Lecture 6. Inverse problems and regularized solutions

J.C. Batsale[1], O. Fudym [2]
Paper of METTI6 revised by C. Le Niliot[3]
[1] I2M Lab., Departement TREFLE, Université Bordeaux 1, CNRS, ENSAM, Bordeaux-INP, France
e-mail: jean-christophe.batsale@u-bordeaux.fr
[2] Université de Toulouse ; Mines Albi; centre RAPSOODEE, Albi, France
e-mail: fudym@mines-albi.fr
[3] IUSTI, Aix Marseille Univ. UMR 7343 CNRS, Marseille, France
e-mail : Christophe.LeNiliot@univ-amu.fr

Abstract. The methods for solving inverse problems must propose some consistent solutions despite their ill-posed character. Regularization is one of the major techniques for stabilizing the solution. We present in this lecture some generic examples as well as the main concepts within the linear estimation frame for the OLS (Ordinary Least Squares) estimator already studied in Lectures 1 and 3. The Singular Value Decomposition of the sensitivity matrix is used in order to analyse the solution. For such finite dimensional problems, the ill-posed behavior results in a bad-conditioned matrix computation.

1. Introduction

The reader could see in Lecture 1, “Getting started with problematic inversions with three basic examples”, some examples of generic inverse problems, which gave rise to envision the main characteristics that make their solution difficult. In Lecture 3, “Basics for linear inversion, the white box case”, the concepts and resolution of linear parameter estimation problems were presented, when using a direct model that computes the output from the knowledge of the input and some inner parameters used in the direct model.
The parameters to be recovered may be as well the passive structural parameters of the model (model identification), the parameters relative to the input variables, initial state, boundary conditions, some thermophysical properties, calibration, etc… For any of these considered cases, the output of the model can be properly computed if all the required information is available.

In a famous book, Hadamard (Hadamard 1923) introduced in 1923 the notion of well-posed problem. This is a problem whose solution:
- exists;
- is unique;
- depends continuously on the data.

Of course, these notions must be specified by the choice of space (and topologies) in which the data and the solution evolve. Problems that are not well-posed in the sense of Hadamard are said to be *ill-posed problems*. Note that the simple inversion of a well-posed problem may be either a well-posed or an ill-posed problem.

The example of 1D steady heat conduction in a wall discussed in Lecture 1, shows how the interpolation problem (that is the computation of $T(x)$ between the sensor location and the well-known boundary condition) is a well-posed problem, while the extrapolation problem (computation of $T(x)$ between the unknown boundary condition to be retrieved and the sensor location) is an ill-posed problem, since the estimation error may increase drastically.

The example of searching the slope of a line with two or more data points, such as discussed in Lecture 3, may be either a well-posed or an ill-posed problem:

- a unique and stable solution exists if all the data points fit on the same line (no noise in the data), and the time zero has not been chosen for some noisy data point. In that very specific case, the problem of finding the slope is well-posed.
- If, due to the noise in the measurement points, the data do not fit on the same line, a solution does not exist and the corresponding inverse problem of finding the slope is ill-posed.
- If the values of time for taking the measurements are not properly chosen (mostly close to zero), the solution is unstable, since the errors in the measurement may increase drastically – see the absolute and relative amplification coefficients such as defined in Lecture 1, and the corresponding inverse problem is ill-conditioned and may be considered as ill-posed.

The parameter estimation problem that consists in finding the vector of parameters by matching the measurements to the model outputs is most often an ill-posed problem, since it is generally over-determined (because the number of measurements m is greater than the number of parameters n), and has no solution because $\mathbf{y} \notin \mathrm{Im}(\mathbf{S})$. When the system is under-determined ($m < n$), it is also ill-posed because there is an infinite number of solutions. Moreover, when $m = n$, the problem may be well-posed if it were stable, but may also be unstable due to the effect of noise in the data.

In the present lecture, we will consider discrete inverse problems, where the number of parameters to be estimated is finite. When the quantities to be estimated are functions instead of discrete values, the corresponding problem is a continuous inverse problem which may be fully ill-posed. However, in many cases, the searched functions can be parametrized and conveniently approximated by a discrete inverse problem. It was typically the case for the 1D transient inverse heat conduction example in section 4 of Lecture 1, where the wall heat flux was to be estimated as a function of time. The heat flux at each time $\boldsymbol{t_i}$ is represented by a stepwise function $\boldsymbol{q_i}$.

The main challenge for such discrete function estimation problem is that the number of unknown is almost the same as the number of measurements, and the least squares approach is quite close to an exact matching procedure where only one observation is available for one estimated value. In this case the solution is highly sensitive to any ill-conditioned behaviour of the sensitivity matrix.

2. Some examples of typical ill-posed problems

We give hereafter some typical examples of ill-posed problems, such as derivation and deconvolution of experimental data. These examples are typical of the case of a parameterized function estimation problem. Instead of having a low number of parameters to be estimated with a high number of measurements, as for the example in Lecture 3 for estimating the slope and intercept of a straight line, the number of parameters to be estimated is herein very large and is quite of the same order as the number of observable data y , which makes the problem highly sensitive to noise. Unfortunately, in this case the inversion is often also amplifying the measurement noise.

2.1 Derivation of a signal

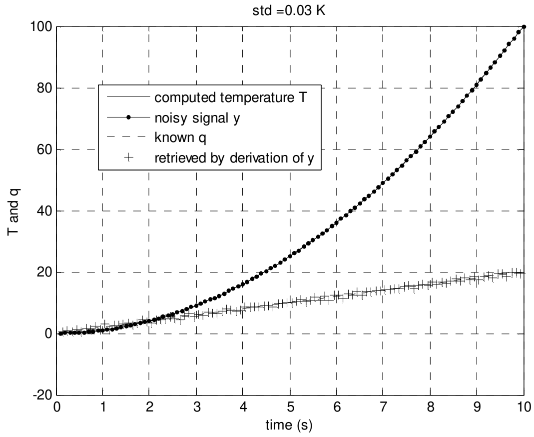
The derivation of a signal is often required for data processing. It is the case of time dependent functions, for instance, when deriving the time evolution of the mass of a product during drying or deducing the velocity of a body from the measurement of its position. A usual case in heat transfer is the problem of estimating the heat flux $q(t)$ exchanged by a body with uniform temperature $T(t)$ and volumetric heat capacity (ρC), with the lumped body approximation for a small thickness e . The heat balance can be written as

$$(\rho Ce)\frac{dT}{dt}=q(t) \qquad \text{with the initial condition } t=0 \quad T=0 \tag{6.1}$$

