21.889	10.554.507	153.272	154.767
14	100	0.17	X	X	21.583	7.620.123	16.113	15.763
14	100	X	X	X	16.288	2.714xE+12	12.001	11.957
3	0	X	X	X	4.283	9.122xE14	8.522	8.617
7	0	X	X	X	7.766	1.979	8.525	10.472
10	0	X	X	X	8.65	3.947xE-12	9.511	10.767
14	0	X	X	X	9.592	131.353	9.701	10.727
50	45	X	X	X	18.755	1.357xE-11	14.825	15.204
50	45	0.1	X	X	21.643	27.841.000	15.314	15.656
50	45	X	0.01	0	201.272	11757.3	17.436	17.212

Table 7.1: Comparison of methods.

Bibliography

[1] Todd K. Moon, *The Expectation-Maximization Algorithm*, IEEE Signal Processing Magazine 1996.

[2] E. Somersalo, *Computational Methods in inverse problems*, Applied Mathematical Sciences, Band 160. Springer Science+Business Media, Inc. 2005.

[3] Nir Friedman, *The Bayesian Structural EM Algorithm*, Proceedings of the Proceedings of the Fourteenth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-98). Morgan Kaufmann. San Francisco, CA. 1998. 129-139.

[4] Y. Vardi and D. Lee, *From image deblurring to optimal investments: Maximum likelihood solutions for positive linear inverse problems*, Journal of the Royal Statistical 1992.

[5] Rafael C. Gonzalez, Richard E. Woods, *Digital Image Processing*, Pearson Education, Inc. 2008.

[6] Mark R. A. K. Katsanggelos, *Digital Image Restoration*. IEEE Signal Processing Magazine 1997.

[7] Patrizio Campisi, Karen Egiazarian, *Blind Image Deconvolution*. Taylor and Francis Group, LLC 2007.

[8] E. Isaacson, H. B. Keller, *Analysis of Numerical Methods*, Wiley, 1966.

[9] Vogel, Curtis R., *Computational methods for inverse problems*. Frontiers in Applied Mathematics Series, Number 23. SIAM, 2002.

[10] Richardson, William Hadley, *Bayesian-Based Iterative Method of Image Restoration*. JOSA 62 (1): 55Ű59, 1972.

[11] Lucy, L. B., *An iterative technique for the rectification of observed distributions*. Astronomical Journal 79 (6) 745Ű754. 1974.

Chapter 8

Reconstruction of Velocity Wind Fields from Horizontal Data by Projection Methods

L. Héctor Juárez[1], María Luisa Sandoval[1],
Jorge López[2]

Abstract

For several meteorological problems and a large number of applications, the knowledge of a realistic wind field over a region is required. A successful method to generate adjusted wind fields from horizontal data is based on the mass conservation equation. This method leads to the solution of an elliptic problem for the multiplier (associated to the mass conservation constraint). However this models require a careful selection of boundary conditions and robust numerical methods to find the complete wind field. In this article we continue with a previous study in which we introduced two ways to solve the problem. Both methods can be considered as projection methods: an orthogonal L_2 projection method, and a reformulation of the problem as a saddle–point problem. For the elliptic problem, we consider a different approach to deal with the truncated boundary. For the saddle point problem, we introduce a preconditioned conjugate gradient algorithm, where the preconditioner is the elliptic problem associated to the first approach.

keywords: Mass conservation, elliptic problem, saddle-point problem, constrained minimization, preconditioned conjugate gradient.

[1]Departamentos de Matemáticas, Universidad Autónoma Metropolitana Iztapalapa, A. P. 55-534, C.P. 09340, D. F., México., hect@xanum.uam.mx, mlss@xanum.uam.mx

[2]División de Ciencias Básicas, Universidad Juárez Autónoma de Tabasco.

8.1 Introduction

In meteorology there are many problems and applications where the knowledge of a velocity vector field over a finite region is required. Some examples include dispersion of air pollutants in the atmosphere [1], [2], realization of wind maps for the design of different urban and general projects [3], among many others. Moreover, velocity fields are also required inputs for air quality models in meteorology [4]. In practice, however only limited horizontal wind field measurements are available, and the calculation of the vertical motion is of fundamental importance. Several models, methods and strategies, with various levels of complexity, have been proposed to address this problem. Diagnostic wind models require available interpolated data (generated by measurements from meteorological stations) to generate wind fields that satisfy some physical constraints. For instance, to assure mass conservation, a simplified steady state version of the continuity equation is imposed and the resulting model is then called a mass–consistent model. Mass–consistent models are attractive because of their simplicity and because they are easy and economical to operate. A review of these models is available in [5] and [6].

In this work we continue with the study, initiated in a recent article [7], of a variational mass–consistent model which is based on the original formulation by Sasaki [8]. This approach is a valuable tool for air quality applications and there have been several developments over the last decades [1], [3], [5], [6], [9], [10], [11], [12], [13]. The variational method proposed by Sasaki, uses the continuity equation in the form

$$\nabla \cdot \mathbf{u} = 0, \tag{1}$$

where $\mathbf{u}$ is the wind velocity vector field that we wan to recover on a given domain Ω from the initial data $\mathbf{u}^0$. This initial wind field $\mathbf{u}^0$ is obtained by interpolation of an initial observed wind field, after the elimination of possible outliers. The vertical component of $\mathbf{u}^0$ is taken as zero because meteorological stations usually do not measure this component. The usual approach to recover $\mathbf{u}$, [5], is based on the minimization of a functional L defined by

$$L(\mathbf{u}, \lambda) = \frac{1}{2} \int_\Omega \left\{ S\left(\mathbf{u} - \mathbf{u}^0\right) \cdot \left(\mathbf{u} - \mathbf{u}^0\right) + \lambda \nabla \cdot \mathbf{u} \right\} dV. \tag{2}$$

The new function λ is a Lagrange multiplier, and S is a diagonal matrix with weighting parameters α_i^2, $i = 1, 2, 3$, called Gaussian precision moduli. The Euler–Lagrange equations associated to the Lagrangian (**2**) are:

$$\mathbf{u} = \mathbf{u}^0 + S^{-1} \nabla \lambda, \tag{3}$$

$$\lambda \, \delta \mathbf{u} \cdot \mathbf{n} = 0. \tag{4}$$

From (**1**) and (**3**) we get the elliptic equation for the multiplier λ:

$$-\nabla \cdot \left(S^{-1} \nabla \lambda\right) = \nabla \cdot \mathbf{u}^0. \tag{5}$$

After complementing this equation with convenient boundary conditions, and solving for the Lagrange multiplier λ, the velocity field $\mathbf{u}$ is recovered using equation (**3**). There are two critical and important practical issues which have not been studied carefully by meteorologist:

- There is no general accepted criterion for choosing or estimating the values of parameters α_i in matrix S.

