Estimation of the Thermophysical Properties of the Soil together with Sensors' Positions by Inverse Problem

Salwa Mansour, Edouard Canot, Renaud Delannay, Ramiro Javier March, José Agustin Cordero, Juan Carlos Ferreri

ARPHYMAT Interdisciplinary Project

ECOS-Sud 2015 Report

Estimation of the Thermophysical Properties of the Soil together with Sensors' Positions by Inverse Problem

Salwa Mansour, INRIA, Rennes, France

Édouard Canot, IRISA, Rennes, France

Renaud Delannay, IPR, Rennes, France

Ramiro March, CReAAH, Rennes, France

Agustin Cordero, Buenos Aires, Argentina

Juan Carlos Ferreri, Buenos Aires, Argentina

Apr 30, 2015

Contents

The work presented here is motivated by the studies of agricultural and archaeological soils. A systematic application of numerical modeling in a particular field of agriculture and archaeology which is the study of seed germination and archaeological hearths is presented. The report is basically divided into two main parts. In the first part, we introduce a numerical strategy in both 1D and 3D axisymmetric coordinate systems to estimate the thermophysical properties of the soil (volumetric heat capacity $(\rho C)_s$, thermal conductivity λ_s and porosity ϕ) of a saturated porous medium where a phase change problem (liquid/vapor) appears due to intense heating from above. Usually ϕ is the true porosity, however when the soil is not saturated (which should concern most cases), ϕ may be taken equal to the part of water in the pores. This is of course an approximation which is correct for the energy balance but which neglects the capillary forces and the migration flow of the liquid inside the porous media; a complete model of such an unsaturated model is out of the scope of this work. In the second part, we present a similar strategy to approximate the value of diffusivity α and the sensors' positions in the case of dry porous medium where no phase change is present.

The investigation of the thermal properties of the soil can have significant practical consequences such as evaluation of optimum conditions for plant growth and development and can be utilized for the control of thermal-moisture regime of soil in the field [LSO13]. These properties influence how energy is partitioned in the soil profile so the ability to monitor them is a tool to manage the soil temperature regime that affects seed germination and growth. It can also provide information about the use of fire by ancient civilizations whether for cooking or heating... [Eck07]. The inverse problem, presented here, consists of the estimation of thermophysical properties of the soil knowing the heating history curves at selected points of the altered soil [MCM12]. In general, the mathematical formulation of inverse problems leads to models that are typically ill-posed [EK05]. In such problems, we usually minimize a discrepancy between some experimental data and some model data [DS83]. In our problem, we use the least square criterion in which the sensitivity coefficients appear and where we try to minimize the discrepancy function which is expressed as the norm of the difference between the experimental temperature and the numerical data obtained by our approximated model [Bjö90]. The system composed of the energy equation together with other boundary initial problems resulting from differentiating the basic energy equation with respect to the unknown parameters must be solved [MMS08].

At the stage of numerical computations, the Damped Gauss Newton method and the Levenberg Marquardt algorithm are used to minimize the least square criterion; that requires the solution of a system of highly nonlinear ordinary differential equations. It is important to note that in our new configuration, the solution is reached after taking into consideration the temperature history at selected points of the domain and at different time steps which was not the case in [MCM12] where the authors reached the solution by taking the temperature history at the final time only and at all the points in the computational domain. Also, we used a scaling technique which sounds to be unevitable since our uknown parameters have different order of magnitude. This global approach is based on the method of lines, where time and space discretizations are considered separately. The space discretization is done using a vertex-centered finite volume method; the discretization in time is done via an ODE solver that uses a BDF scheme and a modified Newton method to deal with the high nonlinearity of the system. The advantage of our configuration to that presented by [MCM12] is that we propose a model which is more realistic and closer to the experimental setup i.e. our synthetic data consists of the calculation of the temperature at few sensors (around 5) during the whole heating duration. At the end of this report, we show the advantage of the inverse problem in 3D axisymmetric coordinate system to that in 1D coordinate system.

Identification of the thermophysical properties of the soil during phase change by inverse problem in 1D

1 Forward problem (1D)

The physical problem consists of heating the soil by a fire. To model this problem, we replace the soil by a perfect porous medium with constant and uniform properties heated from above by a constant temperature T_c (temperature of the fire between 300 C and 700 C). T_c must be greater than T_v (the evaporation or phase change temperature which is normally 100 C). In order to model the heat conduction transfer in the soil, we use the energy equation and we neglect the convection term so that the energy conservation equation for the unknown temperature T is expressed as:

$$(\rho C)_e \frac{\partial T}{\partial t} = \text{div}\left(\lambda_e \vec{\text{grad}} T\right) \tag{1}$$

with the following initial and boundary conditions:

$$T(x,0) = T_0(x) \text{ in } \Omega$$

$$T(x,t) = T^D(x,t) \text{ on } \Gamma^D \times (0, t_{end}] \text{ (Dirichlet)}$$

$$\nabla T(x,t).\nu = 0 \text{ on } \Gamma^N \times (0, t_{end}] \text{ (Neumann)}$$

where T represents the temperature, T_0 is the initial temperature at $t_0 = 0$, T^D is T_c at the fire and T_0 elsewhere; ρ is the density, C is the specific heat capacity, λ is the conductivity, ϕ is the porosity, the subscripts e, f and s indicate the equivalent parameters of the medium, the properties of the fluid and the porous matrix properties respectively. ν indicates the outward unit normal vector along the boundary of Ω. Note that the thermophysical properties of the fluid are temperature dependent and that is why the problem is highly nonlinear.

The effective volumetric heat capacity and the effective conductivity are defined by the equations (2) and (3):

$$(\rho C)_e = \phi(\rho C)_f + (1-\phi)(\rho C)_s \tag{2}$$

$$\lambda_e = \lambda_f \left[\phi + \frac{(1-\phi)\,\beta}{\frac{\lambda_f}{\lambda_s} + \Phi}\right] \tag{3}$$

The effective conductivity in equation (3) is calculated using Kunii and Smith model [MHB+12]

To avoid the tracking of the interface of the phase change problem (liquid/vapor) which appears when the water existing in the soil turns into gas, the Apparent Heat Capacity (AHC) method is used (see chapter 3).

1.1 Numerical strategy

We need to solve the heat diffusion equation (PDE) so we choose the method of lines which is a way of approximating PDEs by ODEs where space and time discretizations are considered separately. The spatial discretization is performed using the vertex-centered finite volume method which conserves the mass locally and preserves continuity of fluxes. To apply the spatial discretization, the computational domain is divided into a finite volume grid or mesh with equal length $h = \Delta x$ as shown in figure 1. In fact, the spatial variable is discretized into N discretization points and each state variable T is transformed into N variables corresponding to its value at each discretization point. It is important to mention that the end points of each interval ($x_{i-\frac{1}{2}}$ and $x_{i+\frac{1}{2}}$) are computed as exactly the middle of two consecutive nodes, i.e.

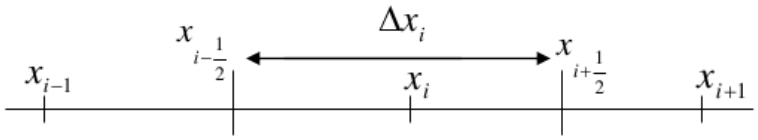

Figure 1: 1 D control volume

$x_{i+\frac{1}{2}} = \frac{1}{2}(x_i + x_{i+1})$. The spatial derivatives are approximated by using a finite volume formula on three points so we end up with a semi-discrete system of ODEs which can be written in the form:

$$\frac{dT}{dt} = B(T)T \tag{4}$$

The ODE coefficient matrix $B(T)$ has a tridiagonal structure due to the 1-D Laplacian discretization. Some tries showed that our ODE system becomes more and more stiff as h becomes smaller. The difficulty with stiff problems is the prohibitive amount of computer time required for their solution by classical ODE solution methods, such as the popular explicit Runge-Kutta and Adams methods. The reason is the excessively small step sizes that these methods must use to satisfy stability requirements due to the high non-linearity of the apparent capacity of the fluid C_f [MCM09]. For this reason, we use an implicit ODE solver (Backward Differentiation Formula) which possesses the property of stability and therefore does not suffer from the stability step size constraint [Hin93]. The BDF implicit scheme requires the calculation of a Jacobian matrix which is calculated and generated by a Computer Algebra System (CAS, Maple or Maxima) and then stored in a sparse format. Note that the numerical calculation is performed with `ddebdf` routine of the SLATEC Fortran library which was modified to use the UMFPACK sparse linear solver [Dav04]. The ODE solver performs time integration by adjusting automatically the time step in the BDF scheme and all these primary libraries are grouped in the easy-to-use MUESLI library [Can].

2 Inverse problem (1D)

In order to solve the parametric inverse problem consisting of finding the volumetric heat capacity $(\rho C)_s$, the conductivity λ_s and the porosity ϕ of the saturated soil, it is necessary to know the values of temperature T_{gi}^f at selected points (sensors) of the porous medium domain for times t^f: $T_{gi}^f = T_g(x_i, t^f)$ where $i = 1, 2, ..., M$ and $f = 1, 2, ..., F$. M and F are the total number of sensors and time steps respectively. We use the least squares criterion to solve this inverse problem so we try to find the soil parameters that minimize the error function which is defined by:

$$S((\rho C)_s, \phi, \lambda_s) = \frac{1}{2}\|T_i^f - T_{gi}^f\|_2^2 \tag{5}$$

where $T_i^f = T(x_i, t^f)$ are the temperatures being the solution of the direct problem for the assumed set of parameters at the point x_i, $i = 1, 2, ..., M$ for the time t^f, $f = 1, 2, ..., F$ and T_{gi}^f is the measured temperature at the same point x_i for time t^f. It is important to mention that the authors in [MCM12] calculated the temperature at the final time only and at all the points of the domain.

2.1 Parameter scaling

In [MCM12], the authors mentioned that the heat equation is not sensitive to the heat capacity in comparison to the other parameters and that it stagnate at its initial guess. In reality, it is due to the fact that the parameters we are investigating are of very different magnitudes so it is necessary to perform parameter scaling or otherwise many searches would not converge. Gradient search techniques generally require parameter scaling to obtain efficient search convergence [DS83].

The first basic rule of scaling is that the variables of the scaled problem should be of similar magnitude and of order unity in the region of interest. If typical values of the variables are known, a problem can be transformed so that the variables are all of the same order of magnitude. The most commonly used transformation is of the form :

$$p = D\tilde{p} \tag{6}$$

where p is the vector of original variables p_j, $\tilde{p}$ is the vector of scaled variables $\tilde{p_j}$ and D is a constant diagonal matrix whose diagonal elements are set to be equal to the order of magnitude of its corresponding variable. We have to keep in mind that when the variables are scaled then the derivatives of the objective function are also scaled [DS83].

2.2 Method of resolution

To illustrate the method of resolution, we define the following vectors:

$$T_g = \begin{pmatrix} T_{g1}^1 \\ \dots \\ T_{g1}^F \\ T_{g2}^1 \\ \dots \\ T_{g2}^F \\ \dots \\ T_{gM}^1 \\ \dots \\ T_{gM}^F \end{pmatrix} \quad g(p^{(k)}) = \begin{pmatrix} T_1^{1,(k)} \\ \dots \\ T_1^{F,(k)} \\ T_2^{1,(k)} \\ \dots \\ T_2^{F,(k)} \\ \dots \\ T_M^{1,(k)} \\ \dots \\ T_M^{F,(k)} \end{pmatrix} \quad p^{(k)} = \begin{pmatrix} (\rho C)_s^{(k)} \\ \lambda_s^{(k)} \\ \phi^{(k)} \end{pmatrix}$$

and

$$r(p^{(k)}) = g(p^{(k)}) - T_g$$

where $r(p^{(k)})$ is the residual vector at the iteration k and $N = M \times F$. In [MCM12], the authors used the Gauss-Newton method to solve the nonlinear least square problem which fails to converge in our case when the temperature is calculated at few sensors only but for the whole simulation time due to some lack of information. Moreover, Gauss-Newton method is not locally convergent on problems that are very non-linear or have very large residuals which is the case in our problem. Since the performance of the Gauss-Newton method is strongly dependent on the residual size, we adopted the use of the Damped Gauss Newton method which is an improved version of the Gauss-Newton algorithm [DS83]. Damped Gauss-Newton method is known to be locally convergent on almost all nonlinear least squares problems including large residual or very nonlinear problems [Bjö90].

The cost function $S((\rho C)_s, \phi, \lambda_s)$ defined by equation (5) can be re-written as:

$$\mathrm{S}(p^{(k)}) = \frac{1}{2} r(p^{(k)})^t r(p^{(k)}) \tag{7}$$

Such necessary condition for the minimization of $S(p^{(k)})$ can be represented in equation (8):

$$\nabla S(p^k) = J(p^k) r(p^k) = 0 \tag{8}$$

where $J(p^k)_{i,j} = \dfrac{\partial r_i(p^k)}{\partial p_j^k}$, $i = 1, 2, ..., N$ and $j = 1, 2, 3$. The sensitivity matrix, $J(p^k)$ is defined by:

$$J(p^{(k)}) = \begin{pmatrix} W_1^{1,(k)} & R_1^{1,(k)} & Z_1^{1,(k)} \\ ... & ... & ... \\ W_1^{F,(k)} & R_1^{F,(k)} & Z_1^{F,(k)} \\ W_2^{1,(k)} & R_2^{1,(k)} & Z_2^{1,(k)} \\ ... & ... & ... \\ W_2^{F,(k)} & R_2^{F,(k)} & Z_2^{F,(k)} \\ ... & ... & ... \\ W_M^{1,(k)} & R_M^{1,(k)} & Z_M^{1,(k)} \\ ... & ... & ... \\ W_M^{F,(k)} & R_M^{F,(k)} & Z_M^{F,(k)} \end{pmatrix} \tag{9}$$

The elements of the sensitivity matrix are called the Sensitivity Coefficients. The sensitivity coefficient $J_{i,j}^f$ is thus defined as the first derivative of the estimated temperature at position i and time f with respect to the unknown parameter p_j [OO00], that is,

$$J_{i,j}^f = \frac{\partial T_i^f}{\partial p_j} \tag{10}$$

where $W_i^{f,(k)} = \dfrac{\partial T_i^f}{\partial (\rho C)_s}|_{(\rho C)_s = (\rho C)_s^{(k)}}$, $R_i^{f,(k)} = \dfrac{\partial T_i^f}{\partial \phi}|_{\phi = \phi^{(k)}}$ and $Z_i^{f,(k)} = \dfrac{\partial T_i^f}{\partial \lambda_s}|_{\lambda_s = \lambda_s^{(k)}}$. The Damped Gauss Newton algorithm iteratively finds the minimum of S. Starting with an initial guess $p^{(0)}$ for the minimum, the method proceeds by the iterations:

$$p^{(k+1)} = p^{(k)} + m^{(k)} \tag{11}$$

$m^{(k)}$ is called the increment vector and is defined by:

$$m^{(k)} = -\alpha_k \left[J(p^{(k)})^t J(p^{(k)}) \right]^{-1} J(p^{(k)})^t r(p^k) \tag{12}$$

α_k is the damping parameter ($0 < \alpha_k \leq 1$). An optimal value of α_k could be obtained using a line search algorithm [DS83]; in our case, we used trial and error to find a suitable constant damping parameter.

2.3 Governed Equations

In the following, we present the heat equation together with the three sensitivity equations resulting from the differentiation of the heat diffusion equation (1) with respect to the soil parameters p_j.

$$\frac{\partial}{\partial p_j} \left[(\rho C)_e \frac{\partial T}{\partial t} \right] = \text{div} \left(\frac{\partial}{\partial p_j} \left[\lambda_e \vec{\text{grad}} T \right] \right) \tag{13}$$

which leads to the general sensitivity equation below:

$$(\rho C)_e(T) \frac{\partial U_j(x,t)}{\partial t} + \frac{d(\rho C)_e(T)}{dp_j} \frac{\partial T(x,t)}{\partial t} =$$
$$\text{div} \left(\lambda_e(T) \vec{\text{grad}} U_j(x,t) \right) + \text{div} \left(\frac{d\lambda_e(T)}{dp_j} \vec{\text{grad}} T(x,t) \right) \tag{14}$$

where $U_j = \partial T/\partial p_j$. The general sensitivity equation is accompanied with the following boundary and initial conditions:

$$t = 0: \quad U_j(x,0) = U_{j0} = 0 \qquad \text{in } \Omega$$

$$U_j(x,t) = U_j^D(x,t) \qquad \text{on } \Gamma^D \times (0, t_{end}] \text{ (Dirichlet)}$$

$$U_j(x,t).\nu = U_j^N(x,t) \qquad \text{on } \Gamma^N \times (0, t_{end}] \text{ (Neumann)}$$

The derivative of $(\rho C)_e$ and λ_e with respect to each soil parameter is given by:

$$\begin{aligned}
\frac{d(\rho C)_e}{d(\rho C)_s} &= \frac{d(\phi(\rho C)_f)}{dT}.\frac{dT}{d(\rho C)_s} + (1-\phi) \\
&= \phi\rho_f \frac{dC_f}{dT}.\frac{dT}{d(\rho C)_s} + \phi C_f \frac{d\rho_f}{dT}\frac{dT}{d(\rho C)_s} + (1-\phi) \\
&= \phi\rho_f \left[(C_v - C_l)\frac{d\sigma}{dT} + L\frac{d^2\sigma}{dT^2}\right].\frac{dT}{d(\rho C)_s} + \phi C_f(\rho_v - \rho_l)\frac{d\sigma}{dT}.\frac{dT}{d(\rho C)_s} + (1-\phi) \\
&= \phi\rho_f \left[(C_v - C_l)\frac{d\sigma}{dT} + L\frac{d^2\sigma}{dT^2}\right].W + \phi C_f(\rho_v - \rho_l)\frac{d\sigma}{dT}.W + (1-\phi) \qquad (15)
\end{aligned}$$

$$\begin{aligned}
\frac{d(\rho C)_e}{d\phi} &= \frac{d(\phi(\rho C)_f}{d\phi} + \frac{d((1-\phi)(\rho C)_s)}{d\phi} \\
&= (\rho C)_f + \phi\frac{d((\rho C)_f)}{dT}\frac{dT}{d\phi} - (\rho C)_s \\
&= (\rho C)_f - (\rho C)_s + \phi\rho_f \left[(C_v - C_l)\frac{d\sigma}{dT} + L\frac{d^2\sigma}{dT^2}\right] Z \\
&+ \phi C_f(\rho_v - \rho_l)\frac{d\sigma}{dT}.Z \qquad (16)
\end{aligned}$$

$$\begin{aligned}
\frac{d(\rho C)_e}{d\lambda_s} &= \frac{d(\rho_C)_e}{dT}\frac{dT}{d\lambda_s} \\
&= \frac{d(\phi(\rho C)_f)}{dT}.\frac{dT}{d\lambda_s} \\
&= \phi\rho_f \left[(C_v - C_l)\frac{d\sigma}{dT} + L\frac{d^2\sigma}{dT^2}\right] R + \phi C_f(\rho_v - \rho_l)\frac{d\sigma}{dT}.R \qquad (17)
\end{aligned}$$

$$\begin{aligned}
\frac{d\lambda_e}{d(\rho C)_s} &= \frac{d\lambda_e}{dT}\frac{dT}{d(\rho C)_s} \\
&= \frac{\lambda_s \frac{d\lambda_f}{dT}(\phi\lambda_s + (1-\phi)\lambda_f) - (1-\phi)\frac{d\lambda_f}{dT}\lambda_f\lambda_s}{(\phi\lambda_s + (1-\phi)\lambda_f)^2}.W \qquad (18)
\end{aligned}$$

$$\frac{d\lambda_e}{d\phi} = \frac{E}{[\phi\lambda_s + (1-\phi)\lambda_f]^2} \qquad (19)$$

where

$$E = [\lambda_s \frac{d\lambda_f}{dT}\frac{dT}{d\phi}][\phi\lambda_s + (1-\phi)\lambda_f] - [\lambda_s - \lambda_f + (1-\phi)\frac{d\lambda_f}{dT}\frac{dT}{d\phi}]\lambda_f\lambda_s$$

$$\frac{d\lambda_e}{d\lambda_s} = \frac{F}{[\phi\lambda_s + (1-\phi)\lambda_f]^2} \tag{20}$$

where

$$F = [\lambda_s \frac{d\lambda_f}{dT}\frac{dT}{d\lambda_s} + \lambda_f][\phi\lambda_s + (1-\phi)\lambda_f] - [\phi + (1-\phi)\frac{d\lambda_f}{dT}\frac{dT}{d\lambda_s}]\lambda_f\lambda_s$$

and

$$\frac{d\lambda_f}{dT} = (\lambda_v - \lambda_l)\frac{d\sigma}{dT}$$

2.3.1 Elimination of the approximation used by [MCM12]

In order to determine the sensitivity coefficients (W, R and Z) appearing in the sensitivity matrix, we must solve the three sensitivity equations without using the approximation:

$$\text{div}\left(\lambda_e \vec{\text{grad}}T\right) \approx \lambda_e \,\text{div}\left(\vec{\text{grad}}T\right) \tag{21}$$

used by [MCM12] and which allowed the authors to write:

$$\frac{(\rho C)_e}{\lambda_e}\frac{\partial T}{\partial t} = \text{div}\left(\vec{\text{grad}}T\right) \tag{22}$$

This approximation leads to an approximated sensitivity matrix (Jacobian) and thus the problem needs more iterations to reach the required solution. Differentiating with respect to $(\rho C)_s$, ϕ and λ_s respectively:

$$\begin{aligned}
&\frac{\partial W}{\partial t} + \frac{1}{(\rho C)_e}\left[\phi\rho_f\left[(C_v - C_l)\frac{d\sigma}{dT} + L\frac{d^2\sigma}{dT^2}\right]W\right. \\
&\quad \left. + \phi C_f(\rho_v - \rho_l)\frac{d\sigma}{dT}W + (1-\phi)\right]\frac{dT}{dt} - \frac{1}{(\rho C)_e}\frac{\partial}{\partial x}\left(\lambda_e\frac{\partial W}{\partial x}\right) \\
&\quad - \frac{1}{(\rho C)_e}\frac{\partial}{\partial x}\left[\left(\frac{\lambda_s\frac{d\lambda_f}{dT}(\phi\lambda_s + (1-\phi)\lambda_f) - (1-\phi)\frac{d\lambda_f}{dT}\lambda_f\lambda_s}{(\phi\lambda_s + (1-\phi)\lambda_f)^2}\right.\right. \\
&\quad \left.\left. W\right)\frac{\partial T}{\partial x}\right]
\end{aligned} \tag{23}$$

$$\begin{aligned}
\frac{\partial R}{\partial t} &+ \frac{1}{(\rho C)_e}\left[(\rho C)_f - (\rho C)_s + \phi\rho_f\left[(C_v - C_l)\frac{d\sigma}{dT} + L\frac{d^2\sigma}{dT^2}\right]R\right. \\
&\left. + \phi C_f(\rho_v - \rho_l)\frac{d\sigma}{dT}R\right]\frac{dT}{dt} - \frac{1}{(\rho C)_e}\frac{\partial}{\partial x}\left(\lambda_e\frac{\partial R}{\partial x}\right) \\
&- \frac{1}{(\rho C)_e}\frac{\partial}{\partial x}\left[\frac{E}{(\phi\lambda_s + (1-\phi)\lambda_f)^2}\frac{\partial T}{\partial x}\right]
\end{aligned} \tag{24}$$

$$\begin{aligned}
\frac{\partial Z}{\partial t} &+ \frac{1}{(\rho C)_e}\left[\phi\rho_f\left[(C_v - C_l)\frac{d\sigma}{dT} + L\frac{d^2\sigma}{dT^2}\right]Z + \phi C_f(\rho_v - \rho_l)\frac{d\sigma}{dT}Z\right]\frac{dT}{dt} \\
&- \frac{1}{(\rho C)_e}\frac{\partial}{\partial x}\left(\lambda_e\frac{\partial Z}{\partial x}\right) - \frac{1}{(\rho C)_e}\frac{\partial}{\partial x}\left(\frac{F}{[\phi\lambda_s + (1-\phi)\lambda_f]^2}\frac{\partial T}{\partial x}\right)
\end{aligned} \tag{25}$$

These three sensitivity equations (23), (24) and (25) are completed with adequate initial and boundary conditions. W, R and Z are the unknowns of the sensitivity equations and T is the temperature.