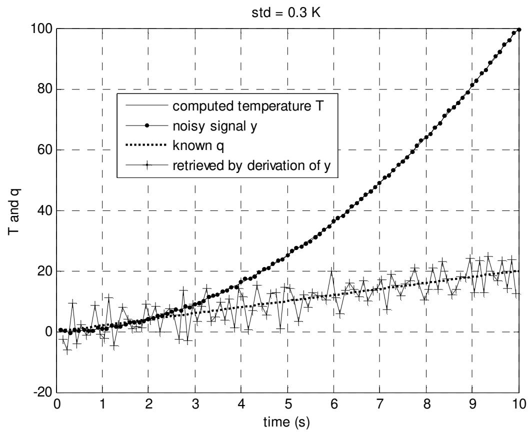
An inversion procedure is sought, for recovering an estimation of $q(t)$ from the measured temperature values $y(t_i)$, for different levels of the measurement noise, based on the following steps:

a. Choose some heat flux function, such as $q(t)=2t$ (arbitraly chosen here)
b. Compute the corresponding analytical solution $T(t) = t^2/(\rho Ce)$.
c. Add some random error, in order to simulate some experimental data, such as $y(t) = T(t) + \varepsilon(t)$
d. Retrieve the estimation by discrete derivation of the signal $\hat{q}(t) = (\rho Ce)\frac{\Delta y}{\Delta t} \approx (\rho Ce)\frac{dT}{dt}$
e. Repeat for different values of the Signal-to-Noise Ratio (characterized by different levels of standard deviation (std))

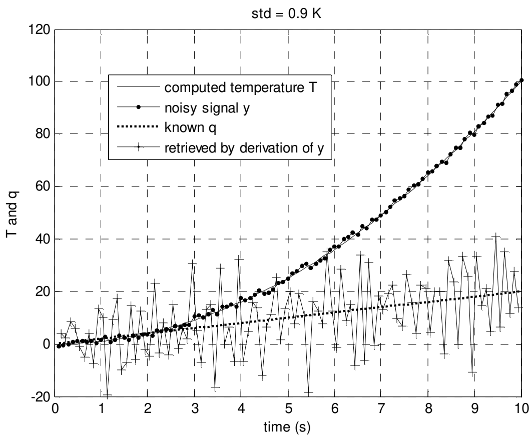
The results are depicted in Figure 1, assuming that $(\rho Ce) = 1$. When the standard deviation of the error is low, the heat flux is conveniently retrieved (Fig. 1a). For case (b), the noise on the signal y remains very low, in the sense it is still almost not visible in the corresponding curve. However, the heat flux is poorly computed. Increasing the level of noise, such as in Fig. 1c, where the std is *0.9 K*, results in a drastically poor computation of the heat flux. Thus, the numerical derivation of an experimental signal in order to retrieve its integral is an ill-posed problem, due to its unstable nature. The numerical derivation consists in computing the difference of successive measurements, divided by the time step. In this case the ill-posed character of the problem could be more dramatic as the time step decreases. As a result, the fluctuations in the identified function could be as important as we would like if Δt tends to 0. It is an illustration of the ill-posed character of the inverse problems: a bounded error can be amplified to infinity and the third condition of Hadamard is not satisfied!



(a)



(b)



(c)

Figure 1 – Derivation of an experimental signal (a) std = 0.03 K (b) std = 0.3 K (c) std = 0.9 K

2.2 Deconvolution of a signal

The deconvolution of a signal is also an operation often required when processing experimental data, for instance when searching the transfer function of a system or sensor, in image processing, optics, geophysics, etc... We give again the heat transfer example of some heat capacity exchanging with convective heat losses with the surrounding medium, such as

$$(\rho Ce)\frac{dT}{dt} = q(t) - hT \quad \text{with the initial condition } t = 0 \quad T = 0 \tag{6.2}$$

We assume here that $(\rho Ce) = 1, h = 1$ and that the area of the boundary surface of the body is 1.

Solving this equation by using the Laplace transformation of the temperature and heat flux with an analytical return to the time domain yields the solution in the form of the following convolution product:

$$T(t) = \int_0^t q(t-\tau)\, exp(-h\tau)\, d\tau = \int_0^t q(\tau)\, exp(-h\,(t-\tau))\, d\tau \tag{6.3}$$

The same approach as in previous example is proposed herein, such as

a. Choose some heat flux function, $q(t)$, called the input. Here the input function is chosen as:

$q(t) = q_0 exp\left(-\left(\frac{t-t_0}{\tau}\right)^2\right)$ with $q_0 = 10; t_0 = 0.5$ and $\tau^2 = 0.05$

b. Compute the corresponding analytical solution, that is the output $T(t)$, of the convolution product above.

c. Add some random error, such as $y(t) = T(t) + \varepsilon(t)$

d. Retrieve the heat flux by inverting the convolution product of this signal by the negative exponential above, that is the corresponding impulse response in this example (a numerical deconvolution). The impulse response can be noted as: $Imp(t) = exp(-h\,t)$.

e. Repeat for different values of the Signal-to-Noise Ratio (different levels of std: σ_ε)

The discrete approximation of expression (6.3) can be considered as a linear matricial expression between an input vector : $\boldsymbol{q}=[q_0\; q_1 \dots q_{n-1}]$ and an output vector $\boldsymbol{T}=[T_1\; T_2 \dots T_n]$, such as

$$\begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_n \end{bmatrix} = \Delta t \begin{bmatrix} Imp_1 & 0 & \vdots & 0 \\ Imp_2 & Imp_1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ Imp_n & \dots & Imp_2 & Imp_1 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ \vdots \\ q_{n-1} \end{bmatrix} \quad \text{or} \quad T_i = \Delta t \sum_{j=0}^{n-1} Imp_{i-j} * q_j \tag{6.3 bis}$$

With $Imp_k = \exp(-ht_k)$ and $t_k = k * \Delta t$ $\quad k = 1\ to\ n$

The sensitivity matrix which here depends on only one vector: $\boldsymbol{Imp}$=$[Imp_1\ Imp_2 \dots Imp_n]$ is called a lower triangular Toeplitz matrix, such as $\boldsymbol{S} = \boldsymbol{Toeplitz(Imp)}\,\Delta t$. Such a matrix is diagonal-constant.