- There is no general consensus about the appropriate boundary conditions for the elliptic equation (**5**) to compute λ.

Actually we believe that this two issues are related and that the introduction of a matrix S, which is different to the identity matrix, is a trick or artifice to compensate the choice of inconsistent boundary conditions on the artificial boundary of the computational truncated domain.

Before we go further, we first introduce some terminology in order to formulate the problem more precisely in mathematical terms. Let Ω be an open, simply connected and bounded region in $\mathbb{R}^d$ ($d = 2$ or 3) with Lipschitz boundary $\Gamma = \partial\Omega$ decomposed as $\Gamma = \Gamma_N \cup \Gamma_D$, where Γ_N is the part of the boundary associated to the surface terrain (topography) and Γ_D is the rest of the boundary, as shown in Figure 8.1. Vector $\mathbf{n}$ is a unit outer normal vector at each point on the boundary. Given

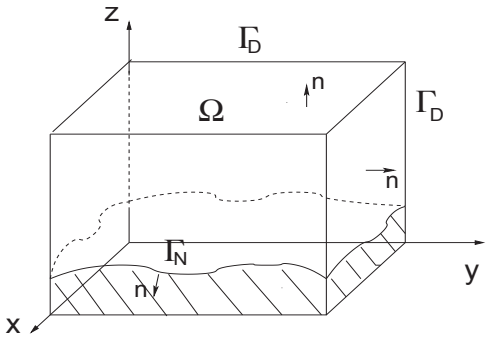

Figure 8.1: General domain Ω.

an *initial vector field* $\mathbf{u}^0$ in Ω, we want to compute a solenoidal vector function $\mathbf{u}$ –called *adjusted field*– as close to $\mathbf{u}^0$ as possible, in a least squares sense, such that $\mathbf{u} \cdot \mathbf{n} = 0$ on Γ_N. Thus the idea is to project $\mathbf{u}^0$ into the space of divergence free vector functions. To formulate the problem in mathematical terms we introduce the following vector function spaces

$$\mathbf{L}_2(\Omega) = L_2(\Omega)^d, \text{ with } d = 2 \text{ or } 3, \quad (\mathbf{6})$$

$$\mathbf{H}(div;\Omega) = \{\, \mathbf{v} \in \mathbf{L}_2(\Omega) : \nabla \cdot \mathbf{v} \in L_2(\Omega) \,\}, \quad (\mathbf{7})$$

$$\mathbf{V} = \{\, \mathbf{v} \in \mathbf{H}(div;\Omega) \;:\; \nabla \cdot \mathbf{v} = 0 \text{ and } \mathbf{v} \cdot \mathbf{n} = 0 \text{ on } \Gamma_N \,\}. \quad (\mathbf{8})$$

Space $\mathbf{V}$ is equipped with the norm $\|\cdot\|_{S,\Omega}$ associated to the inner product

$$\langle \mathbf{u}, \mathbf{v} \rangle = \int_\Omega (S\mathbf{u}) \cdot \mathbf{v} \, d\mathbf{x}, \quad (\mathbf{9})$$

where $\mathbf{v} \cdot \mathbf{w} = \sum_{i=1}^{d} v_i \, w_i$ is the usual scalar product in $\mathbb{R}^d$, and $S = S(\mathbf{x}) \in \mathbb{R}^{d\times d}$ is a symmetric positive definite matrix (usually diagonal). On $\mathbf{V}$ we introduce the following convex quadratic functional:

$$J(\mathbf{v}) = \frac{1}{2} \parallel \mathbf{v} - \mathbf{u}^0 \parallel^2_{S,\Omega} = \frac{1}{2} \int_\Omega \left(S\left(\mathbf{v} - \mathbf{u}^0\right)\right) \cdot (\mathbf{v} - \mathbf{u}^0) \, d\mathbf{x}. \quad (\mathbf{10})$$

Therefore, the problem to generate the adjusted wind field $\mathbf{u}$ from a given initial field $\mathbf{u}^0$ can be stated as follows:

$$\text{Given } \mathbf{u}^0 \in \mathbf{H}(div;\Omega),\ \text{find } \mathbf{u} \in \mathbf{V} \text{ such that } J(\mathbf{u}) \leq J(\mathbf{v}),\ \forall\, \mathbf{v} \in \mathbf{V}. \tag{11}$$

A necessary and sufficient condition for J to have a minimizer $\mathbf{u} \in \mathbf{V}$ is that

$$\int_\Omega \left(S\left(\mathbf{u}-\mathbf{u}^0\right)\right)\cdot \mathbf{v}\, d\mathbf{x} = 0,\ \forall\, \mathbf{v} \in \mathbf{V}. \tag{12}$$

Furthermore, the minimizer $\mathbf{u}$ is unique and it is given by equation (**3**), where $\lambda \in H^1(\Omega)$ is the solution of the elliptic equation (**5**) (see [7] for the details).