2.4 Numerical strategy

The obtained system of coupled equations (heat diffusion equation + 3 sensitivity equations) is a nonlinear system of partial differential equations. To solve this system, we use the same numerical strategy used in the forward problem (method of lines + finite volume method). After spatial discretization, the system of coupled equations can be written in the form:

$$F(t, Y, Y') = 0 \text{ with } Y(t_0) = Y_0 \tag{26}$$

where $Y = (T\ W\ R\ Z)^t$. The system in equation (26) can be solved by an ODE solver as in the forward problem.

2.5 Algorithm

The aim of the inverse problem is the calculation of the vector parameters p that minimizes the cost function S presented in equation (5). The Damped Gauss Newton (DGN) algorithm that we chose to apply to our nonlinear least square problem is as follows:

1. Choose a constant damping parameter α $(0 < \alpha \leq 1)$.
2. Choose an initial value $p^{(0)}$; initialize the iteration $k = 0$
3. Perform the parameters' scaling to obtain $\tilde{p}$;
4. Solve the system (heat equation with phase change + sensitivity equations) using $\tilde{p}^{(k)}$ to define the parameters of the soil. The equivalent parameters of the system are calculated by the apparent heat capacity method (AHC); Deduce $T_i^{f,(k)}$, $\tilde{W}_i^{f,(k)}$, $\tilde{R}_i^{f,(k)}$ and $\tilde{Z}_i^{f,(k)}$ for $i = 1, ..., M$ and $f = 1, ..., F$
5. Calculate $r^{(k)}$ and the Sensitivity matrix $\tilde{J}$
   knowing that $\tilde{J} = J.D$
6. Solve the linear system $\tilde{J}(\tilde{p}^{(k)})^t\tilde{J}(\tilde{p}^{(k)})\tilde{p}^{(k+1)} = \tilde{J}(\tilde{p}^{(k)})^t\tilde{J}(\tilde{p}^{(k)})\tilde{p}^{(k)} - \alpha.\tilde{J}(\tilde{p}^{(k)})^t r^{(k)}$ for $\tilde{p}^{(k+1)}$.
7. If the criteria of convergence are reached, end.
   Calculate the original parameters' vector $p^{(k+1)} = D.\tilde{p}^{(k+1)}$.

   If not, iterate:
   $\tilde{p}^{(k)} \leftarrow \tilde{p}^{(k+1)}$ and go to 4.

2.6 Stopping criteria

Classically, there are three convergence tests used in the algorithms for nonlinear least square problems (e.g [MGH80]). We chose to apply only two of them. The first test is the X-convergence which is based on an estimate of the distance between the current approximation x and the previous solution x* of the problem. If D is the current scaling matrix, then this convergence test attempts to guarantee that:

$$\|D(x - x^*)\| \leq XTOL.\|Dx^*\| \tag{27}$$

where XTOL is a user supplied tolerance (we used XTOL $= 10^{-6}$). The second test, the main convergence test, is based on an estimate of the distance between the Euclidean norm $\|F(x)\|$ of the residuals at the current approximation x and the previous value $\|F(x^*)\|$ at the previous solution x* of the problem. This convergence test (F-convergence) attempts to guarantee that:

$$\|F(x)\| \leq (1 + FTOL).\|F(x^*)\| \tag{28}$$

where FTOL is another user-supplied tolerance (we used FTOL $= 10^{-6}$).

2.7 Code validation

The code validation is based on choosing a plausible example where the soil parameters $\{(\rho C)_s, \phi, \lambda_s\}$ are given constant values. These values are used by the forward problem to calculate the temperature at 5 different positions of the domain (5 sensors). These temperatures are recorded every 24 seconds for 4 hours. In tables 1 and 2, we used the same number of mesh cells in the forward problem (to create the synthetic data) and in the inverse problem. In both tables, we removed the approximation (see equation (21)). In table 1, we did not use the scaling technique while we used it in table 2. The results presented in the two tables prove that scaling is an important factor to obtain the desirable results. On the other hand, if we use scaling together

Table 1: Physical properties of the soil obtained by inverse problem **without** scaling (19 iterations).

	$(\rho C)_s$ $(J/kg.K)$	λ_s (W/m.K)	ϕ
exact	1.95×10^6	0.756	0.20
initial guess	2×10^6	0.8	0.18
calculated	2×10^6	0.7696	0.1979

Table 2: Physical properties of the soil obtained by inverse problem **with** scaling (31 iterations).

	ρC_s $(J/kg.K)$	λ_s (W/m.K)	ϕ
exact	1.95×10^6	0.756	0.20
initial guess	2×10^6	0.8	0.18
calculated	1.9497×10^6	0.7559	0.2000

with the approximation (21) then we will obtain approximately the same results as those in table 2 but the number of iterations is very large (see figure 2). The figures that represent the remaining parameters are similar to figure 2. The target of our work is to perform a numerical simulation that is the closest possible to the real experimental case. For this reason, we generate the synthetic data using a very large number of mesh cells (around 6000) to obtain accurate results. Thereafter, these data play the role of the experimental data in the inverse problem which is run using small number of mesh cells (40, 80, 120, 160 ...). Figure 3 represents the variation of the final residue as function of the number of mesh cells. We can easily notice that the residue decrease as number of mesh cells increase which assures the consistency of our method. Figure 4 represents the convergence of the conductivity for 120 mesh cells in the inverse problem (the figures representing the convergence of the volumetric heat capacity and porosity are similar). We notice that convergence is achieved after few tens of iterations (37 in this case).

2.8 Role of ΔT

As we have seen earlier, the choice of the phase change temperature interval ΔT in the AHC method affects the temperature profile. Recall that ΔT is proportional to h [MCM09] ($\Delta T = kh$) where h is the mesh size and k is a constant chosen in a way to obtain good accuracy with fewer fluctuations in the temperature profile. As a consequence, the value of ΔT plays an important role in the results of the inverse problem. If the initial values of parameters are far from the exact solution then the damped Gauss Newton method might not converge using the optimal value of ΔT [MCM09].To study the effect of this important parameter, we chose the example chosen earlier $\{(\rho C)_s = 1.95 \times 10^6,\ \phi = 0.2,\ \lambda_s = 0.756\}$ and we run the inverse problem using different values of ΔT with 120 as number of mesh cells. We notice

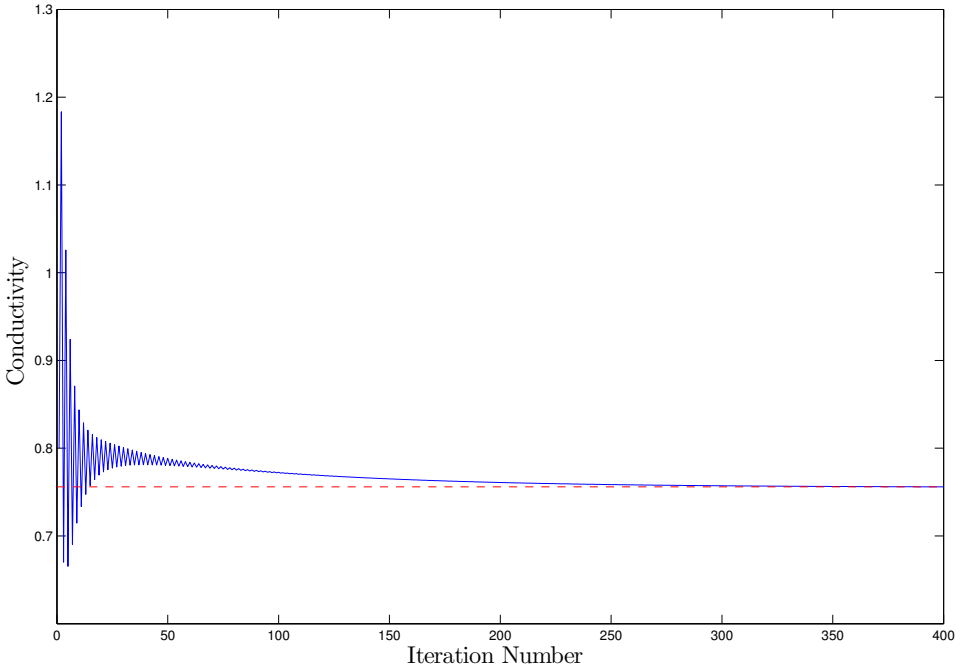

Figure 2: Variation of λ_s as function of iteration number (using scaling and approximation 21). The red line represents the exact value of λ_s.

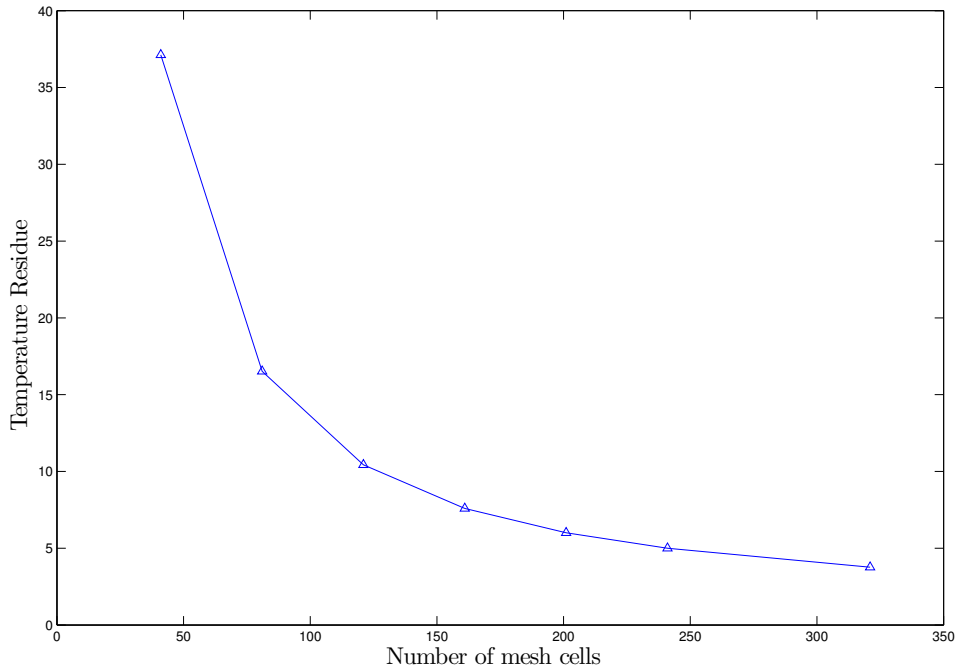

Figure 3: Variation of residue as function of number of mesh cells. The method is consistent (the error decreases as number of mesh cells increase).

that the inverse program fails to converge for $\frac{\Delta T}{kh} = 1$ or 2 and when $\frac{\Delta T}{kh} \geq 11$ whereas it converges for $3 \leq \frac{\Delta T}{kh} \leq 10$. We notice that the values of $(\rho C)_s$, ϕ and λ_s recede from the exact

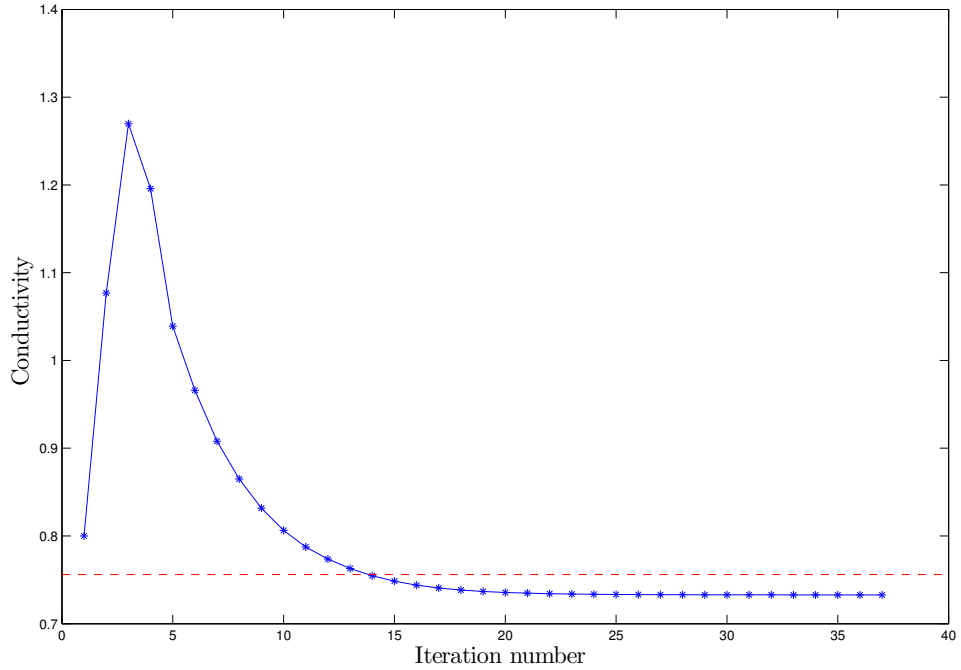

Figure 4: Variation of the conductivity as function of iteration number. The red line represents the exact value of λ_s.

solution and the value of residue increases from 9.588 to 26.7198 as the value ΔT increases from $3 \times \Delta T_{optimum}$ to $10 \times \Delta T_{optimum}$ (see table 3). The results in table 3 are obtained by using

Table 3: Values of soil parameters and residue obtained by varying $\frac{\Delta T}{kh}$. Results become less accurate as $\frac{\Delta T}{kh}$ increases.

$\frac{\Delta T}{kh}$ \ Soil parameter	$(\rho C)_s$	λ_s	ϕ	residue
3	1.149×10^6	0.5362	0.2319	9.5889
4	8.654×10^5	0.4748	0.2436	11.5282
5	6.42×10^5	0.4306	0.2526	13.5174
7	3.351×10^5	0.3770	0.2640	18.0604
10	6.159×10^4	0.3400	0.2724	26.7198

$(\rho C)_s = 3 \times 10^6$, $\lambda_s = 0.4$ and $\phi = 0.12$ as initial guesses. If the results obtained in the first row ($\Delta T = 3 \times \Delta T_{optimum}$) are used as initial guesses then the inverse problem will converge to the exact values using $\Delta T_{optimum}$ if we use the same mesh size in both the forward and inverse problems and toward acceptable values if we use huge mesh size to generate the synthetic data and 120 mesh cells in the inverse problem (check table 4).

3 Levenberg Marquardt Algorithm (1D)

In section 2, we explained that our inverse problem can be viewed as a nonlinear least square minimization problem which is solved by the Damped Gauss Newton Algorithm. In this section, we present a more robust algorithm to solve the nonlinear least square minimization problem known as Levenberg Marquardt Algorithm (LMA). LMA is the most widely used optimization algorithm for the solution of nonlinear least square problems. It outperforms simple gradient

Table 4: Physical properties of the soil obtained by inverse problem using $\Delta T = \Delta T_{optimum}$ using the calculated values in 3 as initial guesses (40 iterations).

	ρC_s $(J/kg.K)$	λ_s (W/m.K)	ϕ
exact	1.95×10^6	0.756	0.20
initial guess	1.1493×10^6	0.5362	0.2319
calculated	1.9387×10^6	0.7328	0.2001

descent and other conjugate gradient methods in a wide variety of problems. It is a blend of original gradient descent and Damped Gauss Newton iteration.

3.1 Introduction to LMA

Levenberg [Lev44] and Marquardt [Mar63] proposed a very elegant algorithm for the numerical solution of equation (5). However, most implementations are either not robust, or do not have a solid theoretical justification. Moré [Mor78] presented a robust and efficient implementation of a version of the Levenberg-Marquardt and show that it has strong convergence properties. In addition to robustness, the main features of this implementation are the proper use of implicitly scaled variables and the choice of the Levenberg-Marquardt parameter via a scheme due to Hebden [Heb73]. The implementation of LMA by Moré that is contained in Minpack [MGH80] has proven to be very successful in practice. Several factors make LMA preferable to DGN: first is that LMA possesses an embedded scaling technique, second it is well defined even when J doesn't have full column rank and finally is that when the Gauss-Newton step is too long, the Levenberg Marquardt step is close to being in the steepest-descent direction $-J^t r$ and is often superior to the DGN step. We use the LMDER1 Minpack subroutine for numerical solution of nonlinear least square problems. LMDER1 is based on Moré's LMA version where the user must provide a subroutine to calculate the functions r_1, r_2, ... r_m and the Jacobian matrix $\frac{\partial r_i(p)}{\partial p_j}$. LMDER1 follows the convergence criteria mentioned in section 3.6.

3.2 Applying LMA on our Inverse Problem: Results

Using the LMDER1 Minpack subroutine (which is embedded in the easy-to-use MUESLI library [Can]) and providing the Jacobian matrix, we obtain the results summarized in table 5. The Jacobian matrix is calculated using Maple.

Table 5: Physical properties of the soil obtained by inverse problem using LMA. Scaling is used implicitly and approximation (21) is removed (11 iterations). Same number of mesh cells is used in both the forward and inverse problems.

	ρC_s $(J/kg.K)$	λ_s (W/m.K)	ϕ
exact	1.95×10^6	0.756	0.20
initial guess	2.0×10^6	0.8	0.18
calculated	1.957×10^6	0.758	0.1996

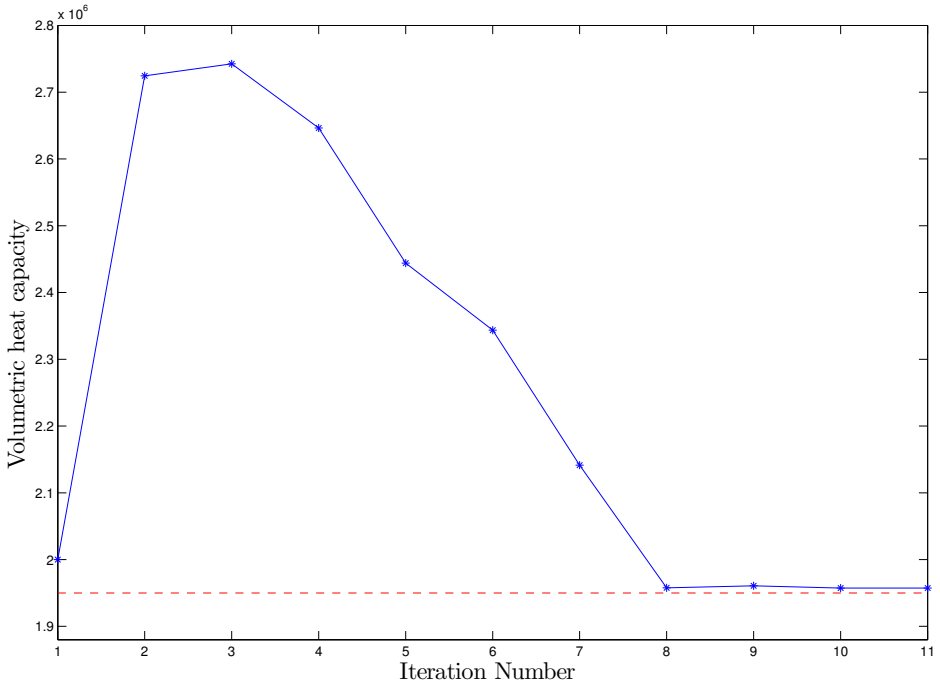

Figure 5: Variation of the volumetric heat capacity as function of iteration number using LMA (Same number of mesh cells in both the forward and inverse problems). The red line represents the exact value of $(\rho C)_s$.

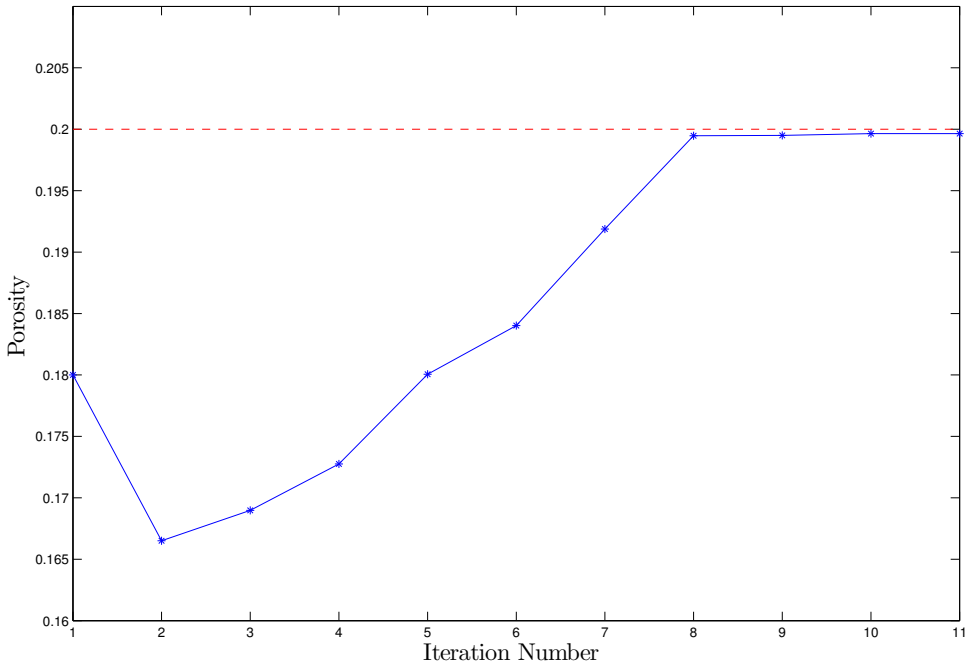

Figure 6: Variation of the porosity as function of iteration number using LMA (Same number of mesh cells in both the forward and inverse problems). The red line represents the exact value of ϕ.