The results from the Matlab script given in Appendix 1 are depicted in Figure 2. For a low standard deviation of the error (*std = 0.01 K*), the heat flux is conveniently retrieved by the deconvolution operation (simple inversion of the Toeplitz matrix). When increasing the noise level (*std = 0.1 K*), the drastic amplification of the errors in the deconvolution operation makes the result absolutely inaccurate. The increase of the noise level, that can be observed in the temperature plots, makes the solution (the input) inaccurate or even unavailable. This example shows that deconvolution of an experimental signal may be an ill-posed problem, depending on the functional form of the impulse response and on the noise level.

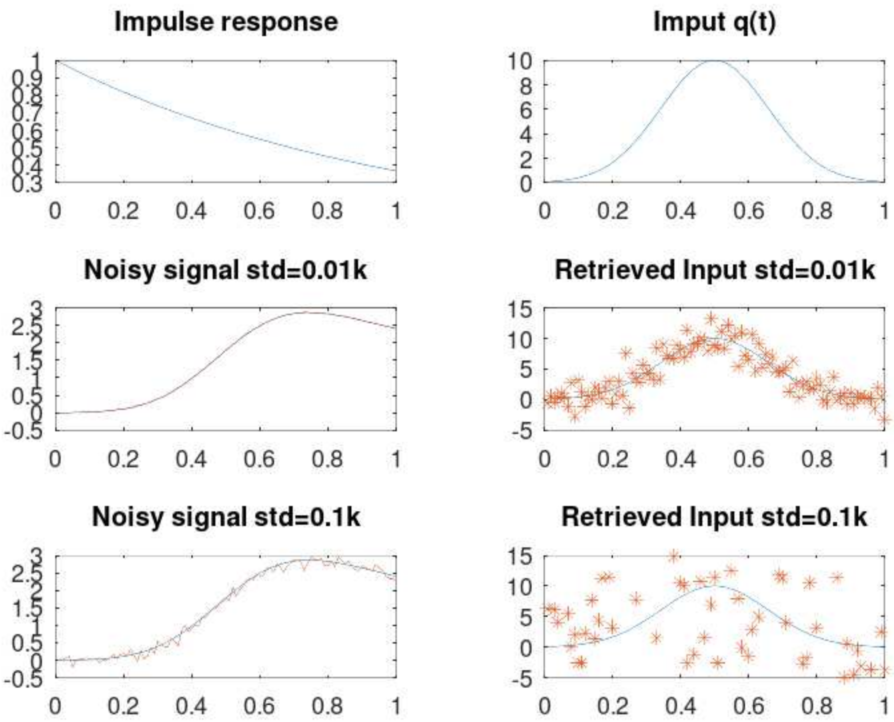


Figure 2 – Effect of the noise level on the deconvolution of a signal
(continuous blue line: exact signal, red line or dotted: noisy or retrieved signal)

3. Structure of a linear transformation and stability

3.1 Singular Value Decomposition of the sensitivity matrix

It was already shown in Lecture 3 of this school that the existence, uniqueness and stability of the solution of a discrete linear parameter estimation problem depends on the characteristics and structure of the rectangular sensitivity matrix S. Moreover, when the overdetermined problem y=Sx is considered as a least square problem given by the normal equations, it appears that the structure of the square information matrix $\boldsymbol{S^t S}$ has a major effect on the

propagation of the errors between the observed data and the output of the model. The anatomy of such a linear transformation is very clearly discussed in the text of S. Tan & C. Fox (Tan 2006).
One approach of interest in order to analyze this problem is to consider the Singular Value Decomposition of S (SVD). We assume herein that $m > n$ (overdetermined system, there is more data than parameters) and that S has only real coefficients. All the details of SVD analysis can be found in (Hansen 1998) and the corresponding routines are available in any of the linear algebra librairies (LAPACK, Num. Recipes, Matlab®, …).
The SVD of the matrix S is then written as

$$\begin{bmatrix} \\ \boldsymbol{S} \\ \\ \end{bmatrix} = \boldsymbol{U}\,\boldsymbol{W}\,\boldsymbol{V}^t = \begin{bmatrix} \\ \boldsymbol{U} \\ \\ \end{bmatrix} \begin{bmatrix} W_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & W_n \end{bmatrix} \begin{bmatrix} & \boldsymbol{V}^t & \\ & & \end{bmatrix} \tag{6.4}$$

where

- $\boldsymbol{U}$ is an orthogonal matrix of dimensions *(m,n)* : its column vectors (the *left* singular vectors of $\boldsymbol{S}$) have a unit norm and are orthogonal by pairs : $\boldsymbol{U}^t\boldsymbol{U} = \boldsymbol{I_n}$, where $\boldsymbol{I_n}$ is the identity matrix of dimension n. Its columns are composed of the first n eigenvectors U_k, ordered according to decreasing values of the eigenvalues of matrix $\boldsymbol{S}\boldsymbol{S^t}$. Let us note that, in the general case, $\boldsymbol{U}\boldsymbol{U^t} \neq \boldsymbol{I}_m$.

- $\boldsymbol{V}$ is a square orthogonal matrix of dimensions *(n,n)* , : $\boldsymbol{V}\,\boldsymbol{V}^t = \boldsymbol{V}^t\,\boldsymbol{V} = \boldsymbol{I_n}$. Its column vectors (the *right* singular vectors of $\boldsymbol{S}$), are the n eigenvectors $\boldsymbol{V_k}$, ordered according to decreasing eigenvalues, of matrix $\boldsymbol{S^t}\boldsymbol{S}$;

- $\boldsymbol{W}$ is a square diagonal matrix of dimensions *(n,n)*, that contains the n so-called singular values of matrix S, ordered according to decreasing values : $W_{\boldsymbol{1}} \geq W_{\boldsymbol{2}} \geq \cdots W_{\boldsymbol{n}}$ The singular values of matrix S are defined as the square roots of the eigenvalues of matrix $\boldsymbol{S^t}\boldsymbol{S}$.

