SOFTWARE IMPLEMENTATION OF
FINITE-DIFFERENCE METHOD FOR
PARAMETER IDENTIFICATION IN
AGROECOLOGICAL MODELLING

N. JUSHCHENKO and V. DENISOV

Klaipėda University
H. Mantų 84, Klaipėda, 5800, Lithuania
E-mail: natalya@ik.ku.lt; vitalij@ik.ku.lt

Received October 3, 2001; revised January 10, 2002

ABSTRACT

A finite difference method for numerical solution of the inverse problem of heat transfer in
a soil is described with an emphasis on algorithmic and software implementation issues. A
mathematical package Matlab v.5.3 is chosen for the method implementation as it includes
a number of matrix procedures with special measures for sparse matrices. Field experiments
data obtained from the Lithuanian Institute of Agriculture are used to provide an input
information and to calculate identification criteria at each iteration step. Typical numerical
results of inverse problem solution and investigation of algorithm convergence process are
presented and discussed in detail.

1. INTRODUCTION

Complex simulation models of agroecosystems dynamics are widely used in
agricultural applications for agronomic decision support and operational crop
management, agricultural market predictions and training of specialists [1; 6; 9]. To be applied in different environments these models should be identified, i.e. their parameters should be estimated according to the local soil, plant
and climate conditions. A method presented below is designed to this purpose
and is described on a particular example of a process of heat transfer in a soil.

In a typical simulation model of agroecosystems dynamics [1; 6] a process
of heat transfer in a soil is described by the equation [5]:

\[ c_s(w) \frac{\partial T_s(t, x)}{\partial t} = \frac{\partial}{\partial x}\left( \lambda_s(w) \frac{\partial T_s(t, x)}{\partial x} \right) \]  

(1.1)
in \( A = \{(x, t) : 0 < x < l, 0 < t \leq T\} \). Boundary conditions are defined on the edge of \( A \):

\[
T_s(t, 0) = \psi(t), \quad T_s(t, l) = \text{const.} \tag{1.2}
\]

The initial condition is given at \( t = 0 \):

\[
T_s(0, x) = \varphi(x). \tag{1.3}
\]

In (1.1) – (1.3) \( T_s(t, x) \) is the soil temperature profile, \( c_s(w) \) – volumetric soil heat capacity, \( \lambda_s(w) \) – soil heat conductivity, \( w \) – volumetric soil moisture, \( l \) – depth of root zone, \( T \) – duration of the vegetation season. For a variety of soil types parameters \( c_s(w) \) and \( \lambda_s(w) \) can be approximated by the following equations [5]:

\[
c_s(w) = c_{sw}w + c_{ss} \rho_s, \tag{1.4}
\]

\[
\lambda_s(w) = c_s \left[ \alpha_1 (w - \alpha_4)^2 + \rho_s \alpha_2 + \alpha_3 \right], \tag{1.5}
\]

where \( c_{sw} \) is the averaged soil water heat capacity, \( c_{ss} \) – averaged heat capacity of soil aggregates, \( \rho_s \) – soil bulk density.

A problem of parameter identification in this case is considered as estimation of vector of parameters \( \beta = (c_{ss}, \alpha_1, \alpha_2, \alpha_3, \alpha_4) \) by minimization of the weighted least squares function of errors \( Q(\beta) \) in the area \( A \):

\[
\min_{\beta} Q(\beta) = \left\{ \left( \mathbf{T} - \hat{\mathbf{T}}(\beta) \right)^\top \mathbf{\Gamma} \left( \mathbf{T} - \hat{\mathbf{T}}(\beta) \right) \right\}, \tag{1.6}
\]

where \( \hat{\mathbf{T}}(\beta) \) – calculated soil temperature profile, \( \mathbf{T} \) – observed soil temperature profile and \( \mathbf{\Gamma} = \text{diag}(\gamma_i) \) – weight coefficients matrix.

2. FINITE-DIFFERENCE SOLUTION OF SOIL HEAT TRANSFER IDENTIFICATION PROBLEM

Let the computational space \( A \) be covered by a grid \( \mathbf{\Omega}_T \times \mathbf{\Omega}_x \):

\[
\mathbf{\Omega}_T = \{t_j = j \tau, \quad j = 0, 1, \ldots, N, \quad t_N = T\}, \quad \mathbf{\Omega}_x = \{x_i = x_{i-1} + h_i, \quad i = 1, \ldots, M, \quad x_0 = 0, \quad x_M = l\}.
\]

We denote the discrete approximations of the function \( T_s(t, x) \) by \( T_i^j = T(t_j, x_i) \), \( \psi(t) \) by \( \psi_i^0 = \psi(t_j) \) and \( \varphi(x) \) by \( \varphi_i^0 = \varphi(x_i) \).
Initial-boundary value problem (1.1) – (1.3) is solved numerically by approximating it with the implicit finite-difference scheme

\[
c_{i}^{j+1} - \frac{T_{i}^{j+1} - T_{i}^{j}}{\tau} = \frac{1}{h_{i+\frac{1}{2}}^{2}} \left( \chi^{j-1}_{i+\frac{1}{2}} \frac{T_{i+1}^{j} - T_{i}^{j}}{h_{i+1}} - \chi^{j-1}_{i-\frac{1}{2}} \frac{T_{i}^{j} - T_{i-1}^{j}}{h_{i}} \right) \]

\[j = 0, 1, \ldots, N - 1, \quad i = 1, 2, \ldots, M - 1, \tag{2.1}\]

\[T_{0}^{j+1} = \psi_{0}^{j+1}, \quad T_{M}^{j+1} = \text{const}, \quad j = 0, 1, \ldots, N - 1, \]