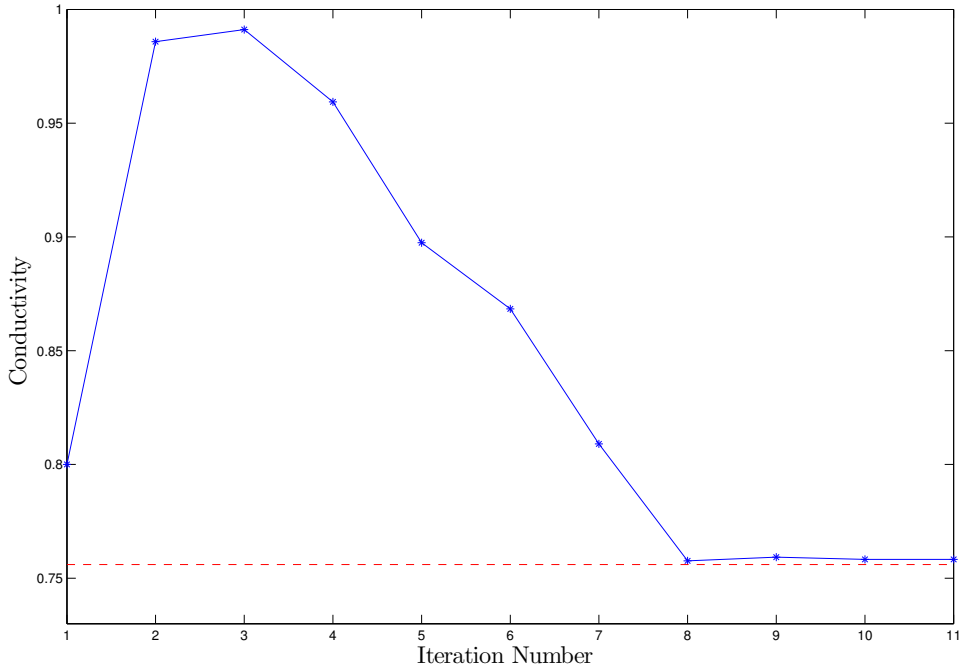

Figure 7: Variation of the conductivity as function of iteration number using LMA (Same number of mesh cells in both the forward and inverse problems). The red line represents the exact value of λ_s.

In figures 5, 6 and 7 we notice that the calculated values are very close to the exact ones due to the fact that we used same number of mesh cells in both direct and inverse problems. Moreover, we notice that convergence is obtained after about 10 iterations.

Identification of the thermophysical properties of the soil during phase change by inverse problem 3D axisymmetric coordinate system

4 Forward problem (3D)

The physical problem consists of heating the soil by a fire. To model this problem, we replace the soil by a perfect porous medium with constant and uniform properties heated from above by a constant temperature T_c (temperature of the fire between 300 C and 700 C). T_c must be greater than T_v (the evaporation or phase change temperature which is normally 100 C).

The energy equation in 3D axisymmetric coordinate system (independent of θ) is given by:

$$(\rho C)_e \frac{\partial T}{\partial t} = \frac{1}{r}\frac{d}{dr}\left(r\lambda_e \frac{dT}{dr}\right) + \frac{d}{dz}\left(\lambda_e \frac{dT}{dz}\right) \tag{29}$$

with the following initial and boundary conditions:

$$T(r_i, z_j, 0) = T_0(r_i, z_j) = T_{0,i,j} \qquad 1 \leq i \leq N,\ 1 \leq j \leq M$$

$$T(r_i, z_j, t) = T^D(r_i, z_j, t) = T^D_{i,j}(t) \qquad i \in \{1, N\} \text{ and } j \in \{1, M\}$$

$$\nabla T(r_i, z_j, t).\nu = 0 \ \text{ (Null Neumann) } \ i \in \{1, N\} \text{ and } j \in \{1, M\}$$

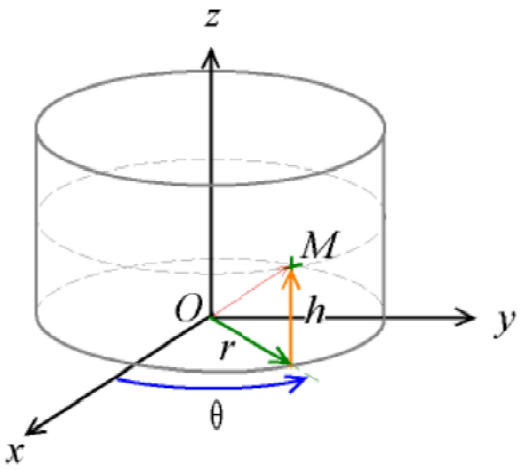

Figure 8: Cylindrical Coordinate System (in our case θ is assumed constant).

where T represents the temperature, T_0 is the initial temperature at $t_0 = 0$, T^D is T_c at the fire and T_0 elsewhere; ρ is the density, C is the specific heat capacity, λ is the conductivity, ϕ is the porosity, the subscripts e, f and s indicate the equivalent parameters of the medium, the properties of the fluid and the porous matrix properties respectively. ν indicates the outward unit normal vector along the boundary of Ω. The computational domain is represented in figure 9.

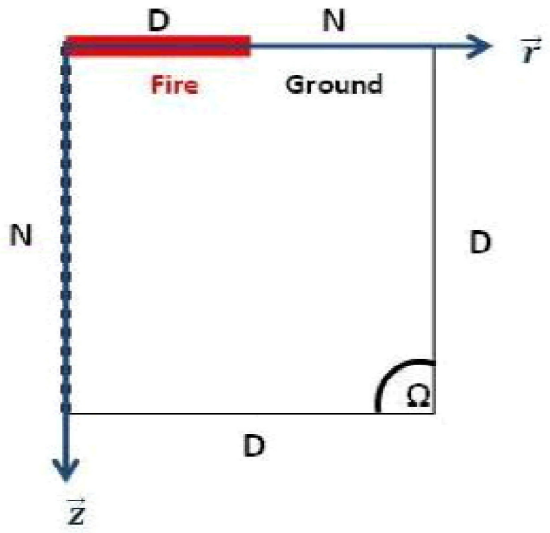

Figure 9: The computational domain in 3D axisymmetric coordinate system. The red part represent the fire. N represents Neumann boundary condition while D represents Dirichlet boundary condition

The effective volumetric heat capacity and the effective conductivity are defined as in the 1D configuration.

As in 1D, to avoid the tracking of the interface of the phase change problem (liquid/vapor) which appears when the water existing in the soil turns into gas, the apparent heat capacity (AHC) method is used.

4.1 Numerical strategy

We need to solve the heat diffusion equation (PDE) so we choose the method of lines which is a way of approximating PDEs by ODEs where space and time discretizations are considered separately. The spatial The discretization is done using the finite volume method, the equation is integrated over a control volume. We use N discretization points in the r direction and M in the z direction so that after discretization the heat equation becomes:

$$\begin{aligned}\frac{dT_{i,j}}{dt} &- \frac{(r_i + r_{i+1})(\lambda_{i,j} + \lambda_{i+1,j})(T_{i+1,j} - T_{i,j})}{4(\rho C)_e r_i \Delta^2 r} \\ &+ \frac{(r_i + r_{i-1})(\lambda_{i,j} + \lambda_{i-1,j})(T_{i,j} - T_{i-1,j})}{4(\rho C)_e r_i \Delta^2 r} \\ &- \frac{(\lambda_{i,j} + \lambda_{i,j+1})(T_{i,j+1} - T_{i,j})}{2(\rho C)_e \Delta^2 z} \\ &+ \frac{(\lambda_{i,j} + \lambda_{i,j-1})(T_{i,j} - T_{i,j-1})}{2(\rho C)_e \Delta^2 z} = 0 \end{aligned} \tag{30}$$

where $\Delta r = r_{i+1} - r_i$ and $\Delta z = z_{j+1} - z_j$. (see figure 10). In our configuration, we used $\Delta r = \Delta z = h$. In fact, the spatial variable is discretized into $N \times M$ discretization points and

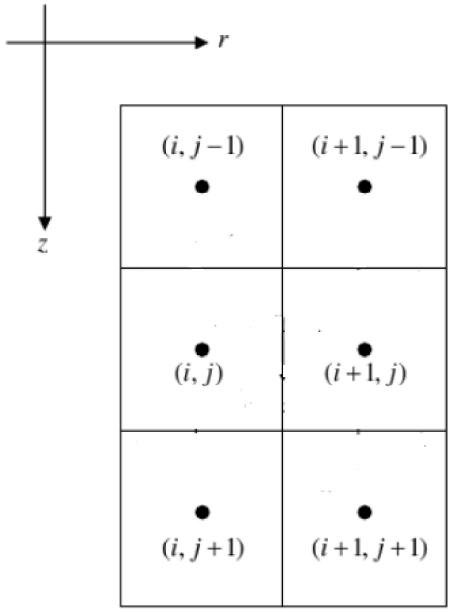

Figure 10: 3D axisymmetric control volumes

each state variable T is transformed into $N \times M$ variables corresponding to its value at each discretization point. After spatial discretization, the system formed using equation 30 is written in the form of an ODE system:

$$\frac{dT}{dt} = B(T)T \tag{31}$$

Some tries showed that our ODE system becomes more and more stiff as h becomes smaller. The difficulty with stiff problems is the prohibitive amounts of computer time required for their solution by classical ODE solution methods, such as the popular explicit Runge-Kutta and Adams methods. The reason is the excessively small step sizes that these methods must use to satisfy stability requirements. For this reason, we use an implicit ODE solver (Backward Differentation Formula) which possesses the property of stiff stability and therefore does not suffer from the stability step size constraint [Hin93]. It is based on polynomial interpolation so at t_n it approximates the solution $T(t)$ by the polynomial $P(t)$ that interpolates T_{n+1} and the

previously computed approximations. The polynomial is to satisfy the ODE at t_{n+1} [SGT03]. The BDF implicit scheme requires the calculation of a Jacobian matrix which is calculated and generated by a Computer Algebra System (CAS, Maple or Maxima) and then stored in a sparse format. Note that the numerical calculation is performed with *ddebdf* routine of the SLATEC Fortran library which was modified to use the UMFPACK sparse linear solver [Dav04]. The ODE solver performs time integration by adjusting automatically the time step in the BDF scheme [Can].

5 Inverse problem (3D)

In order to solve the parametric inverse problem consisting of finding the volumetric heat capacity $(\rho C)_s$, the conductivity λ_s and the porosity ϕ of the saturated soil, it is necessary to know the values of temperature $T^f_{gi,j}$ at selected points (sensors) of the porous medium domain for times t^f: $T^f_{gi,j} = T_g(r_i, z_j, t^f)$ where $i = 1, 2, ..., N$, $j = 1, 2, ..., M$ and $f = 1, 2, ..., F$. F is the total number of time steps. We use the least squares criterion to solve this inverse problem so we try to find the soil parameters that minimize the error function which is defined by:

$$S((\rho C)_s, \phi, \lambda_s) = \frac{1}{2}\|T^f_{i,j} - T^f_{gi,j}\|^2_2 \tag{32}$$

where $T^f_{i,j} = T(r_i, z_j, t^f)$ are the temperatures being the solution of the direct problem for the assumed set of parameters at the point (r_i, z_j), $i = 1, 2, ..., M$, $j = 1, 2, ..., N$ for the time t^f, $f = 1, 2, ..., F$ and $T^f_{gi,j}$ is the measured temperature at the same point (r_i, z_j) for time t^f.

5.1 Method of resolution

To illustrate the method of resolution, we define the following vectors:

$$T_g = \begin{pmatrix} T^1_{g1,1} \\ \dots \\ T^1_{g1,M} \\ T^1_{g2,1} \\ \dots \\ T^1_{g2,M} \\ \dots \\ T^1_{gN,1} \\ \dots \\ T^1_{gN,M} \\ T^2_{g1,1} \\ \dots \\ T^2_{g1,M} \\ T^2_{g2,1} \\ \dots \\ T^2_{g2,M} \\ \dots \\ T^2_{gN,1} \\ \dots \\ T^2_{gN,M} \\ \dots \\ T^F_{g1,1} \\ \dots \\ T^F_{g1,M} \\ T^F_{g2,1} \\ \dots \\ T^F_{g2,M} \\ \dots \\ T^F_{gN,1} \\ \dots \\ T^F_{gN,M} \end{pmatrix} \quad g(p^{(k)}) = \begin{pmatrix} T^{1,(k)}_{1,1} \\ \dots \\ T^{1,(k)}_{1,M} \\ T^{1,(k)}_{2,1} \\ \dots \\ T^{1,(k)}_{2,M} \\ \dots \\ T^{1,(k)}_{N,1} \\ \dots \\ T^{1,(k)}_{N,M} \\ T^{2,(k)}_{1,1} \\ \dots \\ T^{2,(k)}_{1,M} \\ T^{2,(k)}_{2,1} \\ \dots \\ T^{2,(k)}_{2,M} \\ \dots \\ T^{2,(k)}_{N,1} \\ \dots \\ T^{2,(k)}_{N,M} \\ \dots \\ T^{F.(k)}_{1,1} \\ \dots \\ T^{F.(k)}_{1,M} \\ T^{F.(k)}_{2,1} \\ \dots \\ T^{F.(k)}_{2,M} \\ \dots \\ T^{F.(k)}_{N,1} \\ \dots \\ T^{F.(k)}_{N,M} \end{pmatrix} \quad p^{(k)} = \begin{pmatrix} (\rho C)^{(k)}_s \\ \lambda^{(k)}_s \\ \phi^{(k)} \end{pmatrix}$$

and

$$r(p^{(k)}) = g(p^{(k)}) - T_g$$

where $r(p^{(k)})$ is the residual vector at the iteration k and $N' = N \times M \times F$. We adopted the use of the Levenberg Marquardt Algorithm [DS83] which is known to be locally convergent on almost all nonlinear least squares problems including large residual or very nonlinear problems [Bjö90].

The cost function $S((\rho C)_s, \phi, \lambda_s)$ defined by equation (5) can be re-written as:

$$\mathrm{S}(p^{(k)}) = \frac{1}{2} r(p^{(k)})^t r(p^{(k)}) \tag{33}$$

To minimize the least square norm, we need to equate to zero the derivatives of $S(p^{(k)})$ with respect to each of the unknown parameters $[p_1, p_2, p_3] = [(\rho C)_s, \phi, \lambda_s]$, that is :

$$\frac{dS}{dp_1} = \frac{dS}{dp_2} = \frac{dS}{dp_3} = 0 \tag{34}$$

Such necessary condition for the minimization of $S(p^{(k)})$ can be represented in matrix notation by equating the gradient of $S(p^{(k)})$ with respect to the vector of parameters p to zero, that is,

$$\nabla S(p^k) = J(p^k) r(p^k) = 0 \tag{35}$$

where $J(p^k)_{i',j'} = \dfrac{\partial r_{i'}(p^k)}{\partial p_{j'}^{(k)}}$, $i' = 1, 2, ..., N'$ and $j' = 1, 2, 3$. The Sensitivity matrix, $J(p^k)$ is defined by:

$$J(p^{(k)}) = \begin{pmatrix}
W_{1,1}^{1,(k)} & R_{1,1}^{1,(k)} & Z_{1,1}^{1,(k)} \\
... & ... & ... \\
W_{1,M}^{1,(k)} & R_{1,M}^{1,(k)} & Z_{1,M}^{1,(k)} \\
W_{2,1}^{1,(k)} & R_{2,1}^{1,(k)} & Z_{2,1}^{1,(k)} \\
... & ... & ... \\
W_{2,M}^{1,(k)} & R_{2,M}^{1,(k)} & Z_{2,M}^{1,(k)} \\
... & ... & ... \\
W_{N,1}^{1,(k)} & R_{N,1}^{1,(k)} & Z_{N,1}^{1,(k)} \\
... & ... & ... \\
W_{N,M}^{1,(k)} & R_{N,M}^{1,(k)} & Z_{N,M}^{1,(k)} \\
W_{1,1}^{2,(k)} & R_{1,1}^{2,(k)} & Z_{1,1}^{2,(k)} \\
... & ... & ... \\
W_{1,M}^{2,(k)} & R_{1,M}^{2,(k)} & Z_{1,M}^{2,(k)} \\
W_{2,1}^{2,(k)} & R_{2,1}^{2,(k)} & Z_{2,1}^{2,(k)} \\
... & ... & ... \\
W_{2,M}^{2,(k)} & R_{2,M}^{2,(k)} & Z_{2,M}^{2,(k)} \\
... & ... & ... \\
W_{N,1}^{2,(k)} & R_{N,1}^{2,(k)} & Z_{N,1}^{2,(k)} \\
... & ... & ... \\
W_{N,M}^{2,(k)} & R_{N,M}^{2,(k)} & Z_{N,M}^{2,(k)} \\
... & ... & ... \\
W_{1,1}^{F,(k)} & R_{1,1}^{F,(k)} & Z_{1,1}^{F,(k)} \\
... & ... & ... \\
W_{1,M}^{F,(k)} & R_{1,M}^{F,(k)} & Z_{1,M}^{F,(k)} \\
W_{2,1}^{F,(k)} & R_{2,1}^{F,(k)} & Z_{2,1}^{F,(k)} \\
... & ... & ... \\
W_{2,M}^{F,(k)} & R_{2,M}^{F,(k)} & Z_{2,M}^{F,(k)} \\
... & ... & ... \\
W_{N,1}^{F,(k)} & R_{N,1}^{F,(k)} & Z_{N,1}^{F,(k)} \\
... & ... & ... \\
W_{N,M}^{F,(k)} & R_{N,M}^{F,(k)} & Z_{N,M}^{F,(k)}
\end{pmatrix} \tag{36}$$

The elements of the sensitivity matrix are called the Sensitivity Coefficients. The sensitivity coefficient $J_{i',j}^f$ is thus defined as the first derivative of the estimated temperature at position $i' = (i, j)$ and time f with respect to the unknown parameter $p_{j'}$ [OO00], that is,

$$J_{i',j'}^f = \frac{\partial T_{i'}^f}{\partial p_{j'}} \tag{37}$$

where $W_{(i,j)}^{f,(k)} = \dfrac{\partial T_{(i,j)}^f}{\partial (\rho C)_s}|_{(\rho C)_s = (\rho C)_s^{(k)}}$, $Z_{(i,j)}^{f,(k)} = \dfrac{\partial T_{(i,j)}^f}{\partial \phi}|_{\phi=\phi^{(k)}}$ and $R_{(i,j)}^{f,(k)} = \dfrac{\partial T_{(i,j)}^f}{\partial \lambda_s}|_{\lambda_s = \lambda_s^{(k)}}$.

In theory, LMA is based on a trust region approach which is: find $m^{(k)}$ that minimizes $\|r(p^{(k)}) + J^{(k)}.m^{(k)}\|_2$ subject to $\|m^{(k)}\|_2 \leq \Delta$ where Δ is called the step bound [DS83]. On the other side, in Moré's implementation it is $m^{(k)}$ that minimizes

$$\|r(p^{(k)}) + J^{(k)}.m^{(k)}\|_2 \quad \text{subject to} \quad \|D.m^{(k)}\|_2 \leq \Delta \tag{38}$$

where D is a diagonal matrix which takes into account the scaling of the problem. The basis for Levenberg-Marquardt method is that if $m*$ is a solution of equation 38, then $p* = p(\alpha)$ for some $\alpha \geq 0$ where:

$$m(\alpha) = -(J^t J + \alpha D D^t)^{-1} J^t r \tag{39}$$

The Levenberg Marquardt Algorithm iteratively finds the minimum of S. Starting with an initial guess $p^{(0)}$ for the minimum, the method proceeds by the iterations:

$$p^{(k+1)} = p^{(k)} + m^{(k)} \tag{40}$$

5.2 Governed Equations

In the following, we present the heat equation together with the three sensitivity equations resulting from the differentiation of the heat diffusion equation (1) with respect to the soil parameters $p_{j'}$.

$$\frac{\partial}{\partial p_{j'}}\left[(\rho C)_e \frac{\partial T}{\partial t}\right] = \text{div}\left(\frac{\partial}{\partial p_{j'}}\left[\lambda_e \vec{\text{grad}} T\right]\right) \tag{41}$$

which leads to the general sensitivity equation below:

$$\begin{aligned}(\rho C)_e(T)\frac{\partial U_{j'}(x,t)}{\partial t} + \frac{d(\rho C)_e(T)}{dp_{j'}}\frac{\partial T(r,z,t)}{\partial t} = \\ \text{div}\left(\lambda_e\,(T)\vec{\text{grad}} U_{j'}(r,z,t)\right) + \text{div}\left(\frac{d\lambda_e(T)}{dp_{j'}}\vec{\text{grad}} T(r,z,t)\right)\end{aligned} \tag{42}$$

where $U_{j'} = \partial T/\partial p_{j'}$. The general sensitivity equation is accompanied with the following boundary and initial conditions:

$$t = 0: \quad U_{j'}(r,z,0) = U_{j'_0} = 0 \qquad \text{in } \Omega$$

$$U_{j'}(r,z,t) = U^D_{j'}(r,z,t) \qquad \text{on } \Gamma^D \times (0, t_{end}] \text{ (Dirichlet)}$$

$$U_{j'}(r,z,t).\nu = U^N_{j'}(r,z,t) \qquad \text{on } \Gamma^N \times (0, t_{end}] \text{ (Neumann)}$$

Similar to the 1D case, the heat diffusion equation must be differentiated with respect to the soil parameters $(\rho C)_s, \phi$ and λ_s respectively. The derivative of $(\rho C)_e$ and λ_e with respect to each soil parameter is as given in section 2.3.

The sensitivity equations in the 3D axisymmetric coordinate system are as follows:
Differentiating with respect to $(\rho C)_s$:

$$\begin{aligned}\frac{\partial W}{\partial t} &+ \frac{1}{(\rho C)_e}\frac{d(\rho C)_e}{d(\rho C)_s}\frac{dT}{dt} - \frac{1}{(\rho C)_e}\frac{1}{r}\frac{\partial}{\partial r}(r\lambda_e\frac{\partial W}{\partial r}) \\ &- \frac{1}{(\rho C)_e}\frac{\partial}{\partial z}(\lambda_e\frac{\partial W}{\partial z}) \\ &- \frac{1}{(\rho C)_e}\frac{\partial}{\partial r}(r\frac{d\lambda_e}{d(\rho C)_s}\frac{\partial T}{\partial r}) \\ &- \frac{1}{(\rho C)_e}\frac{\partial}{\partial z}(\frac{d\lambda_e}{d(\rho C)_s}\frac{\partial T}{\partial z})\end{aligned} \tag{43}$$

Differentiating with respect to ϕ :

$$\begin{aligned}\frac{\partial R}{\partial t} &+ \frac{1}{(\rho C)_e}\frac{d(\rho C)_e}{d\phi}\frac{dT}{dt} - \frac{1}{(\rho C)_e}\frac{1}{r}\frac{\partial}{\partial r}(r\lambda_e\frac{\partial R}{\partial r}) \\ &- \frac{1}{(\rho C)_e}\frac{\partial}{\partial z}(\lambda_e\frac{\partial R}{\partial z}) \\ &- \frac{1}{(\rho C)_e}\frac{\partial}{\partial r}(r\frac{d\lambda_e}{d\phi}\frac{\partial T}{\partial r}) \\ &- \frac{1}{(\rho C)_e}\frac{\partial}{\partial z}(\frac{d\lambda_e}{d(\phi}\frac{\partial T}{\partial z}) \end{aligned} \tag{44}$$

Differentiating with respect to λ_s :

$$\begin{aligned}\frac{\partial Z}{\partial t} &+ \frac{1}{(\rho C)_e}\frac{d(\rho C)_e}{d\lambda_s}\frac{dT}{dt} - \frac{1}{(\rho C)_e}\frac{\partial}{\partial r}(r\lambda_e\frac{\partial Z}{\partial r}) \\ &- \frac{1}{(\rho C)_e}\frac{\partial}{\partial z}(\lambda_e\frac{\partial Z}{\partial z}) \\ &- \frac{1}{(\rho C)_e}\frac{\partial}{\partial r}(r\frac{d\lambda_e}{d\lambda_s}\frac{\partial T}{\partial r}) \\ &- \frac{1}{(\rho C)_e}\frac{\partial}{\partial z}(\frac{d\lambda_e}{d(\lambda_s}\frac{\partial T}{\partial z}) \end{aligned} \tag{45}$$

These three sensitivity equations (23), (24) and (25) are completed with adequate initial and boundary conditions. W, R and Z are the unknowns of the sensitivity equations and T is the temperature.