In Lecture 3, the Singular value Decomposition of the reduced sensitivity matrix, through the analysis of its singular values, was used to demonstrate that its condition number is a criterion that can be used to measure the degree of ill-posedness of the OLS estimator, regardless of the noise level.

As previously seen in Lecture 3, the Ordinary Least Squares solution is obtained by minimizing the distance between the output of the direct model $\boldsymbol{Sx}$ and the data $\boldsymbol{y}$, which is done by the orthogonal projection of the data on the space spanned by the column vectors of $\boldsymbol{S}$. This is equivalent to minimizing the objective function

$$J_{\,OLS}(\boldsymbol{x}) = \|\boldsymbol{y} - \boldsymbol{Sx}\|^2 = (\boldsymbol{y} - \boldsymbol{Sx})^t\,(\boldsymbol{y} - \boldsymbol{Sx}) \tag{6.5}$$

The minimization of $J_{\,OLS}(\boldsymbol{x})$ yields the OLS estimator, computed with Eq. (3.24) in Lecture 3. Applying the Singular Value Decomposition to the sensitivity matrix yields

$$\hat{x}_{\,OLS} = (\boldsymbol{S^t}\,\boldsymbol{S})^{-1}\,\boldsymbol{S^t}\,\boldsymbol{y} = \boldsymbol{V}\;\boldsymbol{W^{-1}}\,\boldsymbol{U^t}\,\boldsymbol{y} \tag{6.6}$$

In this case, if the standard statistical assumptions hold (see Lecture 3), the covariance matrix of the OLS estimator can be written as

$$cov(\boldsymbol{x}) = {\sigma_\varepsilon}^2 \boldsymbol{V} \ \boldsymbol{W}^{-2} \boldsymbol{V}^t \tag{6.7}$$

Eqs. (6.6) and (6.7) are valid if the sensitivity matrix $\boldsymbol{S}$ is of full rank, which means that its smaller singular value w_n is strictly positive. The condition number is then defined as

$$cond(\boldsymbol{S}) = \frac{w_1}{w_n} \tag{6.8}$$

3.2 Spectral analysis of the OLS estimator

Applying SVD to the normal equations (see Eq. (3.23) in Lecture 3) in order to find the OLS estimator in the diagonal basis yields

$$(\boldsymbol{S}^t \boldsymbol{S}) \ \widehat{\boldsymbol{x}}_{\boldsymbol{OLS}} = \boldsymbol{S}^t \boldsymbol{y} \ \Rightarrow \ \boldsymbol{V} \ \boldsymbol{W} \boldsymbol{U}^t \boldsymbol{U} \boldsymbol{W} \boldsymbol{V}^t \widehat{\boldsymbol{x}}_{\boldsymbol{OLS}} = \boldsymbol{V} \ \boldsymbol{W} \boldsymbol{U}^t \boldsymbol{y} \tag{6.9}$$

where the estimation problem can be reconsidered now with the new parameter vector $\boldsymbol{b} = \boldsymbol{V}^t\boldsymbol{x}$ and a new observable vector : $\boldsymbol{z} = \boldsymbol{U}^t\boldsymbol{y}$, such as

$$\boldsymbol{W} \ \widehat{\boldsymbol{b}}_{\boldsymbol{OLS}} = \boldsymbol{z} \tag{6.10}$$

The unicity of the solution is confirmed here when the sensitivity matrix $\boldsymbol{S}$ is of full rank, i.e. $r = n$, which is possible only if $m \geq n$ (more data than parameters). When $r < n$, the matrix has not full rank, and the number of parameters to be estimated must be reduced, or some parameters must be determined in an arbitrary form.

The linear transformation of the data $\boldsymbol{y}$ also yields a new covariance matrix associated to the observable measurement noise. Hopefully, we can note that this operation does not affect the covariance of the error of the transformed signal $\boldsymbol{z}$ (here for the standard assumptions):

$$cov(\boldsymbol{z}) = \boldsymbol{U}^t cov(\boldsymbol{y}) \ \boldsymbol{U} = {\sigma_\varepsilon}^2 \ \boldsymbol{U}^t \ \boldsymbol{U} = {\sigma_\varepsilon}^2 \boldsymbol{I} \tag{6.11}$$

Hence the covariance matrix of the error of $\widehat{\boldsymbol{b}}_{\boldsymbol{OLS}}$ is computed by

$$cov(\widehat{\boldsymbol{b}}_{\boldsymbol{OLS}}) = {\sigma_\varepsilon}^2 \ \boldsymbol{W}^{-2} \qquad \text{or} \quad cov(\widehat{\boldsymbol{b}}_{\boldsymbol{OLS}}) = \begin{bmatrix} \frac{{\sigma_\varepsilon}^2}{{w_1}^2} & . & \boldsymbol{0} \\ . & . & . \\ \boldsymbol{0} & . & \frac{{\sigma_\varepsilon}^2}{{w_n}^2} \end{bmatrix} \tag{6.12}$$

The above equation shows that an effect of noise amplification appears due to the fact that the eigenvalues have a wide range of orders of magnitude. It is of particular interest to note in Eq. (6.12) that the covariance matrix of the estimator in the diagonal basis is linking the square of the singular values to the variance of noise, that is to the level of uncertainty in the measurement errors.

