Paper CIT04-IL13

INVERSE AND OPTIMIZATION PROBLEMS IN HEAT TRANSFER

Marcelo J. Colaço
Department of Mechanical Engineering (DE/4)
Military Institute of Engineering – IME
Pç. Gen. Tiburcio, 80, RJ, BRAZIL
22290-270, Rio de Janeiro, Brazil
colaco@ime.eb.br

Helcio R. B. Orlande
Department of Mechanical Engineering - POLI/COPPE
Federal University of Rio de Janeiro, UFRJ
Cx. Postal: 68503, 21941-972, Rio de Janeiro, Brazil
helcio@serv.com.ufrj.br

George S. Dulikravich
Department of Mechanical and Materials Engineering, Professor and Chairperson
Florida International University, College of Engineering, Room EC 3474
10555 West Flagler Street, Miami, Florida 33174, U.S.A.
dulikrav@fiu.edu

This paper presents basic concepts of inverse and optimization problems. Deterministic and stochastic minimization techniques in finite and infinite dimensional spaces are revised; advantages and disadvantages of each of them are discussed and a hybrid technique is introduced. Applications of the techniques discussed to inverse and optimization problems in heat transfer are presented.

1. Introduction

Inverse problems of heat transfer rely on temperature and/or heat flux measurements for the estimation of unknown quantities appearing in the mathematical formulation of physical problems in this field [1-11]. As an example, inverse problems dealing with heat conduction have been generally associated with the estimation of an unknown boundary heat flux, by using temperature measurements taken below the boundary surface. Therefore, while in the classical direct heat conduction problem the cause (boundary heat flux) is given and the effect (temperature field in the body) is determined, the inverse problem involves the estimation of the cause from the knowledge of the effect.

The use of inverse analysis techniques represents a *new research paradigm*. The results obtained from numerical simulations and from experiments are not simply compared *a posteriori*, but a close synergism exists between experimental and theoretical researchers during the course of the study, in order to obtain the maximum of information regarding the physical problem under picture [12]. Therefore, in the recent past inverse problems have evolved from a specific theoretical research topic to an important practical tool of engineering analysis [1-21]. Nowadays, at least three international journals are specifically oriented to the community involved with the solution and application of inverse problems, both from the mathematical and from the engineering points of view, namely: *Inverse Problems, Inverse and Ill-posed Problems* and *Inverse Problems in Science and Engineering*. The *International Conference on Inverse Problems in Engineering: Theory and Practice*, organized under the auspices of the Engineering Conferences International (late United Engineering Foundation), is held in a three-year period. Also, several other seminars and symposia have been held in different countries in the past, including the *International Symposium on Inverse Problems in Engineering Mechanics* in Japan, and the *Inverse Problems, Design and Optimization Symposium* in Brazil. More specifically for the heat transfer community, all major conferences in the field, such as the *International Heat Transfer Conference*, have special sessions or mini-symposia dedicated to inverse problems.

Inverse problems can be solved either as a *parameter estimation* approach or as a *function estimation* approach. If some information is available on the functional form of the unknown quantity, the inverse problem can be reduced to the estimation of few unknown parameters. On the other hand, if no prior information is available on the functional form of the unknown, the inverse problem needs to be regarded as a function estimation approach in an infinite dimensional space of functions [1-21].

Inverse problems are mathematically classified as *ill-posed*, whereas standard heat transfer problems are *well-posed*. The solution of a well-posed problem must satisfy the conditions of existence, uniqueness and stability with respect to the input data [22]. The existence of a solution for an inverse heat transfer problem may be assured by physical reasoning. On the other hand, the uniqueness of the solution of inverse problems can be mathematically proved

only for some special cases. Also, the inverse problem is very sensitive to random errors in the measured input data, thus requiring special techniques for its solution in order to satisfy the stability condition.

A successful solution of an inverse problem generally involves its reformulation as an approximate well-posed problem and makes use of some kind of regularization (stabilization) technique. Although the solution techniques for inverse problems do not necessarily make use of optimization techniques, many popular methods are nowadays based on them [1-21].

Despite their similarities, inverse and optimization problems are conceptually different. *Inverse problems are concerned with the identification of unknown quantities appearing in the mathematical formulation of physical problems, by using measurements of the system response*. On the other hand, *optimization problems generally deal with the minimization or maximization of a certain objective or cost function, in order to find design variables that will result on desired state variables*. In addition, inverse and optimization problems involve other different concepts. For example, the solution technique for an inverse problem is required to cope with instabilities resulting from the noisy measured input data, while for an optimization problem the input data is given by the desired response(s) of the system. Differently from inverse problems, the solution uniqueness may not be an important issue for optimization problems, as long as the solution obtained is physically feasible and can be practically implemented. Engineering applications of optimization techniques are very often concerned with the minimization or maximization of different quantities, such as minimum weight (e.g., lighter airplanes), minimum fuel consumption (e.g., more economic cars), maximum autonomy (e.g., longer range airplanes), etc. The necessity of finding the maximum or minimum values of some parameters (or functions) can be governed by economic factors, as in the case of fuel consumption, or design characteristics, as in the case of maximum autonomy of an airplane. Sometimes, however, the decision is more subjective, as in the case of choosing a car model. In general, different designs can be idealized for a given application, but only a few of them will be economically viable. The best design is usually obtained by some Min-Max technique.

In this paper we address solution methodologies for inverse and single-objective optimization problems, based on minimization techniques. Several gradient and stochastic techniques are introduced, together with their basic implementation steps and algorithms. We present some deterministic methods, such as the Conjugate Gradient Method, the Newton Method and the Davidon-Fletcher-Powell Method [1-12, 23-41]. In addition, we present some of the most promising stochastic approaches, such as the Simulated Annealing Method [42,43], the Differential Evolutionary Method [44], Genetic Algorithms [45,46] and the Particle Swarm Method [47-51]. Deterministic methods are in general computationally faster than stochastic methods, although they can converge to a local minima or maxima, instead of the global one. On the other hand, stochastic algorithms can ideally converge to a global maxima or minima, although they are computationally slower than the deterministic ones. Indeed, the stochastic algorithms can require thousands of evaluations of the objective functions and, in some cases, become non-practical. In order to overcome these difficulties, we will also discuss the so-called hybrid algorithm, which takes advantage of the robustness of the stochastic methods and of the fast convergence of the deterministic methods [52-59]. Each technique provides a unique approach with varying degrees of convergence, reliability and robustness at different cycles during the iterative minimization procedure. A set of analytically formulated rules and switching criteria can be coded into the program to automatically switch back and forth among the different algorithms as the iterative process advances. Specific concepts for inverse and optimization problems are discussed and several examples in heat transfer are given in the paper.

2. Objective function

2.1. Basic concept

For the solution of inverse problems, as considered in this paper, we make use of minimization techniques that are of the same kind of those used in optimization problems. Therefore, the first step in establishing a procedure for the solution of either inverse problems or optimization problems is thus the definition of an *objective function*. The objective function is the mathematical representation of an aspect under evaluation, which must be minimized (or maximized). It can be mathematically stated as

$$U = U(\mathbf{x}); \quad \mathbf{x} = \{x_1, x_2, ..., x_N\} \tag{2.1.1}$$

where x_1, x_2, ... , x_N are the variables of the problem under consideration that can be modified in order to find the minimum value of the function U.

The relationship between U and $\mathbf{x}$ can, most of the times, be expressed by a physical / mathematical model. However, in some cases this relationship is impractical or even impossible and the variation of U with respect to $\mathbf{x}$ must be determined by experiments.

2.2. Constraints

Usually the variables x_1, x_2, ... , x_N which appear on the objective function formulation are only allowed to vary between some pre-specified ranges. Such *constraints* are, for example, due to physical or economical limitations.

We can have two types of constraints. The first one is the *equality constraint*, which can be represented by

$$G = G(\mathbf{x}) = \alpha \tag{2.2.1.a}$$

This kind of constraint can represent, for example, the pre-specified power of an automobile.
The second type of constraint is called *inequality constraint* and it is represented by

$$G = G(\mathbf{x}) \leq \alpha \tag{2.2.1.b}$$

This can represent, for example, the maximum temperature allowed in a gas turbine engine.

2.3. Optimization problems

For optimization problems, the objective function U can be, for example, the fuel consumption of an automobile and the variables x_1, x_2, ... , x_N can be the aerodynamic profile of the car, the material of the engine, the type of wheels used, the distance from the floor, etc.

2.4. Inverse problems and regularization

For inverse problems, the objective function usually involves the squared difference between measured and estimated variables. As a result, some statistical aspects regarding the unknown parameters and the measured errors need to be examined in order to select an appropriate objective function. In the examples presented below we assume that temperature measurements are available for the inverse analysis. The eight standard assumptions [2] regarding the *statistical description* of the problem are:

1. The errors are additive, that is,

$$Y_i = T_i + \varepsilon_i \tag{2.4.1.a}$$

where Y_i is the measured temperature, T_i is the actual temperature and ε_i is the random error.

2. The temperature errors ε_i have a zero mean, that is,

$$E(\varepsilon_i) = 0 \tag{2.4.1.b}$$

where $E(\cdot)$ is the expected value operator. The errors are then said to be unbiased.

3. The errors have constant variance, that is,

$$\sigma_i^2 = E\{[Y_i - E(Y_i)]^2\} = \sigma^2 = \text{constant} \tag{2.4.1.c}$$

which means that the variance of Y_i is independent of the measurement.

4. The errors associated with different measurements are uncorrelated. Two measurement errors ε_i and ε_j, where $i \neq j$, are uncorrelated if the covariance of ε_i and ε_j is zero, that is,

$$\text{cov}(\varepsilon_i, \varepsilon_j) \equiv E\{[\varepsilon_i - E(\varepsilon_i)][\varepsilon_j - E(\varepsilon_j)]\} = 0 \qquad \text{for } i \neq j \tag{2.4.1.d}$$

Such is the case if the errors ε_i and ε_j have no effect on or relationship to the other.

5. The measurement errors have a normal (Gaussian) distribution. By taking into consideration the assumptions 2, 3 and 4 above, the probability distribution function of ε_i is given by

$$f(\varepsilon_i) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-\varepsilon_i^2}{2\sigma^2}\right) \tag{2.4.1.e}$$

6. The statistical parameters describing ε_i, such as σ, are known.
7. The only variables that contain random errors are the measured temperatures. The measurement times, measurement positions, dimensions of the heated body, and all other quantities appearing in the formulation of the inverse problem are all *accurately* known.

8.   There is no prior information regarding the quantities to be estimated, which can be either parameters or functions. If such information exists, it can be utilized to obtain improved estimates.

If all of the eight statistical assumptions stated above are valid, the *objective function*, U, that provides minimum variance estimates is the *ordinary least squares norm* [2] (i.e., the sum of the squared residuals) defined as

$$U(\mathbf{x})=[\mathbf{Y}-\mathbf{T}(\mathbf{x})]^T [\mathbf{Y}-\mathbf{T}(\mathbf{x})] \tag{2.4.2}$$

where $\mathbf{Y}$ and $\mathbf{T}(\mathbf{x})$ are the vectors containing the measured and estimated temperatures, respectively, and the superscript T indicates the transpose of the vector. The estimated temperatures are obtained from the solution of the direct problem with estimates for the unknown quantities.

If the values of the standard deviations of the measurements are quite different, the ordinary least squares method does not yield minimum variance estimates [2]. In such a case, the *objective function* is given by the *weighted least squares norm*, U_w, defined as

$$U_W(\mathbf{x}) =[\mathbf{Y}-\mathbf{T}(\mathbf{x})]^T \mathbf{W}[\mathbf{Y}-\mathbf{T}(\mathbf{x})] \tag{2.4.3}$$

where $\mathbf{W}$ is a *diagonal weighting matrix*. Such matrix is taken as the inverse of the covariance matrix of the measurement errors in cases where the other statistical hypotheses remain valid [2].

If we consider that some information regarding the unknown parameters is available, we can use the *maximum a posteriori objective function* in the minimization procedure [2]. Such an objective function is defined as:

$$U_{MAP}(\mathbf{x}) = \left[\mathbf{Y} - \mathbf{T}(\mathbf{x})\right]^T \mathbf{W} \left[\mathbf{Y} - \mathbf{T}(\mathbf{x})\right]+(\boldsymbol{\mu} - \mathbf{x})^T \mathbf{V}^{-1} (\boldsymbol{\mu} - \mathbf{x}) \tag{2.4.4}$$

where $\mathbf{x}$ is assumed to be a random vector with known mean $\boldsymbol{\mu}$ and known covariance matrix $\mathbf{V}$. Therefore, the mean $\boldsymbol{\mu}$ and the covariance matrix $\mathbf{V}$ provide *a priori* information regarding the parameter vector $\mathbf{x}$ to be estimated. Such information can be available from previous experiments with the same experimental apparatus or even from the literature data. By assuming valid the other statistical hypotheses described above regarding the experimental errors, the weighting matrix $\mathbf{W}$ is given by the inverse of the covariance matrix of the measurement errors [2].

If the inverse heat transfer problem involves the estimation of only few unknown parameters, such as the estimation of a thermal conductivity value from the transient temperature measurements in a solid, the minimization of the objective functions given above can be stable. However, if the inverse problem involves the estimation of a large number of parameters, such as the recovery of the unknown transient heat flux components $f(t_i) \equiv f_i$ at times t_i, $i=1,...,I$, excursion and oscillation of the solution may occur. One approach to reduce such instabilities is to use the procedure called *Tikhonov's regularization* [1-21], which modifies the least squares norm by the addition of a term such as

$$U[f(t)]=\sum_{i=1}^{I}(Y_i-T_i)^2+\alpha*\sum_{i=1}^{I} f_i^2 \tag{2.4.5.a}$$

where α^* (> 0) is the *regularization parameter* and the second summation on the right is the *whole-domain zeroth-order regularization term*. In equation (2.4.5.a), f_i is the heat flux at time t_i, which is supposed to be constant in the interval $t_i - \Delta t/2 < t < t_i + \Delta t/2$, where Δt is the time interval between two consecutive measurements. The values chosen for the regularization parameter α^* influence the stability of the solution as the minimization of equation (2.4.5.a) is performed. As $\alpha^* \to 0$ the solution may exhibit oscillatory behavior and become unstable, since the summation of f_i^2 terms may attain very large values and the estimated temperatures tend to match those measured. On the other hand, with large values of α^* the solution is damped and deviates from the exact result.

The *whole-domain Tikhonov's first-order regularization* procedure involves the minimization of the following modified least squares norm:

$$U[f(t)]=\sum_{i=1}^{I}(Y_i-T_i)^2+\alpha*\sum_{i=1}^{I-1}(f_{i+1}-f_i)^2 \tag{2.4.5.b}$$

For $\alpha^* \to 0$, exact matching between estimated and measured temperatures is obtained as the minimization of $U[f(t)]$ is performed and the inverse problem solution becomes unstable. For large values of α^*, when the second summation in equation (2.4.5.b) is dominant, the heat flux components f_i tend to become constant for $i = 1, 2, ..., I$, that is, the first derivative of $f(t)$ tends to zero. Instabilities on the solution can be alleviated by proper selection of the value

of α^*. Tikhonov [1] suggested that α^* should be selected so that the minimum value of the objective function would be equal to the sum of the squares of the errors expected for the measurements, which is know as the *Discrepancy Principle*.

Alternative approaches for Tikhonov's regularization scheme described above is the use of *Beck's Sequential Function Specification Method* [4] or of *Alifanov's Iterative Regularization Methods* [3,5,7]. Beck's sequential function specification method is a quite stable inverse analysis technique. This is due to the averaging property of the least-squares norm and because the measurement at the time when the heat flux is to be estimated, is used in the minimization procedure together with few measurements taken at future time steps [4]. In Alifanov's regularization methods, the number of iterations plays the role of the regularization parameter α^* and the stopping criterion is so chosen that reasonably stable solutions be obtained. Therefore, there is no need to modify the original objective function, as opposed to Tikhonov's or Beck's approaches. The iterative regularization approach is sufficiently general and can be applied to both parameter and function estimations, as well as to linear and non-linear inverse problems [3,5,7].

3. Minimization techniques

In this section, we present deterministic and stochastic techniques for the minimization of an objective function $U(\mathbf{x})$ and the identification of the parameters $x_1, x_2, \ldots , x_N$, which appear on the objective function formulation. Basically, such kind of minimization problem is solved in a space of finite dimension N, which is the number of unknown parameters. For many minimization problems, the unknowns cannot be recast in the form of a finite number of parameters and the minimization needs to be performed in an infinite dimensional space of functions [1-23]. A powerful and straightforward technique for the minimization in a functional space is the conjugate gradient method with adjoint problem formulation [3,5,7,23]. Therefore, such technique is also described below, as applied to the solution of an inverse problem of function estimation.

3.1. Deterministic methods

This type of methods, as applied to non-linear minimization problems, generally rely on establishing an iterative procedure, which, after a certain number of iterations, will hopefully converge to the minimum of the objective function. The iterative procedure can be written in the following general form [2,3,5,24-27,39,40]:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha^k \mathbf{d}^k \qquad (3.1.1)$$

where $\mathbf{x}$ is the vector of design variables, α is the search step size, $\mathbf{d}$ is the direction of descent and k is the number of iterations.

An iteration step is *acceptable* if $U^{k+1} < U^k$. The direction of descent $\mathbf{d}$ will generate an acceptable step if and only if there exists a positive definite matrix $\mathbf{R}$, such that $\mathbf{d} = -\mathbf{R}\nabla U$ [39].

In fact, such requirement results on directions that form an angle greater than 90° with the gradient direction. A minimization method in which the directions are obtained in this manner is called an *acceptable gradient method* [39].

A *stationary point* of the objective function is one at which $\nabla U = 0$. The most that we can hope about any gradient method is that it converges to a stationary point. Convergence to the true minimum can be guaranteed only if it can be shown that the objective function has no other stationary points. In practice, however, one usually reaches the local minimum in the valley where the initial guess for the iterative procedure was located [39].

3.1.1. Steepest descent method

In this section we will introduce the most basic gradient method, which is the Steepest Descent Method. The basic idea of this method is to move downwards on the objective function along the direction of highest variation, in order to locate its minimum value. Therefore, the direction of descent is given by

$$\mathbf{d}^k = -\nabla U(\mathbf{x}^k) \qquad (3.1.2)$$

since the gradient direction is the one that gives the fastest increase of the objective function. Despite being the natural choice for the direction of descent, the use of the gradient direction is not very efficient. Usually the steepest-descent method starts with large variations in the objective function, but, as the minimum value is reached, the convergence rate becomes very low.