5.3 Numerical strategy

The obtained system of coupled equations (heat diffusion equation + 3 sensitivity equations) is a nonlinear system of partial differential equations. To solve this system, we use the same numerical strategy used to solve the 1D inverse problem.

5.4 Algorithm

The aim of the inverse problem is the calculation of the vector parameters p that minimizes the cost function S presented in equation (5). The Levenberg Marquardt algorithm is summarized below:

(a) Given $\Delta^{(k)} > 0$, find $\alpha^{(k)} \geq 0$ such that if: $(J^{(k)})^t J^{(k)} + \alpha^{(k)} D^{(k)} D^{(}(k))^t) m^{(k)} = -J^{(}(k))^t r^{(k)}$ then either $\alpha^{(k)} = 0$ and $\|D^{(k)}.m^{(k)}\|_2 \leq \Delta^{(k)}$ or $\alpha^{(k)} > 0$ and $\|D^{(k)}.m^{(k)}\|_2 = \Delta^{(k)}$.

(b) If $\|r(p^{(k)} + m^{(k)})\|_2 < \|r(p^{(k)})\|_2$ set $p^{(k+1)} = p^{(k)} + m^{(k)}$ and evaluate $J^{(k+1)}$; otherwise set $p^{(k+1)} = p^{(k)}$ and $J^{(k+1)} = J^{(k)}$.

(c) Choose $\Delta^{(k)}$ and $D^{(k+1)}$.

The choice of Δ is explained in details in [Mor78].

The implementation of LMA by Moré that is contained in Minpack [MGH80] has proven to be very successful in practice. Several factors make LMA preferable to DGN: one is that LMA is well defined even when J doesn't have full column rank and another is that when the Gauss-Newton step is too long, the levenberg Marquardt step is close to being in the steepest-descent direction $-J^t r$ and is often superior to the DGN step. We use the LMDER1 Minpack subroutine for numerical solution of nonlinear least square problems. LMDER1 is based on Moré's LMA version where the user must provide a subroutine to calculate the functions r_1, r_2, ... r_m and the Jacobian matrix $\frac{\partial r_i(p)}{\partial p_j}$. LMDER1 is the easy-to-use driver for the core subroutine LMDER where D_0 and Δ_0 are set internally. There are two convergence tests used in the algorithm.

The first test is the X-convergence which is based on an estimate of the distance between the current approximation x and the previous solution x* of the problem. If D is the current scaling matrix, then this convergence test attempts to guarantee that:

$$\|D(x - x^*)\| \leq XTOL.\|Dx^*\| \tag{46}$$

where XTOL is a user supplied tolerance (we used XTOL = 10^{-6}). The second test, the main convergence test, is based on an estimate of the distance between the Euclidean norm $\|F(x)\|$ of the residuals at the current approximation x and the previous value $\|F(x^*)\|$ at the previous solution x* of the problem. This convergence test (F-convergence) attempts to guarantee that:

$$\|F(x)\| \leq (1 + FTOL).\|F(x^*)\| \tag{47}$$

where FTOL is another user-supplied tolerance (we used FTOL = 10^{-6})

5.5 Code validation

The code validation is based on the same plausible example used in the 1D case. In tables 6 and 7 below, we used the same number of mesh cells in the forward problem (to create the synthetic data) and in the inverse problem.

Table 6: Physical properties of the soil obtained by inverse problem (5 iterations).

	ρC_s $(J/kg.K)$	λ_s (W/m.K)	ϕ
exact	1.95×10^6	0.756	0.20
initial guess	2×10^6	0.8	0.18
calculated	1.9499×10^6	0.75596	0.200003

Table 7: Physical properties of the soil obtained by inverse problem (6 iterations).

	ρC_s $(J/kg.K)$	λ_s (W/m.K)	ϕ
exact	1.95×10^6	0.756	0.20
initial guess	3×10^6	0.4	0.12
calculated	1.9499×10^6	0.75595	0.200004

The target of our work is to perform a numerical simulation that is the closest possible to the real experimental case. For this reason, we generate the synthetic data using a very large number of mesh cells (around $150000 = 300 \times 500$) then we run the inverse problem with small number of mesh cells. Figure 11 represents the variation of the final residue as function of the number of mesh cells. We can easily notice that the residue decrease as number of mesh cells increase. Table 8 shows that values of the parameters for different mesh sizes knowing that the synthetic data is generated by 150000 mesh cells.

6 Comparison between 1D and 3D axisymmetric inverse problems

As we see in table 7, the inverse problem succeeds to converge in 3D axisymmetric coordinate even if the initial guess is far from the exact solution which was not the case in 1D. In table 9, we present some examples using different initial guesses where we notice that for certain initial guesses the 3D axisymmetric inverse problem converges while the 1D inverse problem fails to converge using LMA knowing that we used the same number of mesh cells in both the forward (to generate the synthetic data) and inverse problems. This is due to the fact that the inverse problems become more stable as the space dimension of the problem increase.

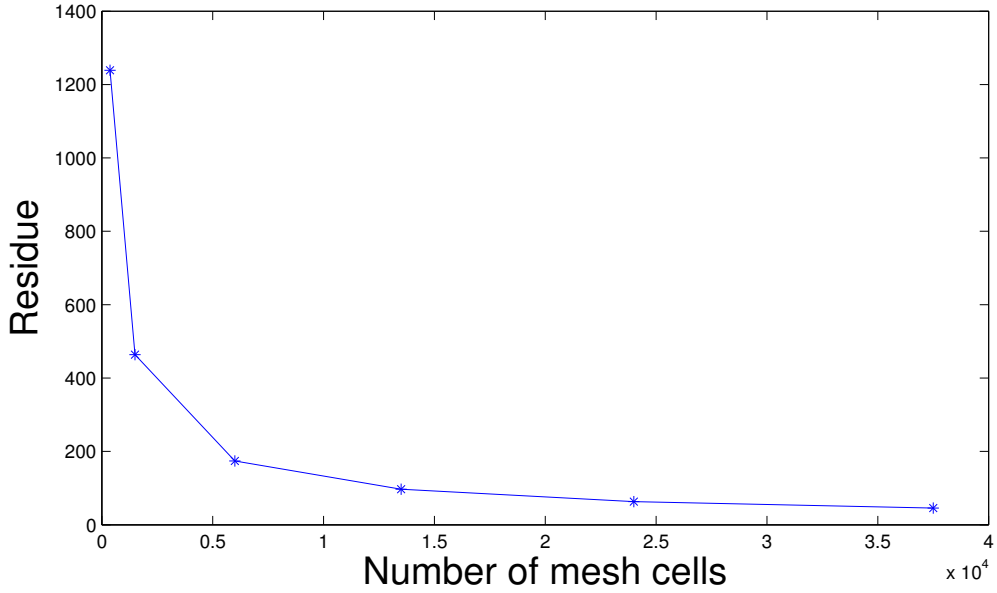

Figure 11: Variation of residue as function of number of mesh cells.

Table 8: Physical properties of the soil obtained by inverse problem using different number of mesh cells where the synthetic data are generated by 150,000 mesh cells.

Mesh	ρC_s $(J/kg.K)$	λ_s (W/m.K)	ϕ	Residue
30×50	1.23×10^6	0.403	0.232	463.9
60×100	1.97×10^6	0.651	0.198	173.8
90×150	2.05×10^6	0.712	0.195	97.2
120×200	2.045×10^6	0.735	0.195	62.9
150×250	1.95×10^6	0.725	0.199	45.6

Table 9: Comparison between the results obtained by 3D axisymmetric and 1D inverse problems. We vary the initial guess to notice the advantage of the 3D configuration.

	ρC_s $(J/kg.K)$	λ_s (W/m.K)	ϕ
exact	1.95×10^6	0.756	0.2
initial guess 1	1.0×10^6	1.1	0.1
calculated by 1D	Fails to converge using $\Delta T_{optimum}$		
calculated by 3D	1.949×10^6	0.7559	0.2
initial guess 2	1.5×10^6	0.35	0.3
calculated by 1D	Fails to converge using $\Delta T_{optimum}$		
calculated by 3D	1.949×10^6	0.7559	0.2

7 Influence of noised measures on the convergence of the inverse problem

In reality, the experimental measurement of temperatures will enclose some errors. To be close to reality, we perturbed the synthetic values of temperature by adding a Gaussian noise of null average and standard deviation equal to 5° C. Tables 10 and 11 represent the convergent values of the heat capacity, thethermal conductivity and the porosity of the soil, estimated by taking into consideration the noised measures in 1D and 3D axisymmetric coordinate systems

respectively.

Table 10: Physical properties of the soil obtained by inverse problem taking into consideration the noised measures in 1D. The same number of mesh cells is used in both the forward (to generate synthetic data) and inverse problems.

	$\rho C_s\ (J/kg.K)$	λ_s (W/m.K)	ϕ
exact	1.95×10^6	0.756	0.20
initial guess	2×10^6	0.8	0.18
calculated	1.952×10^6	0.879	0.189

Table 11: Physical properties of the soil obtained by inverse problem taking into consideration the noised measures in 3D. The same number of mesh cells is used in both the forward (to generate synthetic data) and inverse problems.

	$\rho C_s\ (J/kg.K)$	λ_s (W/m.K)	ϕ
exact	1.95×10^6	0.756	0.20
initial guess	2×10^6	0.8	0.18
calculated	1.954×10^6	0.767	0.20

We can see clearly that the inverse problem converges in both 1D and 3D axisymmetric coordinate system but toward different converging values which are not very far away from the values obtained without taking the noise into consideration. Figure 12 reveals that the residue decrease as function of the iteration number when noise is taken into consideration.

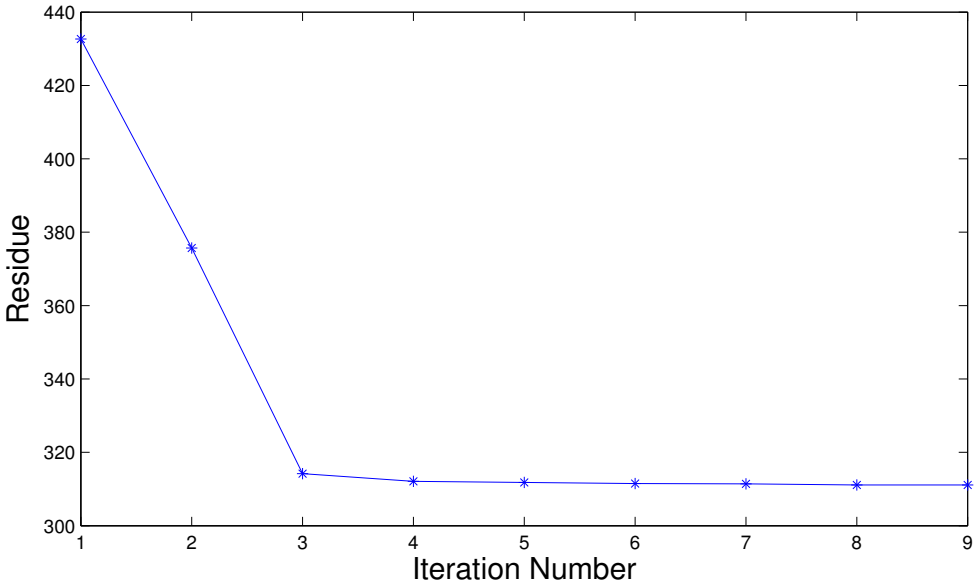

Figure 12: Variation of residue as function of iteration number.

8 Sensitivity study of the thermophysical parameters

Sensitivity analysis can split model parameters in two sets: sensitive and insensitive parameters. Sensitive parameters are characterized as parameters where variation in the parameter values

invoke a significant change in the model output, while a change of insensitive parameters has a negligible impact on the model output.

Sensitivity Analysis

In our model, the parameters have different order of magnitude so it is better to use scaled parameters. Also, since our model parameters (ρC_s, λ_s, ϕ) and model output (temperature) have different units, it may be advantageous to compute relative sensitivities defined as:

$$J = \frac{\partial T}{\partial p}\frac{p}{T} \tag{48}$$

In this study, we present a method (discussed in [PEZ+09]) uses singular value decomposition of the sensitivity matrix J followed by QR factorization. The sensitivity matrix in our model (see 36) need to be reformulated to match with equation (48). After that, the singular value decomposition of J ($J = USV^T$) is used to obtain a numerical rank for J. This numerical rank is then used to determine ρ parameters that can be identified given the model output T. To estimate the number of uncorrelated parameters we used an error estimate in our computation of the Jacobian as a lower bound on acceptable singular values. For example, in the study analyzed here we used ODE solver with an absolute error tolerance of 10^{-6} , i.e., the error of the numerical model solution is of order 10^{-6} and the error in the Jacobian matrix is approximately $\sqrt{10^{-6}} = 10^{-3}$. Consequently, singular values should not be smaller than 10^{-3}. Once the number of identifiable parameters has been determined, we find the most dominant parameters by performing a QR decomposition with column pivoting on the most dominant right singular vectors. Below we summarize subset selection method as an algorithm.
Subset selection algorithm:

1- Given an initial parameter estimate, p_0 , compute the Jacobian, $J(p_0)$ and the singular value decomposition $J = USV^T$, where S is a diagonal matrix containing the singular values of J in decreasing order, and V is an orthogonal matrix of right singular vectors.

2- Determine , the numerical rank of J. This can be done by determining a smallest allowable singular value.

3- Partition the matrix of eigenvectors in the form $V = [V_\rho V_{n-\rho}]$.

4- Determine a permutation matrix P by constructing a QR decomposition with column pivoting, for V_ρ^T . That is, determine P such that: $V_\rho^T P = QR$; where Q is an orthogonal matrix and the first ρ columns of R form an upper triangular matrix with diagonal elements in decreasing order.

5- Use the matrix P to re-order the parameter vector $\hat{p}$ according to $\hat{p} = P^T p$. The first ρ elements of $\hat{p}$ are identifiable. They are in ordered from most sensitive to less sensitive.

Applying the algorithm described above on our inverse problem in 3D axisymmetric coordinate system using synthetic data using J as the final sensitivity matrix before convergence, the singular values of J are $\sigma_1 = 4.86 \times 10^3$, $\sigma_2 = 4.24 \times 10^2$ and $\sigma_3 = 26.79$. This means that the 3 parameters are identifiable. Applying a QR decomposition with column pivoting for V^T we find the permutation matrix P

$$P = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

From all what preceded, we deduce that ϕ is the most identifiable parameter then λ_s and finally $(\rho C)_s$. Figure 13 represents the singular value ratio for the 3 parameters.

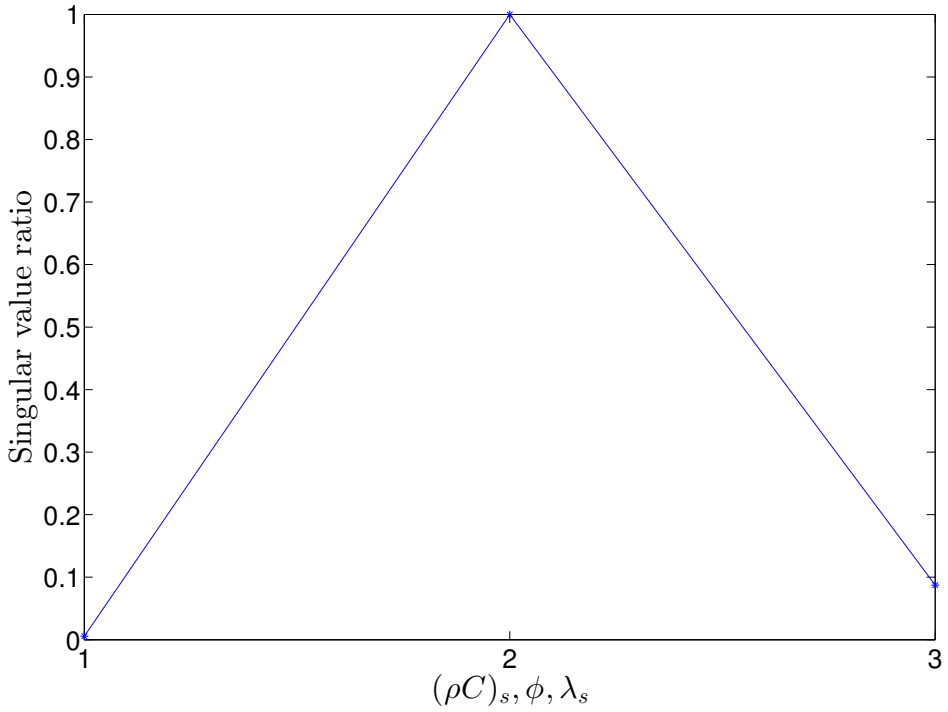

Figure 13: Sensitivity Analysis: ratio of singular values $\frac{\sigma_k}{\sigma_1}$.

9 Inverse problem in 3D axisymmetric coordinate system using real experimental data

9.1 Experimental hearth: materials and design

Modeling the evaporation of water in the soil (due to phase change) is not an easy procedure. We need to perform an experimental hearth where we replace the soil by a perfect porous medium. Figure 14 represents the experimental setup.

Materials needed:

- A stainless-steel box of size $50 \times 50 \times 30$ cm.
- Thermocouples of type K (Chromel/Alumel).
- An acquisition card.
- Labview software (we specify the number of thermocouples and measurements time interval and a stop watch displays total duration of measures and their numbers).
- A heating plate.
- A voltage converter for feeding the plate.
- Length measurement tools.
- Fontainebleau sand.

Procedure: We weigh a certain amount of sand and then we add 22% water by mass. We mix everything in a bucket then we pour it into the experimental box. The thermocouples are placed on the vertical axis which passes through the center of the heating plate(i.e they are

Figure 14: Experimental hearth.

plunged horizontally 25 cm) at different depths such that at least one of the thermocouples is in contact with the bottom of the heating plate which is used to control its temperature. We record the initial temperature T_0 then we heat the plate till it attains 400 C and we set the software to record the temperature every 30 seconds for 6 hours. When the experiment is over, we delve carefully to locate the position of the thermocouples. We measure the distance with respect to the two perpendicular walls of the box (x and y coordinates) and the height from the upper edge in order to deduce the depth of the thermocouples. In this experiment, thermocouples were found at the depths: 0 cm, 1.4 cm, 2.3 cm, 3.2 cm, 6.2 cm and 10.6 cm. Note that the Cartesian coordinates are transformed into 3D-axisymmetric coordinates (r-z).

9.2 Inverse problem

To solve the inverse problem consisting of finding the thermophysical properties of the saturated soil: heat capacity $(\rho C)_s$, the conductivity λ_s and the porosity ϕ during phase change and using real experimental data in 3D axisymmetric coordinate system, we use the same numerical strategy explained in section 5. We recall that the numerical method used in section 5 proved to be efficient and accurate using synthetic data but unfortunately it is not the case with real experimental data. In fact, the physical parameters attain negative values during the iterative procedure of the LMA algorithm and as a consequence the algorithm fails to converge. To overcome this obstacle, we used LMA with bound constraints over parameters to force the parameters values to stay in a certain physical domain [Sha08]. In other words, given an arbitrary point x, the projection of x onto the feasible bounded region is defined as follows. The i^{th} component of the projection of x is given by:

$$P(x,l,u)_i = \begin{cases} {}_i \text{ if } x_i < l_i \\ x_i \text{ if } x_i \in [l_i, u_i] \\ u_i \text{ if } x_i > u_i \end{cases} \tag{49}$$

where l_i and u_i represent the lower and upper bound of x_i respectively.

9.2.1 Results

The results obtained using the experiment performed by Szubert during his Masters project [Szu10] are summarized in the table 12:

Table 12: Physical properties of the soil obtained by inverse problem using real experimental data.

	$(\rho C)_s$ $(J/kg.K)$	λ_s (W/m.K)	ϕ	residue
initial guess	1.95×10^6	0.33	0.22	
calculated(mesh: 50×36)	1.0×10^5	0.1267	0.320	840
calculated(mesh: 100×72)	1.0×10^5	0.1120	0.3003	834
calculated(mesh: 150×108)	1.0×10^5	0.1124	0.3026	832

The bound constraints over the parameters used in this example are: $(\rho C)_s \in [1.0 \times 10^5, 1.0 \times 10^7]$, $\lambda_s \in [0.1, 9.0]$ and $\phi \in [0.1, 0.9]$. We notice that $(\rho C)_s$ converges to its lower bound whereas the exact physical value is of order 10^6, λ_s converges to 0.11 whereas the exact physical value is of order 0.3. Moreover, changing the box constrained used to: $(\rho C)_s \in [1.0 \times 10^3, 1.0 \times 10^7]$, $\lambda_s \in [0.1, 9.0]$ and $\phi \in [0.1, 0.9]$ gives totally different results which are summarized in the table 13.

	$(\rho C)_s$ $(J/kg.K)$	λ_s (W/m.K)	ϕ	residue
initial guess	1.95×10^6	0.33	0.22	
calculated(mesh: 150×108)	1.0×10^3	2.35	0.592	733

Table 13: Physical properties of the soil obtained by inverse problem using real experimental data.

Comparing tables 12 and 13, we notice that the results are not homogenous and thus our algorithm failed to converge in case experimental data is used. This failure is due to the fact that the physical model used during phase change is not close enough to reality because:

- The convection term, gravity and capillary forces are neglected.
- Radiation is neglected although strong heating is used.
- A saturated model is used which is experimentally impossible.
- A boiling model (temperature ¿ 100 C).

As a consequence, a more complex model is needed in order to approximate the thermophysical properties of the soil during phase change but this is out of the scope of this work. On the other hand, in the next part of this chapter we will study in details an inverse problem to approximate the thermophysical properties in case of a dry porous medium where no phase change is present.