Concerning the boundary conditions for the multiplier λ in the elliptic problem (**5**), the most common choices (among the meteorological community) are (**a**) $\lambda = 0$ on *open* or "*flow through*" *boundaries*, and (**b**) $\partial\lambda/\partial\mathbf{n} = 0$ on *closed* or "*no flow through*" *boundaries* [5]. However these boundary conditions are not physically nor mathematically justified, also they produce solutions which are poorly adjusted near the boundaries, and therefore they degrade the solutions, sometimes by several orders of magnitude. This is demonstrated in [7], where the solution obtained by solving (**5**), with different boundary conditions, is compared with the *saddle–point formulation* of the problem where the multiplier λ is not forced to take boundary values on the boundary. The saddle–point formulation is obtained from the Lagrangian (see equation **2**)

$$L(\mathbf{v},q) = J(\mathbf{v}) + \langle q, \nabla\cdot\mathbf{v}\rangle = \frac{1}{2}\int_\Omega \left(S(\mathbf{v}-\mathbf{u}^0)\right)\cdot(\mathbf{v}-\mathbf{u}^0)\, d\mathbf{x} + \int_\Omega q\nabla\cdot\mathbf{v}\, d\mathbf{x}, \tag{13}$$

defined on $\mathbf{V}_N \times L_2(\Omega)$, where $\mathbf{V}_N$ is the space of vector functions defined by

$$\mathbf{V}_N = \left\{\, \mathbf{v} \in H(div;\Omega)\ :\ \mathbf{v}\cdot\mathbf{n} = 0\ \text{ on }\ \Gamma_N \,\right\}. \tag{14}$$

A point $(\mathbf{u},\ \lambda) \in \mathbf{V}_N \times L_2(\Omega)$ is a stationary point of the Lagrangian (**13**) if and only if it is solution of the saddle–point problem:

$$\int_\Omega S\mathbf{u}\cdot\mathbf{v}\, d\mathbf{x} + \int_\Omega \lambda\nabla\cdot\mathbf{v}\, d\mathbf{x} = \int_\Omega S\mathbf{u}^0\cdot\mathbf{v}\, d\mathbf{x}, \quad \forall\, \mathbf{v} \in \mathbf{V}_N, \tag{15}$$

$$\int_\Omega q\nabla\cdot\mathbf{u}\, d\mathbf{x} = 0, \quad \forall\, q \in L_2(\Omega), \tag{16}$$

where λ **need not to satisfy boundary conditions**. The solution $\mathbf{u}$ is the minimizer of J, but now it has to be obtained from the enlarged space $\mathbf{V}_N$ where divergence free is not satisfied. Instead, condition (**1**) is relaxed by introducing a Lagrange multiplier λ, so that $\mathbf{u}$ must satisfy the weaker condition (**16**). This saddle–point problem is solved by an iterative *conjugate gradient algorithm* introduced in [7]. This algorithm does not degrade the solution near artificial boundaries, (vertical and top boundaries), and produce accurate numerical solutions which satisfy the constraint (**1**) almost to the desired tolerance, in a weak sense. However, to obtain a good accuracy about one thousand iterations are required.

In this work we mainly discuss another way to deal with the boundary conditions for the elliptic problem on truncated boundaries. On the other hand, we introduce a preconditioned conjugate gradient algorithm in order to accelerate convergence of the iterative method for the saddle–point problem. The organization of this article is as follows: in Section 2 we consider the elliptic problem (**5**), and we propose a new way to deal with the boundary conditions on the truncated boundary. In Section 3 we demonstrate that the elliptic problem (**5**) is actually an optimal preconditioner of the conjugate gradient algorithm introduced in [7]. Then, we present a preconditioned conjugate gradient algorithm with a mixed finite element discretization. We still working on the computer implementation of this preconditioned algorithm. However, we can anticipate that the number of iterations should be reduced from one thousand to about twenty, as it occurs when we solve problems in fluid mechanics. Finally, in Section 4 we establish some concluding remarks.

8.2 Elliptic problem: a different approach

8.2.1 Formulation of the elliptic problem

Given that $\mathbf{u}$ is of the form (**3**) and it satisfies $\nabla \cdot \mathbf{u} = 0$ and $\mathbf{u} \cdot \mathbf{n} = 0$ on Γ_N, then the Lagrange multiplier λ satisfies the following equations

$$-\nabla \cdot (S^{-1}\nabla\lambda) = \nabla \cdot \mathbf{u}^0 \text{ in } \Omega, \tag{17}$$

$$-(S^{-1}\nabla\lambda) \cdot \mathbf{n} = \mathbf{u}^0 \cdot \mathbf{n} \text{ on } \Gamma_N. \tag{18}$$

Observe that the boundary condition (**18**) is imposed only on the physical boundary Γ_N. We do not impose an explicit boundary condition on the artificial truncated boundary Γ_D, since the bounded domain Ω is arbitrary and it could be smaller or larger, depending of available information such as $\mathbf{u}^0$. Instead, we enforce that the flux across the entire boundary be zero in order to guaranty mass conservation, i.e.

$$0 = \int_\Gamma \mathbf{u} \cdot \mathbf{n}\, d\Gamma = \int_\Gamma (\mathbf{u}^0 + S^{-1}\nabla\lambda) \cdot \mathbf{n}\, d\Gamma\,.$$

Now, this condition and (**18**) imply

$$\int_{\Gamma_D} (\mathbf{u}^0 + S^{-1}\nabla\lambda) \cdot \mathbf{n}\, d\Gamma = 0. \tag{19}$$

Then, the physical condition (**19**) must be enforced in some way when computing the solution of the elliptic problem. Equations (**17**)–(**19**) imply the identity

$$\int_\Omega \nabla \cdot \mathbf{u}^0\, d\mathbf{x} - \int_\Gamma \mathbf{u}^0 \cdot \mathbf{n}\, d\Gamma = 0.$$

Actually this is also the compatibility condition associated to the Poisson–Neumann–like problem (**17**)–(**19**), Therefore, the problem has a unique solution λ, up to a constant, in $H^1(\Omega)$. If we define

$$\Lambda = \left\{ q \in H^1(\Omega) \,\middle|\, \int_\Omega q\, dx = 0 \right\}, \tag{20}$$

then, the variational formulation of the problem is: Given $\mathbf{u}^0 \in \mathbf{L}_2(\Omega)$, find $\lambda \in \Lambda$ such that

$$\int_\Omega (S^{-1}\nabla\lambda)\cdot\nabla q\, d\mathbf{x} = -\int_\Omega \mathbf{u}^0\cdot\nabla q\, d\mathbf{x} + \int_{\Gamma_D} q\,(\mathbf{u}^0 + S^{-1}\nabla\lambda)\cdot\mathbf{n}\, d\Gamma\,, \quad \forall\, q\in\Lambda. \qquad (\mathbf{21})$$

Observe that if q were a constant function in Ω, we would recover (**19**).