A small perturbation applied to a single component k of $\boldsymbol{z}$, such as

$$\boldsymbol{\delta z} = \delta z_k \boldsymbol{U_k} \tag{6.13}$$

yields the following variation to the OLS estimator

$$\boldsymbol{\delta \widehat{b}}_{\boldsymbol{OLS}} = \frac{\delta z_k}{w_k} \boldsymbol{V_k} \tag{6.14}$$

which implies a relative variation corresponding to

$$\frac{\|\boldsymbol{\delta \widehat{b}}_{\boldsymbol{OLS}}\|}{\|\boldsymbol{\delta z}\|} = \frac{1}{w_k} \tag{6.15}$$

Thus the singular values indicate how the same perturbation yields different effects on the components of the estimator. Moreover, this relative variation may increase drastically when the singular values are close to zero. The relative variation between two components of respective index k and h is given by the ratio $\frac{w_k}{w_h}$. Hence the maximum relative variation factor is obtained between the first and the last component, such as $\frac{w_1}{w_n} = cond(\boldsymbol{S})$, which is the condition number of the sensitivity matrix, as seen in Eq. (6.8). If is not too large, the problem is said to be well-conditioned and the solution is stable with respect to small variations of the data. Otherwise the problem is said to be ill-conditioned. It is clear that the separation between well-conditioned and ill-conditioned problems is not very sharp and that the concept of well-conditioned problem is more vague than the concept of well-posed problem.

3.3 Example of a simple ill-conditioned matrix

$$\begin{bmatrix} 1 & 1 \\ 1 & 1.01 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$ the inversion yields $$\begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Let's give a perturbation of 1% on the second data point, such as

$$\begin{bmatrix} 1 & 1 \\ 1 & 1.01 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1.01 \end{bmatrix}$$ the inversion yields $$\begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Hence the perturbation of the data makes the solution of the matrix inversion, that is the solution of the square linear system of 2 equations with two unknowns, surprisingly as far as possible from the original solution. The solution is quite unstable.

The eigenvalues are (2.005, 0.005), and the condition number is 402 >> 1.

4. Regularization

In the previous section it was shown how the ill-posed estimation problem is turned into an ill-conditioned problem by the least squares approach. Equations. (6.6) and (6.7) show that the unstable behavior of the pseudo-inverse of the sensitivity matrix can be straightly addressed by the means of the singular values diagonal matrix $\boldsymbol{W}$. Regularization is a process for searching some acceptable solution, by reducing the effect of measurement errors on the estimate. Several approaches may be used for this purpose. The main idea is to reduce the effect of the "small" singular values on the obtained solution, while trying to avoid that this "smooth" solution to be too different from the true-but-untractable solution. The two main approaches are

(i) *Apply some prior information as a constraint*
The most usual methods are truncation of the diagonal basis and the parametrization of the solution (reduction of the number of parameters)

(ii) *Apply some penalization to the objective function*
Some weighted prior information is included in the objective function

4.1 Truncated SVD (TSVD)

TSVD of order α is obtained through replacing in Eq. (6.6) the inverse of the matrix $\boldsymbol{W}$ by the truncated inverse $\boldsymbol{W_\alpha}^{-1}$ where the smaller singular values are removed (their inverse put to zero), such as

$$\boldsymbol{W_\alpha}^{-1} = \begin{bmatrix} 1/w_1 & & & & 0 & & \\ & 1/w_2 & & & & & \\ & & \ddots & & & & \\ & & & 1/w_\alpha & & & \\ & 0 & & & 0 & & \\ & & & & & \ddots & \\ & & & & & & 0 \end{bmatrix} \tag{6.16}$$

The regularized TSVD estimator is:

$$\hat{\boldsymbol{x}}_{\boldsymbol{\alpha}}^{\boldsymbol{TSVD}} = \boldsymbol{V}\ \boldsymbol{W_\alpha}^{-1}\,\boldsymbol{U}^t\,\boldsymbol{y} \tag{6.17}$$

We note that only $\boldsymbol{W_\alpha}^{-1}$ can be computed, and not matrix $\boldsymbol{W_\alpha}$ since an infinite value has been assigned to the $n-\alpha$ smallest singular values $W_{\alpha+1}$, $W_{\alpha+2}$, … , W_{n-1} , W_n .

Equation (6.17) may be written using the left and right singular column vectors $\boldsymbol{U_k}$ and $\boldsymbol{V_k}$ defined in section 1, such as

$$\hat{\boldsymbol{x}}_{\boldsymbol{\alpha}}^{\boldsymbol{TSVD}} = \sum_{k=1}^{\alpha} \frac{1}{w_k}\ \left(\boldsymbol{U_k}^t\boldsymbol{y}\right)\boldsymbol{V_k} \tag{6.18}$$

The choice of the truncation order α can be considered by taking into account the observable measurement noise such as:

$$W_\alpha \geq \sigma_\varepsilon > \ W_{\alpha+1} \tag{6.19}$$

Unfortunately, it is not always easy to determine the standard deviation of the measurement noise for example in the case of non-stationary signal. The choice of the truncation order without the knowledge of σ_ε will be discussed in section 4.4.

4.2 Tikhonov regularization of zero order

The important idea in introducing some regularization by penalizing the objective function is the will to include some prior knowledge relative to the parameters to be retrieved. For instance, the parameter should not be very far from a reference value, or the time history of the function to be estimated should be smooth… A widespread regularization method by penalization of the OLS objective function is Tikhonov regularization.

We present herein the Tikhonov regularization of order zero, which yields the minimization of the following objective function:

$$J_{\mu}(\boldsymbol{x}) = \|\boldsymbol{y} - \boldsymbol{S}\boldsymbol{x}\|^2 + \mu\left\|\boldsymbol{x} - \boldsymbol{x}_{prior}\right\|^2 = (\boldsymbol{y} - \boldsymbol{S}\boldsymbol{x})^{\boldsymbol{t}}(\boldsymbol{y} - \boldsymbol{S}\boldsymbol{x}) + \mu\left(\boldsymbol{x} - \boldsymbol{x}_{prior}\right)^{\boldsymbol{t}}\left(\boldsymbol{x} - \boldsymbol{x}_{prior}\right) \quad (6.20)$$

where the real positive number μ is the regularization parameter. The value μ =0 yields the OLS solution where no regularization applies. Increasing μ tends to force the solution to be close to the prior estimate $\boldsymbol{x}_{prior}$