10 Identification of the thermophysical properties of the soil in a dry porous medium (No Phase Change)

The soil is treated as a continuous medium subject to a purely diffusive transfer. This last hypothesis is a priori questionable because the heating is intense and the temperature gradients

can be significant (both vertical component and horizontal component), so we could have a radiative or convective transfer. The energy balance is simply reflected as we have seen earlier by the heat equation:

$$(\rho C)_e \frac{\partial T}{\partial t} = \text{div}\left(\lambda_e \vec{\text{grad}} T\right) \tag{50}$$

Where T is the temperature, $(\rho C)_e$ is the heat capacity per unit volume and λ_e is the thermal conductivity of the medium.

If the medium is homogeneous and isotropic, and that we neglect the variation of λ_e with temperature then one obtains:

$$\frac{\partial T}{\partial t} = \alpha \,\text{div}\left(\vec{\text{grad}} T\right) \tag{51}$$

Where $\alpha = \frac{\lambda_e}{(\rho C)_e}$ is the thermal diffusivity. We see that through all these assumptions (homogeneous effective medium, transfer by conduction only with conductivity independent of temperature T), the transfer is completely defined by the geometry of the medium, its diffusivity, boundary conditions and initial conditions. For a dry medium, which constitute of a solid phase of volumetric heat capacity $(\rho C)_s$ and thermal conductivity λ_s and fluid phase (dry air) of volumetric heat capacity $(\rho C)_f$ and thermal conductivity λ_f, the effective volumetric heat capacity of the medium can be wasily calculated by a simple linear formulation:

$$(\rho C)_e = \phi(\rho C)_f + (1-\phi)(\rho C)_s \tag{52}$$

Where ϕ represents the posority of the medium. The volumetric heat capacity of the medium can be easily calculated using the previous equation. On the contrary, the expression of the effective thermal conductivity as function of that of solid and fluid is uknown. Approximating the diffusivity by an inverse problem similar to that used previously will be a good way to approximate λ_e.

10.1 Experimental hearth: materials and design

To perform an experimental hearth, we replace the soil by a perfect dry porous medium. Figure 15 represents the experimental setup.

The materials used are exactly the same described in the experiment done in the case of phase change but we used the small heating plate (radius = 5 cm) instead of the big one.

Procedure:

We weigh a certain amount of dry sand and pour it into the experimental box. The thermocouples are placed randomly at different depths such that at least one of the thermocouples is in contact with the bottom of the heating plate which is used to control its temperature. We record the initial temperature T_0 then we heat the plate till it attains 600 C and we set the software to record the temperature every 30 seconds for 6 hours. When the experiment is over, we delve carefully to locate the position of the thermocouples. We measure the distance with respect to the two perpendicular walls of the box (x and y coordinates) and the height from the upper edge in order to deduce the depth of the thermocouples. In this study we explore two different experiments. Note that, porosity was estimated from the density of the sand (Fontainebleau sand) and the density of the solid alone. The value of the latter (2540 kg/m^3) was determined from a measurement in water (measure of volume of solid for a given mass of sand). Porosity ϕ represents the volume of pores to that of the total volume.

10.2 Methods used to approximate diffusivity α

10.2.1 Laloy and Massard

In their paper [LM84], Laloy and Massard presented a way to estimate the duration of fire. They assumed that the medium is homogeneous and isotropic and that physical characteristics such as conductivity, heat capacity and diffusivity are independent of temperature and of spatial

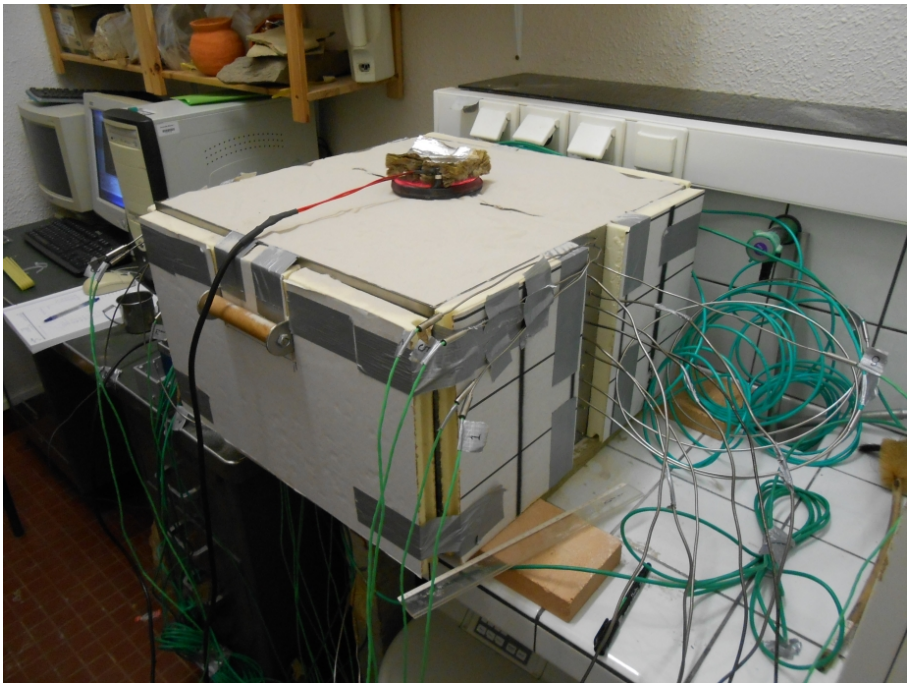

Figure 15: Experimental hearth.

coordinates. In addition, the assumed that the dimension of fire is larger than the height of the altered soil. These hypothesesthese allows a simple study of flat fire in a semi-infinite domain and hence solving analytically the heat diffusion equation in the case of a 1D geometry (function of the depth z).

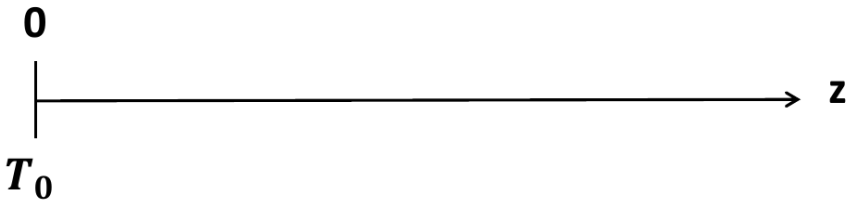

Figure 16: 1D problem.

The problem consists of finding $T = f(z,t)$ which is the solution of equation (51) where $\text{div}\left(\vec{\text{grad}}T\right) = \frac{\partial^2 T}{\partial z^2}$ with $\alpha = constant$. If T_0 is the temperature of fire (at t = 0 and z = 0), T_i is the initial temperature of the domain then:

$$\frac{T_0 - T(z,t)}{T_0 - T_i} = \frac{2}{\sqrt{\pi}} \int_0^x e^{-u^2}\, du \quad \text{where} \quad x = \frac{z}{2\sqrt{\alpha t}} \tag{53}$$

Knowing that:

$$e^{-2x^2} \leq 1 - \frac{4}{\pi}\left[\int_0^x e^{-u^2}\, du\right]^2 \leq e^{-x^2}$$

Laloy and Massard approximated the expression $1 - \frac{4}{\pi}\left[\int_0^x e^{-u^2}\, du\right]^2$ using a geometric average and thus:

$$1 - \frac{4}{\pi}\left[\int_0^x e^{-u^2}\, du\right]^2 \approx e^{\frac{-3}{2}x^2}$$

Consequently:

$$z^2 \approx \frac{-8\alpha t}{3} ln \left[1 - \left(\frac{T_0 - T(z,t)}{T_0 - T_i}\right)^2\right] \tag{54}$$

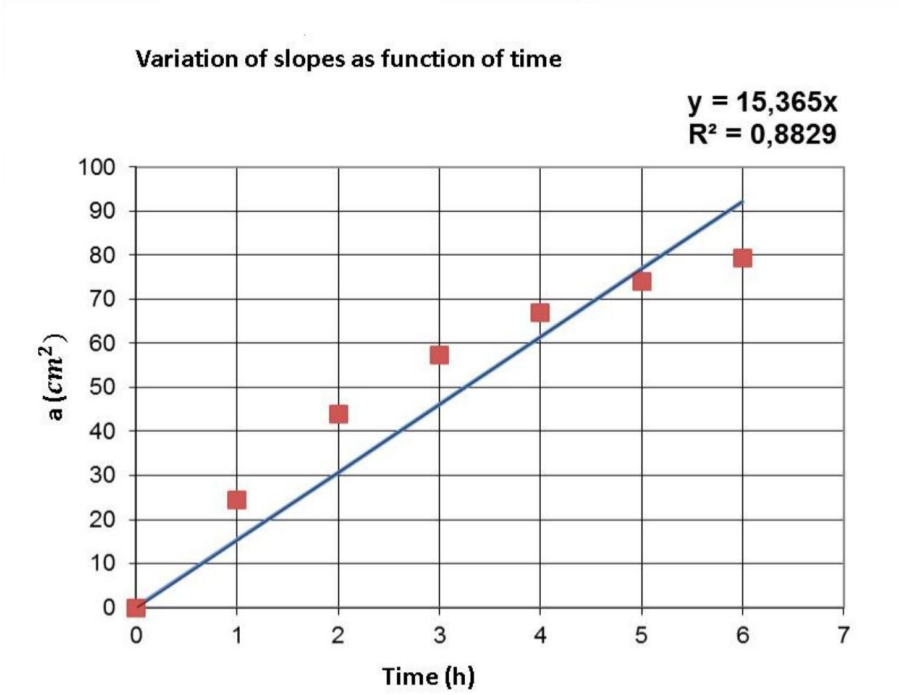

Figure 17: Variation of slope as function of time.

To determine α, according to Laloy and Massard, we simply need to know the variation of temperature in z-position at each instant t ($T(z,t)$) and then plot, for different values of z, z^2 as function of $-ln\left[1 - \left(\frac{T_0 - T(z,t)}{T_0 - T_i}\right)^2\right]$. If the assumptions are true, for each instant t, we should get a straight line passing through the origin, of slope $(\frac{8}{3}\alpha t)$. Normally only one of these lines is enough to obtain α (if t is known). To improve accuracy and test hypotheses, we must determine, at different times, the slope $a = \frac{8}{3}\alpha t$ and verify that it is a straight line of slope $\frac{8}{3}\alpha$. Using this simple way, we can deduce the value of the diffusivity α. (see figure 17).

Important Notes:

- Laloy and Massard method is valid in 1D only (see figure 18 to see the difference between the 1D and 3D-axisymmetric cases).
- Laloy and Massard is accurate if the temperature is measured at a depth close to the fire.

Results

	First Experiment	Second Experiment
Diffusivity α in m^2/s	2.5×10^{-7}	2.34×10^{-7}

Table 14: The values of diffusivity (Fontainebleau Sand) obtained by Laloy and Massard method.

The two experiments are done using small heating plate of radius 5 cm and a box of dimensions 50 cm $\times$ 50 cm $\times$ 30 cm so the computational domain is in 3D axisymmetric coordinate

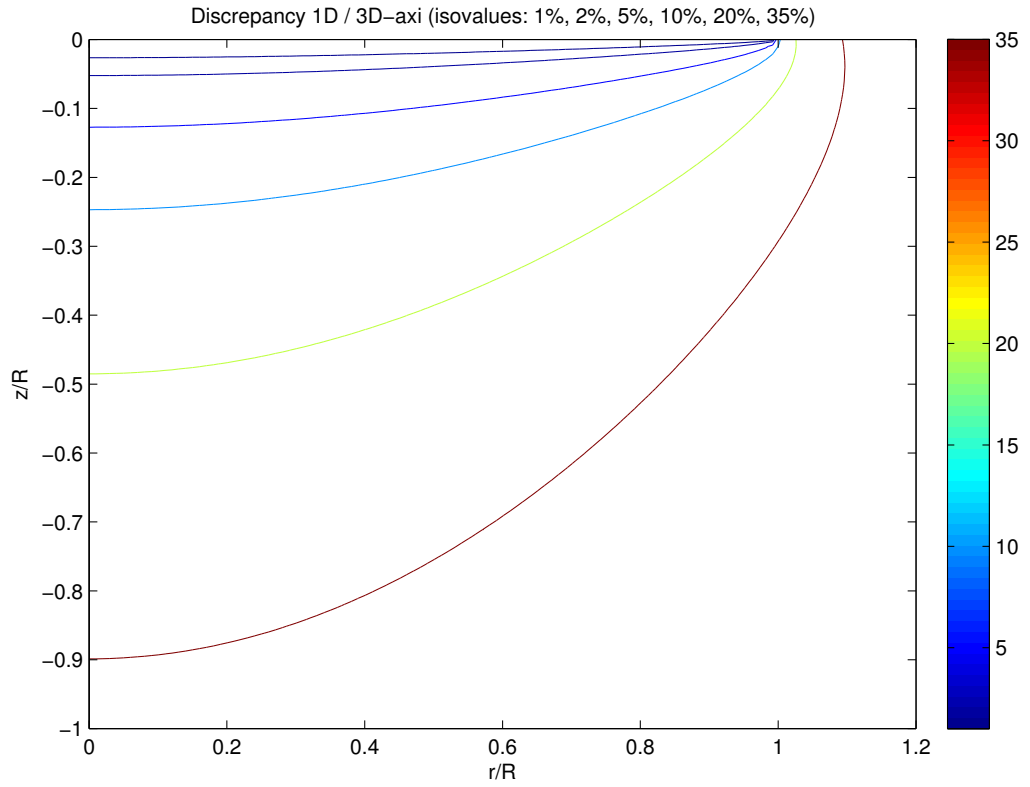

Figure 18: The difference in temperature profile for 1D and 3D-axisymmetric cases.

system and hence Laloy and Massard method will not be accurate due to the fact that we lost the 1 D property. Moreover, the sensors are not all close to the fire and the axis and as we noticed in figure 18 that the error between 1D and 3D axisymmetric configurations is reduced close to the fire and next to the axisymmetric axis whereas it increases away from the axis and as the depth increases. For all what proceeded, Laloy and Massard method is not accurate enough for an experiment done in 3D axisymmetric coordinate system.

10.2.2 Laplace Transform

In this part, we solve the heat diffusion equation in 1D using Laplace transform in time $u(x,t) \rightarrow U(x,p) = \mathcal{L}\left\{u(x,t)\right\}(p)$.

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2} \tag{55}$$

with $T(x,0) = 0$ and $T(0,t) = T_0(t)$. Using Laplace transform, the heat equation (55) becomes:

$$p\bar{T}(x,p) = \alpha \frac{\partial^2 \bar{T}(x,p)}{\partial x^2} \tag{56}$$

The solution of equation (56) is given by : $\bar{T}(x,p) = A(p)\ \exp\left(-\sqrt{\frac{p}{\alpha}}x\right)$ (see [CJ59]), where $A(p)$ comes from the boundary excitation $T_0(t)$.

Methodology

Assume that we have two two sensors at two different depths z_1 and z_2 then we can say that $\bar{T}_1(p)$ and $\bar{T}_2(p)$ at z_1 and z_2 respectively are given by:

$$\bar{T}_1(p) = A(p)\ \exp\left(-\sqrt{\frac{p}{\alpha}}z_1\right) \tag{57}$$

$$\bar{T_2}(p) = A(p)\ \exp\left(-\sqrt{\frac{p}{\alpha}}z_2\right) \tag{58}$$

then dividing equations 58 and 57 we get:

$$\frac{\bar{T_2}(p)}{\bar{T_1}(p)} = ratio(p) = \exp\left(-\sqrt{\frac{p}{\alpha}}\,(z_2 - z_1)\right) \tag{59}$$

we end up with:

$$(\Delta z)^2 p = \alpha\left[\log\left(ratio(p)\right)\right]^2 \quad \text{where} \quad \Delta z = z_2 - z_1 \tag{60}$$

Simply, to approximate α, we plot the graph of $(\Delta z)^2 p$ as function of $[\log(ratio(p))]^2$ and we calculate its slope.

Results

As we mentioned earlier, we need a couple of sensors only to approximate α. To ensure the validity of our method we use synthetic data where $\alpha = 1.0 \times 10^{-7}\ m^2/s$ and we try to refind α using Laplace transform. The computational domain is in 3D axisymmetric coordinate system where $r = z = 45\ cm$ and the fire radius is $R = 15\ cm$. Figure 19 shows the variation of the Laplace of the temperature at two different sensors as function of p, figure 20 is a check for the validity of the ratio expressed in equation (59), figure 21 allows us to calculate the slope which eventually equal to the diffusivity α. We can see clearly that the error between the exact and the calculated solution is 0.4%.

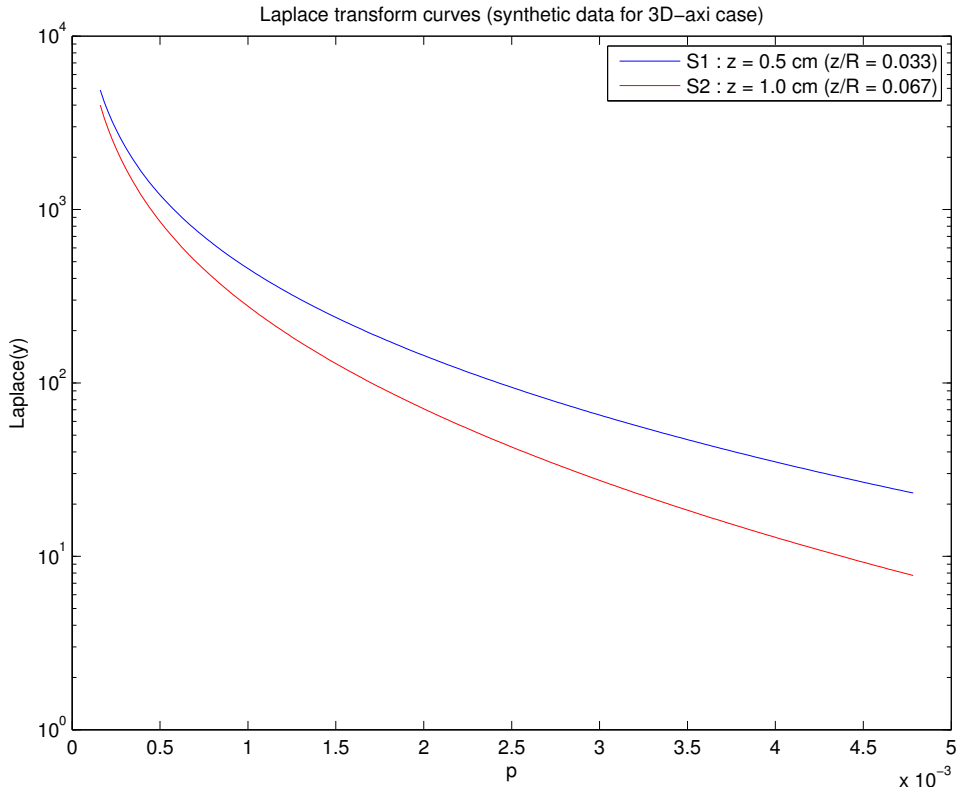

Figure 19: Laplace transform curves for two different sensors

Laplace transform method is valid for 1D configuration, so to check its precision in 3D axisymmetric coordinate system. We try to detect the variation of error obtained upon the choice of different sensors' positions. First, we study the effect of the sensor's depth on the accuracy

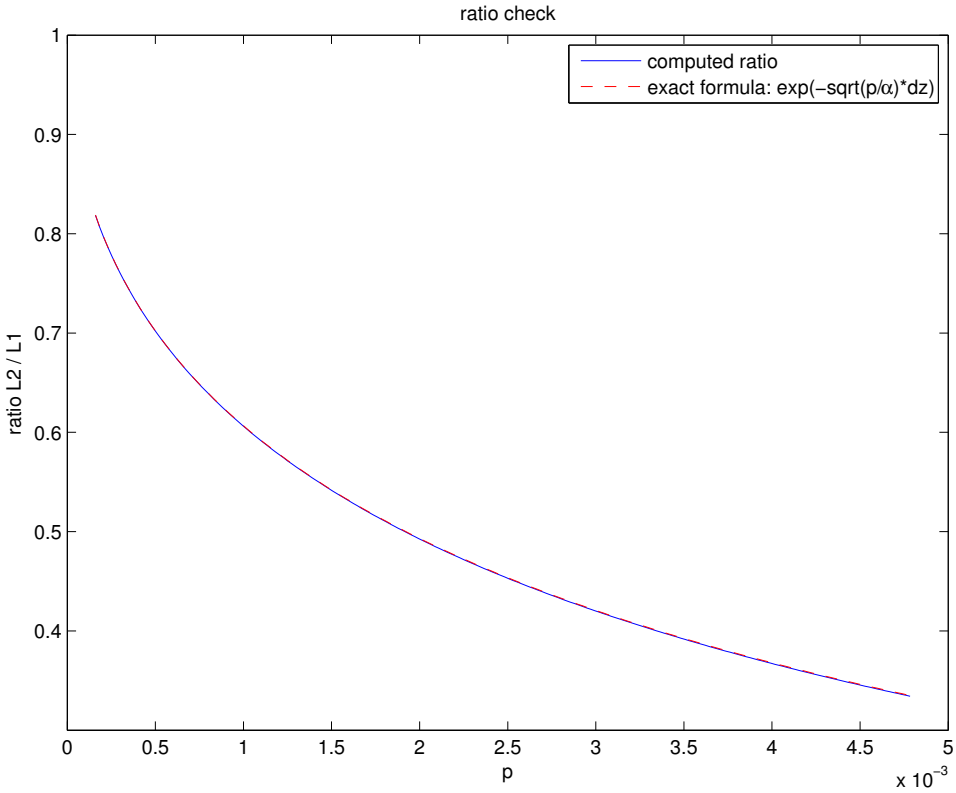

Figure 20: Ratio check

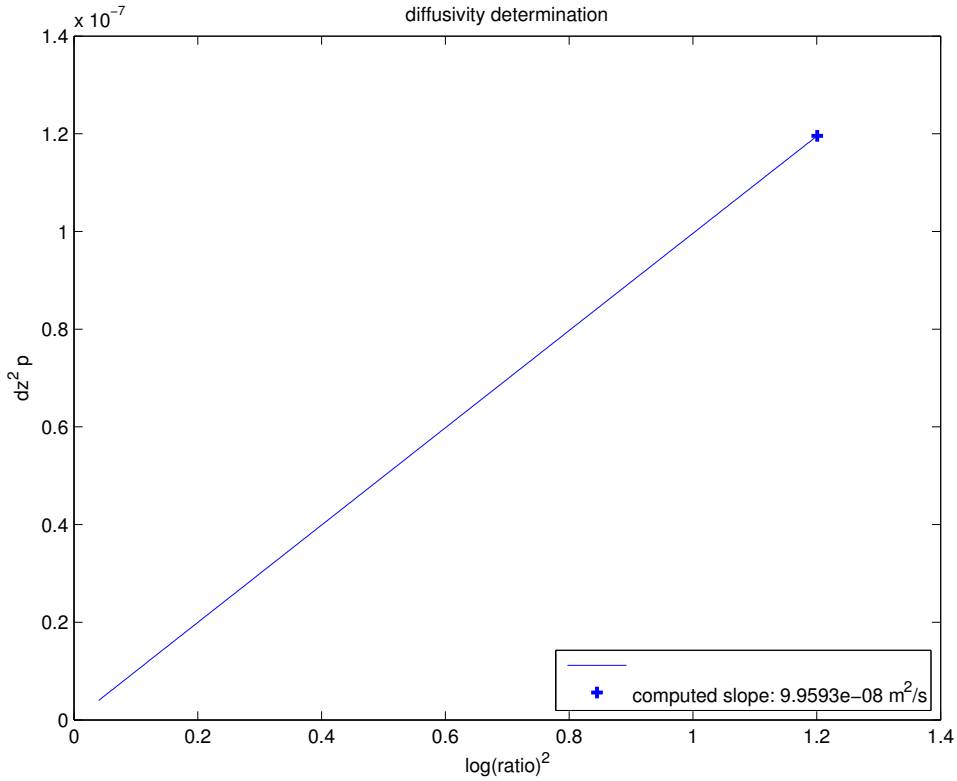

Figure 21: Diffusivity Determination

for results. In figure 22, we took into consideration 8 sensors located on the axisymmetric axis at different depths such that the ratio of depth to that of fire radius varies from 0.033 to 1.333

$(0.033 \leq \frac{z}{R} \leq 1.33)$. In figure 23, we notice that the error between the different calculated values and the exact one (1.0×10^{-7}) is between 0.4% and 0.44%.