8.2.2 Finite element approximation

We solve problem (**21**) by the finite element method. Let $\mathcal{T}_h$ be a finite element triangulation of $\overline{\Omega}\subset\mathbb{R}^2$, where h is the space discretization step [14]. Denote by P_1 the space of polynomials of degree less or equal than 1. Then, spaces $\mathbf{L}_2(\Omega)$ and Λ, are approximated by the following finite dimensional spaces

$$\mathbf{L}_h = \left\{\, \mathbf{v}_h\in\mathcal{C}^0(\Omega)\times\mathcal{C}^0(\Omega)\ :\ \mathbf{v}_h|_T\in P_1\times P_1,\ \forall\, T\in\mathcal{T}_h \,\right\}, \qquad (\mathbf{22})$$

$$\Lambda_h = \left\{\, q\in\mathcal{C}^0(\Omega)\ :\ q|_T\in P_1 \text{ and } \int_\Omega q\, dx = 0,\ \forall\, T\in\mathcal{T}_h \right\}, \qquad (\mathbf{23})$$

respectively. Thus, the discrete version of (**21**) is: Given $\mathbf{u}_h^0\in\mathbf{L}_h$, find $\lambda_h\in\Lambda_h$ such that

$$\int_\Omega (S^{-1}\nabla\lambda_h)\cdot\nabla q\, d\mathbf{x} = -\int_\Omega \mathbf{u}_h^0\cdot\nabla q\, d\mathbf{x} + \int_{\Gamma_D} q\,(\mathbf{u}_h^0 + S^{-1}\nabla\lambda_h)\cdot\mathbf{n}\, d\Gamma\,, \quad \forall\, q\in\Lambda_h, \qquad (\mathbf{24})$$

where $\mathbf{u}_h^0\in\mathbf{L}_h$ is the interpolant of the initial velocity field, $\mathbf{u}^0$. Observe that the above discrete variational problem produces a nonsymmetric algebraic problem, because the boundary integral on the right hand side has the unknown λ_h. We do not want to deal with this nonsymmetric problem, since this would require about twice the memory than the one required for the symmetric case. So, in order to keep the problem symmetric we consider two choices:

1. **Ghost Nodes**. The finite dimensional space Λ_h is generated from the set of piecewise linear "hat" functions $\phi_i = \phi_i(\mathbf{x})$ (up to a contant) defined by

   $$\phi_i(\mathbf{x}_j) = \delta_{ij},$$

   where $\{\mathbf{x}_i\}_{i=1}^N$ is the set of N nodes associated to the triangulation $\mathcal{T}_h$ of Ω. If $\mathbf{x}_i$ is an interior vertex, then the boundary integral in (**24**) with $q=\phi_i$ is zero, and

   $$\int_\Omega (S^{-1}\nabla\lambda_h)\cdot\nabla\phi_i\, d\mathbf{x} = -\int_\Omega \mathbf{u}_h^0\cdot\nabla\phi_i\, d\mathbf{x}. \qquad (\mathbf{25})$$

   So, if we introduce a layer of ghost nodes around and beyond Γ_D, and we define $\lambda_h = 0$ or $\partial\lambda_h/\partial\mathbf{n} = 0$ on those ghost nodes, then we only have to solve the symmetric algebraic system associated to (**25**). At the end we discard the solution on the ghost nodes and we only keep the solution values on the actual nodes. Actually this is a well known way to deal with PDE in domains with truncated boundaries.

2. **Iteration**. The solution obtained by the introduction of ghost nodes may be improved by the following iteration: We denote the initial solution for λ_h^0, and then for each $k \geq 1$, we get λ_h^k solving the following problem

$$\int_\Omega (S^{-1}\nabla\lambda_h^k)\cdot\nabla q\; d\mathbf{x} = -\int_\Omega \mathbf{u}_h^0\cdot\nabla q\; d\mathbf{x} + \int_{\Gamma_D} q\,(\mathbf{u}_h^0 + S^{-1}\nabla\lambda_h^{k-1})\cdot\mathbf{n}\; d\Gamma\,, \quad \forall\, q\in\Lambda_h. \tag{26}$$

We hope that about two iterations are sufficient. Once λ_h is computed, the numerical approximation $\mathbf{u}_h$ to the adjusted field $\mathbf{u}$ is calculated using the weak version of (**3**).

8.2.3 Numerical example

We consider the two dimensional solenoidal vector field $\mathbf{u}(x,\,y) = (x,\,-y)$ defined in $\Omega = (1,\,2)\times(0,\,1)$, which satisfies the conditions $\nabla\cdot\mathbf{u} = 0$ in Ω and $\mathbf{u}\cdot\mathbf{n} = 0$ on Γ_N. Assuming that we have the horizontal component $\mathbf{u}^0(x,\,z) = (x,\,0)$ as the initial wind field, we want to see how much we can recover of the vertical component of $\mathbf{u}$. For this calculation we divide Ω into a triangular mesh of size 80×80, and we choose the values $\alpha_1 = 1$ and $\alpha_3 = 0.001$ for the Gaussian precision modula. We apply the previous algorithm, incorporating two layers of **ghost nodes**, and without doing iterations. Figure 8.2 shows the exact and computed velocity fields, which are indistinguishable at a first look. To measure the global difference between the exact field $\mathbf{u}$ and the computed adjusted field $\mathbf{u}_h$ we use the following formula for the relative error

$$e_r = \frac{||\mathbf{u}-\mathbf{u}_h||_2}{||\mathbf{u}||_2}, \tag{27}$$

For the present example we obtain $e_r = 2.1\times 10^{-5}$. We also computed a mean value of the divergence of $\mathbf{u}_h$, defined as

$$mdiv = \underset{\mathbf{x}_i}{\text{mean}}\{\ \nabla\cdot\mathbf{u}_h(\mathbf{x}_i)\mid \mathbf{x}_i \text{ is a interior vertex of the computational mesh}\} \tag{28}$$

where the point–wise divergence is computed in a weak sense. More precisely, if ϕ_i is the piece–wise linear base function associated to node $\mathbf{x}_i$, the weak divergence at the interior node $\mathbf{x}_i$ is defined as

$$\nabla\cdot\mathbf{u}_h(\mathbf{x}_i) = -\int_\Omega \mathbf{u}_h\cdot\nabla\phi_i\; d\mathbf{x}\,. \tag{29}$$

The weak divergence of the computed solution for the present example is $mdiv = 1.6\times 10^{-6}$.