Equation (6.20) is solved by:

$$\widehat{\boldsymbol{x}}_{\boldsymbol{\mu}}^{Tik0} = (\boldsymbol{S}^{\boldsymbol{t}}\,\boldsymbol{S} + \mu\boldsymbol{I}_{\boldsymbol{n}})^{-1}\left(\boldsymbol{S}^{\boldsymbol{t}}\,\boldsymbol{y} + \mu\boldsymbol{x}_{prior}\right) \quad (6.21)$$

Applying SVD to the sensitivity matrix $\boldsymbol{S}$ and using $\boldsymbol{V}\boldsymbol{V}^{\boldsymbol{t}} = \boldsymbol{I}_{\boldsymbol{n}}$ yields:

$$\widehat{\boldsymbol{x}}_{\boldsymbol{\mu}}^{Tik0} = \boldsymbol{V}\left(\boldsymbol{W}^{\boldsymbol{2}} + \mu\boldsymbol{I}_{\boldsymbol{n}}\right)^{-1}\left(\boldsymbol{W}\boldsymbol{U}^{\boldsymbol{t}}\,\boldsymbol{y} + \mu\boldsymbol{V}^{\boldsymbol{t}}\boldsymbol{x}_{prior}\right) \quad (6.22)$$

Equation (6.22) clearly shows that the regularization parameter will cancel the noise amplification effect of the smallest singular values in the diagonal matrix $\left(\boldsymbol{W}^{\boldsymbol{2}} + \mu\boldsymbol{I}_{\boldsymbol{n}}\right)$ to be inverted. Nevertheless, the cost of this stabilization is also obvious, since the non-zero regularization parameter value yields that the information of the experimental data in $\boldsymbol{y}$ is biased by the prior information ($\boldsymbol{x}_{prior}$). Hence let's point out that the regularized solution aims to balance accuracy and stability requirements.

4.3 Example: Regularization for deconvolution

The experimental derivation and deconvolution example given in section 2 can be solved as a linear estimation problem. The function estimation problems are highly sensitive to noise, since the number of unknown matches the number of function components to be retrieved (exact matching: the sensitivity matrix is a square matrix).

Expression (6.17) (TSVD method) and (6.21) (Thikonov method) are used for the regularization of the output of the model given by equations (6.2) and (6.3), for $h = \rho C e = 1$ and with the same input as the one presented in figure 2.

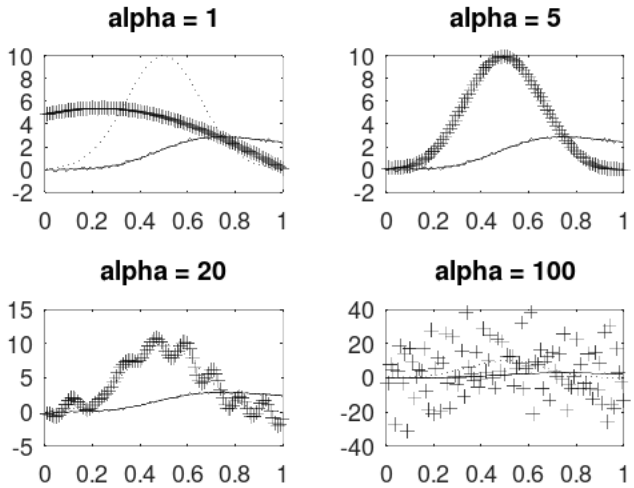


Figure 3 – Deconvolution and inversion with TSVD regularization
— temperature (exact and noisy); - - exact input ; + estimated input.

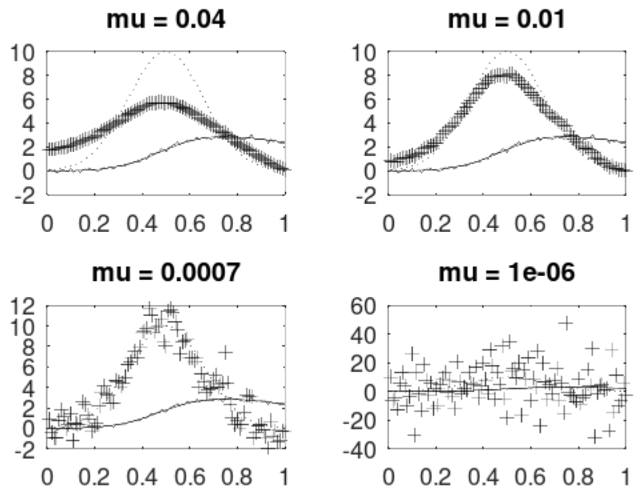


Figure 4 – Deconvolution and inversion with Tikhonov regularization
— temperature (exact and noisy); - - exact input ; + estimated input.

Figure 3 and 4 show how increasing the value of the regularization parameter has a positive effect regarding the stabilization of the heat flux time history to be retrieved, while this effect is counter balanced by the apparition of a bias with the original solution. It is of great interest to point out that the correct possible values of the regularization parameter are related to the signal to noise ratio. In the case of the TSVD method, the truncation parameter α is near 5 and the corresponding singular values are : $w_3 = 0.124$ $w_4 = 0.898$ and $w_5 = 0.706$. In the case of the Thikonov method, the regularisation parameter μ is quite close to the variance of the measurement error (here $\sigma_\varepsilon{}^2$=0.01K).

The Matlab codes related to the figures 3 and 4 are given in appendix 2 and 3.