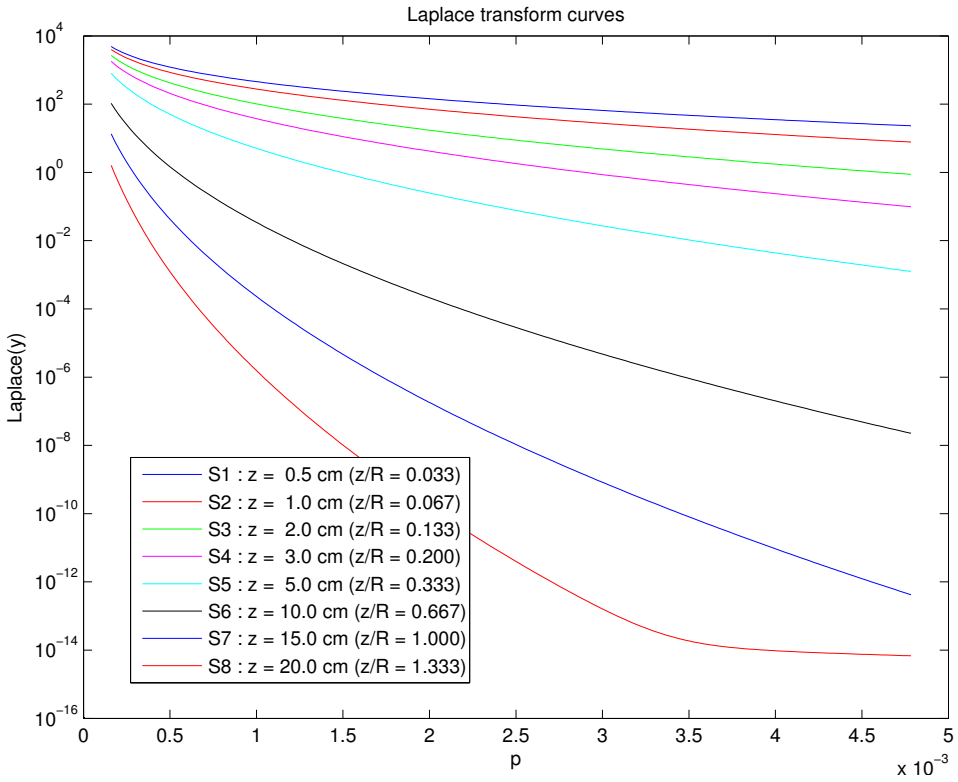

Figure 22: Laplace transform curves for different sensors at different depths

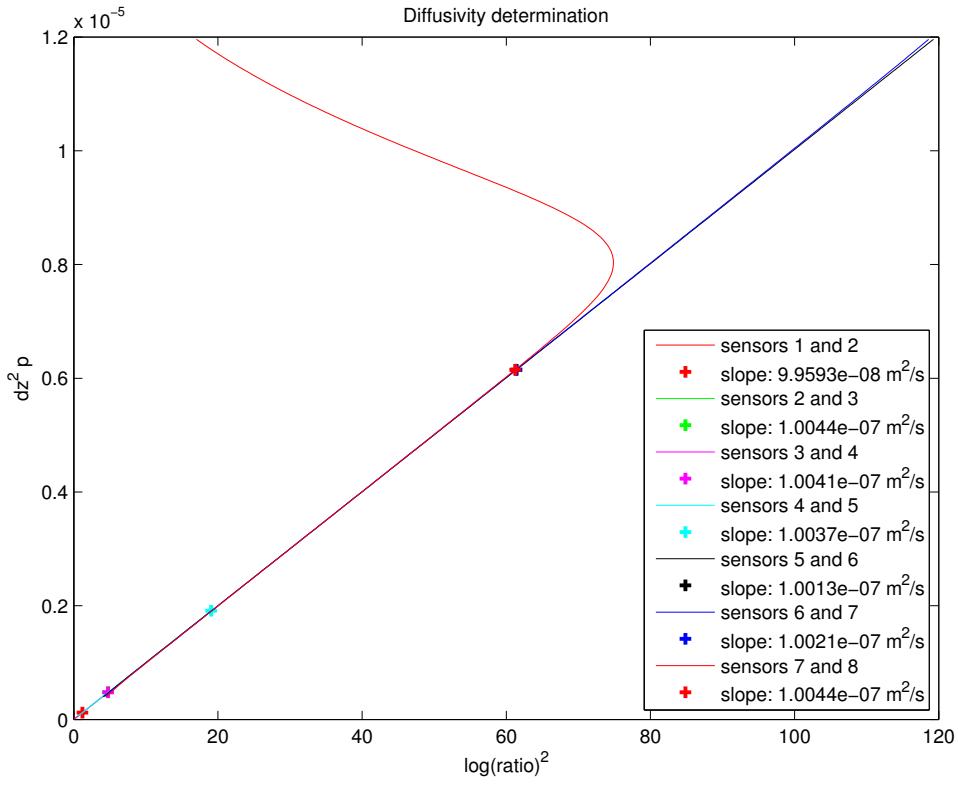

Figure 23: Diffusivity Determination for sensors at different depths

In figure 24, we took into consideration 8 sensors that share the same property ($r = z$) but their positions vary from $r = z = 4\ cm$ to $r = z = 18\ cm$, the values of α obtained using various sensors combinations are shown in figure 25 where we can see clearly that as the sensor's depth and distance from axis increase the error increases where it varies from 0.27% to 33.58%.

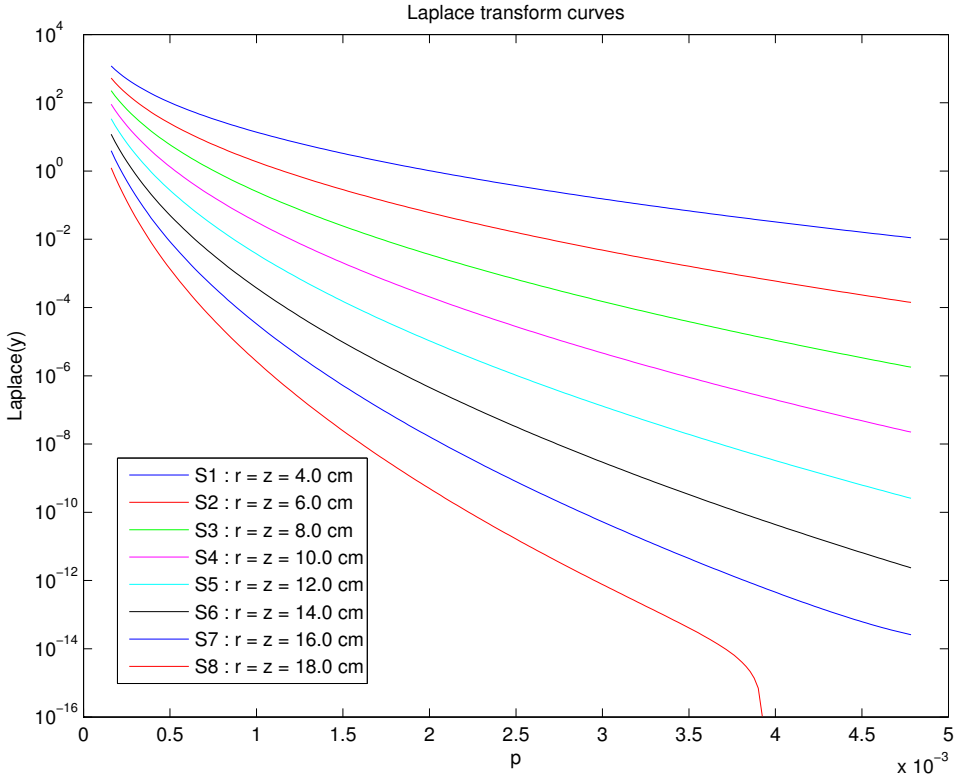

Figure 24: Laplace transform curves for different sensors having same r and z positions

Looking at the results obtained by synthetic data, we conclude that to obtain accurate result while using Laplace Transform method in 3D axisymmetric coordinate system, we must choose the sensors close to the axis to avoid large errors. Table 15 and table 16 represent different values of α obtained using different couples of sensors using real experimental data precisely experiments number 1 and 2. In the first experiment, we choose 3 couples of sensors that are not far from the axis where $0.32 \leq \frac{r}{R} \leq 0.46$ and the same time not far from the fire where $0.24 \leq \frac{z}{R} \leq 0.76$. For the second experiment, we also choose 3 couples of sensors that are not far from the axis where $0.28 \leq \frac{r}{R} \leq 0.42$ and the same time not far from the fire where $0.3 \leq \frac{z}{R} \leq 0.6$.

Sensors	1-2	2-3	7-8
Diffusivity α in m^2/s	3.21×10^{-7}	1.92×10^{-7}	2.77×10^{-7}

Table 15: The values of diffusivity (Fontainebleau Sand) obtained by Laplace transform method using the first experiment.

Sensors	10-14	7-11	11-13
Diffusivity α in m^2/s	3.28×10^{-7}	2.04×10^{-7}	2.02×10^{-7}

Table 16: The values of diffusivity (Fontainebleau Sand) obtained by Laplace transform method using the second experiment.

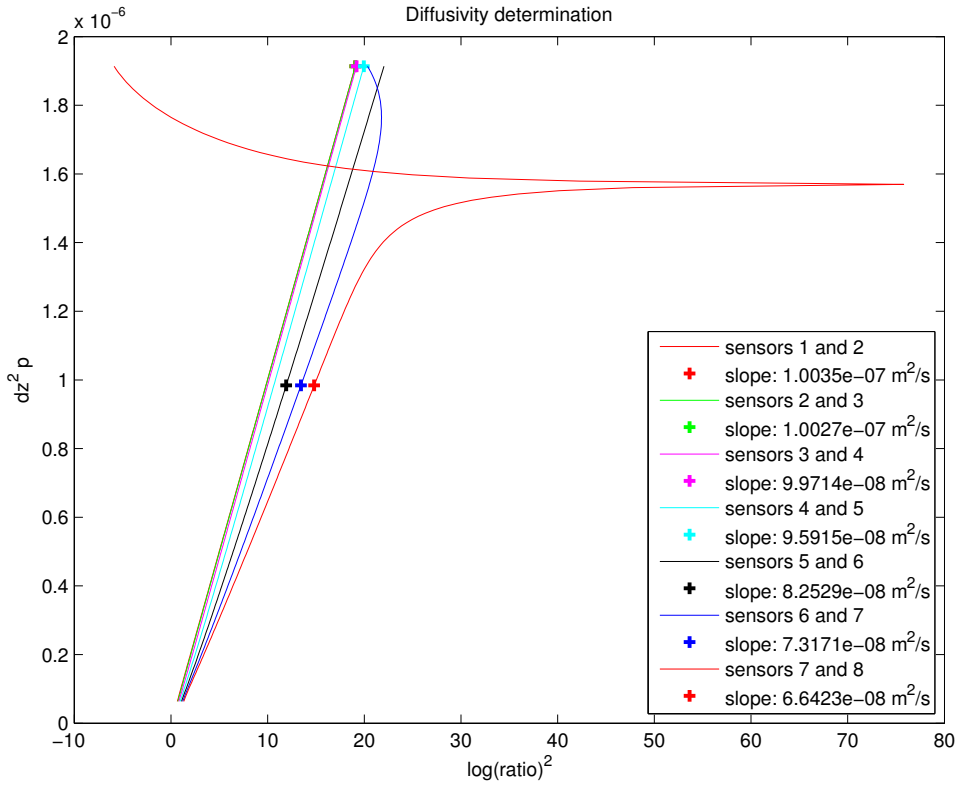

Figure 25: Diffusivity Determination for sensors having same r and z positions

The average value of α obtained using the first experiment is $2.63 \times 10^{-7} m^2/s$ and that using the second experiment is 2.44×10^{-7}.

10.3 Inverse Problem in 3D-axisymmetric coordinate system

Objective

Similar to the inverse problem visited earlier, we need to estimate the thermal diffusivity α of the soil by inverse problem knowing the history curves at selected few sensors of the domain. To do this, we use the least square criterion where we try to minimize the error function S which represents the difference between the experimental temperature and the numerical temperature:

$$S(\alpha) = \frac{1}{2} \sum_{i=1}^{M} \sum_{j=1}^{F} \left(T^f_{i,num} - T^f_{i,exp} \right)^2 \tag{61}$$

where $T^f_{i,num} = T(x_i, t^f)$ are the temperatures being the solution of the direct problem for the assumed parameter(α) at the point x_i, $i = 1, 2, ..., M$ for the time t^f, $f = 1, 2, ..., F$ and $T^f_{i,exp}$ is the measured experimental temperature at the same point x_i for time t^f.

10.3.1 Forward Problem

The physical problem consists of heating the dry soil by a fire (no phase change). To model this problem, we replace the soil by a perfect porous medium with constant and uniform properties heated from above by a constant temperature T_c. In this case, we assume that the thermal conductivity λ_e is independent of temperature and that the heat diffusion equation simplify to:

$$\frac{\partial T}{\partial t} = \alpha \left[\frac{1}{r} \frac{d}{dr} \left(r \frac{dT}{dr} \right) + \frac{d}{dz} \left(\frac{dT}{dz} \right) \right] \tag{62}$$

with the following initial and boundary conditions:

$$T(r_i, z_j, 0) = T_0(r_i, z_j) = T_{0,i,j} \qquad 1 \leq i \leq N,\ 1 \leq j \leq M$$

$$T(r_i, z_j, t) = T^D(r_i, z_j, t) = T^D_{i,j}(t) \qquad i \in \{1, N\} \text{ and } j \in \{1, M\}$$

$$\nabla T(r_i, z_j, t).\nu = 0 \ \text{ (Null Neumann) } \ i \in \{1, N\} \text{ and } j \in \{1, M\}$$

where T represents the temperature, T_0 is the initial temperature at $t_0 = 0$, T^D is T_c at the fire and T_0 elsewhere; ν indicates the outward unit normal vector along the boundary of Ω. The computational domain is similar to that presented in figure 9.

10.3.2 Method of Resolution

The method of resolution is similar to that presented earlier in section 5.1 where $p^{(k)} = (\alpha^{(k)})$ and the Jacobian(sensitivity) matrix ended up to be a vector:

$$J(p^{(k)}) = \begin{pmatrix} U^{1,(k)}_{\alpha 1,1} \\ \dots \\ U^{1,(k)}_{\alpha,M} \\ U^{1,(k)}_{\alpha 2,1} \\ \dots \\ U^{1,(k)}_{\alpha 2,M} \\ \dots \\ U^{1,(k)}_{\alpha N,1} \\ \dots \\ U^{1,(k)}_{\alpha N,M} \\ U^{2,(k)}_{\alpha 1,1} \\ \dots \\ U^{2,(k)}_{\alpha 1,M} \\ U^{2,(k)}_{\alpha 2,1} \\ \dots \\ U^{2,(k)}_{\alpha 2,M} \\ \dots \\ U^{2,(k)}_{\alpha N,1} \\ \dots \\ U^{2,(k)}_{\alpha N,M} \\ \dots \\ U^{F,(k)}_{\alpha 1,1} \\ \dots \\ U^{F,(k)}_{\alpha 1,M} \\ U^{F,(k)}_{\alpha 2,1} \\ \dots \\ U^{F,(k)}_{\alpha 2,M} \\ \dots \\ U^{F,(k)}_{\alpha N,1} \\ \dots \\ U^{F,(k)}_{\alpha N,M} \end{pmatrix} \tag{63}$$

where $U^{f,(k)}_{\alpha(i,j)} = \dfrac{\partial T^f_{(i,j)}}{\partial \alpha}|_{\alpha=\alpha^{(k)}}$. We try to find α by minimizing $S(\alpha)$ using LMA.

10.3.3 Goverened Equations and Numerical Strategy

As in 5.2, we need to differentiate the heat diffusion equation in dry case (62) with respect to the unknown parameter α:

$$\frac{\partial}{\partial \alpha}\left[\frac{\partial T}{\partial t}\right] = \frac{\partial}{\partial \alpha}\left(\alpha \operatorname{div}\left[\vec{\operatorname{grad}} T\right]\right) \tag{64}$$

which leads to the α sensitivity equation below:

$$\frac{\partial U_\alpha(x,t)}{\partial t} = \operatorname{div}\left(\vec{\operatorname{grad}} T(x,t)\right) \tag{65}$$

The initial and the boundary conditions for the sensitivity equation are similar to those in 5.2. The obtained system of coupled equations (heat diffusion equation and the sensitivity equation with respect to α) forms a system of partial differential equations. This system is solved using the same strategy explained in 1.1 and 2.4..

10.3.4 Results

We test our inverse problem using real experimental data because using synthetic data we are sure that the inverse problem will provide accurate results. The two experiments studied are experiments number 13 and 12 performed by José Augustin Cordero in the archeology lab of Rennes CREAAH. Both experiments are performed in the same manner as described in 10.1 using same box, heating plate and Fontainebleau sand but different sensors positions. The value of porosity (measured in the laboratory) for both experiments is $\phi = 0.4$. Using experiment 13 data, we obtained the results summarized in table 17.

Mesh	$\alpha \ \ m^2/s$	Residue
30×50	3.909×10^{-5}	6.581×10^3
120×200	3.203×10^{-5}	6.584×10^3
300×500	3.060×10^{-5}	6.582×10^3

Table 17: The values of diffusivity (Fontainebleau Sand) obtained by Inverse Problem using different mesh sizes.

As we have seen in previous sections, using Laloy/Massard and Laplace methods, the value of diffusivity is of order 10^{-7}. Also, examining the literature, the value of α for Fontainebleau sand is always found to be of order 10^{-7}. In [Com99], it is reported that the density of Fontainebleau sand is 1480 kg/m^3 and in other references, we can find that its volumetric heat capacity is 1.041×10^6 J/kgK and its thermal conductivity is 0.32 W/mK so we can simply deduce that 3.07×10^{-7} m^2/s. Moreover, [NYKE10] indicates that the thermal conductivity of Fontainebleau sand ranges between 0.28 and 0.42 W/mK and hence its diffusivity will be of order 10^{-7}. Comparing the results obtained in table 17 and what we mentioned now, we can see that the error is of order 10^2 which is huge.

We suspect that the error obtained is due to various experimental errors:

1. The measurement of sensors' positions is done after the end of the experiment where we start removing the sand by delving using a brush to locate the position of the thermocouples.

2. The position of the thermocouple is located by measuring the distance with respect to two perpendicular walls of the box (coordinates x and y) and the height to the upper edge thereof, with a string (having previously identified, in the same manner, the height of the sand surface, we deduce the depth of the thermocouples).

3. The Cartesian coordinates are then transformed into 3D axisymmetric coordinates.

4. Observing the measured initial temperatures, we notice that it is not uniform.

5. The circular heating plate used in the experiment do not have a uniform temperature.

To treat the error resulted from measuring the positions of sensors we propose to add the sensors' positions as unknowns in addition to diffusivity and we use a curve fit to treat the problem of having non-uniform initial temperature.

10.4 Estimation of α and sensors' positions

The analytical solution of the heat equation in 1D semi-infinite domain [CJ59] is of the form:

$$T(z,t) = k\mathrm{erf}\left(\frac{\mathrm{z}}{\sqrt{\alpha \mathrm{t}}}\right) \tag{66}$$

where T is the temperature, k is a constant, z is the position, α is the diffusivity and t is the time. Based on equation 66, it is not possible to find both the diffusivity of a solid medium and the position of the sensors, because there are an infinite number of solutions. This arise both in 1D problems, but also in 2D or 3D configurations.

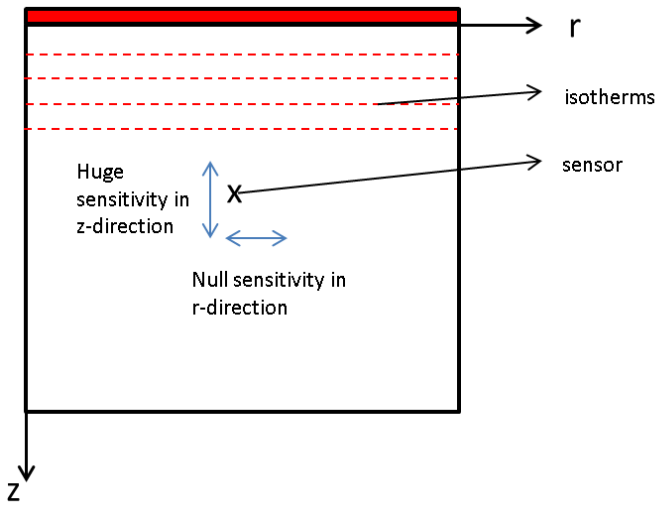

Figure 26: Simple 1D heating, showing the horizontal isothermal lines; there is a null sensitivity along the r-axis.