The optimum choice for the search step size, α, is the one that minimizes the objective function along the direction of descent at each iteration. Thus, a univariate search method need to be employed in order to find the search step size at each iteration. In the case of a unimodal function some classical procedures can be used, such as the Dichotomous

Search, Fibonacci Search, Golden Search and Cubic Interpolation, among others. However, for some realistic cases, the variation of the objective function with the search step size is not unimodal and, then, more robust techniques are required, such as the exhaustive search method or a technique based on exhaustive interpolation [24-27,39,40].

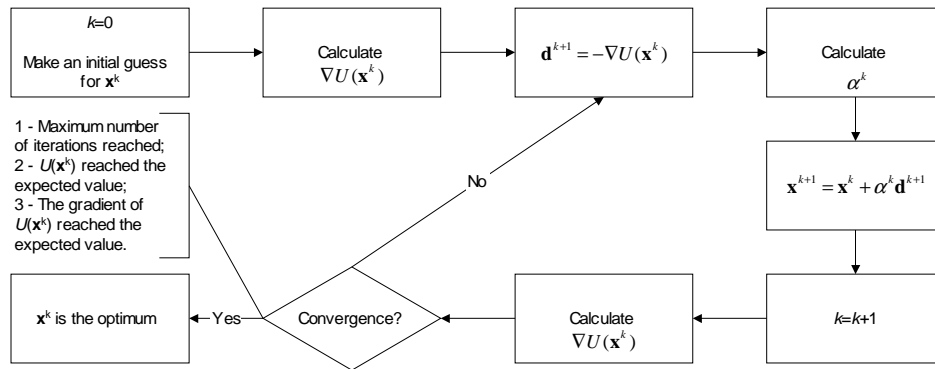Figure 3.1.1 illustrates the iterative procedure for the Steepest Descent Method.

Figure 3.1.1. Iterative procedure for the Steepest Descent Method.

3.1.2. Conjugate gradient method

The Conjugate Gradient Method improves the convergence rate of the Steepest Descent Method, by choosing directions of descent that are a linear combination of the gradient direction with directions of descent of previous iterations [3,5,7,23-33]. In a general form, the direction of descent is given by:

$$\mathbf{d}^k = -\nabla\left(\mathbf{x}^k\right) + \gamma^k \mathbf{d}^{k-1} + \psi^k \mathbf{d}^q \tag{3.1.3}$$

where γ^k and ψ^k are the conjugation coefficients. This direction of descent is used in the iterative procedure specified by equation (3.1.1).

The superscript q in equation (3.1.3) denotes the iteration number where a restarting strategy is applied to the iterative procedure of the conjugate gradient method. Restarting strategies were suggested for the conjugate gradient method of parameter estimation in order to improve its convergence rate [28].

Different versions of the Conjugate Gradient Method can be found in the literature depending on the form used for the computation of the direction of descent given by equation (3.1.3) [3,5,7,23-33]. In the Fletcher-Reeves version, the conjugation coefficients γ^k and ψ^k are obtained from the following expressions:

$$\gamma^k = \frac{\left\|\nabla U\left(\mathbf{x}^k\right)\right\|^2}{\left\|\nabla U\left(\mathbf{x}^{k-1}\right)\right\|^2} \text{ , with } \gamma^0 = 0 \text{ for } k=0 \tag{3.1.4.a}$$

$$\psi^k = 0 \text{ , for } k = 0,1,2,\ldots \tag{3.1.4.b}$$

where $\| \, . \, \|$ denotes the Euclidian norm in the vector space.

In the Polak-Ribiere version of the Conjugate Gradient Method the conjugation coefficients are given by:

$$\gamma^k = \frac{[\nabla U\left(\mathbf{x}^k\right)]^T \left[\nabla U\left(\mathbf{x}^k\right) - \nabla U\left(\mathbf{x}^{k-1}\right)\right]}{\left\|\nabla U\left(\mathbf{x}^{k-1}\right)\right\|^2} \text{ , with } \gamma^0 = 0 \text{ for } k=0 \tag{3.1.5.a}$$

$$\psi^k = 0 \text{ , for } k = 0,1,2,\ldots \tag{3.1.5.b}$$

Based on a previous work by Beale [32], Powell [28] suggested the following expressions for the conjugation coefficients, which gives the so-called Powell-Beale's version of the Conjugate Gradient Method:

$$\gamma^k = \frac{[\nabla U(\mathbf{x}^k)]^T [\nabla U(\mathbf{x}^k) - \nabla U(\mathbf{x}^{k-1})]}{[\mathbf{d}^{k-1}]^T [\nabla U(\mathbf{x}^k) - \nabla U(\mathbf{x}^{k-1})]} \text{ , with } \gamma^0 = 0 \text{ for } k=0 \tag{3.1.6.a}$$

$$\psi^k = \frac{[\nabla U(\mathbf{x}^k)]^T [\nabla U(\mathbf{x}^{q+1}) - \nabla U(\mathbf{x}^q)]}{[\mathbf{d}^q]^T [\nabla U(\mathbf{x}^{q+1}) - \nabla U(\mathbf{x}^q)]} \text{ , with } \psi^0 = 0 \text{ for } k=0 \tag{3.1.6.b}$$

In accordance with Powell [28], the application of the conjugate gradient method with the conjugation coefficients given by equations (3.1.6.a,b) requires restarting when gradients at successive iterations tend to be non-orthogonal (which is a measure of the local non-linearity of the problem) and when the direction of descent is not sufficiently downhill. Restarting is performed by making $\psi^k=0$ in equation (3.1.3).

The non-orthogonality of gradients at successive iterations is tested by the following equation:

$$ABS\left([\nabla(\mathbf{x}^{k-1})]^T \nabla(\mathbf{x}^k)\right) \geq 0.2 \left\|\nabla\left(\mathbf{x}^k\right)\right\|^2 \tag{3.1.7.a}$$

where ABS (.) denotes the absolute value.

A non-sufficiently downhill direction of descent (i.e., the angle between the direction of descent and the negative gradient direction is too large) is identified if either of the following inequalities is satisfied

$$[\mathbf{d}^k]^T \nabla\left(\mathbf{x}^k\right) \leq -1.2 \left\|\nabla\left(\mathbf{x}^k\right)\right\|^2 \tag{3.1.7.b}$$

or

$$[\mathbf{d}^k]^T \nabla\left(\mathbf{x}^k\right) \geq -0.8 \left\|\nabla\left(\mathbf{x}^k\right)\right\|^2 \tag{3.1.7.c}$$

We note that the coefficients 0.2, 1.2 and 0.8 appearing in equations (3.1.7.a-c) are empirical and are the same used by Powell [28].

In Powell-Beale's version of the conjugate gradient method, the direction of descent given by equation (3.1.3) is computed in accordance with the following algorithm for $k \geq 1$ [28]:

<u>STEP 1</u>: Test the inequality (3.1.7.a). If it is true set $q = k-1$.
<u>STEP 2</u>: Compute γ^k with equation (3.1.6.a).
<u>STEP 3</u>: If $k = q+1$ set $\psi^k = 0$. If $k \neq q+1$ compute ψ^k with equation (3.1.6.b).
<u>STEP 4</u>: Compute the search direction $\mathbf{d}^k$ with equation (3.1.3).
<u>STEP 5</u>: If $k \neq q+1$ test the inequalities (3.1.7.b,c). If either one of them is satisfied set $q = k-1$ and $\psi^k=0$. Then recompute the search direction with equation (3.1.3).

The Steepest Descent Method, with the direction of descent given by the negative gradient equation, would be recovered with $\gamma^k = \psi^k=0$ for any k in equation (3.1.3). The same procedures used for the evaluation of the search step size in the Steepest Descent Method are employed for the Conjugate Gradient Method.

Figure 3.1.2 shows the iterative procedure for the Fletcher-Reeves version of the Conjugate Gradient Method.
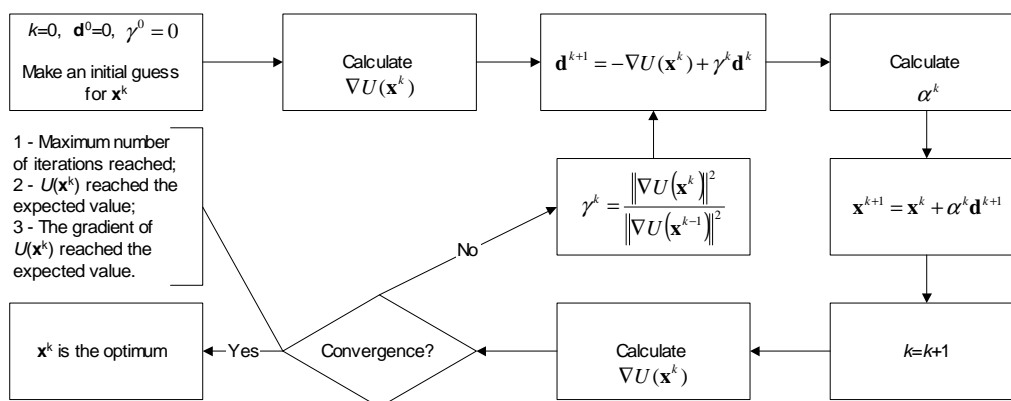

Figure 3.1.2. Iterative procedure for the Fletcher-Reeves version of the Conjugate Gradient Method.

3.1.3. Newton-Raphson method

While the Steepest Descent and the Conjugate Gradient Methods use gradients of the objective function in their iterative procedures, the Newton-Raphson Method also uses information of the second derivative of the objective function in order to achieve a faster convergence rate (which doesn't mean a lower computational cost).

Let us consider an objective function $U(\mathbf{x})$, which is, at least, twice differentiable. The Taylor expansion of $U(\mathbf{x})$ around a vector $\mathbf{x}^k$, where $\mathbf{x} - \mathbf{x}^k = \mathbf{h}$, is given by [2,24,26,35-40]:

$$U(\mathbf{x}^k + \mathbf{h}) = U(\mathbf{x}^k) + \nabla U(\mathbf{x}^k)^T \mathbf{h} + \frac{1}{2}\mathbf{h}^T D^2 U(\mathbf{x}^k)\mathbf{h} + O(\mathbf{h}^3) \qquad (3.1.8)$$

where $\nabla U(\mathbf{x})$ is the gradient (vector of 1st order derivatives) of the objective function and $D^2 U(\mathbf{x})$ is the Hessian (matrix of 2nd order derivatives) of the objective function.

If the objective function $U(\mathbf{x})$ is twice differentiable, the Hessian is symmetric, and we can write

$$\nabla U(\mathbf{x}^k + \mathbf{h}) \cong \nabla U(\mathbf{x}^k) + D^2 U(\mathbf{x}^k)\mathbf{h} \qquad (3.1.9)$$

A necessary condition for the minimization of the objective function is that its gradient vanishes. Therefore, from equation (3.1.9) we have:

$$\mathbf{h} \cong -\left[D^2 U(\mathbf{x}^k)\right]^{-1} \nabla U(\mathbf{x}^k) \qquad (3.1.10)$$

and the vector that locally optimizes the function $U(\mathbf{x})$ is given by

$$\mathbf{x} \cong \mathbf{x}^k - \left[D^2 U(\mathbf{x}^k)\right]^{-1} \nabla U(\mathbf{x}^k) \qquad (3.11)$$

Therefore, the iterative procedure of the Newton-Raphson method, given by equation (3.1.11), can be written in the same form of equation (3.1.1) by setting $\alpha^k = 1$ and

$$\mathbf{d}^k = -\left[D^2 U(\mathbf{x}^k)\right]^{-1} \nabla U(\mathbf{x}^k) \qquad (3.1.12.a)$$

However, the Newton-Raphson method converges to the extreme point closer to the initial guess, disregarding if it is a maximum, a minimum or a saddle point. For this reason it is common to introduce a search step size to be taken along the direction of descent for this method, so that it is written as:

$$\mathbf{d}^k = -\alpha^k \left[D^2 U(\mathbf{x}^k)\right]^{-1} \nabla U(\mathbf{x}^k) \qquad (3.1.12.b)$$

Figure 3.1.3 shows the iterative procedure for the Newton-Raphson Method.

Although the convergence rate of the Newton-Raphson is quadratic, the calculation of the Hessian is computationally expensive. As a result, other methods that approximate the Hessian with simpler and faster-computing forms have been developed. Some of these methods are described next.
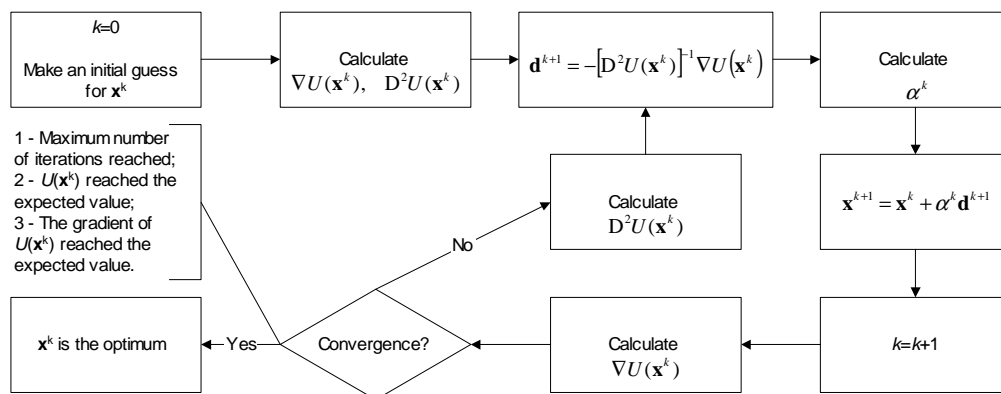

Figure 3.1.3. Iterative procedure for the Newton-Raphson Method.

3.1.4. Quasi Newton methods

In this kind of methods, the Hessian matrix appearing in the Newton-Raphson's Method is approximated in such a way that it does not involve second derivatives. Usually, the approximations for the Hessian are based on first derivatives. As a result, the Quasi-Newton Methods have a slower convergence rate than the Newton-Raphson Method, but they are computationally faster [2,24,26,35-40].

Let us define a new matrix **H**, which is an approximation for the inverse of the Hessian, that is,

$$\mathbf{H}^k \cong \left[\mathbf{D}^2 U(\mathbf{x}^k)\right]^{-1} \tag{3.1.13}$$

The direction of descent for the Quasi-Newton methods is thus given by:

$$\mathbf{d}^{k+1} = -\mathbf{H}^k \nabla U\!\left(\mathbf{x}^k\right) \tag{3.1.14}$$

and the matrix **H** is iteratively calculated as

$$\mathbf{H}^k = \mathbf{H}^{k-1} + \mathbf{M}^{k-1} + \mathbf{N}^{k-1} \ \text{ for } k = 1,2,\ldots \tag{3.1.15.a}$$

$$\mathbf{H}^k = \mathbf{I} \ \text{ for } k = 0 \tag{3.1.15.b}$$

where **I** is the identity matrix. Note that, for the first iteration, the method starts as the Steepest Descent Method.

Different Quasi-Newton methods can be found in the literature depending on the choice of the matrices **M** and **N**. For the Davidon-Fletcher-Powell (DFP) Method [33,34], such matrices are given by

$$\mathbf{M}^{k-1} = \alpha^{k-1}\,\frac{\mathbf{d}^{k-1}\left(\mathbf{d}^{k-1}\right)^T}{\left(\mathbf{d}^{k-1}\right)^T \mathbf{Y}^{k-1}} \tag{3.1.16.a}$$

$$\mathbf{N}^{k-1} = -\frac{\left(\mathbf{H}^{k-1}\mathbf{Y}^{k-1}\right)\left(\mathbf{H}^{k-1}\mathbf{Y}^{k-1}\right)^T}{\left(\mathbf{Y}^{k-1}\right)^T \mathbf{H}^{k-1}\mathbf{Y}^{k-1}} \tag{3.1.16.b}$$

where

$$\mathbf{Y}^{k-1} = \nabla U\!\left(\mathbf{x}^k\right) - \nabla U\!\left(\mathbf{x}^{k-1}\right) \tag{3.1.16.c}$$

Note that, since the matrix **H** is iteratively calculated, some errors can be propagated and, in general, the method needs to be restarted after some number of iterations. Also, since the matrix **M** depends on the choice of the search step size α, the method is very sensitive to its choice. A variation of the DFP method is the Broyden-Fletcher-Goldfarb-Shanno (BFGS) Method [33-36], which is less sensitive to the choice of the search step size. For this method, the matrices **M** and **N** are calculated as

$$\mathbf{M}^{k-1} = \left(1 + \frac{\left(\mathbf{Y}^{k-1}\right)^T \mathbf{H}^{k-1}\mathbf{Y}^{k-1}}{\left(\mathbf{Y}^{k-1}\right)^T \mathbf{d}^{k-1}}\right)\frac{\mathbf{d}^{k-1}\left(\mathbf{d}^{k-1}\right)^T}{\left(\mathbf{d}^{k-1}\right)^T \mathbf{Y}^{k-1}} \tag{3.1.17.a}$$

$$\mathbf{N}^{k-1} = -\frac{\mathbf{d}^{k-1}\left(\mathbf{Y}^{k-1}\right)^T \mathbf{H}^{k-1} + \mathbf{H}^{k-1}\mathbf{Y}^{k-1}\left(\mathbf{d}^{k-1}\right)^T}{\left(\mathbf{Y}^{k-1}\right)^T \mathbf{d}^{k-1}} \tag{3.1.17.b}$$

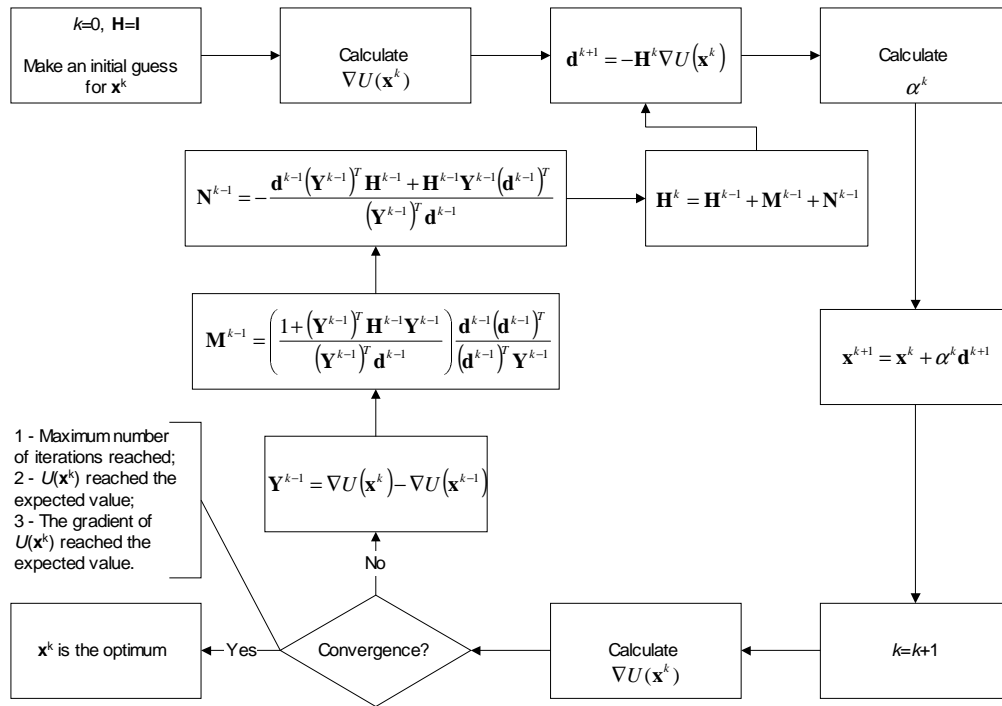Figure 3.1.4 shows the iterative procedure for the BFGS Method.