We compare this solution with the one obtained in reference [7], where two different sets of boundary conditions on Γ_D were considered. For reference, from now on we denote by ***BCE*** the boundary conditions considered in this work. Thus the different sets of boundary conditions for comparison are:

BC1: $-S^{-1}\nabla\lambda\cdot\mathbf{n} = \mathbf{u}^0\cdot\mathbf{n}$ on Γ_N, $\qquad\qquad \lambda = 0$ on Γ_D,

BC2: $-S^{-1}\nabla\lambda\cdot\mathbf{n} = \mathbf{u}^0\cdot\mathbf{n}$ on Γ_N, $\qquad -S^{-1}\nabla\lambda\cdot\mathbf{n} = 0$ on Γ_V, $\qquad \lambda = 0$ on Γ_T,

BCE: $-S^{-1}\nabla\lambda\cdot\mathbf{n} = \mathbf{u}^0\cdot\mathbf{n}$ on Γ_N, $\qquad$ with two layers of ghost nodes on Γ_D,

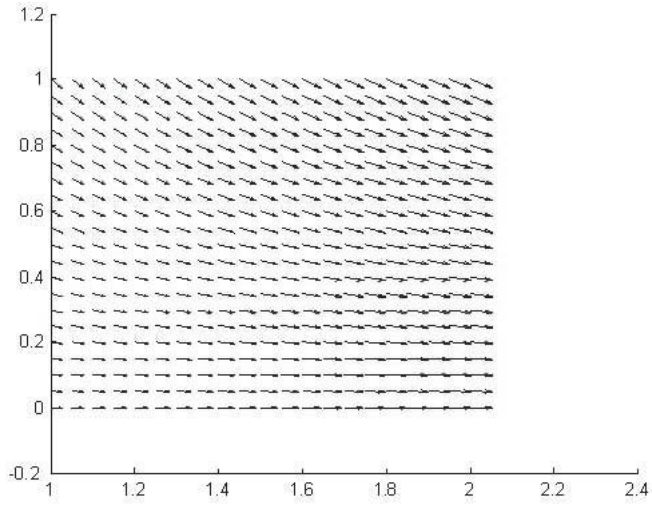

Figure 8.2: Exact field $\mathbf{u} = (x, -z)$ in red; adjusted field in blue.

where Γ_D was divided in two parts: the vertical artificial boundary Γ_V, and the top artificial boundary Γ_T. Table 8.1 shows how boundary conditions degrade numerical calculations. It is observed that the solution improves each time the Dirichlet boundary condition $\lambda = 0$ is applied to a smaller section of the non-physical boundary. This is not surprising, since this boundary condition introduces a large artificial gradient when calculating the term $\nabla\lambda$ in order to get $\mathbf{u} = \mathbf{u}^0 + S^{-1}\nabla\lambda$ at the corresponding boundary nodes.

Case	e_r	$mdiv$
BC1	1.9×10^{-2}	4.1×10^{-2}
BC2	4.0×10^{-4}	1.8×10^{-2}
BCE	2.1×10^{-5}	1.6×10^{-6}

Table 8.1: Comparison of numerical solutions obtained with different sets of boundary conditions.

8.3 Preconditioned conjugate gradient algorithm

8.3.1 An operator for the Lagrange multiplier

There are some effective numerical techniques to solve saddle–point problems like the problem (**15**)–(**16**). Here, we shall adapt a numerical technique, introduced by Glowinski [15], to solve generalized Stokes problems. This approach has proven to provide accurate solutions efficiently

when it is applied to problems in fluid mechanics. We proceed as follows: Assume that $(\mathbf{u}, \lambda)$ is solution of problem (**15**)–(**16**) with

$$\mathbf{u} = \mathbf{u}^0 + \mathbf{u}_\lambda, \tag{30}$$

where the initial velocity is tangent to Γ_N. Then $\mathbf{u}_\lambda \in \mathbf{V}_N$ solves

$$\int_\Omega (S\,\mathbf{u}_\lambda)\cdot\mathbf{v}\,d\mathbf{x} = -\int_\Omega \lambda \nabla\cdot\mathbf{v}\,d\mathbf{x} \quad \forall\,\mathbf{v}\in\mathbf{V}_N. \tag{31}$$

Since $\nabla\cdot\mathbf{u}=0$, then from (**30**) it follows that $-\nabla\cdot\mathbf{u}_\lambda = \nabla\cdot\mathbf{u}^0$, and this equation can be expressed in the functional form

$$A\,\lambda = \nabla\cdot\mathbf{u}^0, \tag{32}$$

where $A : L_2(\Omega)\to L_2(\Omega)$ is an operator defined by

$$A\,q = -\nabla\cdot\mathbf{u}_q, \tag{33}$$

with $\mathbf{u}_q\in\mathbf{V}_N$ the solution of

$$\int_\Omega (S\,\mathbf{u}_q)\cdot\mathbf{v}\,d\mathbf{x} = -\int_\Omega q\,\nabla\cdot\mathbf{v}\,d\mathbf{x} \quad \forall\,\mathbf{v}\in\mathbf{V}_N. \tag{34}$$

Operator A is linear, and taking $\mathbf{v}=\mathbf{u}_{q'}$, then (**33**) and (**34**) imply

$$\int_\Omega (A\,q')\,q\,d\mathbf{x} = -\int_\Omega q\,\nabla\cdot\mathbf{u}_{q'}\,d\mathbf{x} = \int_\Omega (S\,\mathbf{u}_q)\cdot\mathbf{u}_{q'}\,d\mathbf{x}.$$

The above properties imply that the operator A is self-adjoint and strongly elliptic. Therefore, equation (**32**) can be solved by the conjugate gradient algorithm:

1. **Initialization**: λ^0 given, $\quad g^0 = A\lambda^0 - \nabla\cdot\mathbf{u}^0, \quad d^0 = -g^0$.
2. **Descent**: For $k\geq 0$, assuming we know λ^k, g^k, d^k, find λ^{k+1}, g^{k+1}, d^{k+1} by
$$\lambda^{k+1} = \lambda^k + \alpha_k\,d^k \quad \text{where} \quad \alpha_k = \langle g^k,\,g^k\rangle/\langle d^k,\,A\,d^k\rangle,$$
$$g^{k+1} = g^k + \alpha_k\,A\,d^k.$$
3. **Test of convergence and new conjugate direction**:
$$\text{If} \quad \langle g^k,\,g^k\rangle \leq \epsilon\langle g^0,\,g^0\rangle, \quad \text{take} \quad \lambda = \lambda^{k+1} \quad \text{and stop.}$$
If not, compute
$$d^{k+1} = -g^{k+1} + \beta_k\,d^k \quad \text{with} \quad \beta_k = \frac{\langle g^{k+1},\,g^{k+1}\rangle}{\langle g^k,\,g^k\rangle}.$$
Do $k=k+1$ and return to 2.

The nice properties of this iterative algorithm are discussed in [7], and it is shown how it produces excellent computational results when using a mixed finite element method to compute $A\,d^k$ at each iteration. Actually most of the computational cost is due to this calculation which involves the solution of the integral equation (**34**) with $q=d^k$. For the two dimensional problem, considered in the previous section, the obtained solution satisfies $e_r = 5.9\times 10^{-4}$ and $mdiv = -5.3\times 10^{-12}$, after 1214 iterations.

8.3.2 Preconditioned conjugate gradient

The conjugate gradient algorithm is very effective in reducing the average divergence of the wind velocity field. However the number of iterations required for convergence is more than one thousand. So, in order to obtain a more efficient iterative algorithm we need to reduce the number of iterations, and one way to do that is introducing a good preconditioner.

Let $B : L_2(\Omega) \to L_2(\Omega)$ be an operator defined by

$$Bq = \phi_q, \tag{35}$$

where ϕ_q solves the problem

$$\int_\Omega \left(S^{-1}\nabla\phi_q\right)\cdot\nabla\psi\, d\mathbf{x} = \int_\Omega q\,\psi\, d\mathbf{x} \qquad \forall\,\psi \in H^1(\Omega). \tag{36}$$

The operator B is self-adjoint, positive definite, and satisfies $ABq = q$ for every $q \in L^2(\Omega)$. Therefore, we can take B^{-1} as a *preconditioner* for A. Then, the preconditioned conjugate gradient algorithm to solve problem (**15**)–(**16**) is as follows:

1. **Initialization**: λ^0 given, $\quad g^0 = A\,\lambda^0 - \nabla\cdot\mathbf{u}^0, \quad \widehat{g}^0 = B\,g^0, \quad d^0 = -\widehat{g}^0.$
2. **Descent**: For $k \geq 0$, assuming we know λ^k, g^k, $\widehat{g}^k$, d^k, find λ^{k+1}, g^{k+1}, $\widehat{g}^{k+1}$, d^{k+1} by
$$\begin{aligned}
&\lambda^{k+1} = \lambda^k + \alpha_k\, d^k \quad \text{where} \quad \alpha_k = \langle g^k,\, g^k\rangle / \langle d^k,\, A\,d^k\rangle,\\
&g^{k+1} = g^k + \alpha_k\, A\, d^k,\\
&\widehat{g}^{k+1} = \widehat{g}^k + \alpha_k\, B(A\,d^k).
\end{aligned}$$
3. **Test of convergence and new conjugate direction**:

   $$\text{If} \quad \langle g^k,\, \widehat{g}^k\rangle \leq \epsilon\langle g^0,\, \widehat{g}^0\rangle, \quad \text{take} \quad \lambda = \lambda^{k+1} \quad \text{and stop.}$$

   If not, compute

   $$d^{k+1} = -\widehat{g}^{k+1} + \beta_k\, d^k \quad \text{with} \quad \beta_k = \frac{\langle g^{k+1},\, \widehat{g}^{k+1}\rangle}{\langle g^k,\, \widehat{g}^k\rangle}.$$

   Do $k = k + 1$ and return to 2.

The previous algorithm can be expressed in more detail when we take into account equations (**33**)–(**34**), for the definition of operator A, and equations (**35**)–(**36**) for the definition of operator B. Then, the detailed conjugate gradient algorithm with preconditioning is as follows:

Initialization

1. Given $\lambda^0 \in L_2(\Omega)$, solve for $\mathbf{u}_\lambda^0 \in \mathbf{V}_N$

   $$\int_\Omega (S\,\mathbf{u}_\lambda^0)\cdot\mathbf{v}\, d\mathbf{x} = -\int_\Omega \lambda^0\nabla\cdot\mathbf{v}\, d\mathbf{x},\ \forall\,\mathbf{v} \in \mathbf{V}_N.$$

2. Let $g^0 = \nabla \cdot \left(\mathbf{u}_\lambda^0 + \mathbf{u}^0\right)$.
3. Solve for $\phi^0 \in H^1(\Omega)/\mathbb{R}$

$$\int_\Omega (S^{-1}\nabla\phi^0) \cdot \nabla\psi \, d\mathbf{x} = \int_\Omega g^0\psi \, d\mathbf{x}, \; \forall\, \psi \in H^1(\Omega).$$

4. Let $\hat{g}^0 = \phi^0$, $d^0 = \hat{g}^0$.

Descent

For $m \geq 0$, assuming λ^m, g^m, d^m, $\mathbf{u}^m$ are known, compute λ^{m+1}, g^{m+1}, d^{m+1} and $\mathbf{u}^{m+1}$, using the following steps:

5. Solve for $\bar{\mathbf{u}}^m \in \mathbf{V}_N$

$$\int_\Omega (S\,\bar{\mathbf{u}}^m) \cdot \mathbf{v} \, d\mathbf{x} = -\int_\Omega d^m \nabla \cdot \mathbf{v}^m \, d\mathbf{x}, \; \forall\, \mathbf{v} \in \mathbf{V}_N.$$