To obtain a unique solution, we must add some kind of constraints. As shown in figure 26, the 1D inverse problem has a null sensitivity along the r-axis. For the 3D-axi configuration, the isothermal lines during the heating process may be well approximated by a family of confocal ellipsoidal curves (see figure 27). As a result, we introduce a constraint by imposing the displacement normal to these isothermal lines. This writes:

$$\sin(\phi_i)\,\delta_{r_i} = \cos(\phi_i)\,\delta_{z_i} \tag{67}$$

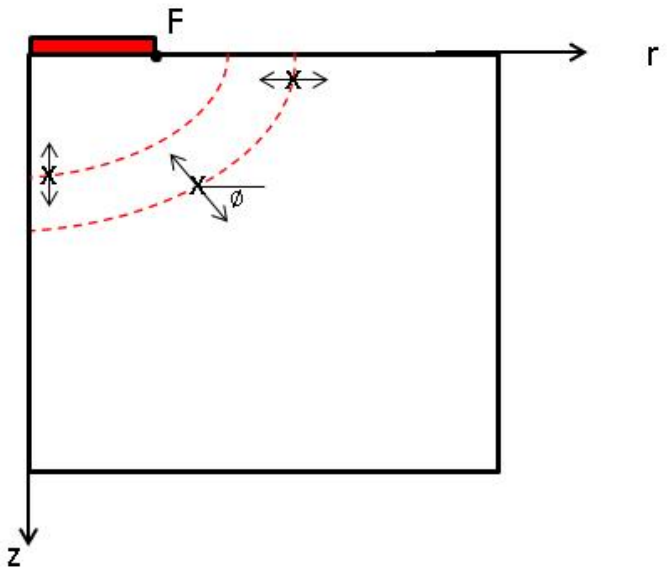

Figure 27: In a 3D-axi heating, the isothermal lines are very close to ellipsoidal curves; there is a null sensitivity in a direction tangent to these curves.

for all sensors i. In equation 67, ϕ_i denotes the angle of the isothermal lines at sensor number i. δ_{r_i} (resp. δ_{z_i}) is the (unknown) displacement of the sensor i in the r direction (resp. z direction).

Furthermore, we add new constraints about the unknown position of sensors, which specify that their mean displacement is zero, both in r and z direction. This comes from the assumption that the position errors obey a normal centered distribution law and it is generally true for a great number of sensors. This writes:

$$\sum_{i=1}^{n} \delta_{r_i} = 0 \quad \text{and} \quad \sum_{\mathrm{i}=1}^{\mathrm{n}} \delta_{\mathrm{z_i}} = 0 \tag{68}$$

for all sensors i.

Lastly, we suspect that a bias may be present in the measures.In any statistical investigation, we can always attribute some of the variation in data to measurement error, part of which can result from the measurement instrument itself. But human mistakes, especially recording errors (e.g., misreading a dial, incorrectly writing a number, not observing an important event, misjudging a particular behavior), can also often contribute to the variability of the measurement and thus to the results of a study. In our experiments, the way the sensors' positions were measured might be affected by a bias (see figure 28).

The bias is represented by a shift in both directions and it should also be added to the unknowns of the whole inverse problem. The new positions of the sensors can write:

$$\begin{aligned} \tilde{r_i} &= r_i + \delta_{r_i} + shift_r \\ \tilde{z_i} &= z_i + \delta_{z_i} + shift_z \end{aligned} \tag{69}$$

for all sensors i. $shift_r$ and $shift_z$ are the also unknowns, but global to our problem.

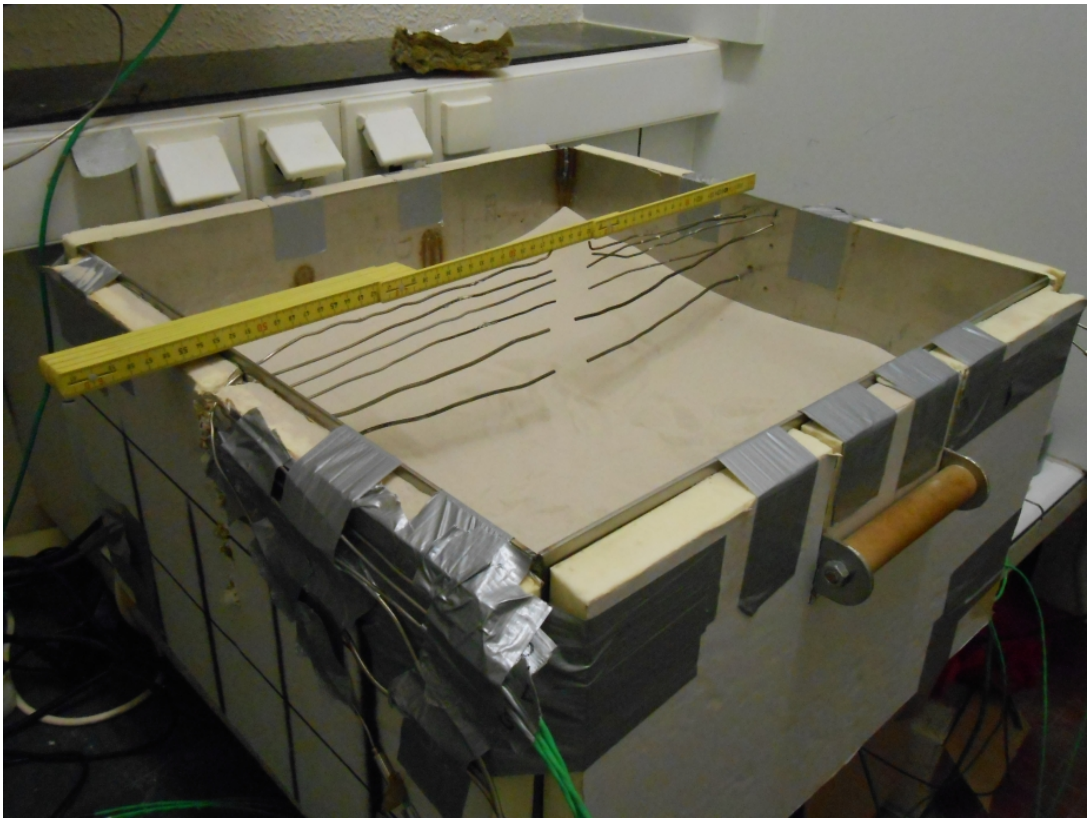

Figure 28: A bias in the measures might result from the way the sensors' position were calculated.

Reformulation of our inverse problem

The unknowns of our inverse problem are:

- Diffusivity α.
- The displacement in both r-direction and z-direction for each sensor i (δ_{r_i} and δ_{z_i}).
- The bias in both r and z directions ($shift_r$ and $shift_z$).

The constraints applied to our inverse problem are:

- $\sum_{i=1}^{n} \delta_{r_i} = 0 \quad \text{and} \quad \sum_{\mathrm{i}=1}^{\mathrm{n}} \delta_{\mathrm{z_i}} = 0.$
- For each sensor, $\sin(\phi_i)\, \delta_{r_i} = \cos(\phi_i)\, \delta_{z_i}$.
- Box constraints over the new sensors' position to ensure that they are in the physical domain under study.
- Box constraint over the value of α to assure that it will attain a positive value.

10.4.1 Numerical Strategy

The obtained system of coupled equations (heat diffusion equation 62 + sensitivity equation 65) is a system of partial differential equations. To solve this system, we use the same numerical strategy explained earlier (method of lines + finite volume method). After spatial discretization, the system of coupled equations can be written in the form of a system of first order implicit ODEs:

$$F(t, Y, Y') = 0 \text{ with } Y(t_0) = Y_0 \tag{70}$$

where $Y = [T,\ U_\alpha]$. The system in equation (70) can be solved by an ODE solver as in 1.1. The ODE solver will provide us with the values of T and U_α which are not enough to solve our inverse problem where the sensors positions are unknowns. To approximate $U_{r_i} = \frac{\partial T}{\partial r}$ and $U_{z_i} = \frac{\partial T}{\partial z}$ at each sensor i, we use an interpolation of order 2 based on a biquadratic interpolation (9-point stencil).

10.4.2 Solving the constrained inverse problem

The Levenberg Marquardt algorithm used in previous sections solves non-linear least square unconstrained problems. Our inverse problem here is a constrained non-linear least square problem with linear constraints (Equality constraints): $\sum_{i=1}^{n} \delta_{r_i} = 0$ and $\sum_{\mathrm{i}=1}^{\mathrm{n}} \delta_{\mathrm{z_i}} = 0$ and $\sin(\phi_i)\,\delta_{r_i} = \cos(\phi_i)\,\delta_{z_i}$. By using simple variable substitution, we transform the constrained non-linear least square problem into an unconstrained non-linear least square problem with less number of unknowns which can be easily solved using LMA with parameters' scaling (see 5.4).

10.4.3 Results

As we mentioned earlier, we are going to apply our inverse problem to two experiments.

First Experiment: The thermocouples in the first experiment are presented in figure 29, the fire radius is $R = 5\ cm$, the dimensions of the computational domain are $r = 25\ cm$ and $z = 30\ cm$. We can see clearly that the sensors are close to the z-axis and their distance from the axisymmetric axis with respect to fire radius is $(0.18 \leq \frac{r}{R} \leq 0.54)$ where as their depths with respect to the fire radius is $(0.18 \leq \frac{z}{R} \leq 1.96)$. Figure 30 shows the initial temperature profile for the first experiment: we can check that it is nearly uniform in space (the initial temperature varies from 24 to 26.5 C).

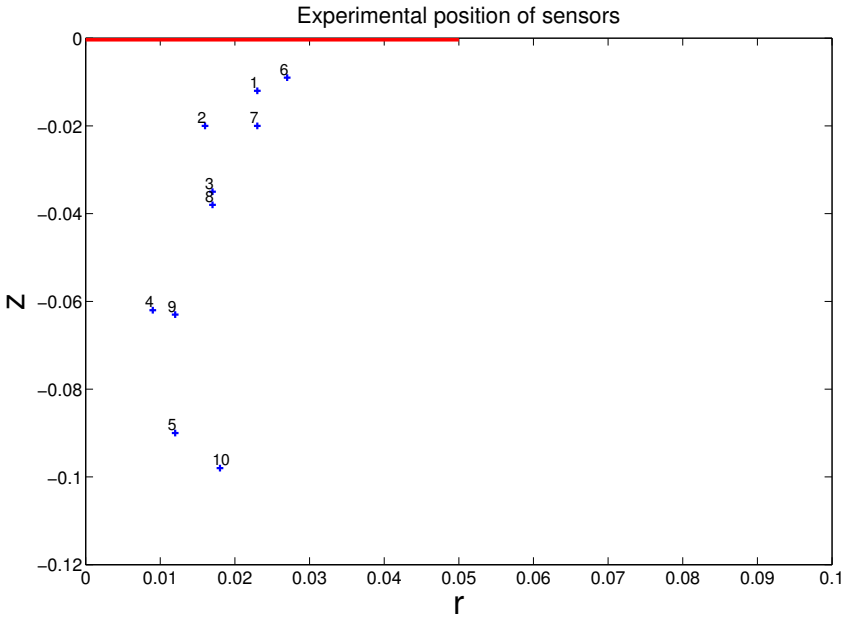

Figure 29: Experimental positions of 10 sensors in blue. The red bar represents the fire (unit is meter for both axes).

Using a numerical mesh of 300x500, we obtain a diffusivity $\alpha = 1.73 \times 10^{-7}\ m^2/s$ which is acceptable for Fontainebleau sand. The standard deviation for both δ_{r_i} and δ_{z_i} are respectively 0.8 mm and 5.5 mm, showing that the z position of the sensor is more difficult to be obtained

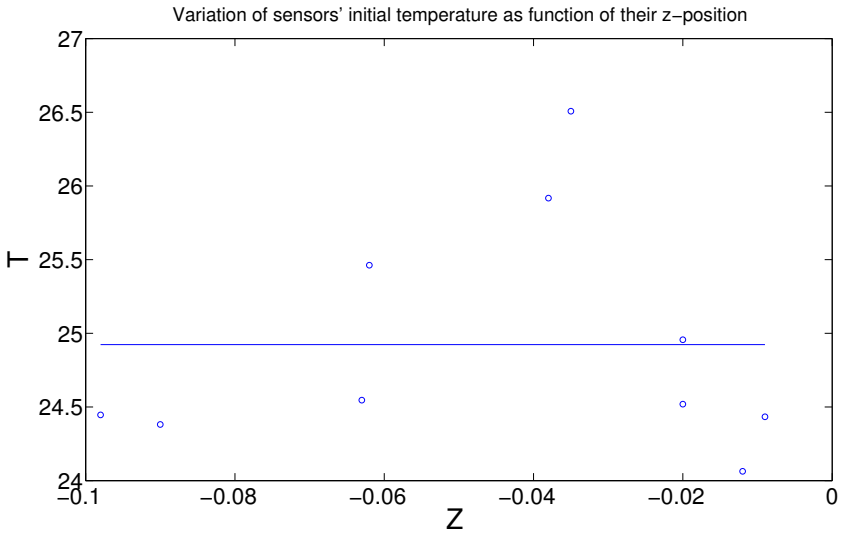

Figure 30: Initial temperature profile of the first experiment, showing that the temperature is nearly uniform.

accurately.

Besides, the shifts in r and z directions are respectively 1.4 mm and 9.8 mm; the last value appears to be somewhat too large to be attributed to an experimental protocol error. Therefore, the inverse problem must be investigated further on. However, we are confident of the numerical results, because the convergence curves during the iteration process are good (see figures 31 and 32).

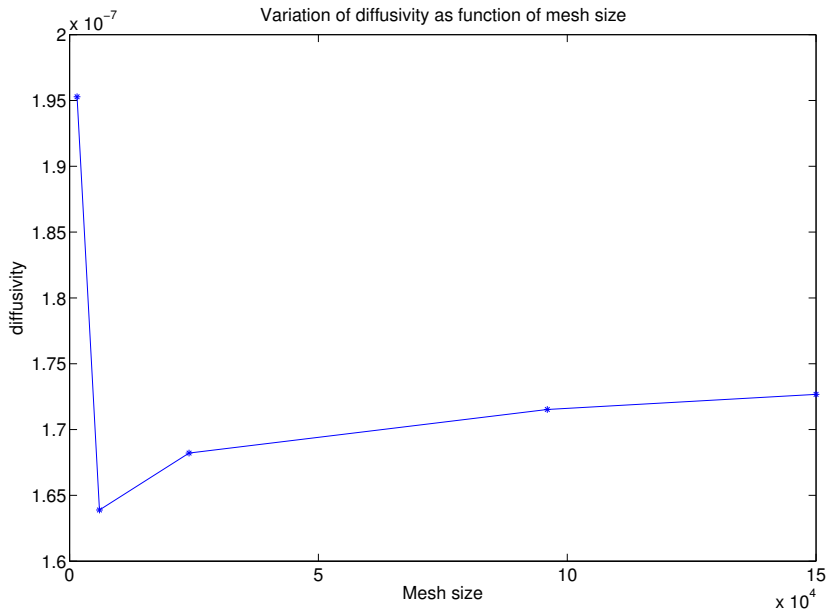

Figure 31: Variation of diffusivity as function of mesh size.The mesh size varies from 30×50 to 300×500. The value of diffusivity converges to 1.73×10^{-7}

Figure 33 represents the experimental (old) positions of sensors, the new numerical positions and the direction of displacement.

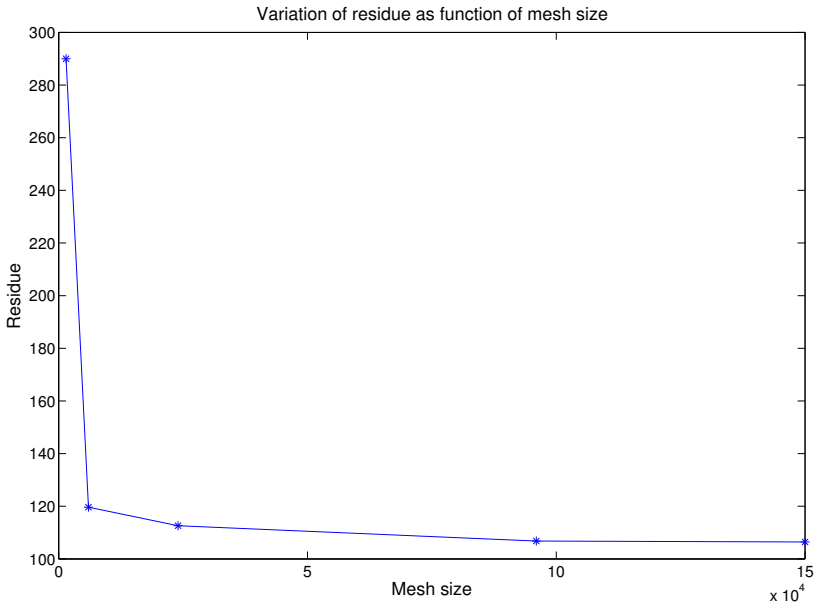

Figure 32: Variation of residue as function of mesh size. The mesh size varies from 30×50 to 300×500. The value of residue converges to 106.4

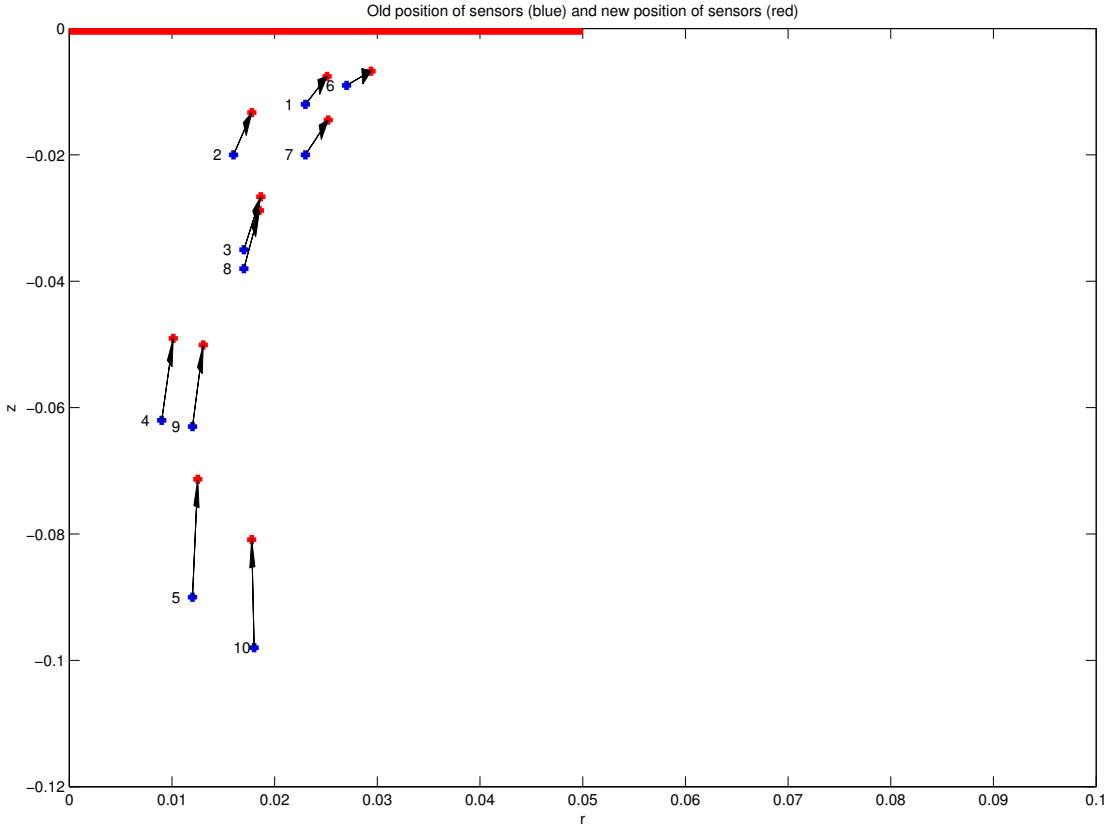

Figure 33: The experimental positions of sensors are in blue and the new calculated positions are in red (unit is meter for both axes).

Second Experiment: The thermocouples in the second experiment are presented in figure 34, the fire radius is $R = 5\ cm$, the dimensions of the computational domain are $r = 25\ cm$ and $z = 30\ cm$. We can notice that the sensors are spread in the domain unlike the first experiment where they were approximately close to the axisymmetric axis ($0.28 \leq \frac{r}{R} \leq 1.44$) and ($0.04 \leq \frac{z}{R} \leq 1.9$).

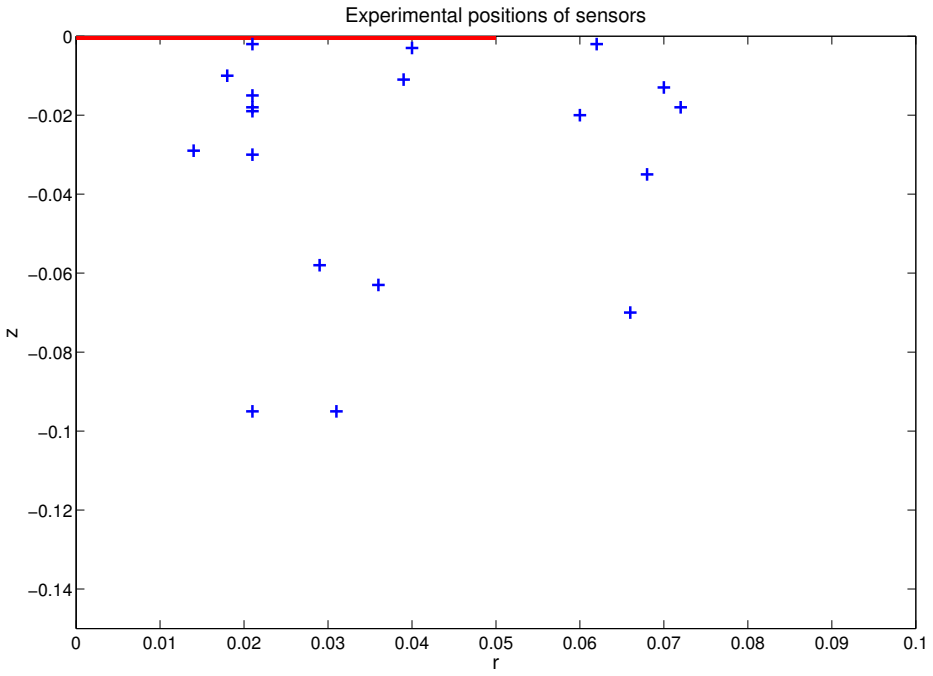

Figure 34: Experimental positions of 19 sensors in blue. The red bar represents the fire (unit is meter for both axes).

Figure 35 shows that the initial temperature for the second experiment is not uniform, it varies from 25.5 to 32 C). In figure 35 we modeled the initial temperature by constant mean of temperatures and in figure 36 we modeled the initial temperature profile by an exponential law. This kind of initial profile may reveal that the soil was not in equilibrium and that the experiment occurred during a slow transient heating of the medium due to an increase in the room temperature. Nevertheless, the fit of the temperature data to an exponential law works quite well.

Results: uniform initial temperature

Figures 37 and 38 presents that variation of thermal diffusivity as function of mesh size and the variation of residue as function of mesh size respectively for the case where we used a uniform initial temperature. Unfortunately, the curves are not well-convergent which incite us to use an exponential fit instead.

Results: Modeling initial temperature by an exponential fit

The use of an exponential fit to model the initial temperature results in very good converging curves presented in figures 39 and 40 respectively.