Figure 3.1.4. Iterative procedure for the BFGS Method.

3.1.5. Levenberg-Marquardt method

The Levenberg-Marquardt Method was first derived by Levenberg [37] in 1944, by modifying the ordinary least squares norm. Later, in 1963, Marquardt [38] derived basically the same technique by using a different approach. Marquardt's intention was to obtain a method that would tend to the Gauss method in the neighborhood of the minimum of the ordinary least squares norm, and would tend to the steepest descent method in the neighborhood of the initial guess used for the iterative procedure. This method actually converts a matrix that approximates the Hessian into a positive definite one, so that the direction of descent is acceptable.

The method rests on the observation that if $\mathbf{P}$ is a positive definite matrix, then $\mathbf{A} + \lambda\,\mathbf{P}$ is positive definite for sufficiently large λ. If $\mathbf{A}$ is an approximation for the Hessian, we can choose $\mathbf{P}$ as a diagonal matrix whose elements coincide with the absolute values of the diagonal elements of $\mathbf{A}$ [39].

The direction of descent for the Levenberg-Marquardt method is given by [39]:

$$\mathbf{d}^k = -(\mathbf{A}^k + \lambda^k \mathbf{P}^k)^{-1} \nabla U(\mathbf{x}^k) \tag{3.1.18}$$

and the step size is taken as $\alpha^k = 1$. Note that for large values of λ^k a small step is taken along the negative gradient direction. On the other hand, as λ^k tends to zero, the Levenberg-Marquardt method tends to an approximation of Newton's method based on the matrix $\mathbf{A}$. Usually, the matrix $\mathbf{A}$ is taken as that for the Gauss method [2,7,39].

3.2. Evolutionary and stochastic methods

In this section some Evolutionary and Stochastic Methods like Genetic Algorithms, Differential Evolution, Particle Swarm and Simulated Annealing will be discussed. Evolutionary methods, in opposition to the deterministic methods, don't rely, in general, on strong mathematical basis and do not make use of the gradient of the objective function as a direction of descent. They tend to mimic the nature in order to find the minimum of the objective function, by selecting, in a fashionable and organized way, the points where such function is going to be computed.

3.2.1. Genetic algorithms

Genetic algorithms are heuristic global optimization methods that are based on the process of natural selection. Starting from a randomly generated population of designs, the optimizer seeks to produce improved designs from one generation to the next. This is accomplished by exchanging genetic information between designs in the current population, in what is referred to as the crossover operation. Hopefully this crossover produces improved designs, which are then used to populate the next generation [45,46].

The basic genetic algorithm works with a collection or population of potential solutions to the minimization problem. The algorithm works in an iterative manner. At each iteration, also called generation, three operators are applied to the entire population of designs. These operators are selection, crossover, and mutation. For the operators to be effective, each potential solution or design must be represented as a collection of finite parameters, also called genes. Each design must have a unique sequence of these parameters that define it. This collection of genes is often called the chromosome. The genes themselves are often encoded as binary strings though they can be represented as real numbers. The length of the binary string determines how precisely the value, also know as the allele of the gene, is represented.

The genetic algorithm applied to an optimization problem proceeds as follows. The process begins with an initial population of random designs. Each gene is generated by randomly generating 0's and 1's. The chromosome strings are then formed by combining the genes together. This chromosome defines the design. The objective function is evaluated for each design in the population. Each design is assigned a fitness value, which corresponds to the value of the objective function for that design. In the case of minimization, a higher fitness is assigned to designs with lower values of the object function.

Next, the population members are selected for reproduction, based upon their fitness. The selection operator is applied to each member of the population. The selection operator chooses pairs of individuals from the population who will mate and produce offspring. In the tournament selection scheme, random pairs are selected from the population and the individual with the higher fitness of each pair is allowed to mate.

Once a mating pair is selected, the crossover operator is applied. The crossover operator essentially produces new designs or offspring by combining the genes from the parent designs in a stochastic manner. In the uniform crossover scheme, it is possible to obtain any combination of the two parent's chromosomes. Each bit in each gene in the chromosome is assigned a probability that crossover will occur (for example, 50 % for all genes). A random number between 0 and 1 is generated for each bit in each gene. If a number greater than 0.5 is generated then that bit is replaced by the corresponding bit in the gene from the other parent. If it is less than 0.5, the original bit in the gene remains unchanged. This process is repeated for the entire chromosome for each of the parents. When complete, two offspring are generated, which may replace the parents in the population.

The mutation process follows next. When the crossover procedure is complete and a new population is formed, the mutation operator is applied. Each bit in each gene in the design is subjected to a chance for a change from 0 to 1, or vice versa. The chance is known as the mutation probability, which is usually small. This introduces additional randomness into the process, which helps to avoid local minima. Completion of the mutation process signals the end of a design cycle. Many cycles may be needed before the method converges to an optimum design.

For more details or for the numerical implementation of Genetic Algorithms, the reader is referred to [45,46].

3.2.2. Differential evolution

The Differential Evolution Method is an evolutionary method based on Darwin's Theory for the Evolution of the Species. It was created in 1995 by two researchers from Berkeley (Kenneth Price and Rainer Storn) [44] as an alternative to the Genetic Algorithm Method. Following the Theory for the Evolution of the Species, the strongest members of a population will be more capable to survive to a certain environmental condition. During the matting process, the chromosomes of two individuals of the population are combined in a process called crossover. During this process mutations can occur, which can be good (individual with a better objective function) or bad (individual with a worst objective function). The mutations are used as a way to escape from local minima. However, their excessive usage can lead to a non-convergence of the method.

The method starts with a randomly generated population in the domain of interest. Then, successive combinations of chromosomes and mutations are performed, creating new generations until an optimum value is found.

The iterative process of the Differential Evolution Method is given by:

$$\mathbf{x}_i^{k+1} = \delta_1 \mathbf{x}_i^k + \delta_2 \left[\boldsymbol{\alpha} + F(\boldsymbol{\beta} - \boldsymbol{\gamma})\right] \tag{3.2.1}$$

where

$\mathbf{x}_i$ is the *i*-th individual vector of parameters.

$\boldsymbol{\alpha}, \boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ are three members of the population matrix **P**, randomly chosen.

F is a weight function, which defines the mutation ($0.5 < F < 1$).

k is the number of generations.

δ_1 and δ_2 are Dirac delta functions that define the mutation.

In the minimization process, if $U(\mathbf{x}^{k+1}) < U(\mathbf{x}^k)$, then $\mathbf{x}^{k+1}$ replaces $\mathbf{x}^k$ in the population matrix **P**. Otherwise, $\mathbf{x}^k$ is kept in the population matrix.

The binomial crossover is given as

$$\delta_1 = \begin{cases} 0, & \text{if } R < CR \\ 1, & \text{if } R > CR \end{cases} \qquad \delta_2 = \begin{cases} 1, & \text{if } R < CR \\ 0, & \text{if } R > CR \end{cases} \tag{3.2.2.a,b}$$

where CR is a factor that defines the crossover (0.5 < CR < 1) and R is a random number with uniform distribution between 0 and 1.

Figure 3.2.1 shows the iterative procedure for the Differential Evolution Method.
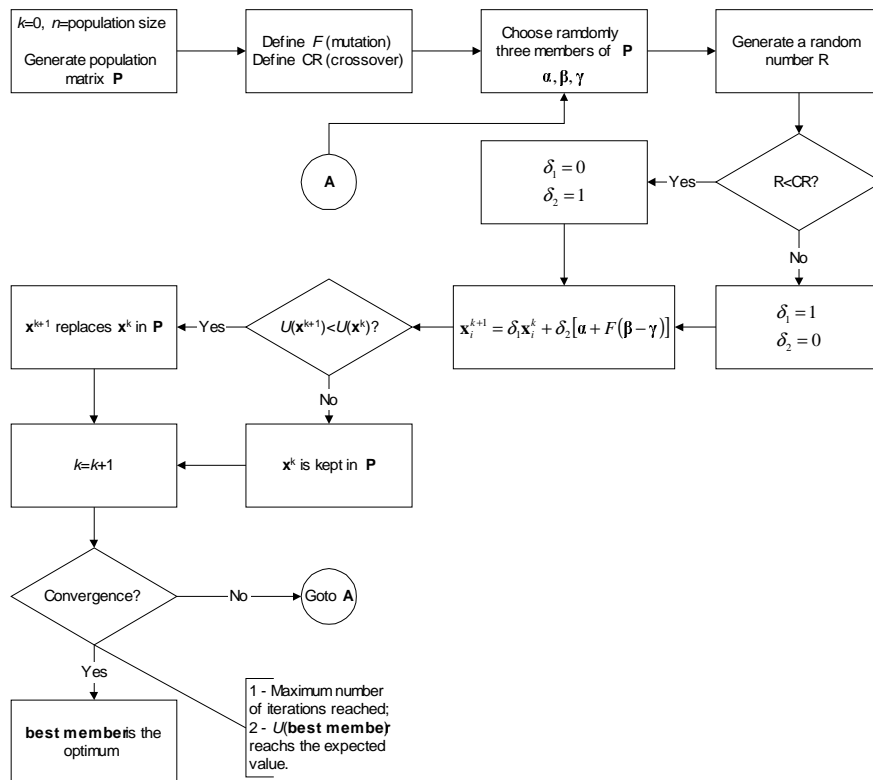

Figure 3.2.1. Iterative procedure for the Differential Evolution Method.

3.2.3. Particle swarm

Another evolutionary method is the one that uses the concepts of Particle Swarm. Such method was created in 1995 by an Electrical Engineer (Russel Eberhart) and a Social Psychologist (James Kennedy) [47-50] as an alternative to the Genetic Algorithm Method. This method is based on the social behavior of various species and tries to equilibrate the individuality and sociability of the individuals in order to locate the optimum of interest. The original idea of Kennedy and Eberhart came from the observation of birds looking for nesting places. When the individuality, is increased the search for alternative places for nesting is also increased. However, if the individuality becomes too high, the individual might never finds the best place. On the other hand, when the sociability is increased, the individual learns more from its neighbors' experience. However, if the sociability becomes too high, all the individuals might converge to the first minima found, which is possibly a local minima.

In this method, the iterative procedure is given by:

$$\mathbf{x}_i^{k+1} = \mathbf{x}_i^k + \mathbf{v}_i^{k+1} \tag{3.2.3.a}$$

$$\mathbf{v}_i^{k+1} = \alpha \mathbf{v}_i^k + \beta \mathbf{r}_{1i}\left(\mathbf{p}_i - \mathbf{x}_i^k\right) + \beta \mathbf{r}_{2i}\left(\mathbf{p}_g - \mathbf{x}_i^k\right) \tag{3.2.3.b}$$

where:
$\mathbf{x}_i$ is the *i*-th individual vector of parameters.
$\mathbf{v}_i = 0$, for k=0.
$\mathbf{r}_{1i}$ and $\mathbf{r}_{2i}$ are random numbers with uniform distribution between 0 and 1.
$\mathbf{p}_i$ is the best value found for the vector $\mathbf{x}_i$.
$\mathbf{p}_g$ is the best value found for the entire population.
$0 < \alpha < 1; 1 < \beta < 2$

In equation (3.2.3.b), the second term on the right hand side represents the individuality and the third term the sociability. The first term on the right-hand side represents the inertia of the particles and, in general, must be decreased as the iterative process runs. In this equation, the vector $\mathbf{p}_i$ represents the best value ever found for the i-th component vector of parameters $\mathbf{x}_i$ during the iterative process. Thus, the individuality term involves the comparison between the current value of the i-th individual $\mathbf{x}_i$ with its best value in the past. The vector $\mathbf{p}_g$ is the best value ever found for the entire population of parameters (not only the i-th individual). Thus the sociability term compares $\mathbf{x}_i$ with the best value of the entire population in the past.

Figure 3.2.2 shows the iterative procedure for the Particle Swarm Method.
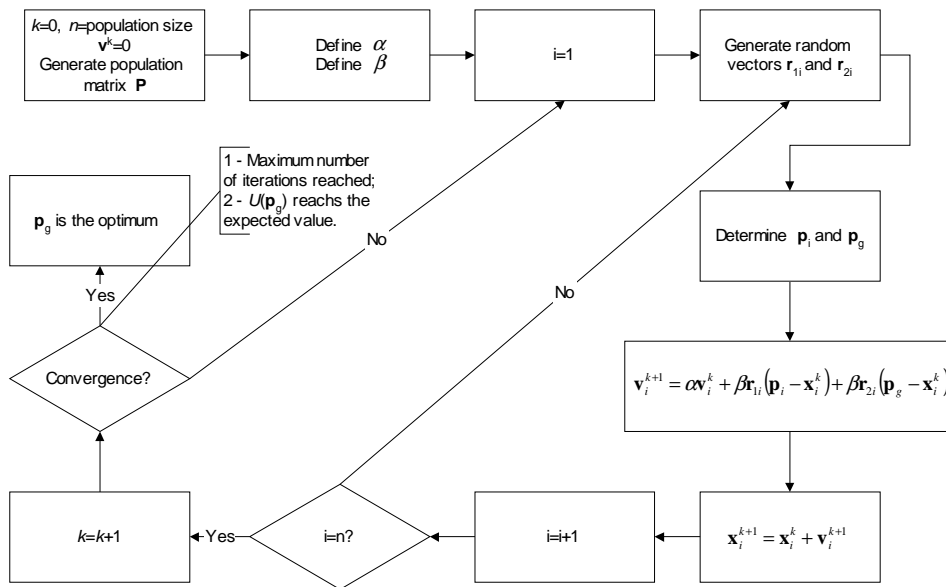
Figure 3.2.2. Iterative procedure for the Particle Swarm Method.

3.2.4. Simulated annealing

This method is based on the thermodynamics of the cooling of a material from a liquid to a solid phase [42,43]. If a liquid material (e.g. liquid metal) starts being slowly cooled down and left for a sufficiently long time close to the phase change temperature, a perfect crystal will be created, which has the lowest internal energy state. On the other hand, if the liquid material is not left for a sufficient long time close to the phase change temperature, or, if the cooling process is not sufficiently slow, the final crystal will have several defects and a high internal energy state. This is similar to the quenching process used in metallurgical applications.

We can say that gradient-based methods "cool down too fast", going rapidly to an optimum location which, in most cases, is not the global but the local one. Differently from the gradient-based methods, the Simulated Annealing Method can move in any direction, escaping from possible local minimum or maximum values. Consider, for example, the Boltzmann probability function

$$\text{Prob}(E) \quad \propto \quad e^{(-E/KT)} \tag{3.2.4}$$

This equation expresses the idea that a system in thermal equilibrium has its energy distributed probabilistically among different energy states E. In this equation, K is the Boltzmann constant. Such equation tells us that, even at low temperatures, there is a chance, even small, that the system is at a high energy level, as illustrated in figure 3.2.3. Thus, there is a chance of the system to get out of this local minimum and search for a global one.
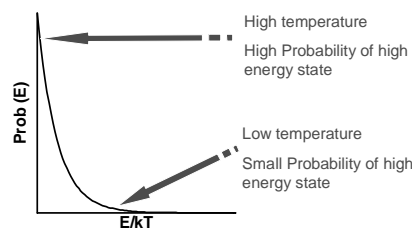

Figure 3.2.3. Schematic representation of eq. (3.2.4).

Figure 3.2.4 shows the iterative procedure for the Simulated Annealing Method. The procedure starts by generating a population of individuals of the same size of the number of variables ($n=m$), in such a way that the population matrix is a square matrix. Then, the initial temperature (T), the reducing ratio (RT), the number of cycles (N_s) and the number of iterations of the annealing process (N_{it}) are selected. After N_S*n function evaluations, each element of the step length V is adjusted, so that approximately half of all function evaluations are accepted. The suggested value for the number of cycles is 20. After $N_{it}*N_s*n$ function evaluations, the temperature (T) is changed by the factor RT. The value suggested for the number of iterations by Corana et al [42] is MAX(100, 5*n).

The iterative process is given by the following equation:

$$x_i^1 = x_i^0 + RV_i \qquad (3.2.5)$$

where R is a random number with uniform distribution between 0 and 1 and V is a step-size which is continuously adjusted.

To start, it randomly chooses a trial point within the step length V (a vector of length N) of the user selected starting point. The function is evaluated at this trial point (x_i^1) and its value is compared to its value at the initial point (x_i^0). In a minimization problem, all downhill moves are accepted and the algorithm continues from that trial point. Uphill moves may be accepted; the decision is made by the Metropolis criteria. It uses T (temperature) and the size of the downhill move in a probabilistic manner

$$P = e^{\left[U\left(x_i^1\right) - U\left(x_i^0\right) \right] / T} \qquad (3.2.6)$$

The smaller T and the size of the uphill move are, the more likely that move will be accepted. If the trial is accepted, the algorithm moves on from that point. If it is rejected, another point is chosen for a trial evaluation.