4.4 The regularization parameter

The optimal choice of the value of the regularization parameter is a nontrivial problem for which numerous solutions have been proposed. Such problem is accentuated if the variance of the measurement noise is poorly known. The L-curve method (due to Hansen,1998) has become a popular method, which is implemented by the graphical analysis of a log-log plot (or ordinary plot) obtained by varying the value of the regularization parameter, as shown in figure 5. For each value of μ, the norm of the distance between the data and the model is reported on the horizontal axis, while the distance of $\boldsymbol{x}$ to $\boldsymbol{x}_{prior}$ is reported on the vertical axis. Very often the vector $\boldsymbol{x}_{prior}$ is set to zero initially. An iterative process can be further implemented. The L-curve selection criterion consists in locating the value which maximizes the curvature, that is the L-curve corner which separates the two regions: under-regularized on the left, over-regularized on the right

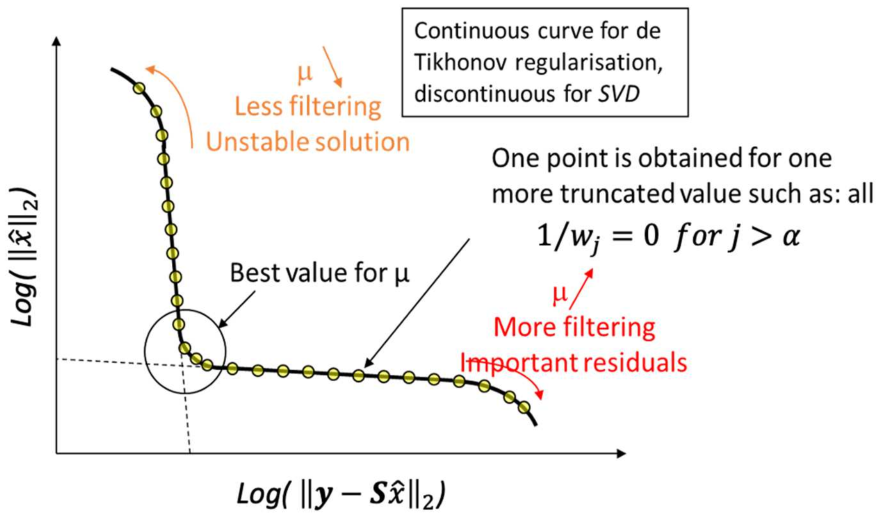


Figure 5 – L Curve, choice of the Tikhonov regularization parameter μ, comparison with truncated SVD solution

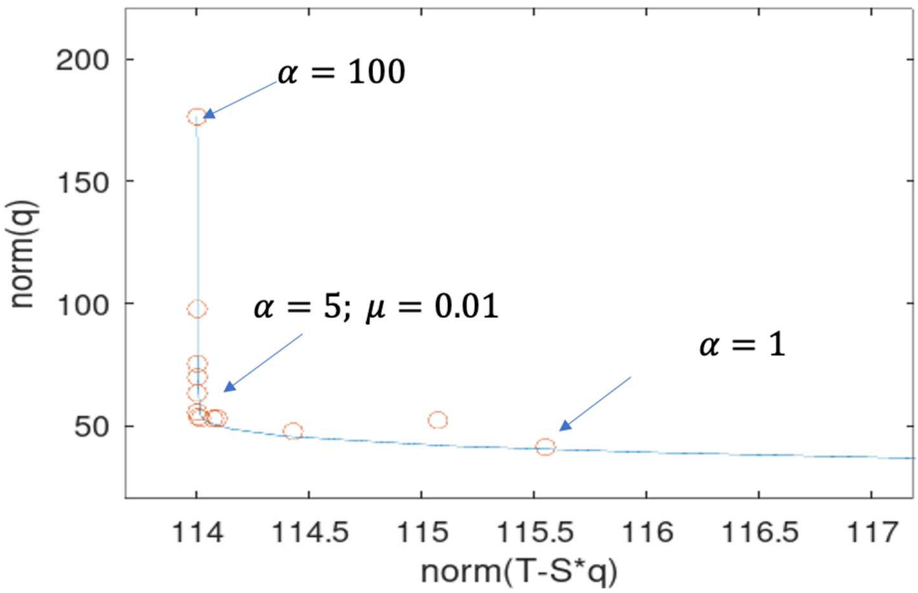


Figure 6 – L-curve obtained for the regularization of the deconvolution example

The L-curve is plotted for the previous regularized deconvolution example on figure 6. The red circles are the regularisation parameters obtained with the TSVD method (α from 1 to n). The continuous line is related to the parameter μ. The best regularization parameter μ is near the same value as the variance of the measurement noise. The best truncation number α is near 5 . It is difficult to distinguish the results with $\alpha = 4$ or $\alpha = 6$. All these points are very close to the point that maximizes the curvature of the L-curve.

The Matlab code related to figure 6 is given in appendix 4.

6. Conclusions

Regularization is an important step for solving ill-posed problems. When the inverse problem is of finite dimension, which is the case for discrete estimation problems, the existence of a solution is achieved by the least squares approach, and the problem is in fact ill-conditioned. For function estimation problems, the parametrization of the function to be retrieved tends to exact matching, where the number of experimental data is equal to the number of parameters. This case is generally highly sensitive to the measurement noise. Regularization stabilizes the solution by removing the effect of the smallest singular values which amplify the effect of these measurement errors. However the cost of regularization is a biased stabilized solution, hence the value of the regularization parameter (Tikhonov parameter or truncation level) must be carefully chosen.

References

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

Aster R. C., Borchers B., Thurber C. H., *Parameter Estimation and Inverse Problems*, Elsevier Academic Press, 2005

Beck, J. V. and Arnold K. J., *Parameter estimation in engineering and science*, John Wiley & Sons, 1977

THERMAL MEASUREMENTS AND INVERSE TECHNIQUES, Edited by Helcio R. B. Orlande, Olivier Fudym, Denis Maillet, Renato M. Cotta, CRC Press, New York, ISBN : 978-1-4398-4555-4, 2011

Ozisik, M.N. and H.R.B. Orlande. 2000. Inverse Heat Transfer: Fundamentals and Applications, Taylor and Francis, New York.

Campbell, S. L. and C. D. Jr Meyer. 1991. Generalized Inverses of Linear Transformations. New York: Dover.

Hansen, P.C., 1998. Rank Deficient and Ill-Posed Problems: Numerical Aspects Of Linear Inversion, SIAM.

Jenkins, G.M. and D.G. Watts. 1998. Spectral analysis and its applications, Emerson-Adams Press.

Tan, S., C. Fox and G. Nicholls. 2006. Inverse Problems, Course Notes for Physics 707, University of Auckland, 2006. http://www.math.auckland.ac.nz/%7Ephy707/

Tikhonov, A. and V. Arsénine. 1976. Méthodes de résolution des problèmes mal-posés, Editions de Moscou.