6. Let $\bar{g}^m = \nabla \cdot \bar{\mathbf{u}}^m$.
7. Solve for $\phi^m \in H^1(\Omega)/\mathbb{R}$

$$\int_\Omega (S^{-1}\nabla\phi^m) \cdot \nabla\psi \, d\mathbf{x} = \int_\Omega \bar{g}^m\psi \, d\mathbf{x}, \; \forall\, \psi \in H^1(\Omega).$$

8. Let $\alpha_m = \int_\Omega g^m \hat{g}^m d\mathbf{x} \,/\, \int_\Omega \bar{g}^m d^m d\mathbf{x}$.
9. Set

$$\begin{aligned}
\lambda^{m+1} &= \lambda^m - \alpha_m \, d^m, \\
\mathbf{u}^{m+1} &= \mathbf{u}^m - \alpha_m \, \bar{\mathbf{u}}^m, \\
g^{m+1} &= g^m - \alpha_m \, \bar{g}^m, \\
\hat{g}^{m+1} &= \hat{g}^m - \alpha_m \, \phi^m.
\end{aligned}$$

Test of convergence and new descent direction

If $\int_\Omega g^{m+1}\hat{g}^{m+1} \, d\mathbf{x} \,/\, \int_\Omega g^0\hat{g}^0 \, d\mathbf{x} < \varepsilon$, then do $\lambda = \lambda^{m+1}$, $\mathbf{u} = \mathbf{u}^{m+1}$ and stop. Otherwise, do the following

10. Compute $\beta_m = \int_\Omega g^{m+1}\hat{g}^{m+1} d\mathbf{x} \,/\, \int_\Omega g^m \hat{g}^m \, d\mathbf{x}$.
11. Set $d^{m+1} = \hat{g}^{m+1} + \beta_m \, d^m$.
12. Do $m = m + 1$ and return to 5.

Note that in this algorithm, $\mathbf{u}$ and λ are *computed simultaneously*. Additional steps in this algorithm with respect the conjugate gradient algorithm without preconditioning are mainly steps 3 and 7. Then, the additional cost of this algorithm, at each iteration, is the solution of the elliptic problem in step 7. However, this additional cost is offset by two nice properties: a) the preconditioning reduces dramatically the number of iterations (in CFD, the average number of iterations is between 10 and 20); b) there is a significant reduction of degrees of freedom in the discrete version of the elliptic problem in steps 3 and 7. We shall clarify this last point after having discretized the algorithm by the following mixed finite element method.

8.3.3 Discretization by a mixed finite element method

To approximate the functions belonging to the spaces $\mathbf{V}_N$ and $L_2(\Omega)$, we make use of the *Bercovier–Pironneau* finite element approximation [16]. This a stable mixed method where the vector functions on $\mathbf{V}_N$, such as $\mathbf{u}_\lambda^0$, $\mathbf{u}^m$ and $\bar{\mathbf{u}}^m$, are approximated by continuous piecewise linear polynomials on a fine triangulation $\mathcal{T}_h$ of Ω. On the other hand, scalar functions on $L_2(\Omega)$, such as λ^m, g^m, $\bar{g}^m$, $\hat{g}^m$, d^m, are also approximated with piecewise linear polynomials, but this time on a triangulation twice as coarse, $\mathcal{T}_{2h}$ of Ω. The fine triangulation $\mathcal{T}_h$ is obtained from the coarse triangulation through a regular subdivision of each triangle $T \in \mathcal{T}_{2h}$, as shown in Figure 8.3. Then, the function spaces $\mathbf{V}_N$

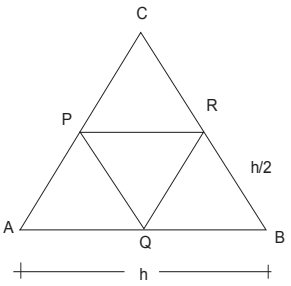

Figure 8.3: Element in $\mathcal{T}_{2h}$: triangle ABC. Elements in $\mathcal{T}_h$: triangles AQP, PRC, PQR and QBR

and $L_2(\Omega)$ are approximated by the following finite dimensional subspaces

$$\mathbf{V}_{Nh} = \left\{ \mathbf{v}_h \in C^0(\bar{\Omega})^2 \ : \ \mathbf{v}_h|_T \in P_1 \times P_1, \, \forall T \in \mathcal{T}_h, \, \mathbf{v}_h \cdot \mathbf{n} = 0 \text{ on } \Gamma_N \right\}, \qquad \textbf{(37)}$$

and

$$L_{2h} = \left\{ q_h \in C^0(\bar{\Omega}) \ : \ q_h|_T \in P_1, \, \forall T \in \mathcal{T}_{2h} \right\}, \qquad \textbf{(38)}$$

respectively. We apply the mixed method described above, particularly in steps 1 and 5, as well as in the weak version of the steps 2 and 6 of the *PCG–Algorithm*.

Concerning the elliptic problems in steps 3 and 7, they are approximated over the coarse triangulation $\mathcal{T}_{2h}$. Scalar functions on $H^1(\Omega)$, such as ϕ^0 and ϕ^m are approximated by continuous piecewise linear polynomials on each of the triangles of $\mathcal{T}_{2h}$. Then, $H^1(\Omega)$ is approximated by means of the finite dimensional space

$$H_{2h}^1 = \left\{ q_h \in C^0(\bar{\Omega}) \ : \ q_h|_T \in P_1, \, \forall T \in \mathcal{T}_{2h} \right\}. \qquad \textbf{(39)}$$

Finally, scalar functions, such as g^0, $\bar{g}^m$ are approximated by functions of L_{2h} defined in (**38**), as we have mentioned before.