Each element of V is periodically adjusted, so that half of all function evaluations in that direction are accepted. The number of accepted function evaluations is represented by the variable N_i. Thus the variable r represents the ratio of accepted over total function evaluations for an entire cycle N_s and it is used to adjust the step length V.

A decrease in T is imposed upon the system with the RT variable by using

$$T(i+1) = RT * T(i) \qquad (3.2.7)$$

where i is the i-th iteration. Thus, as T declines, uphill moves are less likely to be accepted and the percentage of rejections rises. Given the scheme for the selection for V, V falls. Thus, as T declines, V falls and the algorithm focuses upon the most promising area for optimization.

The parameter T is crucial for the successful use of the algorithm. It influences V, the step length over which the algorithm searches for the minimum. For a small initial T, the step length may be too small; thus not enough values of the function will be evaluated to find the global minimum. To determine the starting temperature that is consistent with optimizing a function, it is worthwhile to run a trial run first. The user should set $RT = 1.5$ and $T = 1.0$. With $RT > 1.0$, the temperature increases and V rises as well. Then it must be selected the T that produces a large enough V.
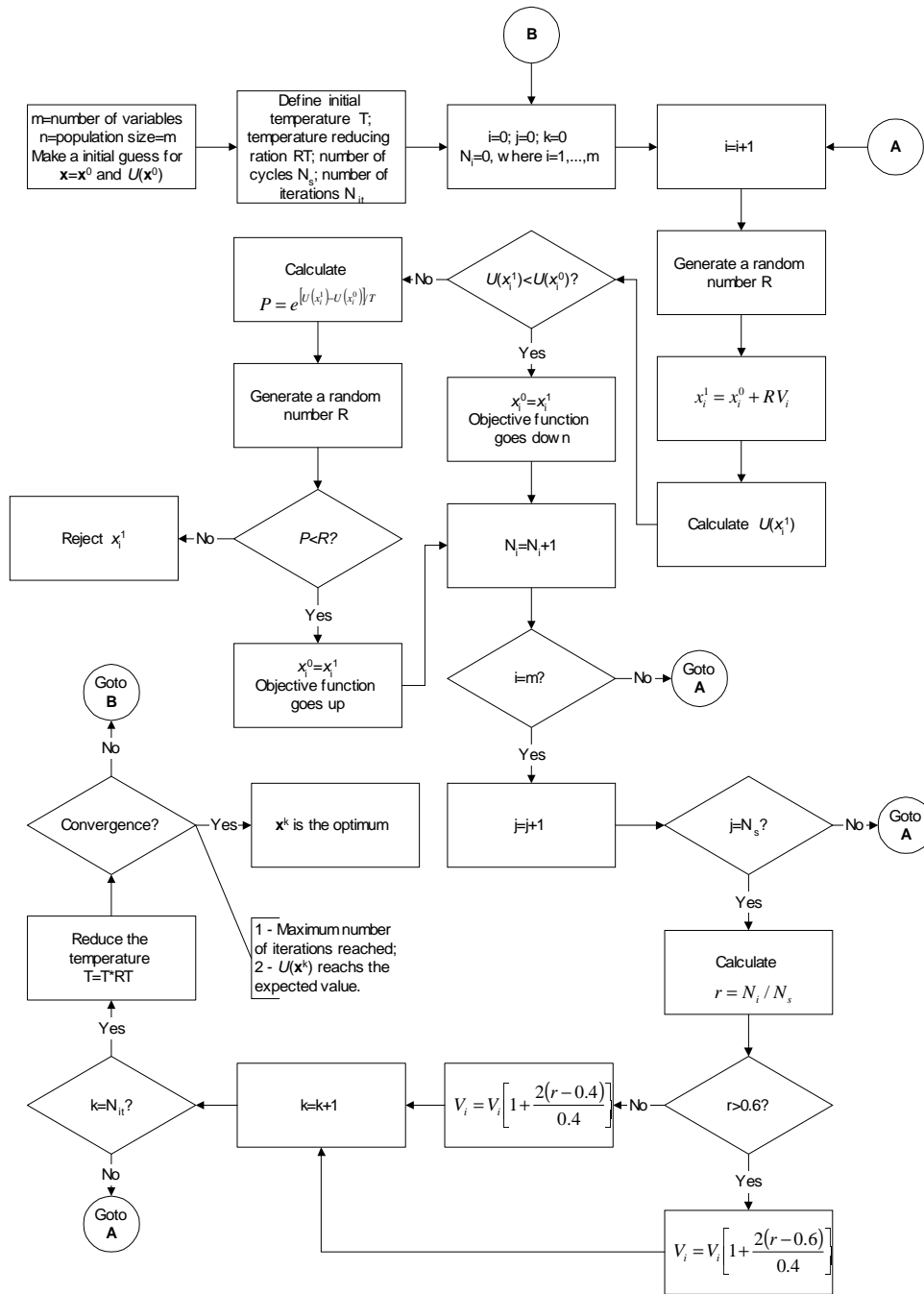
Figure 3.2.4. Iterative procedure for the Simulated Annealing Method.

3.3. Hybrid methods

The so-called Hybrid Methods are a combination of the deterministic and the evolutionary/stochastic methods, in a sense that the advantages of each one of them are used. Hybrid Methods usually employ an evolutionary/stochastic method to locate the region where the global minimum is located and then switches to a deterministic method to get closer to the exact point faster.

As an example, consider the Hybrid Method illustrated in figure 3.3.1. The main module is the Particle Swarm Method, which does almost the entire optimization task. When some percentual of the particles find a minima (lets say, some birds already found their best nesting place), the algorithm switches to the Differential Evolution Method and the particles (birds) are forced to breed. If there is an improvement on the objective function, the algorithm returns to the Particle Swarm Method, meaning that some other region is more prone to have a global minimum. If there is no improvement on the objective function, this can indicate that this region already contains the global minimum expected and the algorithm switches to the BFGS Method in order to locate more precisely its location. Finally, the algorithm

returns again to the Particle Swarm Method in order to check if there are any changes in the minimum location and the entire procedure is repeated for few more iterations (e.g., 5).

More involved Hybrid Methods, dealing with the application of other deterministic and stochastic methods, can be found in references [52-59].
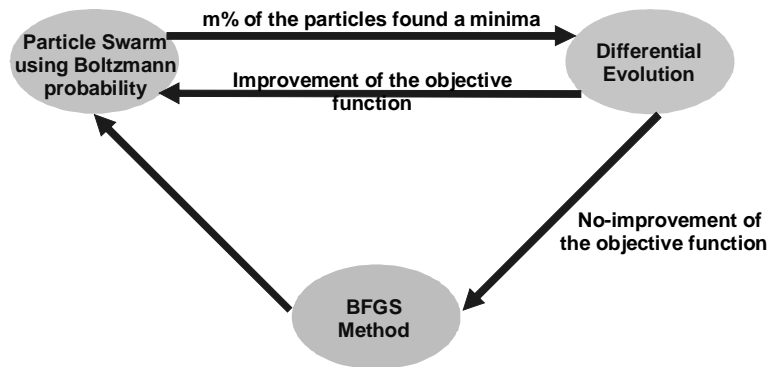
Figure 3.3.1. Global procedure for the Hybrid Method.

3.4. Function estimation approach

The methods described above were applied for the minimization of an objective function $U(\mathbf{x})$, where $\mathbf{x} = [x_1, x_2, \ldots, x_N]$ is the vector with the parameters of the problem under consideration that can be modified in order to find the minimum of $U(\mathbf{x})$. Therefore, the minimization is performed in a parameter space of dimension N. Several optimization or inverse problems rely on functions, instead of parameters, which need to be selected for the minimization of an objective function. In these cases, the minimization needs to be performed in an infinite dimensional space of functions and no *a priori* assumption is required regarding the functional form of the unknown functions, except for the functional space that they belong to [3,5,7]. A common selection is the Hilbert space of square integrable functions in the domain of interest.

In this paper we use the conjugate gradient method with adjoint problem formulation for the minimization of the objective function. We illustrate here the function estimation approach as applied to the solution of a specific inverse problem involving the following diffusion equation [59]:

$$C*(\mathbf{r}*)\frac{\partial T*(\mathbf{r}*,t*)}{\partial t*} = \nabla \cdot [D*(\mathbf{r}*)\nabla T*] + \mu*(\mathbf{r}*)T* \qquad (3.4.1)$$

where $\mathbf{r}*$ denotes the vector of coordinates and the superscript $*$ denotes dimensional quantities.

Equation (3.4.1) can be used for the modeling of several physical phenomena, such as heat conduction, groundwater flow and tomography. We focus our attention here to a one-dimensional version of equation (1) written in dimensionless form as

$$\frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(D(x)\frac{\partial T}{\partial x}\right) + \mu(x)T \quad \text{in } 0 < x < 1, \text{ for } t > 0 \qquad (3.4.2.a)$$

and subject to the following boundary and initial conditions.

$$\frac{\partial T}{\partial x} = 0 \quad \text{at } x = 0 \text{ for } t > 0 \qquad (3.4.2.b)$$

$$D(x)\frac{\partial T}{\partial x} = 1 \quad \text{at } x = 1 \text{ for } t > 0 \qquad (3.4.2.c)$$

$$T = 0 \quad \text{for } t = 0 \quad \text{in } 0 < x < 1 \qquad (3.4.2.d)$$

Notice that in the *direct problem*, the diffusion coefficient function $D(x)$ and the source term distribution function $\mu(x)$ are regarded as known quantities, so that a direct (analysis) problem is concerned with the computation of $T(x,t)$.

3.4.1. Inverse problem

For the *inverse problem* of interest here, the functions $D(x)$ and $\mu(x)$ are regarded as unknown. Such functions will be simultaneously estimated by using measurements of $T(x,t)$ taken at appropriate locations in the medium or on its boundaries. Such measurements may contain random errors. These measurement errors are assumed to be uncorrelated, additive, normally distributed, with zero mean, and with a known constant standard deviation.

Practical applications of this inverse problem include the identification of non-homogeneities in the medium, such as inclusions, obstacles or cracks, determination of thermal diffusion coefficients and distribution of heat sources, groundwater flow and tomography physical problems, in which both $D(x)$ and $\mu(x)$ vary.

The basic steps of the conjugate gradient method of function estimation are discussed below.

3.4.2. Conjugate gradient method with adjoint problem

For the simultaneous estimation of the functions $D(x)$ and $\mu(x)$ with the conjugate gradient method with adjoint problem, we consider the minimization of the following objective functional

$$U[D(x),\mu(x)] = \frac{1}{2} \int_{t=0}^{t_f} \sum_{m=1}^{M} \{T[x_m,t;D(x),\mu(x)] - Y_m(t)\}^2 dt$$

(3.4.3)

where $Y_m(t)$ are the transient measurements of $T(x,t)$ taken at the positions x_m, $m = 1,\ldots, M$. The estimated dependent variable $T[x_m,t;D(x),\mu(x)]$ is obtained from the solution of the direct problem (3.4.2.a-d) at the measurement positions x_m, $m = 1,\ldots, M$, with estimates for $D(x)$ and $\mu(x)$. When dealing with an optimization problem, instead of an inverse problem, $Y_m(t)$ represents the desired temperature at the positions x_m, $m = 1,\ldots, M$. We note in equation (3.4.3) that, for simplicity in the analytical analysis developed below, the measurements $Y_m(t)$ are assumed to be continuous in the time domain.

The use of the conjugate gradient method with an adjoint problem for the minimization of the objective functional (3.4.3) requires the solution of auxiliary problems, known as *sensitivity and adjoint problems*.

The *sensitivity function*, solution of the sensitivity problem, is defined as the directional derivative of $T(x,t)$ in the direction of the perturbation of the unknown function [3,5,7]. Since the present problem involves two unknown functions, two sensitivity problems are required for the estimation procedure, resulting from perturbations in $D(x)$ and $\mu(x)$.

The sensitivity problem for $\Delta T_D(x,t)$ is obtained by assuming that the dependent variable $T(x,t)$ is perturbed by $\varepsilon\Delta T_D(x,t)$ when the diffusion coefficient $D(x)$ is perturbed by $\varepsilon\Delta D(x)$, where ε is a real number. The sensitivity problem resulting from perturbations in $D(x)$ is then obtained by applying the following limiting process:

$$\lim_{\varepsilon \to 0} \frac{L_\varepsilon(D_\varepsilon) - L(D)}{\varepsilon} = 0$$

(3.4.4)

where $L_\varepsilon(D_\varepsilon)$ and $L(D)$ are the direct problem formulations written in operator form for perturbed and unperturbed quantities, respectively. The application of the limiting process given by equation (3.4.4) results in the following sensitivity problem:

$$\frac{\partial \Delta T_D}{\partial t} = \frac{\partial}{\partial x}\left(D(x)\frac{\partial \Delta T_D}{\partial x} + \Delta D(x)\frac{\partial T}{\partial x}\right) + \mu(x)\Delta T_D \quad \text{in } 0 < x < 1 \quad \text{for } t > 0$$

(3.4.5.a)

$$\frac{\partial \Delta T_D}{\partial x} = 0 \qquad\qquad \text{at } x = 0 \quad \text{for } t > 0$$

(3.4.5.b)

$$\Delta D(x)\frac{\partial T}{\partial x} + D(x)\frac{\partial \Delta T_D}{\partial x} = 0 \quad \text{at } x = 1 \quad \text{for } t > 0$$

(3.4.5.c)

$$\Delta T_D = 0 \quad \text{in } 0 \le x \le 1 \quad \text{for } t = 0$$

(3.4.5.d)

A limiting process analogous to that given by equation (3.4.4), obtained from the perturbation $\varepsilon\Delta\mu(x)$, results in the following sensitivity problem for $\Delta T_\mu(x,t)$

$$\frac{\partial \Delta T_\mu}{\partial t} = \frac{\partial}{\partial x}\left(D(x)\frac{\partial \Delta T_\mu}{\partial x}\right) + \mu(x)\Delta T_\mu + \Delta\mu(x)T \quad \text{in } 0 < x < 1 \quad \text{for } t > 0 \tag{3.4.6.a}$$

$$\frac{\partial \Delta T_\mu}{\partial x} = 0 \quad \text{at } x = 0 \text{ and } x = 1 \quad \text{for } t > 0 \tag{3.4.6.b,c}$$

$$\Delta T_\mu = 0 \quad \text{in } 0 \le x \le 1 \text{ ; for } t = 0 \tag{3.4.6.d}$$

A Lagrange multiplier $\lambda(x,t)$ is utilized in the minimization of the functional (3.4.3) because the estimated dependent variable $T[x_m,t;D(x),\mu(x)]$ appearing in such functional needs to satisfy a constraint, which is the solution of the direct problem. Such Lagrange multiplier, needed for the computation of the gradient equations (as will be apparent below), is obtained through the solution of problems *adjoint* to the sensitivity problems, given by equations (3.4.5.a-d) and (3.4.6.a-d) [3]. Despite the fact that the present inverse problem involves the estimation of two unknown functions, thus resulting in two sensitivity problems as discussed above, one single problem, adjoint to problems (3.4.5.a-d) and (3.4.6.a-d), is obtained.

In order to derive the adjoint problem, the governing equation of the direct problem, equation (3.4.2.a), is multiplied by the Lagrange multiplier $\lambda(x,t)$, integrated in the space and time domains of interest and added to the original functional (3.4.3). The following extended functional is obtained:

$$U[D(x),\mu(x)] = \frac{1}{2}\int_{x=0}^{1}\int_{t=0}^{t_f}\sum_{m=1}^{M}[T-Y]^2\delta(x-x_m)dt\,dx +$$

$$\int_{x=0}^{1}\int_{t=0}^{t_f}\left[\frac{\partial T}{\partial t} - \frac{\partial}{\partial x}\left(D(x)\frac{\partial T}{\partial x}\right) - \mu(x)T\right]\lambda(x,t)\,dt\,dx \tag{3.4.7}$$

where δ is the Dirac delta function.

Directional derivatives of $U[D(x),\mu(x)]$ in the directions of perturbations in $D(x)$ and $\mu(x)$ are respectively defined by

$$\Delta U_D[D,\mu] = \lim_{\varepsilon\to 0}\frac{S[D_\varepsilon,\mu] - S[D,\mu]}{\varepsilon} \tag{3.4.8.a}$$

$$\Delta U_\mu[D,\mu] = \lim_{\varepsilon\to 0}\frac{S[D,\mu_\varepsilon] - S[D,\mu]}{\varepsilon} \tag{3.4.8.b}$$

where $U[D_\varepsilon,\mu]$ and $U[D,\mu_\varepsilon]$ denote the extended functional (3.4.7) written for perturbed $D(x)$ and $\mu(x)$, respectively.

After letting the above directional derivatives of $U[D(x),\mu(x)]$ go to zero, which is a necessary condition for the minimization of the extended functional (3.4.7), and after performing some lengthy but straightforward manipulations [3,5,7], the following adjoint problem for the Lagrange multiplier $\lambda(x,t)$ is obtained

$$-\frac{\partial \lambda}{\partial t} - \frac{\partial}{\partial x}\left(D(x)\frac{\partial \lambda}{\partial x}\right) - \mu(x)\lambda + \sum_{m=1}^{M}[T-Y]\delta(x-x_m) = 0 \quad \text{in } 0 < x < 1 \text{ , for } t > 0 \tag{3.4.9.a}$$

$$\frac{\partial \lambda}{\partial x} = 0 \quad \text{at } x = 0 \text{ and } x = 1 \quad \text{for } t > 0 \tag{3.4.9.b,c}$$

$$\lambda = 0 \quad \text{in } 0 \le x \le 1 \text{ for } t = t_f \tag{3.4.9.d}$$

During the limiting processes used to obtain the adjoint problem, applied to the directional derivatives of $S[D(x),\mu(x)]$ in the directions of perturbations in $D(x)$ and $\mu(x)$, the following integral terms are respectively obtained:

$$\Delta U_D[D,\mu] = \int\limits_{x=0}^{1} \int\limits_{t=0}^{t_f} \Delta D(x)\frac{\partial T}{\partial x}\frac{\partial \lambda}{\partial x} dt\, dx \tag{3.4.10.a}$$

$$\Delta U_\mu[D,\mu] = -\int\limits_{x=0}^{1} \int\limits_{t=0}^{t_f} \Delta\mu(x)\lambda(x,t)T(x,t)dt\, dx \tag{3.4.10.b}$$

By invoking the hypotheses that $D(x)$ and $\mu(x)$ belong to the Hilbert space of square integrable functions in the domain $0 < x < 1$, it is possible to write [3,5,7]:

$$\Delta U_D[D,\mu] = \int\limits_{x=0}^{1}\nabla S[D(x)]\Delta D(x)\, dx \tag{3.4.11.a}$$

$$\Delta U_\mu[D,\mu] = \int\limits_{x=0}^{1}\nabla S[\mu(x)]\Delta\mu(x)\, dx \tag{3.4.11.b}$$

Hence, by comparing equations (3.4.10.a,b) and (3.4.11.a,b) we obtain the gradient components of $U[D,\mu]$ with respect to $D(x)$ and $\mu(x)$, respectively, as

$$\nabla U[D(x)] = \int\limits_{t=0}^{t_f} \frac{\partial T}{\partial x}\frac{\partial \lambda}{\partial x} dt \tag{3.4.12.a}$$

$$\nabla U[\mu(x)] = -\int\limits_{t=0}^{t_f} \lambda(x,t)T(x,t)dt \tag{3.4.12.b}$$

An analysis of equations (3.4.12.a) and (3.4.9.b.c) reveals that the gradient component with respect to $D(x)$ is null at $x = 0$ and at $x = 1$. As a result, the initial guess used for $D(x)$ is never changed by the iterative procedure of the conjugate gradient method at such points, which can create instabilities in the inverse problem solution in their neighborhoods.