Using a 300×500 mesh, we used the inverse problem to find the value of diffusivity, δ_{r_i}, δ_{z_i}, $shift_r$ and $shift_z$ (i.e. the new sensors' positions). The values obtained using uniform (constant) and non-uniform (exponential fit) initial temperature are shown in table 18

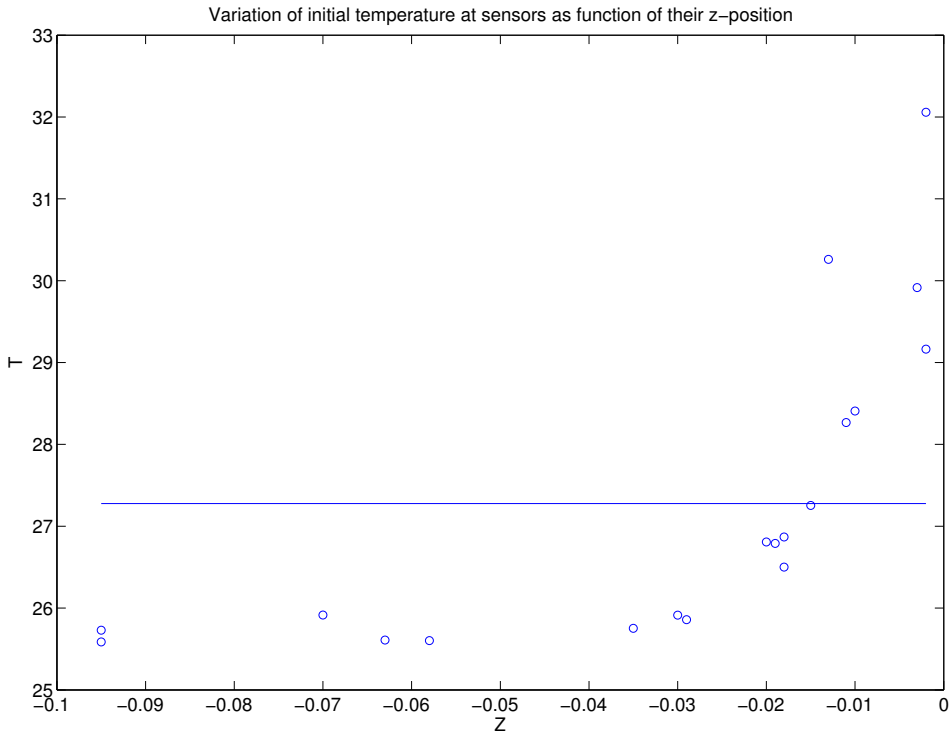

Figure 35: Modeling initial temperature by a constant temperature $T = 27.28\,\mathrm{C}$ (length is in meter)

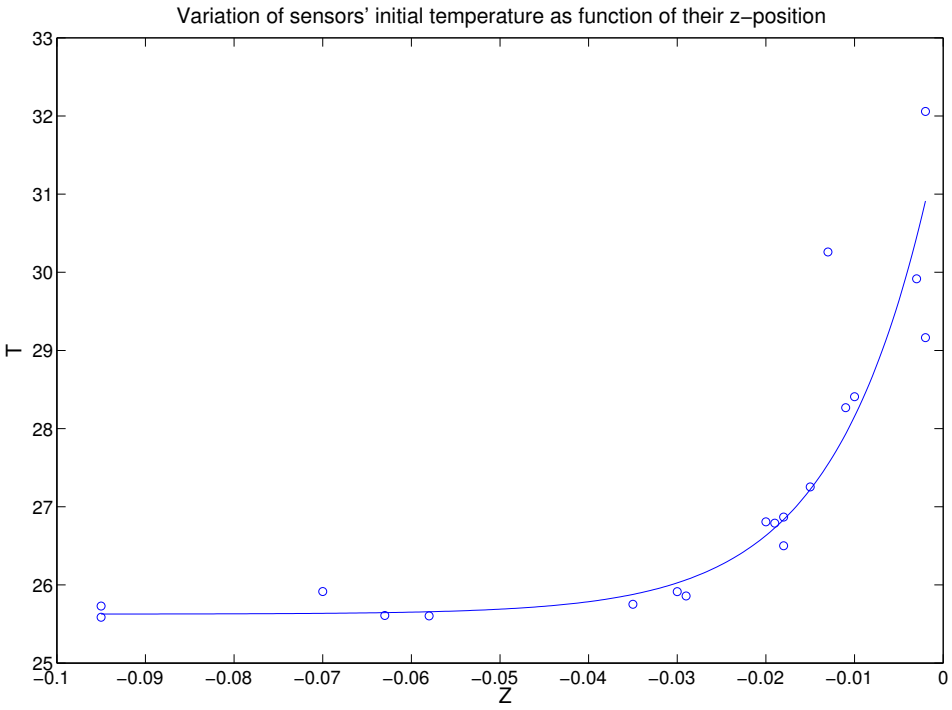

Figure 36: Modeling initial temperature by exponential fit $5.93\ \exp(70.95\ z) + 25.51$ (length is in meter, temperature in Celsius)

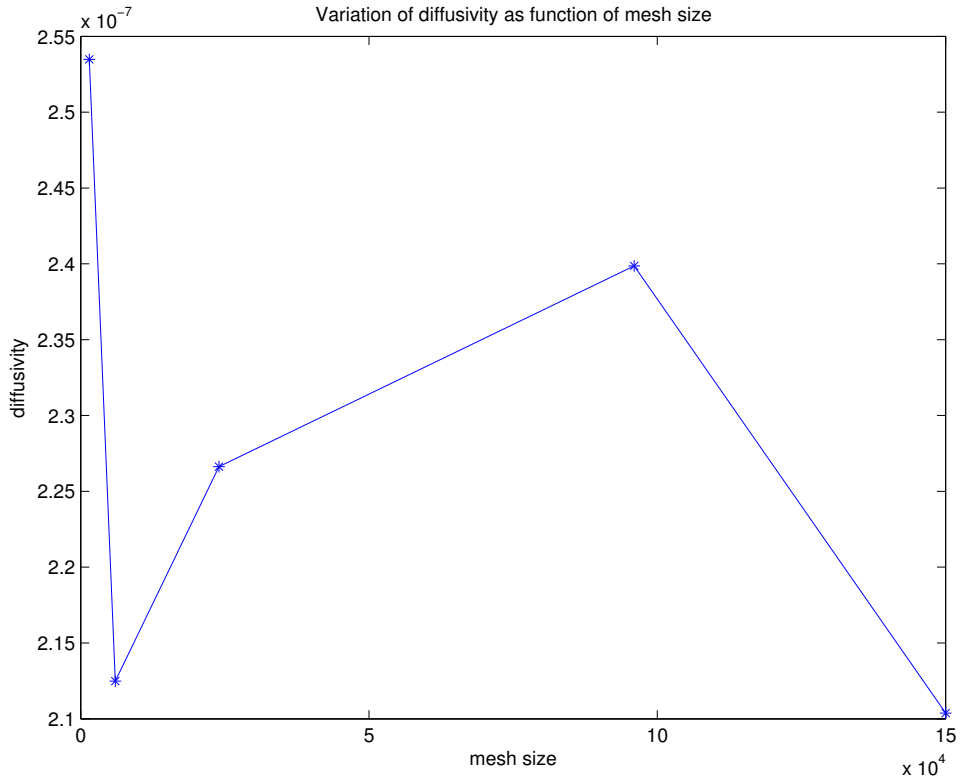

Figure 37: Variation of diffusivity as function of mesh size using constant initial temperature. Mesh varies from 30×50 to 300×500

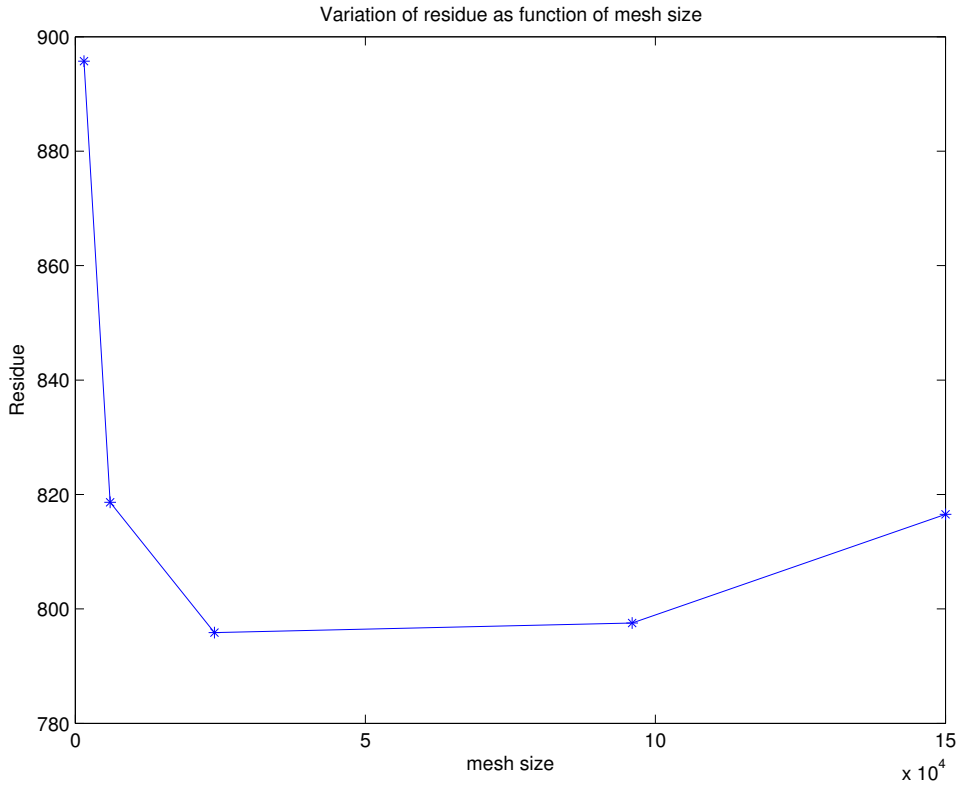

Figure 38: Variation of residue as function of mesh size using constant initial temperature.

The new value of diffusivity is $\alpha = 2.42 \times 10^{-7}\ m^2/s$ which is acceptable, even if two

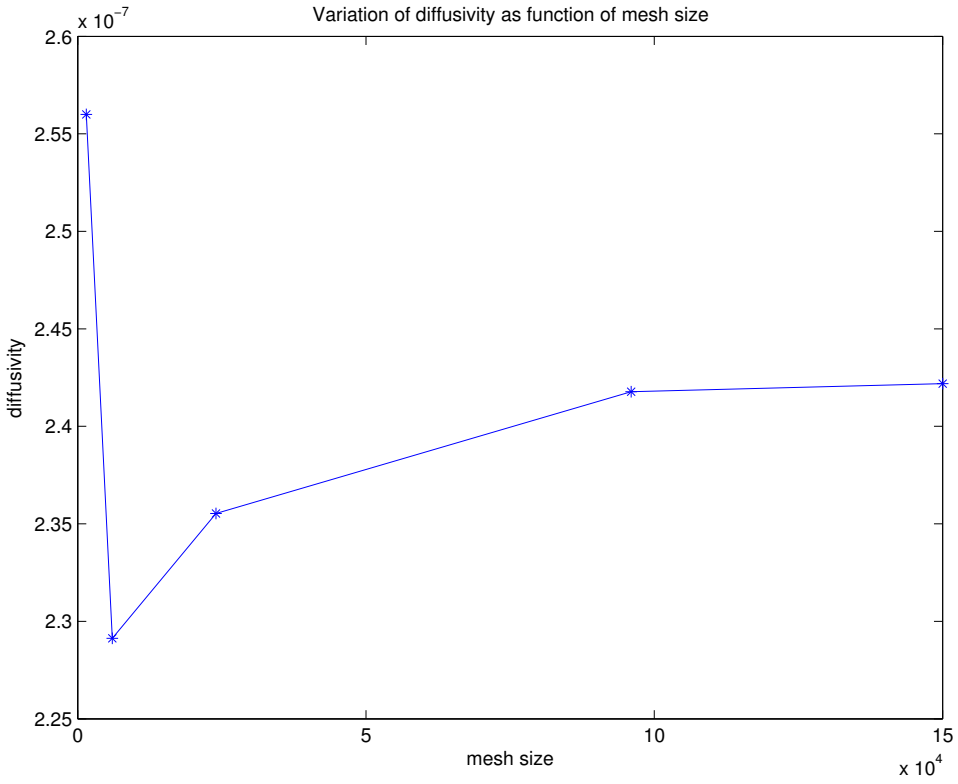

Figure 39: Variation of diffusivity as function of mesh size using an exponential fit for initial temperature.

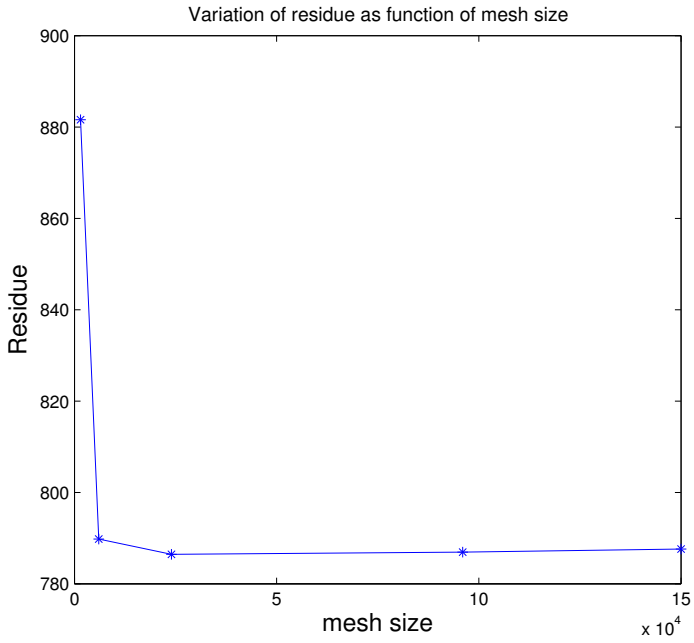

Figure 40: Variation of residue as function of mesh size using an exponential fit for initial temperature.

different values for the diffusivity have been obtained from experiment 1 and 2. Indeed, it is not uncommon to get some variations from one experiment to another, which may come from the ambient humidity of the room. In this second experiment, we are confident of the numerical results because we obtain diffusivity values close to the ones obtained by previously presented

	Uniform	Non-uniform
$\alpha(m^2/s)$	2.1×10^{-7}	2.42×10^{-7}
$shift_r(mm)$	-0.14	-6.64
$shift_z(mm)$	4.73	1.87
Standard deviation for $\delta_r(mm)$	6.58	1.97
Standard deviation for $\delta_z(mm)$	5.44	4.44
Residue	816.5	787.6

Table 18: Results obtained using the second experiment.

methods. The experimental (old) positions of sensors, the new numerical positions and the direction of displacement are presented in figure 41.

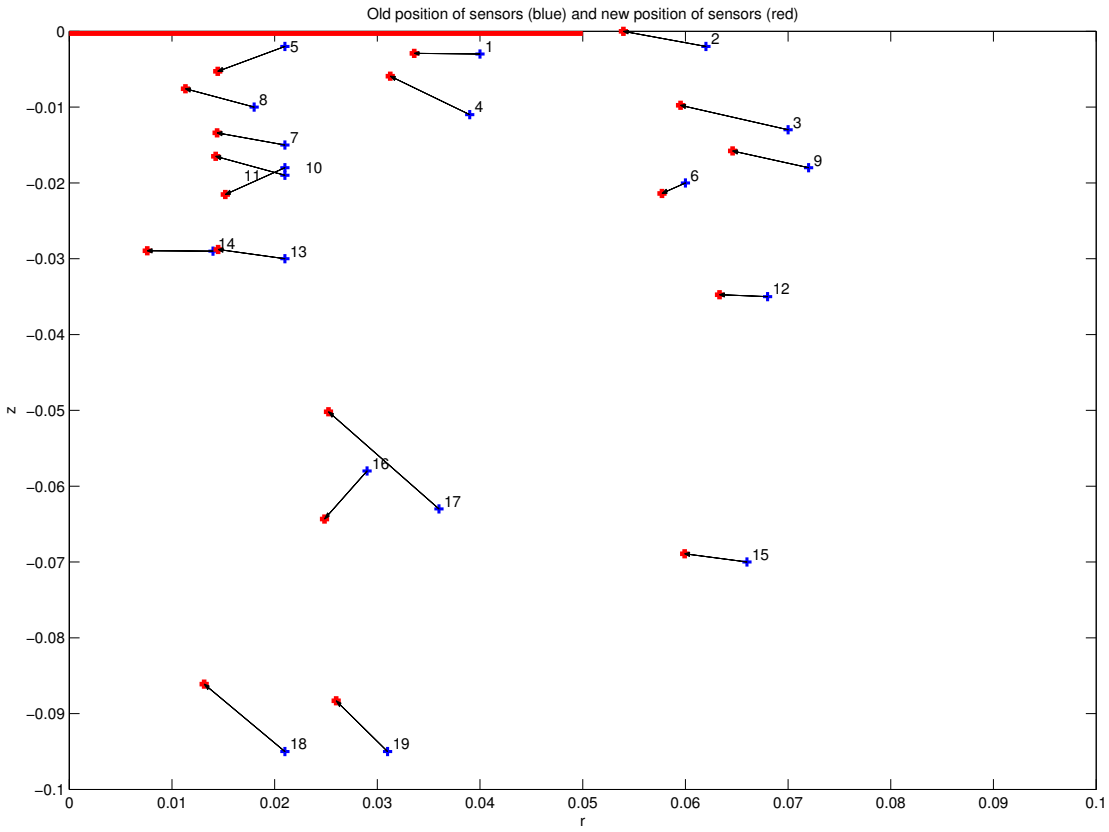

Figure 41: The experimental positions of sensors are in blue and the new calculated positions are in red (unit is meter).

11 Conclusion

As a conclusion, we saw that the use of statistical and box constraints were unavoidable since the thermocouples position is difficult to be measured accurately due to the experimental setup. Moreover, each experiment has its own conditions and thus data should be treated accordingly.

In both experiments, the Fontainebleau sand has been used, so we expect to obtain the same value for the thermal diffusivity. Unfortunately, we obtained different values ($\alpha = 1.73 \times 10^{-7}\ m^2/s$ and $\alpha = 2.42 \times 10^{-7}\ m^2/s$). We think that this discrepency comes from the fact that it is difficult to have well controlled experiments. However, it is worth noting that the magnitude order match quite well with our results obtained with other methods (Laloy & Massard, and the Laplace transform) and also with literature data.

References

[Bjö90] A. Björck. *Numerical methods for least squares problems*. Siam, 1990.

[Can] É. Canot. *Muesli Reference Manual - Fortran 95 implementation*. Available at http://people.irisa.fr/Edouard.Canot/muesli.

[CJ59] H. S. Carslaw and J. C. Jaeger. *Conduction of heat in solids*. Oxford: Clarendon Press, second edition, 1959.

[Com99] O. Combarieu. Caractérisation mécanique d'un massif de sable compacté : Cohérence des essais réalisés. Report NT 4233, Laboratoire régional des Ponts et Chaussées de Rouen, January-February 1999.

[Dav04] T. A. Davis. Algorithm 832: Umfpack, an unsymmetric-pattern multifrontal method. *ACM Transactions on Mathematical Software*, 30(2):196–199, 2004.

[DS83] J. E. Dennis and R. B. Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations*. Prentice-Hall, INC. New Jersey, 1983.

[Eck07] E. Eckmeier. *Detecting prehistoric fire-based farming using biogeochemical markers*. Doctoral thesis, University of Zurich, Faculty of Science, 2007.

[EK05] H. W. Engl and P. Kugler. Nonlinear inverse problems: theoretical aspects and some industrial applications. *Multidisciplinary Methods for Analysis, Optimization and Control of Complex systems*, 6:3–47, 2005.

[Heb73] M.D. Hebden. An algorithm for minimization using exact second derivatives. Report TP515, Atomic Energy Research Establishment, Harwell, England, 1973.

[Hin93] A. C. Hindmarsh. Description and use of LSODE, the Livermore solver for ordinary differential equations. Report 113855, Lawrence Livermore National Laboratory, Livermore, CA, 1993.

[Lev44] K. Levenberg. A method for the solution of certain nonlinear problems in least squares. *Quarterly of Applied Mathematics*, 2:164–168, 1944.

[LM84] J. Laloy and P. Massard. Nouvelle méthode thermique d'étude des foyers préhistoriques. *Revue d'Archéométrie*, 8:33–40, 1984.

[LSO13] M. A. Landunjoye, O. A. Sanuade, and A. A. Olaojo. Variability of soil thermal properties of a seasonally cultivated agricultural teaching and research farm. *Global Journal of Science Frontier Research Agriculture and Veterinary*, 13:40–64, 2013.

[Mar63] D. W. Marquardt. An algorithm for least squares estimation of nonlinear parameters. *SIAM Journal of Applied Mathematics*, 11:431–441, 1963.

[MCM09] M. Muhieddine, É. Canot, and R. March. Various approaches for solving problems in heat conduction with phase change. *International Journal on Finite Volumes*, 6(1):1–20, 2009.

[MCM12] M. Muhieddine, É. Canot, and R. March. Heat transfer modeling in saturated porous media and identification of the thermophysical properties of the soil by inverse problem. *J. Applied Numerical Mathematics*, 62:1026–1040, 2012.

[MGH80] J. J. Moré, B. S. Garbow, and K. E. Hillstrom. User guide for MINPACK-1. Report ANL-80-74, Argonne National Laboratory, 1980.

[MHB+12] G. Malherbe, J-F. Henry, A. El Bakali, C. Bissieux, and S. Fohanno. Measurement of thermal conductivity of granular materials over a wide range of temperatures. comparison with theoretical models. *Journal of Physics*, 395, 2012.

[MMS08] E. Majchrzak, B. Mochnaki, and J.S. Suchy. Identification of substitute thermal capacity of solidifying alloy. *Journal of Theoretical and Applied Mechanics*, 46(2):257–268, 2008.

[Mor78] J. J. Moré. The Levenberg-Marquardt algorithm: implementation and theory. *Numerical Analysis, Lecture Notes in Mathematics, Springer, Berlin*, 630, 1978.

[NYKE10] G. A. Narsilio, T. S. Yun, J. Kress, and T. M. Evans. Hydraulic and thermal conduction phenomena in soils at the particle-scale: Towards realistic fem simulations. *Materials Science and Engineering*, 10, 2010.

[OO00] M. N. Ozisik and H. R. B. Orlande. *Inverse heat transfer*. Taylor and Francis, 2000.

[PEZ+09] S. R. Pope, L. M. Ellwein, C. L. Zapata, V. Novak, C. T. Kelley, and M. S. Olufsen. Estimation and identification of parameters in a lumped cerebrovascular model. *Mathematical Biosciences and Engineering*, 6(1):93–115, 2009.

[SGT03] L. F. Shampine, I. Gladwell, and S. Thompson. *Solving ODEs with MATLAB*. Cambridge University Press, 2003.

[Sha08] S. Shan. *A Levenberg-Marquardt Method For Large-Scale Bound-Constrained Nonlinear Least-Squares*. Master thesis, The University of British Columbia, The Faculty of Graduate Studies(Computer Science), 2008.

[Szu10] D. Szubert. Étude de transferts de chaleur dans un foyer préhistorique expérimental. Rapport de stage de master 1, Université de Rennes 1, 2010.