Note that since $\mathbf{u}$ is obtained on the fine mesh, its resolution is the same as that obtained with the traditional algorithm. Also, if the trapezoidal rule is applied to calculate the integrals on the left hand side in steps 1 and 5, we obtain a system of algebraic equations with diagonal matrix, and the cost to solve them is only a vector multiplication. Then, the additional cost of the *PCG–Algorithm*

compared to the cost of the *CG–Algorihm* is the solution of these elliptic problems in steps 3 and 7, but these problems are solved in a mesh twice as coarse. So, in a two-dimensional problem, the number of degrees of freedom (number of unknowns) in the resulting algebraic problem is about four times less than the number of degrees of freedom obtained when solving the elliptic problem with the traditional method described in Section 2. In a three-dimensional problem the number of degrees of freedom is about eight times less. According to this, the *PCG–Agorithm* algorithm saves memory on the matrix storage, compared with the *EE–algorithm* algorithm. This matrix can be pre-calculated before starting to iterate because it remains constant throughout the calculation process. Regarding the numerical calculations with this algorithm, work is under development and we hope to report numerical results soon.

8.4 Concluding remarks

Table 1 shows that boundary conditions can significantly affect numerical solutions, and that they may degrade the solution to a greater or lesser degree, depending of how we treat artificial truncated boundaries. We have proposed two additional ways of dealing with this problem: "ghost nodes", and an iterative method, which we should explore more deeply. We think that the choice of appropriate boundary conditions is more crucial than the choice of parameters in the matrix S to obtain good solutions. A bad choice of boundary conditions on non physical boundaries, such as the Dirichlet boundary condition $\lambda = 0$, produces poor results due to the introduction of spurious high gradients by the term $S^{-1}\nabla\lambda$ in formula (**3**), particularly on each node near the corresponding part of the boundary. The best result we can expect with this type of boundary conditions is to obtain a numerical solution with a weak divergence of the order of 10^{-2} and an overall relative accuracy of the same order.

Concerning the second approach, it is based on the iterative conjugate gradient algorithm applied to the functional equation obtained from the saddle point problem. In previous work we have shown that this method gives very good results. However, the number of iterations for convergence, in some problems, is typically on the order of several hundred, and sometimes around a thousand. This slow convergence motivated us to find a good preconditioner to accelerate convergence of the iterative method. It turned out that the preconditioner is an elliptic operator, which involves solving a Poisson problem. This preconditioner was derived following the idea of Cahouet and Chabard [17]. The extra work introduced by the preconditioner in the conjugate gradient method, is mainly the solution of elliptic problem in each iteration. However, this elliptic problem is solved in a coarse mesh, and it is four times smaller than the elliptic problem for the multiplier λ. Again, we need to consider appropriate boundary conditions to solve these elliptic problems. In short, we want to combine the preconditioned conjugate gradient with an efficient elliptic solver, but without degrading the numerical solutions. Clearly, the boundary conditions for the elliptic operator are important again, and we continue working on this problem.

Bibliography

[1] Sherman, C. A., *A mass–consistent model for wind fields over complex terrain*, J. Appl. Meteor. **17**, 312–319 (1978).

[2] Finardi, S., Tinarelli, G., Nanni, A., Brusasca, G. and Carboni, G., *Evaluation of a 3–D flow and pollutant dispersion modelling system to estimate climatological ground level concentrations in complex coastal sites*, International Journal of Environment and Pollution **16**, 472–482 (2001).

[3] Castino, F., Rusca, L. and Solari, G., *Wind climate micro–zoning: a pilot application to Liguria Region (North–Western Italy)*, Journal of Wind Engineering and Industrial Aerodynamics **91**, 1353–1375 (2003).

[4] R. Daley, *Atmospheric data analysis*, (Cambridge University Press, New York, (1991).

[5] Ratto, C. F., Festa, R., Romeo, C., Frumento, O. A. and Galluzzi, M., *Mass–consistent models for wind fields over complex terrain: The state of the art*, Environ. Software **9**, 247–268 (1994).

[6] Ratto, C. F., *An overview of mass-consistent models*. Modeling of Atmosphere Flow Fields, D. P. Lalas and C. F. Ratto, Eds, World Scientific Publications, 379-400 (1996).

[7] Flores C. F., Juárez, L. H., Nuñez, M. A. and Sandoval, M. L., *Algorithms for vector field generation in mass consistent models*. Journal of Numerical Methods for Partial Differential Equations, 26-4, pp. 826–842, (2009).

[8] Sasaki, Y., *An objective analysis based on the variational method*, Journal Met. Soc. Japan, **36**:77–88 (1958).

[9] Ross, D. G., I. N. Smith, P. C. Manins and D. G. Fox, *Diagnostic wind field modeling for complex terrain: Model development and testing*, J. Appl. Meteor., 27, 785-796 (1988).

[10] Kitada, T., Kaki, A., Ueda H. and Peters L. K., *Estimation of the vertical air motion from limited horizontal wind data–A numerical experiment*, Atmos. Environ. **17**, 2181–2192 (1983).

[11] Kitada, T. and Igarashi, K., *Numerical Analysis of Air Pollution in a Combined Field of Land/Sea Breeze and Mountain/Valley Wind*, Climite and Applied Met. **25**, 767–784 (1986).

[12] Núñez, M. A., Flores, C. and Juárez, H., *A study of hydrodynamic mass–consistent models*, Journal of Computational Methods in Science and Engineering, **6**, 365–385 (2006).

[13] Núñez, M. A., Flores, C. and Juárez, H., *Interpolation of hydrodynamic velocity data with the continuity equation*, Journal of Computational Methods in Science and Engineering,.vol. **7** (1), 21–42, (2007).

[14] Ciarlet, P. G., *The Finite Element Method for Elliptic Problems*, North–Holland Amsterdam (1970), re–edited as Vol. 40, SIAM, Classics in Applied Mathematics, Philadelphia, PA., (2002).

[15] Glowinski, R., *Numerical Methods for Fluids (Part 3), Handbook of Numerical Analysis, volume IX*, North–Holland, Amsterdam, (2003).

[16] Bercovier, M. and Pironneau O., *Error estimates for the finite element method solution of the Stokes problem in the primitive variables*, Numer. Math. **33**, 211–224 (1979).

[17] J. Cahouet and J. P. Chabard, *Some Fast 3d Finite Element Solvers for the Generalized Stokes Problem*, Int. J. Numer. math. Fluids, 8, pp. 869–895, (1988).

First Symposium on Inverse Problems and its Applications se terminó de imprimir en los talleres gráficos de S y G editores en mayo de 2011.