For the simultaneous estimation of $D(x)$ and $\mu(x)$, the iterative procedure of the conjugate gradient method is written respectively as [3,5,7]

$$D^{k+1}(x) = D^k(x) + \alpha_D^k d_D^k(x) \tag{3.4.13.a}$$

$$\mu^{k+1}(x) = \mu^k(x) + \alpha_\mu^k d_\mu^k(x) \tag{3.4.13.b}$$

where $d_D^k(x)$ and $d_\mu^k(x)$ are the directions of descent for $D(x)$ and $\mu(x)$, respectively; α_D^k and α_μ^k are the search step sizes for $D(x)$ and $\mu(x)$, respectively; and k is the number of iterations.

For the iterative procedure for each unknown function, the direction of descent is obtained as a linear combination of the gradient direction with directions of descent of previous iterations. The directions of descent for the conjugate gradient method for $D(x)$ and $\mu(x)$ can be written respectively as

$$d_D^k(x) = -\nabla S[D^k(x)] + \gamma_D^k d_D^{k-1}(x) + \psi_D^k d_D^{qD} \tag{3.4.14.a}$$

$$d_\mu^k(x) = -\nabla S[\mu^k(x)] + \gamma_\mu^k d_\mu^{k-1}(x) + \psi_\mu^k d_\mu^{q\mu} \tag{3.4.14.b}$$

where γ_D^k, γ_μ^k, ψ_D^k and ψ_μ^k are the conjugation coefficients. The superscripts qD and $q\mu$ in equations (3.4.14.a,b) represent the iteration numbers where a restarting strategy is applied to the iterative procedure for the estimation of $D(x)$ and $\mu(x)$, respectively [28].

Different versions of the conjugate gradient method can be found in the literature, depending on how the conjugation coefficients are computed. In this work we use the so-called Powell-Beale's version of the conjugate gradient method because of its superior robustness and convergence rate in non-linear problems [59]. The conjugation coefficients for this version of the conjugate gradient method are given by:

$$\gamma_D^k = \frac{\int\limits_{x=0}^{1} \left\{ \nabla S[D^k(x)] - \nabla S[D^{k-1}(x)] \right\} \nabla S[D^k(x)]\,dx}{\int\limits_{x=0}^{1} \left\{ \nabla S[D^k(x)] - \nabla S[D^{k-1}(x)] \right\} d_D^{k-1}(x)\,dx} \tag{3.4.15.a}$$

$$\gamma_\mu^k = \frac{\int\limits_{x=0}^{1} \left\{ \nabla S[\mu^k(x)] - \nabla S[\mu^{k-1}(x)] \right\} \nabla S[\mu^k(x)]\,dx}{\int\limits_{x=0}^{1} \left\{ \nabla S[\mu^k(x)] - \nabla S[\mu^{k-1}(x)] \right\} d_\mu^{k-1}(x)\,dx} \tag{3.4.15.b}$$

$$\psi_D^k = \frac{\int\limits_{x=0}^{1} \left\{ \nabla S[D^{qD+1}(x)] - \nabla S[D^{qD}(x)] \right\} \nabla S[D^k(x)]\,dx}{\int\limits_{x=0}^{1} \left\{ \nabla S[D^{qD+1}(x)] - \nabla S[D^{qD}(x)] \right\} d_D^{qD}(x)\,dx} \tag{3.4.16.a}$$

$$\psi_\mu^k = \frac{\int\limits_{x=0}^{1} \left\{ \nabla S[\mu^{q\mu+1}(x)] - \nabla S[\mu^{q\mu}(x)] \right\} \nabla S[\mu^k(x)]\,dx}{\int\limits_{x=0}^{1} \left\{ \nabla S[\mu^{q\mu+1}(x)] - \nabla S[\mu^{q\mu}(x)] \right\} d_\mu^{q\mu}(x)\,dx} \tag{3.4.16.b}$$

where $\gamma_D^k = \gamma_\mu^k = \psi_D^k = \psi_\mu^k = 0$, for $k = 0$.

Powell-Beale's version of the conjugate gradient method is restarted by making the conjugation coefficient $\psi_D^k = 0$ (or $\psi_\mu^k = 0$) if gradients at successive iterations are too far from being orthogonal (which is a measure of the nonlinearity of the problem) or if the direction of descent is not sufficiently downhill. For further details, the reader is referred to [59].

The search step sizes α_D^k and α_μ^k, appearing in the expressions of the iterative procedures for the estimation of $D(x)$ and $\mu(x)$, equations (3.4.13.a,b), respectively, are obtained by minimizing the objective functional at each iteration along the specified directions of descent. If the objective functional given by equation (3.4.3) is linearized with respect to α_D^k and α_μ^k, closed form expressions can be obtained for such quantities as follows [3,5,7]:

$$\alpha_d^k = \frac{F_1 A_{22} - F_2 A_{12}}{A_{11} A_{22} - A_{12}^2} \quad ; \quad \alpha_\mu^k = \frac{F_2 A_{11} - F_1 A_{12}}{A_{11} A_{22} - A_{12}^2} \tag{3.4.17.a,b}$$

where

$$A_{11} = \int\limits_{t=0}^{t_f} \sum_{m=1}^{M} [\Delta T_D^k(x_m, t)]^2\,dt \tag{3.4.18.a}$$

$$A_{22} = \int\limits_{t=0}^{t_f} \sum_{m=1}^{M} [\Delta T_\mu^k(x_m, t)]^2\,dt \tag{3.4.18.b}$$

$$A_{12} = \int\limits_{t=0}^{t_f} \sum_{m=1}^{M} \Delta T_D^k(x_m, t)\Delta T_\mu^k(x_m, t)\,dt \tag{3.4.18.c}$$

$$F_1 = \int\limits_{t=0}^{t_f} \sum_{m=1}^{M} [Y_m^k - T^k(x_m, t)][\Delta T_D^k(x_m, t)]\,dt \tag{3.4.18.d}$$

$$F_2 = \int\limits_{t=0}^{t_f} \sum_{m=1}^{M} [Y_m^k - T^k(x_m, t)][\Delta T_\mu^k(x_m, t)]\,dt \tag{3.4.18.e}$$

In equations (3.4.18.a-e), $\Delta T_D^k(x,t)$ and $\Delta T_\mu^k(x,t)$ are the solutions of the sensitivity problems given by equations (3.4.5.a-d) and (3.4.6.a-d), respectively, obtained by setting $\Delta D^k(x) = d_D^k(x)$ and $\Delta \mu^k(x) = d_\mu^k(x)$.

The use of the conjugate gradient method for the simultaneous estimation of $D(x)$ and $\mu(x)$ can be suitably arranged in a systematic and straightforward computational procedure, which is omitted here for the sake of brevity, but can be readily adapted from those found in reference [7]. The conjugate gradient method of function estimation belongs to the class of *iterative regularization methods* [3]. For this class of methods, the stopping criterion for the computational procedure is chosen so that sufficiently accurate and smooth solutions are obtained for the unknown functions. Although different approaches can be used for the specification of the tolerance for the stopping criterion, we use in this work the *discrepancy principle* [3].

With the use of the discrepancy principle, the iterative procedure of the conjugate gradient method is stopped when the difference between measured and estimated variables is of the order of the standard deviation, σ, of the measurements, that is, when

$$\left| T(x_m,t;D,\mu) - Y_m(t) \right| \approx \sigma \qquad (3.4.19)$$

Therefore, the iterative procedure is stopped when

$$U[D(x),\mu(x)] < \varphi \qquad (3.4.20)$$

where the tolerance, φ, is obtained by substituting equation (3.4.19) into the definition of the functional given by equation (3.4.3), that is,

$$\varphi = \frac{1}{2} M \sigma^2 t_f \qquad (3.4.21)$$

4. Applications

We now present some applications of inverse and optimization problems in heat transfer. The first example deals with the simultaneous estimation of the spatial distributions of the diffusion coefficient and of the source-term in a 1D diffusion problem. Such is the same kind of inverse problem for which the conjugate gradient method of function estimation was derived above. The other four examples of applications deal with convective heat transfer. The first two examples involve the solution of inverse problems in irregularly shaped channels (forced convection) and cavities (natural convection), respectively, while the other two involve the optimization of externally induced electro or magnetic fields in order to control the solidification characteristics of melts.

4.1. Simultaneous estimation of spatially-dependent diffusion coefficient and source term in a nonlinear diffusion problem [59]

This work deals with the simultaneous estimation of the spatially varying diffusion coefficient and of the source term distribution in a one-dimensional nonlinear diffusion problem. The problem formulation is given by equations (3.4.2.a-d). This work can be physically associated with the detection of material non-homogeneities such as inclusions, obstacles or cracks, heat conduction, groundwater flow detection, and tomography. Two solution techniques are applied in this work to the inverse problem under consideration, namely: the conjugate gradient method with adjoint problem and a hybrid optimization algorithm. For the hybrid optimization algorithm, stabilization for the solution of the inverse problem was obtained with Tikhonov's first order regularization. Thus, the objective function was rewritten in the form

$$U[D(x),\mu(x)] = \frac{1}{2} \int_{t=0}^{t_f} \sum_{m=1}^{M} \{T[x_m,t;D(x),\mu(x)] - Y_m(t)\}^2 dt +$$
$$+ \alpha_1 \sum_{m=1}^{M-1} \left\{ [D_{i+1}(x) - D_i(x)]^2 + [\mu_{i+1}(x) - \mu_i(x)]^2 \right\} \qquad (4.1.1)$$

where α_1 is the first-order Tikhonov's regularization parameter.

The test cases examined below in dimensionless form are physically associated with a heat conduction problem in a homogeneous steel bar of length 0.050 *m*. The diffusion coefficient and the spatial distribution of the source term are supposed to vary from base values of $D(x) = 54$ *W/mK* and $\mu(x) = 10^5$ *W/m³K*, which result in dimensionless base values of $D_c = 1$ and $\mu_c = 5$, respectively. The base values for the diffusion coefficient and source term distribution are associated with solid-solid phase transformations in steels. The final time is assumed to be 60 seconds, resulting in a dimensionless value of $t_f = 0.36$, and 50 measurements are supposed available per temperature sensor.

Figure 4.1.1 shows the results obtained with the conjugate gradient method and with the measurements of two non-intrusive sensors, for a step variation of $D(x)$ and for constant $\mu(x)$. The simulated measurements in this case contained random errors with standard deviation $\sigma = 0.01\ Y_{max}$, where Y_{max} is the maximum absolute value of the measured variable. The initial guesses used for the iterative procedure of the conjugate gradient method and for the hybrid optimizer were $D(x) = 0.9$ and $\mu(x) = 4.5$. We note in Figure 4.1.1 that quite good results were obtained for such a strict test case involving a discontinuous variation for $D(x)$, even with only two non-intrusive sensors, by using the conjugate gradient method of function estimation.
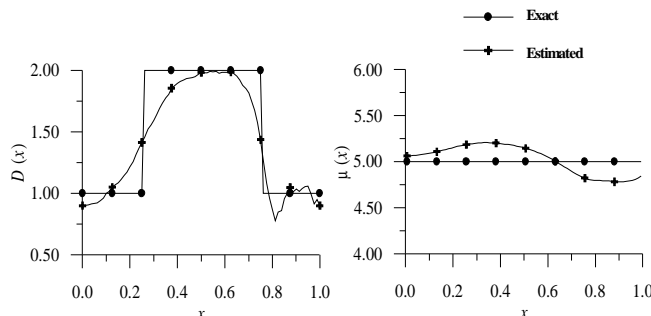
Figure 4.1.1. Estimation of $\mu(x)$ and $D(x)$ obtained by the CGM with
two non-intrusive sensors with standard deviation ($\sigma = 0.01\ Y_{max}$).

Figure 4.1.2 shows the results obtained for the same test case shown in Figure 1, but with 10 sensors. We note in Figure 2 that more accurate results are obtained for both $D(x)$ and $\mu(x)$ when more sensors are used in the inverse analysis.
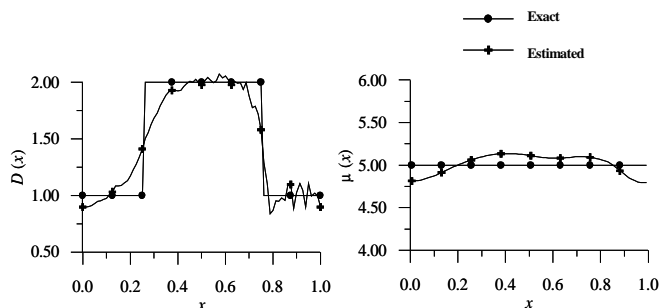
Figure 4.1.2. Estimation of $\mu(x)$ and $D(x)$ obtained by the CGM
with ten sensors with standard deviation ($\sigma = 0.01\ Y_{max}$).

Figure 4.1.3 shows results similar to those presented in Figure 4.1.1, obtained with the hybrid optimization algorithm by using 2 non-intrusive sensors, with the regularization parameter set as zero. The x-axis in this figure represents the ith value of the control volume, that is, $i = 80$ represents $x = 1.0$. Note that the estimated functions suffer from large oscillations resulting from the ill-posed character of the inverse problem.
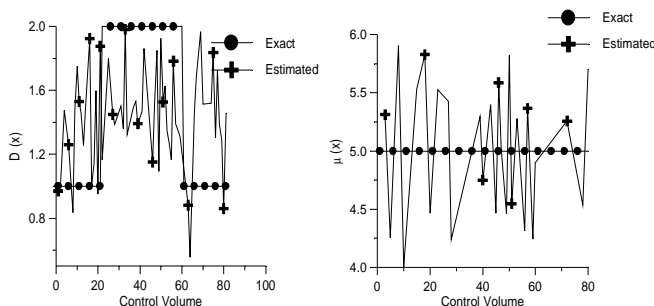
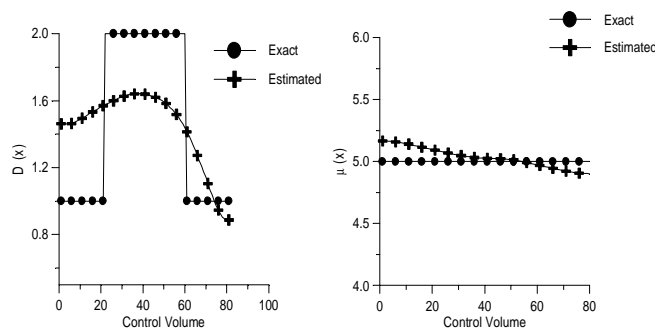

Figure 4.1.3. Estimation of $\mu(x)$ and $D(x)$ by the hybrid optimizer
without regularization and with 2 sensors ($\sigma = 0.01\ Y_{max}$).

In order to find the best value of the first order Tikhonov's regularization parameter α_1, we used the L-shape curve as shown in Figure 4.1.4. In this figure, the x-axis represents the second term appearing on the RHS of equation (4.1.1)

and the *y*-axis represents the first term appearing on the RHS of equation (4.1.1). The best choice for α_1 is the one that minimizes both terms represented by the *x* and *y*-axis. The optimum value for α_1 in this case was 0.0001.
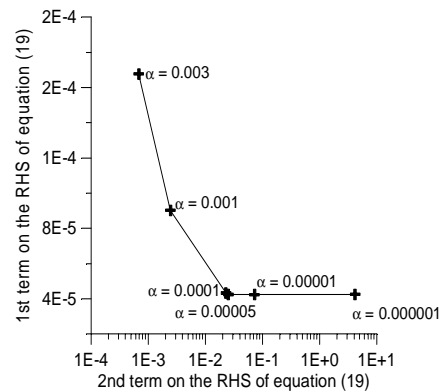
Figure 4.1.4. L-shape curve for choice of the regularization parameter using 2 sensors.

Figure 4.1.5 shows the estimated functions obtained with such value of the regularization parameter and the hybrid optimization approach, with the measurements of two non-intrusive sensors. Note that the oscillations are eliminated because of the stabilization introduced by the first order regularization term. However, the estimated function for $D(x)$ is in very bad agreement with the exact one. This is probably due to the fact that this case, involving 50 measurements per sensor and only two sensors, is underdetermined, that is, the number of unknown parameters is less than the number of measurements. Note that for the hybrid optimization approach, 2 parameters are estimated for each of the control-volumes used for the discretization of the domain, corresponding to the values of $D(x)$ and $\mu(x)$ at the control volume. Therefore, a total of 160 parameters were estimated in this case. A comparison of the functions estimated for $D(x)$ with the conjugate gradient method and with the hybrid optimization approach by using only two sensors (see figures 4.1.1 and 4.1.5, respectively) shows that the conjugate gradient method is not as sensitive to the fact that the problem is undetermined. This is because of the fact that the measured data is used in the source-function term of the adjoint problem in the function estimation approach with the conjugate gradient method, so that the information from the sensors at the boundaries is extended to the rest of the domain through the adjoint function $\lambda(x,t)$. On the other hand, the accuracies of the functions estimated for $\mu(x)$ with the conjugate gradient method and with the hybrid optimization approach, with the measurements of two non-intrusive sensors (see figures 4.1.1 and 4.1.5, respectively), are similar. This is due to the fact that the exact $\mu(x)$ is constant and the initial guess used for both approaches was relatively near the exact function.