Shenfelt, J.R., R. Luck, R.P. Taylor and J.T. Berry. 2002. Solution to inverse heat conduction problems employing SVD and model reduction. IJHMT 45:67-74.

Fudym, O., C. Carrère-Gée, D. Lecomte and B. Ladevie. 2003. Drying kinetics and heat flux in thin layer conductive drying. Int. Com. on Heat and Mass Transfer, 30 (3) 335-349.

Maillet, D., S. Andre, J.C. Batsale , A. Degiovanni and C. Moyne. 2000. Thermal quadrupoles-Solving the heat eq. through int. transforms, John Wiley.

D. Petit, D. Maillet, *Techniques inverses et estimation de paramètres (Inverse techniques and parameter estimation),* Editeur : Techniques de l'Ingénieur, Paris. Thème : Sciences Fondamentales, base : Physique-Chimie, rubrique : Mathématiques pour la physique.
- Dossier AF 4515, pp. 1- 18, janvier 2008.
- Dossier AF 4516, pp. 1-24, janvier 2008.

Appendix 1

```
%Calculations related to the figure 2
% Deconvolution influence of the noise
% fi=fi0*exp(-((t-t0)/tau)*2)
% dT/dt = q - hT
% T=conv(q,exp(-ht))
% Tr = T + noise

N=100;dt=0.01;t0=dt*N/2;t=dt*(1:N);
q0=10;h=1;q=q0*exp(-20*(t-t0).^2);

% noise
std1=0.01;, std2=0.1;
noise1=std1*randn(size(t));noise2=std2*randn(size(t));

Imp=exp(-h*(t(1:N)));%Impulse response
X=dt*toeplitz(Imp, zeros(1,N)); % Sensitivity matrix

T=X*q'; % Direct model (convolution)
Tr1=T'+noise1;Tr2=T'+noise2;% noise simulation
G=inv(X'*X);
qr1=G*X'*Tr1'; qr2=G*X'*Tr2'; %OLS inversion

subplot(3,2,1), plot(t,Imp),title('Impulse response');
subplot(3,2,2), plot(t,q),title('Imput q(t)');
subplot(3,2,3), plot(t,T,t,Tr1), title(['Noisy signal std=' num2str(std1) 'k'])
subplot(3,2,4), plot(t,q,t,qr1,'*'), title(['Retrieved Input std=' num2str(std1) 'k'])
subplot(3,2,5), plot(t,T,t,Tr2), title(['Noisy signal std=' num2str(std2) 'k'])
subplot(3,2,6), plot(t,q,t,qr2,'*'), title(['Retrieved Input std=' num2str(std1) 'k'])
```

Appendix 2

```
%Calculations related to the figure 3
% Deconvolution with the TSVD method

N=100;dt=0.01;t0=dt*N/2;t=dt*(1:N);
q0=10;h=1;q=q0*exp(-20*(t-t0).^2);

% noise
std=0.1;
noise=std*randn(size(t));

Imp=exp(-h*(t(1:N)));%Impulse response
S=dt*toeplitz(Imp, zeros(1,N)); % Sensitivity matrix

T=S*q'; % Direct model (convolution)
Tr=T'+noise;% noise simulation

%TSVD

alph=[1 5 20 100];

for p=1:4
```

```
 [U,W,V]=svds(S,alph(p));
  qr=V*diag(1./diag(W))*U'*Tr';
 %  J(p)=norm(Tr-S*qr);
  % K(p)=norm(qr);

subplot(2,2,p),plot(t,T,'k',t,Tr,'k',t,q,'k:',t,qr,'k+'),
   title(['alpha = ',num2str(alph(p))])
   figure(gcf);
end
```

Appendix 3

```
%Calculations related to the figure 4
% Deconvolution with the Thikonnov method

N=100;dt=0.01;t0=dt*N/2;t=dt*(1:N);
q0=10;h=1;q=q0*exp(-20*(t-t0).^2);

% noise
std=0.1; %standard deviation of the noise
noise=std*randn(size(t));

Imp=exp(-h*(t(1:N)));%Impulse response
S=dt*toeplitz(Imp, zeros(1,N)); % Sensitivity matrix

T=S*q'; % Direct model (convolution)
Tr=T'+noise;% noise simulation

%TSVD

mu=[0.04 0.01 0.0007 0.000001];

for p=1:4

  %Thikonov regularization
   G=inv(S'*S+mu(p)*eye(N));
    qr=G*S'*Tr';

 %  J(p)=norm(Tr-S*qr);
 %  K(p)=norm(qr);

subplot(2,2,p),plot(t,T,'k',t,Tr,'k',t,q,'k:',t,qr,'k+'),
   title(['mu = ',num2str(mu(p))])
   figure(gcf);
end
```

Appendix 4

```
% Deconvolution and inversion with regularization related to figure 6
%L-Curve and comparisons netween TSVD and Thikonov regularization methods

clear
```

```
N=100;dt=0.01;t0=dt*N/2;t=dt*(1:N);
q0=10;h=1;q=q0*exp(-20*(t-t0).^2);

% noise
std=0.1;
noise=std*randn(size(t));

mu=[ 0.04  0.03  0.02  0.01  0.005  0.002  0.0015  0.001  0.0008  0.0005  0.0001  0.00005  5e-10   ];%
     Regularization parameter
nu=[1 2 3 4 5 6 10 20 30 40 50 70 100];%Truncation parameter

S=dt*toeplitz(exp(-h*(t(1:N))), zeros(1,N)); % Sensitivity matrix
T=S*q'; % Direct model (convolution)
Tr=T'+noise;

for i=1:length(mu)
%Thikonov regularization
   G=inv(S'*S+mu(i)*eye(N));
    qr=G*S'*Tr';
%TSVD regularization
   [U,W,V]=svds(S,nu(i));
   qrs=V*diag(1./diag(W))*U'*Tr';
 %norms
   nres(i)=norm(Tr-S*qr);
   nqr(i)=norm(qr);
   nress(i)=norm(Tr-S*qrs);
   nqrs(i)=norm(qrs);
end

hold off
plot(nres,nqr),xlabel('norm(T-S*q)'), ylabel('norm(q)'), hold on,plot(nress,nqrs,'o'),  figure 2
```