Figure 4.1.5. Estimation of $\mu(x)$ and $D(x)$ by the hybrid optimizer
with regularization and with 2 sensors ($\sigma = 0.01\ Y_{max}$).

Figure 4.1.6 shows the estimated functions obtained with the hybrid optimization approach and the measurements of 10 sensors evenly spread in the medium and with $\alpha_1 = 0.0001$. It is interesting to note that the function estimated for $\mu(x)$ with 10 sensors is in much better agreement with the exact one than that obtained with 2 sensors. A comparison of figures 4.1.2 and 4.1.6 shows that similar results are obtained for the simultaneous estimation of $D(x)$ and $\mu(x)$, by using either the conjugate gradient method or the hybrid optimization approach, when ten sensors are used in the inverse analysis.
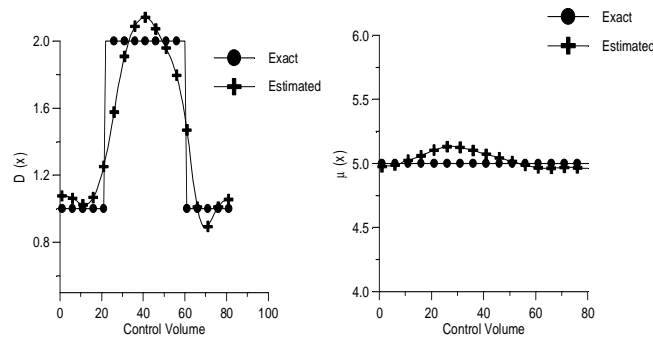
Figure 4.1.6. Estimation of $\mu(x)$ and $D(x)$ by the hybrid optimizer
with regularization and with 10 sensors ($\sigma = 0.01\ Y_{max}$).

We note that the use of the regularization parameter, based on the Tikhonov's technique, produced the regularization necessary to obtain stable results for the estimation of both functions with the hybrid optimization approach. In fact, completely unstable results were obtained if the regularization technique was not used, as a result of the ill-posed character of the inverse problem under picture. Also, the results presented above show that the two solution approaches examined in this paper are not sensitive to measurement errors. In fact, qualitatively similar results were obtained by using errorless simulated measurements.

Finally, Figure 4.1.7 shows the convergence history of the hybrid optimizer for the estimation of the functions presented in Figure 4.1.6, where one can see that no further reduction in the cost function is obtained after approximately 500 iterations.
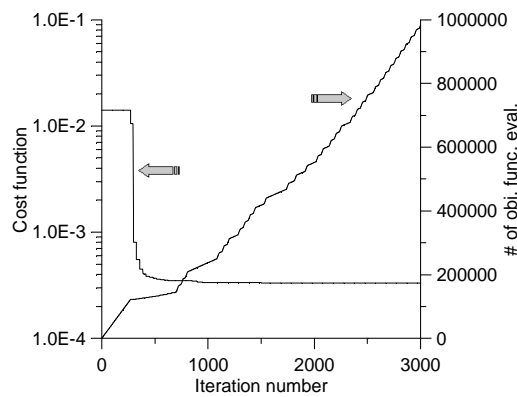
Figure 4.1.7. Convergence history for the simultaneous estimation of $\mu(x)$ and $D(x)$
by the hybrid optimizer with regularization and with 10 sensors.

4.2. Inverse forced convection problem of simultaneous estimation of two boundary heat fluxes in irregularly shaped channels [60]

This work deals with the use of the conjugate gradient method of function estimation for the simultaneous identification of two unknown boundary heat fluxes in channels with laminar flows. The irregularly shaped channel in the physical domain is transformed into a parallel plate channel in the computational domain, by using an elliptic scheme of numerical grid generation. The direct problem, as well as the auxiliary problems and the gradient equations, required for the solution of the inverse problem with the conjugate gradient method, are formulated in terms of generalized boundary-fitted coordinates. Therefore, this solution approach can be readily applied to forced convection boundary inverse problems in channels of any shape. Direct and auxiliary problems are solved with finite-volumes. The numerical solution for the direct problem is validated by comparing the results obtained here with benchmark solutions for smoothly expanding channels.

For the results presented below, we considered as an example of application of the present solution approach, test-cases involving the laminar flow of water ($\rho = 1000.52\ kg/m^3$, $\mu = 0.001\ kg/m\ s$, $k = 0.597\ W/m$, $C_p = 4.1818 \times 10^3\ J/kg$ $^{\circ}$C) inside a channel with a smooth expansion, as illustrated in figure 4.2.1.
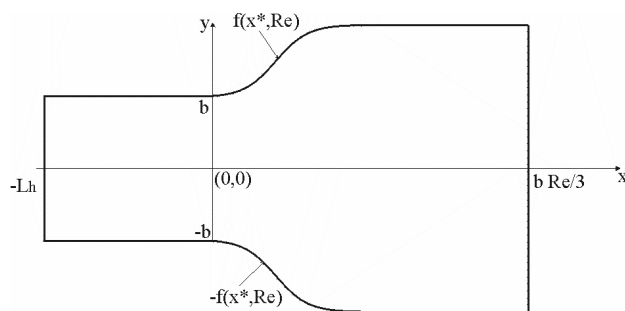
Figure 4.2.1. Geometry for the channel with smooth expansion.

Test-cases involving the estimation of the time and spatial variations for the heat fluxes were examined. For the estimation of the unknown functions we considered available the readings of 28 sensors uniformly distributed along the channel and located at the second control-volumes distant from each of the walls. Simulated measurements with a standard-deviation $\sigma = 0.01\ T_{max}$ were used for the inverse analysis. Figures 4.2.2.a and 4.2.2.b show that the estimated functions are quite accurate, even for the strict exact functions tested here, and for measurements with random errors. These figures present the results obtained for $t = 999$ s and 1665 s, respectively.
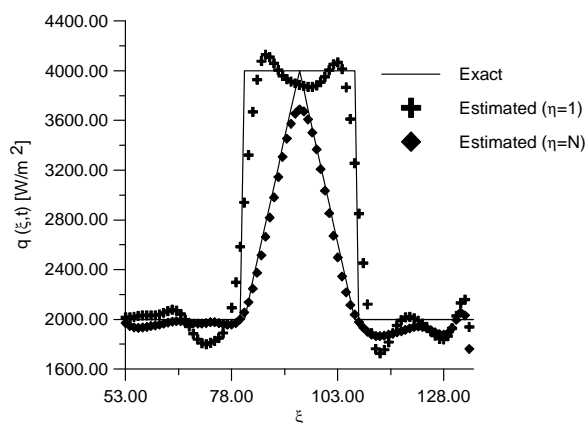
Figure 4.2.2.a. Estimated and exact heat fluxes for t=999 s.
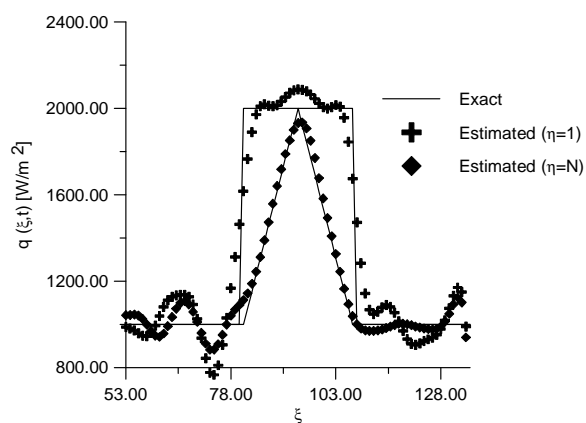
Figure 4.2.2.b. Estimated and exact heat fluxes for t=1665 s.

4.3. Inverse natural convection problem of simultaneous estimation of two boundary heat fluxes in irregular cavities [61]

This paper deals with the use of the Conjugate Gradient Method of function estimation with Adjoint Problem for the simultaneous identification of two boundary conditions in natural convection inverse problems in two-dimensional irregular cavities. This is a much more involved inverse problem solution than that addressed above for forced convection. Such is the case because, for natural convection, the energy equation is coupled with continuity and momentum equations. As a result, the counterparts of these equations for the sensitivity and adjoint problems are also coupled.

The unknown boundary conditions are estimated with no *a priori* information about their functional forms. Irregular geometries in the physical domain are transformed into regular geometries in the computational domain by using an elliptic scheme of numerical grid generation. The methodology is applied to cases involving an annular cavity, as depicted in figure 4.3.1, where the position and time dependent heat fluxes are unknown at the inner and outer surfaces. These two surfaces are supposed to be maintained at the constant temperatures T_h and T_c, respectively.

For the results presented here we considered natural convection of air with physical properties: ρ=1.19 kg/m^3; μ=1.8 x 10^{-5} kg/m s; β=0.00341 K^{-1}; Pr=0.70; K=0.2624 W/m K; C_P=1020.4 J/Kg ^{o}C. The test-cases analyzed below correspond to a Rayleigh number of 5 x 10^4, where the characteristic length used was $L=R_2 - R_1$. For this Rayleigh number, R_2 was taken as 54.4 mm and R_1 as 22.9 mm, while the temperatures at the walls $\eta=N$ and $\eta=1$ were taken as $T_h = 30$ ^{o}C and $T_c = 20$ ^{o}C, respectively.
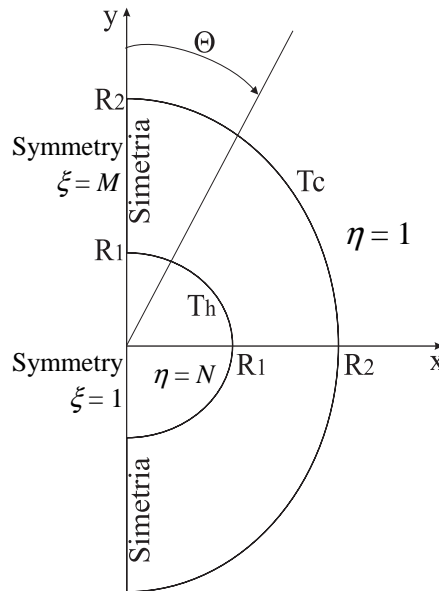
Figure 4.3.1. Geometry for the irregular cavity.

We now examine the results obtained with simulated measurements containing random errors of standard deviation $\sigma = 0.6$ ^{o}C. For this test-case, the measurements of 27 sensors, located near each of the boundaries, were assumed available for the inverse analysis. The sensors were located 1.08 mm below the surface at $\eta=1$ and 1.25 mm below the surface at $\eta=N$. Figures 4.3.2.a,b show that the estimated functions are in very good agreement with the exact ones for such test-case, although some oscillations are observed in the inverse problem solution, specially near the sharp variation around 2 seconds at $\eta=N$ (see figure 4.3.2.a).
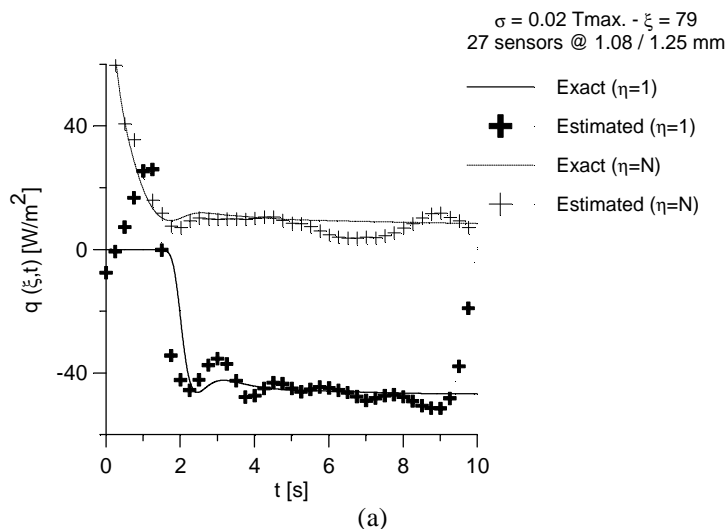

(a)
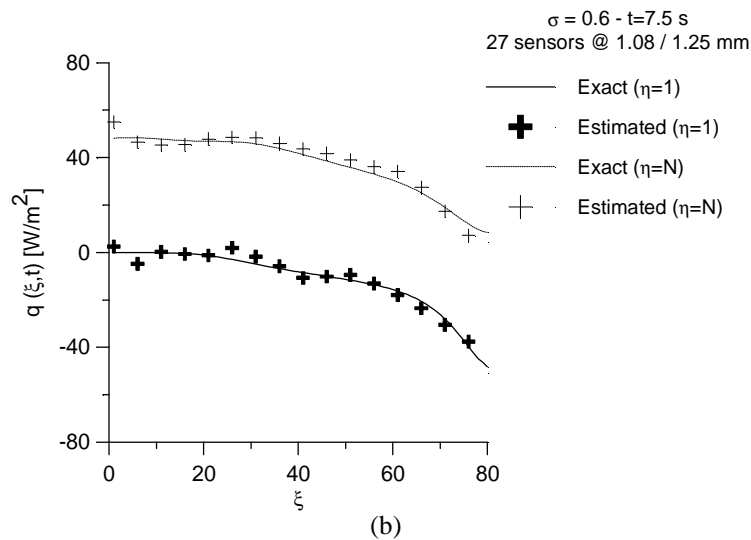Figure 4.3.2. Results for: (a) ξ=79, (b) t =7.5 seconds.

Figure 4.3.2. (Continued)..

4.4. Optimization problem in magnetohydrodynamics [54-58]

This section presents the optimization problem for achieving desired features of a melt undergoing solidification, involving the application of an external magnetic field, whose intensity and spatial distribution are obtained by the use of a Hybrid Method. The intensities of the magnets along the boundaries of the container are described as B-splines. The inverse problem is then formulated as to find the magnetic boundary conditions (the coefficients of the B-splines) in such a way that the gradients of temperature along the gravity acceleration direction are minimized.

Transient Navier-Stokes and Maxwell equations were discretized using the Finite Volume Method in a generalized curvilinear non-orthogonal coordinate system. For the phase change problems, an enthalpy formulation was used. The code was validated against analytical and numerical benchmark results with very good agreement in both cases.

First, let us demonstrate the inverse determination of the magnetic boundary conditions that create certain pre-specified flow-field within some domain. Figure 4.4.1 shows the geometry and the boundary conditions for the test cases considered here. The height and length of the square container were equal to 0.15 m. The top and bottom walls were kept thermally insulated. The left boundary was kept at a "hot" temperature while the right wall was kept at a "cold" temperature. For the first test case, there was no phase change, since the "hot" and "cold" temperatures were above the melting temperature.
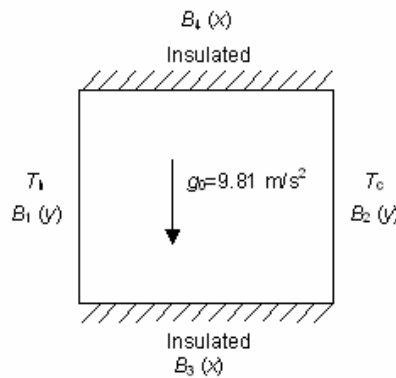

Figure 4.4.1. Geometry and boundary conditions for MHD.

The four walls were subjected to unknown magnetic field distributions whose directions were made orthogonal to each wall. In order to satisfy the magnetic flux conservation equation

$$\nabla \bullet \mathbf{B} = 0 \qquad (4.4.1)$$

The following periodic conditions were imposed

$$B_1(y) = B_2(y) \qquad (4.4.2)$$

$$B_3(x) = B_4(x) \tag{4.4.3}$$

The objective was to minimize the natural convection effects by reducing the gradient of temperature along the *y* direction, thus trying to obtain a temperature profile similar to those obtained for pure conduction. The objective function to be minimized is then formulated as:

$$F = \sqrt{\frac{1}{\#cells} \sum_{i=1}^{\#cells} \left(\frac{\partial T_i}{\partial y_i} \right)^2} \tag{4.4.4}$$

The magnetic field boundary conditions were inversely determined at either four or six points equally spaced along each of the four boundaries and interpolated using B-splines for the other points at those boundaries. The magnetic boundary conditions at $x = 0.15$ m and $y = 0.15$ m were then obtained using the periodic conditions from equation (4.4.2) and equation (4.4.3).

The fluid analyzed was silicon. For the first test case, the temperature difference T_h-T_c was set equal to 0.65 K, which gives a Rayleigh number of 10^5.

Figure 4.4.2 shows streamlines, isotherms and heat fluxes on all four boundaries, predicted without any magnetic flux applied and no phase change (left column), as well as streamlines, isotherms and heat fluxes on all four boundaries resulting from the optimized magnetic boundary conditions with six points on each boundary. One can see that the gradients of temperature in the *y* direction are reduced. Figure 4.4.3 shows the optimized magnetic field boundary conditions for $x = 0$ and $y = 0$ and figure 4.4.4 shows the convergence history of the process.
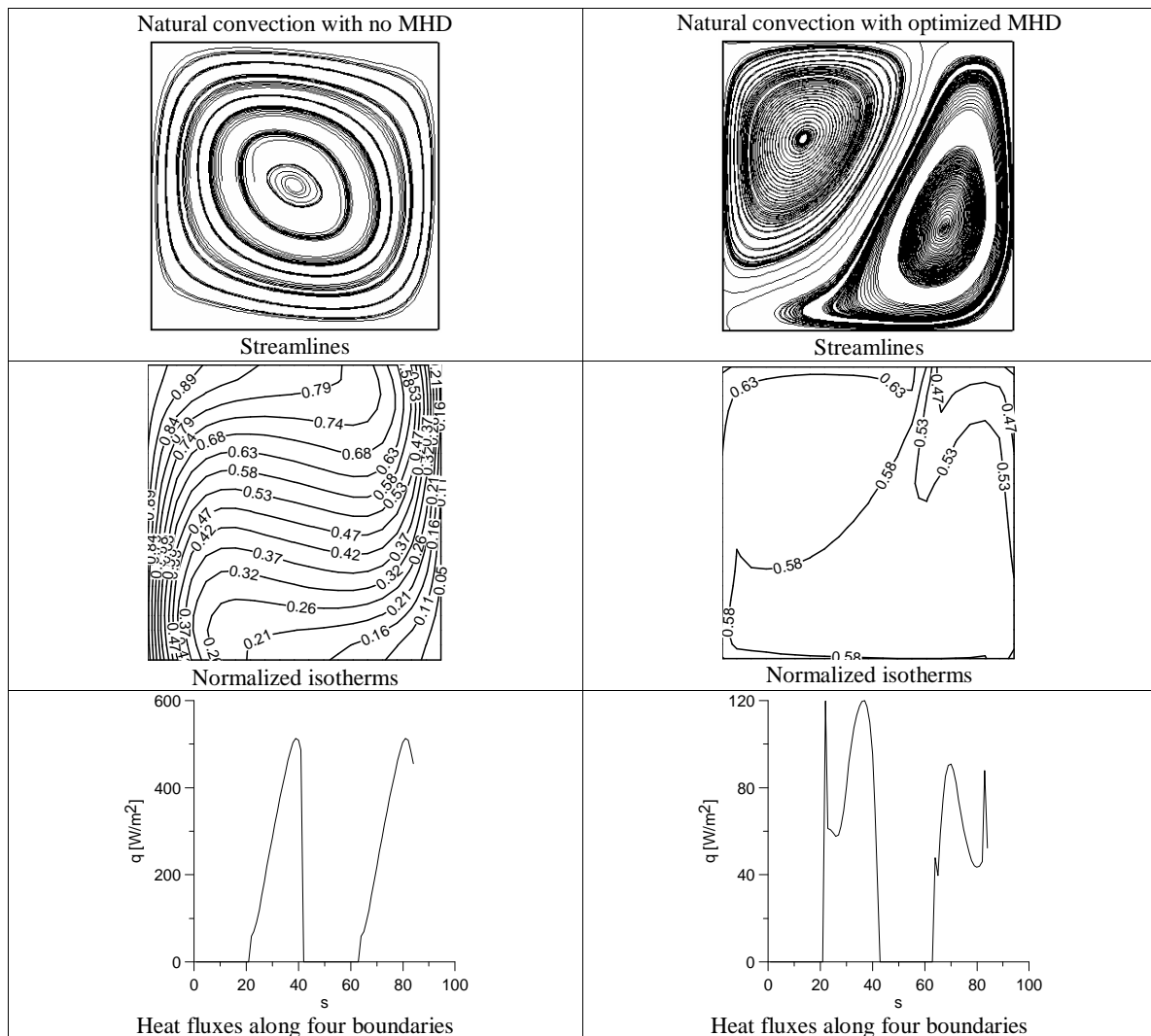

Figure 4.4.2. Natural convection; results of analysis with no magnetic field (left column) and results with optimized magnetic boundary conditions (right column) for Ra = 10^5. The parameter *s* is measured counterclockwise along the boundaries of the rectangular container starting from lower left corner.
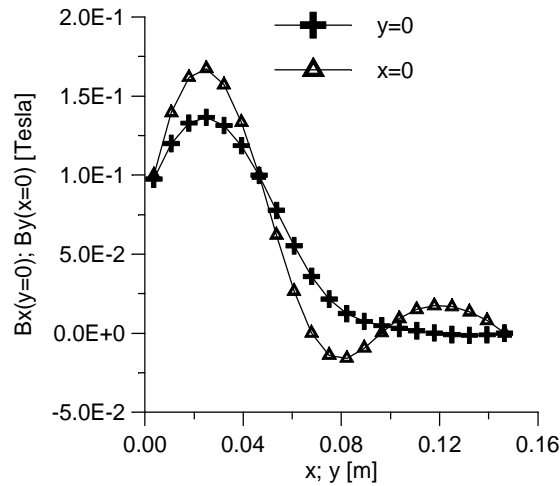
Figure 4.4.3. Natural convection; optimized magnetic boundary conditions
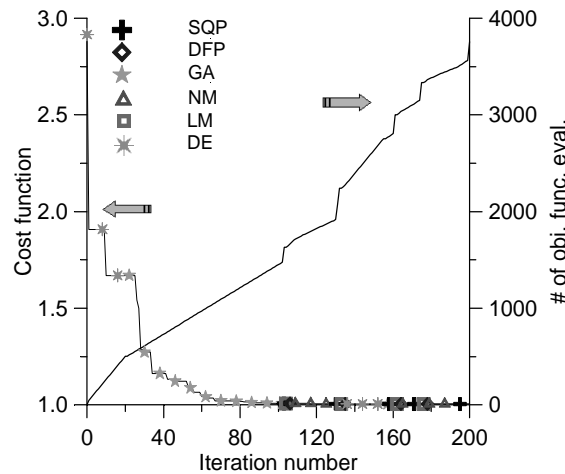at $x = 0$ and $y = 0$ with the estimation of B at six points per boundary

Figure 4.4.4. Natural convection; optimization convergence history for the
estimation of B at six points per boundary.

As a second test case, we minimized the curvature of the isotherms in a solidifying process after a pre-specified time from the start of the solidification process. The temperature difference T_h-T_c was set equal to 6.5 K (T_h = 1686. K, T_c = 1676.5 K) and the length of the square container was taken as 0.069 m, which gives a Rayleigh number of 10^5. The *solidus* and *liquidus* temperatures were equal to 1681.0 K and 1686.0 K, respectively. Thus, a mushy region exists between the phases. The initial condition was set as $T_0 = T_h$. Then, the solidifying process started at the right wall, where $T = T_c$.

Figure 4.4.5 shows the streamlines, isotherms and heat fluxes on all four boundaries for this test case without any magnetic flux applied, predicted at 500 seconds, (left column) as well as the streamlines, isotherms and heat fluxes on all four boundaries resulting from the optimization of six B-spline points for the estimation of the magnetic boundary conditions on each boundary (right column). The boundary conditions at the other points were interpolated using B-splines. One can see that the curvature of the isotherms is smaller than in the case without any magnetic fields applied.

Figure 4.4.6 shows the optimized variation of the magnetic fields orthogonal to $x = 0$ and $y = 0$ boundaries. Figure 4.4.7 shows the convergence history of the optimization process.

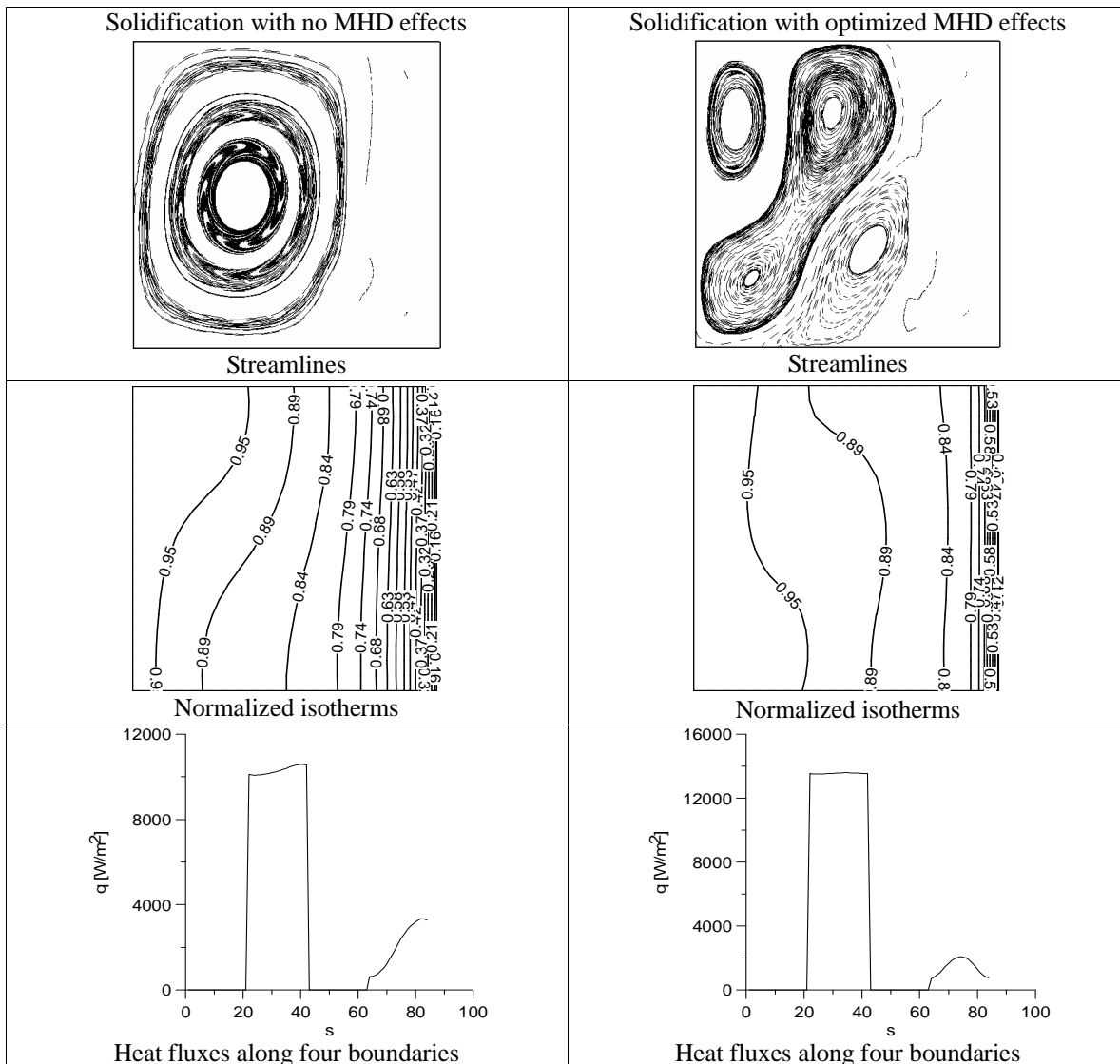| Solidification with no MHD effects	Solidification with optimized MHD effects
Streamlines	Streamlines
Normalized isotherms	Normalized isotherms
Heat fluxes along four boundaries	Heat fluxes along four boundaries

Figure 4.4.5. Natural convection with solidification; results of analysis with no magnetic field (left column) and results with optimized magnetic boundary conditions (right column) for $Ra = 10^5$. The parameter s is measured counterclockwise along the boundaries of the rectangular container starting from lower left corner.
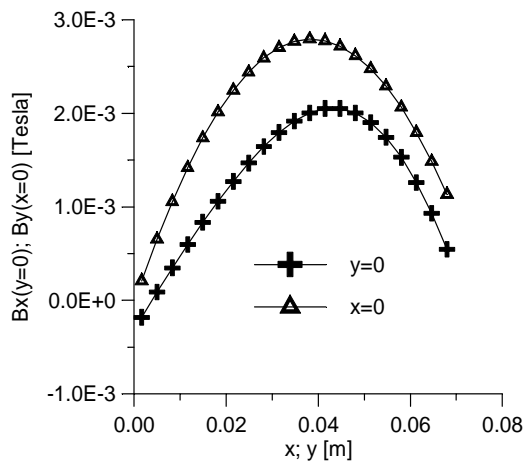
Figure 4.4.6. Optimized magnetic boundary conditions on $x = 0$ and $y = 0$ boundaries with the estimation of B at six points per boundary in case with solidification.
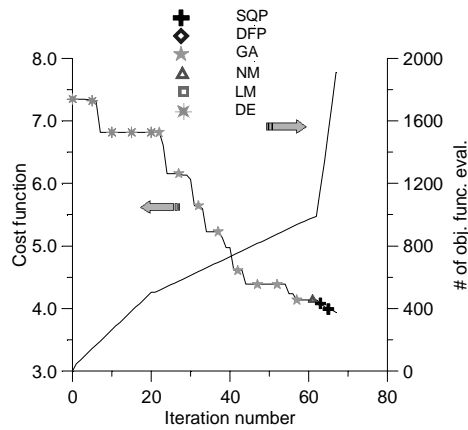
Figure 4.4.7. Optimization convergence history for the estimation of B at
six points per boundary in case with solidification.

4.5. Optimization problem in electrohydrodynamics [52,53]

In another test case we dealt with the inverse determination of the electric boundary conditions that create some pre-specified flow-field features within some region. Figure 4.5.1 shows the geometry and the boundary conditions for the configuration considered here.
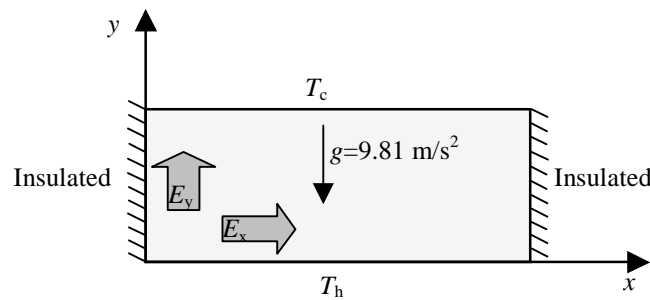

Figure 4.5.1. Geometry and boundary conditions for EHD.

The height and length of the closed container filled with the electrically conducting liquid were equal to 33.3 mm and 66.7 mm, respectively. The vertical walls were kept thermally insulated. The bottom boundary was kept at a "hot" temperature while the top wall was kept at a "cold" temperature. A slightly triangular temperature profile was applied to the bottom wall in order to create a preferential direction for the thermally induced fluid flow.

The vertical walls were subjected to unknown electric potential boundary conditions. The electric charged particles were assumed to enter the fluid from the walls where the electric potential was applied. The objective was to minimize the natural convection effects by reducing the gradient of temperature along the x direction, thus trying to obtain a temperature profile similar to those obtained for pure conduction. The objective function to be minimized was then formulated as

$$F = \sqrt{\frac{1}{\#cells} \sum_{i=1}^{\#cells} \left(\frac{\partial T_i}{\partial x_i}\right)^2}$$

(4.5.1)

The electric boundary conditions were inversely determined at six points equally spaced along each of the vertical walls and parameterized using B-splines for the other points of these boundaries. In this case we considered natural convection of gallium arsenide. The temperature difference T_h-T_c was set equal to 1.0 K, which gives a Rayleigh number of 1.9×10^4.

For the first test case, there was no phase change, since the "hot" and "cold" temperatures were above the melting temperature ($T_h = 1521.5$ K; $T_c = 1520.5$ K).

Figure 4.5.2 shows the streamlines, isotherms and heat fluxes on all four walls predicted for the first test case without any electric field applied and no phase change (left column). Figure 4.5.2 also shows the streamlines, isotherms, and heat fluxes on the four walls when using six points on each vertical wall for the estimation of the electric

boundary conditions (right column). One can see that the gradients of temperature in the *x* direction are reduced close to the top and bottom walls. One can see that the isotherms start to become horizontal which is similar to those obtained if the gravity vector were acting in the horizontal direction.
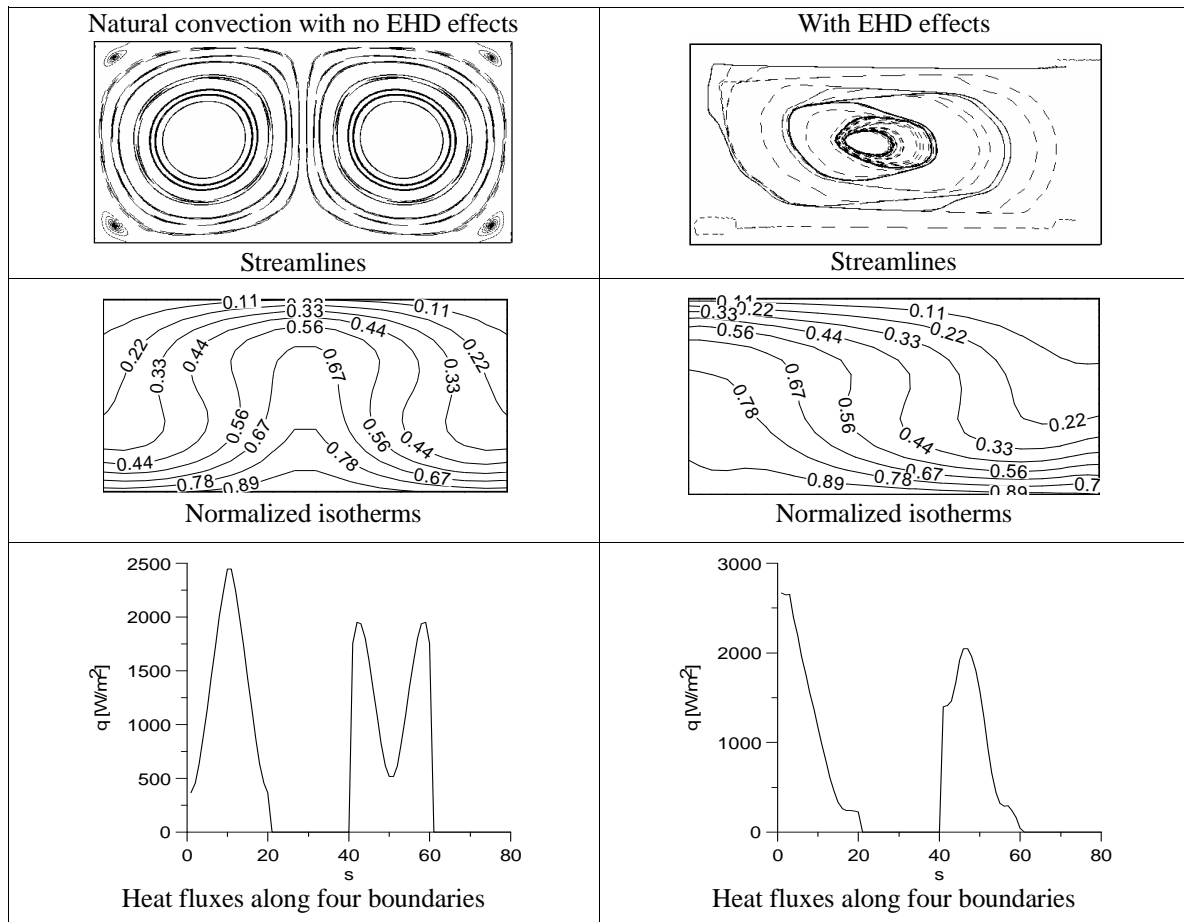
Natural convection with no EHD effects	With EHD effects
Streamlines	Streamlines
Normalized isotherms	Normalized isotherms
Heat fluxes along four boundaries	Heat fluxes along four boundaries

Figure 4.5.2. Natural convection with no solidification; results of analysis with no electric field (left column) and results of optimized electric boundary conditions (right column) for Ra = 1.9×10^4. The parameter *s* is measured counterclockwise along the boundaries of the rectangular container starting from lower left corner.

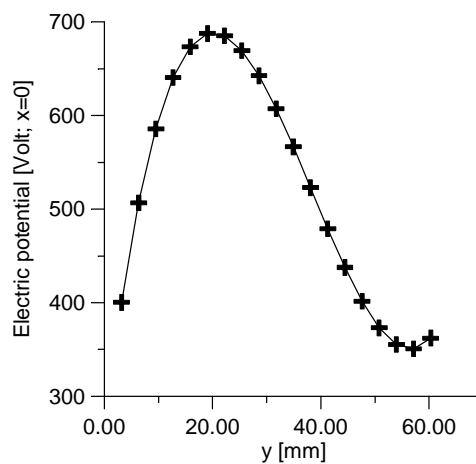Figure 4.5.3 shows the optimized electric potential and figure 4.5.4 shows the convergence history.

Figure 4.5.3. Optimized electric field potential at *x* = 0 and *y* = 0 with the estimation of E at six points per boundary.
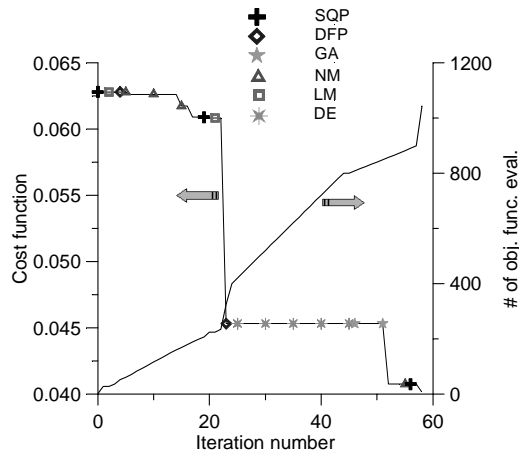
Figure 4.5.4. Optimization convergence history for the
estimation of E at six points per boundary.

In a second test case, we tried to minimize the curvature of the isotherms in a solidifying process after a pre-specified time from the start of the solidifying process. Figure 4.5.5 shows (left column) the results obtained for a Rayleigh number equal to 1.9×10^4 without any electric field applied. In this case, the "hot" and "cold" temperatures were equal to 1510.5 K and 1511.5 K, respectively. Figure 4.5.5 also shows the results obtained with an optimized electric potential acting in the horizontal direction. Note that the isotherms are smoother than those for a case without any electric field applied.

Figure 4.5.6 shows the optimized electric potential and figure 4.5.7 shows the convergence history for the hybrid optimizer.
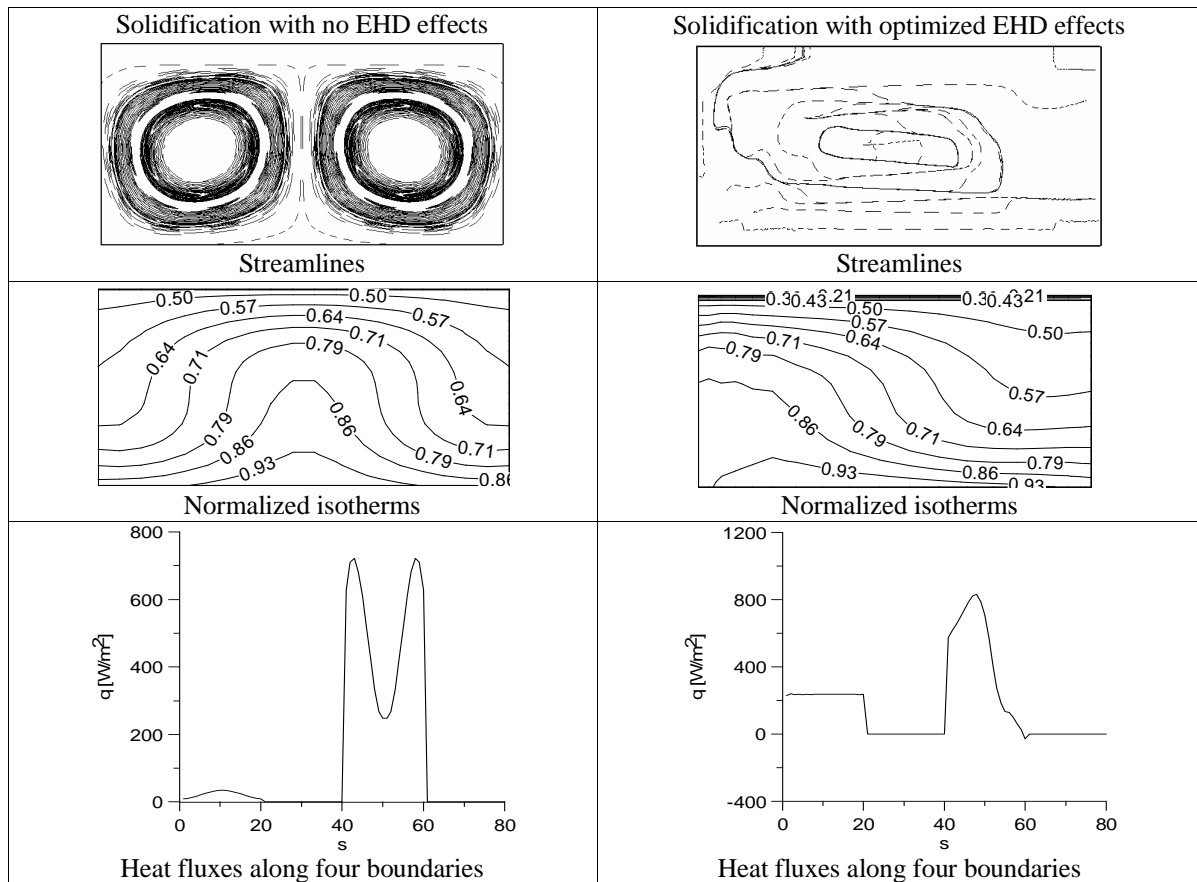

Figure 4.5.5. Natural convection with solidification; results of analysis with
no electric field (left column) and results of optimized electric boundary conditions
(right column) for Ra = 1.9×10^4. The parameter *s* is measured counterclockwise along the
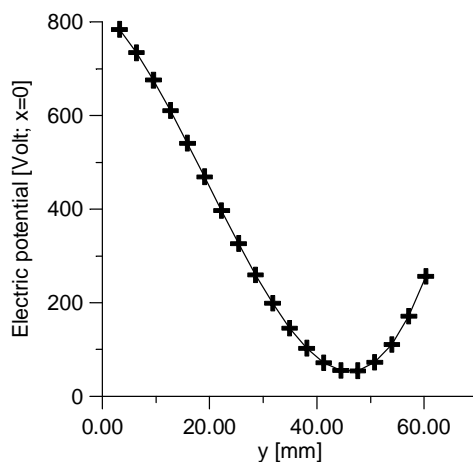boundaries of the rectangular container starting from lower left corner.

Figure 4.5.6. Solidification case: optimized electric potential boundary conditions
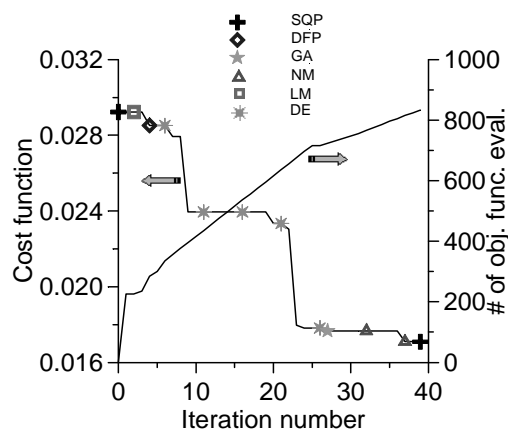at $x = 0$ with Ra = $1.9\text{x}10^4$ and estimation of E at six points.

Figure 4.5.7. Solidification case: convergence history for Ra = $1.9\text{x}10^4$
and optimization of E at six points at $x = 0$.

Acknowledgements

References

[1] Tikhonov, A. N. and Arsenin, V. Y., Solution of Ill-Posed Problems, Winston & Sons, Washington, DC (1977).
[2] Beck, J. V. and Arnold, K. J., Parameter Estimation in Engineering and Science, Wiley Interscience, New York (1977).
[3] Alifanov, O. M., Inverse Heat Transfer Problems, Springer-Verlag, New York (1994).
[4] Beck, J. V., Blackwell, B. and St. Clair, C. R., Inverse Heat Conduction: Ill-Posed Problems, Wiley Interscience, New York (1985).
[5] Alifanov, O. M., Artyukhin, E. and Rumyantsev, A., Extreme Methods for Solving Ill-Posed Problems with Applications to Inverse Heat Transfer Problems, Begell House, New York (1995).
[6] Dulikravich, G. S. and Martin, T. J., "Inverse Shape and Boundary Condition Problems and Optimization in Heat Conduction", Chapter 10 in Advances in Numerical Heat Transfer, **1**, 381-426, Minkowycz, W. J. and Sparrow, E. M. (eds.), Taylor and Francis (1996).
[7] Ozisik, M.N. and Orlande, H.R.B., Inverse Heat Transfer: Fundamentals and Applications, Taylor and Francis, New York (2000).
[8] Kurpisz, K. and Nowak, A. J., Inverse Thermal Problems, WIT Press, Southampton, UK (1995).
[9] Woodbury, K., Inverse Engineering Handbook, CRC Press, Boca Raton, 2002.

[10] Murio, D. A., The Mollification Method and the Numerical Solution of Ill-Posed Problems, Wiley Interscience, New York (1993).

[11] Trujillo, D. M. and Busby, H. R., Practical Inverse Analysis in Engineering, CRC Press, Boca Raton (1997).

[12] Beck, J.V., Sequential Methods in Parameter Estimation, 3rd International Conference on Inverse Problems in Engineering, Tutorial Session, Port Ludlow, WA.

[13] Sabatier, P. C., (ed.), Applied Inverse Problems, Springer Verlag, Hamburg (1978).

[14] Hensel, E., Inverse Theory and Applications for Engineers, Prentice Hall, New Jersey (1991).

[15] Denisov, A. M., Elements of the Theory of Inverse Problems, VSP, Netherlands (1999).

[16] Yagola A. G., Kochikov, I.V., Kuramshina, G. M. and Pentin, Y. A., Inverse Problems of Vibrational Spectroscopy, VSP, Netherlands (1999).

[17] Ramm, A. G. , Shivakumar, P.N. and Strauss, A. V. (eds.), Operator Theory and Applications, Amer. Math. Soc., Providence (2000).

[18] Morozov, V. A., Methods for Solving Incorrectly Posed Problems, Springer Verlag, New York (1984).

[19] Zubelli, J. P., An Introduction to Inverse Problems: Examples, Methods and Questions, Institute of Pure and Applied Mathematics, Rio de Janeiro, Brazil (1999).

[20] Kirsch, A., An Introduction to the Matheamtical Theory of Inverse Problems, Applied Mathematical Sciences, vol. 120, Springer, (1996).

[21] Isakov, V., Inverse Problems for Partial Differential Equations, Applied Mathematical Sciences, vol. 127, Springer, (1998).

[22] Hadamard, J., Lectures on Cauchy's Problem in Linear Differential Equations, Yale University Press, New Haven, CT (1923).

[23] Daniel, J.W., The Approximate Minimization of Functionals, Prentice-Hall, Englewood Cliffs, 1971.

[24] Jaluria, Y., Design and Optimization of Thermal Systems, McGraw Hill, 1998.

[25] Stoecker, W. F., Design of Thermal Systems, McGraw Hill, 1989.

[26] Belegundu, A.D and Chandrupatla, T. R., Optimization Concepts And Applications In Engineering, Prentice Hall, 1999.

[27] Fletcher, R., Practical Methods of Optimization, John Wiley & Sons, 2000.

[28] Powell, M.J.D., Restart Procedures for the Conjugate Gradient Method, Mathematical Programming, vol. 12, pp.241-254, 1977.

[29] Fletcher, R. and Reeves, C.M., Function Minimization by Conjugate Gradients, Comput. Journal, vol.7, pp.149-154, 1964.

[30] Hestenes, M.R. and Stiefel, E., Method of Conjugate Gradients for Solving Linear Systems, J. Res. Nat. Bur. Standards Sect. B, vol. 49, pp.409-436, 1952.

[31] Polak, E., Computational Methods in Optimization, Academic Press, New York, 1971.

[32] Beale, E.M.L., A Derivation of Conjugate Gradients, Numerical Methods for Nonlinear Optimization, F. A. Lootsma (editor), pp. 39-43, 1972

[33] Davidon, W.C., Variable Metric Method for Minimization, Argonne Natl. Lab., ANL-5990, 1959.

[34] Fletcher, R. and Powell, M.J.D., A Rapidly Convergent Descent Method for Minimization, Computer J., vol. 6, pp. 163-168, 1963.

[35] Broyden, C.G., A Class of Methods for Solving Nonlinear Simultaneous Equations, Math. Comp., vol. 19, pp. 577-593, 1965.

[36] Broyden, C.G., Quasi-Newton Methods and Their Applications to Function Minimization, Math. Comp., vol. 21, pp. 368-380, 1967.

[37] Levenberg, K., "A Method for the Solution of Certain Non-linear Problems in Least Squares", Quart. Appl. Math., **2**, 164-168, 1944.

[38] Marquardt, D. W., "An Algorithm for Least Squares Estimation of Nonlinear Parameters", J. Soc. Ind. Appl. Math, **11**, 431-441, 1963.

[39] Bard, Y. B., Nonlinear Parameter Estimation, Acad. Press, New York, 1974.

[40] Dennis, J. and Schnabel, R., Numerical Methods for Unconstrained Optimization and Nonlinear Equations, Prentice Hall, 1983.

[41] Moré, J. J., "The Levenberg-Marquardt Algorithm: Implementation and Theory", in Numerical Analysis, Lecture Notes in Mathematics, Vol. 630, G. A. Watson (ed.), Springer-Verlag, Berlin, 105-116, 1977.

[42] Corana, A., Marchesi, M., Martini, C. and Ridella, S., Minimizing Multimodal Functions of Continuous Variables with the 'Simulated Annealing Algorithm', ACM Transactions on Mathematical Software, vol. 13, pp. 262-280, 1987.

[43] Goffe, W.L., Ferrier, G.D. and Rogers, J., Global Optimization of Statistical Functions with Simulated Annealing, Journal of Econometrics, vol. 60, pp. 65-99, 1994.

[44] Storn, R. and Price, K.V., Minimizing the Real Function of the ICEC'96 Contest by Differential Evolution, IEEE Conf. On Evolutionary Comutation, pp. 842-844, 1996.

[45] Goldberg, D.E., Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Pub Co, 1989.

[46] Deb, K., Multi-Objective Optimization Using Evolutionary Algorithms, John Wiley & Sons, 2002

[47] Kennedy, J. and Eberhart, R.C., Particle Swarm Optimization, Proceedings of the 1995 IEEE International Conference on Neural Networks, vol. 4, pp. 1942-1948, 1995.

[48] Kennedy, J. Small Worlds and Mega-Minds: Effects of Neighborhood Topology on Particle Swarm Performance, Proceedings of the 1999 Congress of Evolutionary Computation, vol. 3, pp. 1931-1938, IEEE Press, 1999.

[49] Naka, S., Yura, T.G. and Fukuyama, T., Practical Distribution State Estimation using Hybrid Particle Swarm Optimization, Proceedings IEEE Power Engineering Society, Winter Meeting, January 28-February 1st, 2001, Columbus, OH.

[50] Eberhart, R., Shi, Y. and Kennedy, J., Swarm Intelligence, Morgan Kaufmann, 2001.

[51] Dorigo, M. and Stützle, T., Ant Colony Optimization, MIT Press, 2004.

[52] Colaço, M.J., Dulikravich, G.S. and Martin, T. J., Optimization of Wall Electrodes for Electro-Hydrodynamic Control of Natural Convection Effects During Solidification, In: ASME International Mechanical Engineering Congress & Exposition, November, 2003.

[53] Colaço, M.J., Dulikravich, G.S. and Martin, T. J., Optimization of Wall Electrodes for Electro-Hydrodynamic Control of Natural Convection Effects During Solidification, Materials and Manufacturing Processes, in press, 2004.

[54] Colaço, M.J., Dulikravich, G.S. and Martin, T.J. Reducing Convection Effects in Solidification by Appling Magnetic Fields Having Optimized Intensity Distribution (Keynote Lecture), In: ASME Summer Heat Transfer Conference, July, 2003.

[55] Dulikravich, G.S., Colaço, M.J., Martin, T.J. and Lee, S., Magnetized Fiber Orientation and Concentration Control in Solidifying Composites, In: Symposium on Materials Processing Under the Influence of Electrical and Magnetical Fields, TMS Annual Meeting, San Diego, CA, 2003.

[56] Dulikravich, G.S., Colaço, M.J., Martin, T.J. and Lee, S., An Inverse Method Allowing User-Specified Layout of Magnetized Micro-Fibers in Solidifying Composites, Journal of Composite Materials, vol. 37, no. 15, pp. 1351-1365, 2003.

[57] Dulikravich, G.S., Colaço, M.J., Dennis, B.H., Martin, T.J. and Lee, S., Optimization of Intensities, and Orientations of Magnets Controlling Melt Flow During Solidification, In: Symposium on Materials Processing Under the Influence of Electrical and Magnetical Fields, TMS Annual Meeting, San Diego, CA, 2003.

[58] Dulikravich, G.S., Colaço, M.J., Dennis, B.H., Martin, T.J. and Lee, S., Optimization of Intensities, and Orientations of Magnets Controlling Melt Flow During Solidification, Materials and Manufacturing Processes, vol. 19, iss. 4, 2004.

[59] Colaço, M.J., Orlande, H.R.B., Dulikravich, G.S. and Rodrigues, F. A., A Comparison of Two Solution Techniques for the Inverse Problem of Simultaneously Estimating the Spatial Variations of Diffusion, In: ASME International Mechanical Engineering Congress & Exposition, November, 2003.

[60] Colaço, M.J., Orlande, H.R.B., Inverse Forced Convection Problem Of Simultaneous Estimation Of Two Boundary Heat Fluxes In Irregularly Shaped Channels, Numerical Heat Transfer, Part A, Vol. 39, pp. 737-760 , 2001

[61] Colaço, M.J., Orlande, H.R.B, Inverse natural convection problem of simultaneous estimation of two boundary heat fluxes in irregular cavities,