\[T_{i}^{0} = \psi_{i}^{0}, \quad i = 0, 1, \ldots, M, \]

or in a matrix form:

\[
C^{j}T^{j+1} = D^{j}T^{j}, \quad j = 0, 1, \ldots, N - 1, \tag{2.2}\]

where \(T^{j}\) and \(T^{j+1}\) are respectively current and calculated vectors of state variables, which are given in the form:

\[
T^{j} = (T_{0}^{j}, T_{2}^{j}, \ldots, T_{M}^{j}), \quad D^{j} = \text{diag}(-h_{i+\frac{1}{2}} h_{i+1} c_{si}),
\]

\[
h_{i+\frac{1}{2}} = \frac{h_{i} + h_{i+1}}{2}, \quad \chi^{j}_{i+\frac{1}{2}} = \lambda \left( \frac{w^{j}_{i+\frac{1}{2}}}{h_{i} + h_{i+1}} \right), \quad \frac{w^{j}_{i+\frac{1}{2}}}{h_{i} + h_{i+1}} = \frac{w^{j}_{i+1} h_{i+1} + w^{j}_{i} h_{i}}{h_{i} + h_{i+1}}
\]

and the matrix \(C^{j}\) has a tridiagonal structure, with elements:

\[
c^{j}_{i,i-1} = \tau h_{i+\frac{1}{2}} \chi^{j}_{i-\frac{1}{2}}, \quad c^{j}_{i,i+1} = \tau h_{i} \chi^{j}_{i+\frac{1}{2}}, \tag{2.3}\]

\[c^{j}_{i,i} = -\left(c^{j}_{i,i-1} + c^{j}_{i,i+1} + h_{i+\frac{1}{2}} h_{i+1} c_{si} \right), \quad 0 < i < M,
\]

and appropriately changed for boundary equations. For the whole time interval (i.e., vegetation period of crop) system (2.2) can be written in a compact block form:

\[
GT = X \tag{2.4}
\]

or, in an explicit form:

\[
\begin{bmatrix}
C^{0} & 0 & 0 & \cdots & 0 & 0 \\
-D^{1} & C^{1} & 0 & \cdots & 0 & 0 \\
0 & \cdots & \cdots & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & \cdots & \cdots & \cdots & 0 & 0 \\
0 & \cdots & \cdots & \cdots & 0 & 0 \\
\end{bmatrix}
\times
\begin{bmatrix}
T^{1} \\
T^{2} \\
T^{j+1} \\
\vdots \\
T^{L} \\
\end{bmatrix}
= \begin{bmatrix}
D^{0}T^{0} \\
0 \\
0 \\
\vdots \\
0 \\
\end{bmatrix}.
\]
A solution of a whole direct problem therefore is given by

$$T = G^{-1}X.$$  \hspace{1cm} (2.5)

3. NUMERICAL SOLUTION OF IDENTIFICATION PROBLEM

Condition (1.6) leads to the system of normal equations:

$$\frac{\partial Q(\beta)}{\partial \beta} = 0$$  \hspace{1cm} (3.1)

or in an explicit form

$$\frac{\partial T(\beta)}{\partial \beta} T \left( T - \hat{T}(\beta) \right) = \frac{\partial \left( G^{-1}(\beta) X(\beta) \right)}{\partial \beta} T \left( T - G^{-1}(\beta) X(\beta) \right) = 0,$$

which is nonlinear with respect to $\beta$. So it is advisable to use gradient optimization methods for minimization of $Q(\beta)$. However it is difficult to use them directly due to huge dimension of matrices in (3.1). A recurrent method proposed below allow to calculate gradient and approximation of Hessian iteratively when solving direct problem and therefore to apply gradient optimization for solving the inverse problem. A modified Marquard method with regularization is chosen to solve the system (3.1), so the iteration process is [8]:

$$\beta^{i+1} = \beta^{i} + (P^T P + \mu R)^{-1} P^T T \left( T - \hat{T}(\beta^{i}) \right),$$  \hspace{1cm} (3.2)

where $i$ - iteration number, $\mu$ - regularization step parameter,

$$P = \left. \frac{\partial T(\beta)}{\partial \beta} \right|_{\beta = \beta_{i}} = \frac{\partial \left( G^{-1}(\beta_{i}) X(\beta_{i}) \right)}{\partial \beta} = \left( -G^{-1} \frac{\partial G(\beta_{i})}{\partial \beta} G^{-1} X + G^{-1} \frac{\partial X(\beta_{i})}{\partial \beta} \right),$$

$P$ is the matrix of derivatives of solution with regard to parameters, $R = \text{diag}(P^T P)$ or $R = I$, $\Gamma$ is weight matrix.

By applying Frobenius formula for inverse block matrix and a blocked analogue of an adjacent matrix method it is possible to show, that $G^{-1}$ elements can be calculated by recurrent formula [3]:

$$P^j_k = (C^j)^{-1} \left[ D^j_k P^{j-1}_k + F^j_k T^{j-1} + B^j_k T^j \right],$$  \hspace{1cm} (3.3)

where

$$V_k = \frac{\partial G(\beta)}{\partial \beta_k} T(\beta), \quad B^j_k = \frac{\partial C^j}{\partial \beta_k}, \quad F^j_k = \frac{\partial D^j}{\partial \beta_k}, \quad k = 1, \ldots, m, \quad j = 1, \ldots, N.$$
Having the formula (3.3) for recurrent calculation of matrix $P$ elements at $j$ time step of a direct problem it is easy to see that the values of elements of Hessian matrix approximation

$$
H = P^T TP = \begin{pmatrix}
    h_{11} & h_{12} & \ldots & h_{1m} \\
    \vdots & \vdots & \ddots & \vdots \\
    h_{m1} & h_{m2} & \ldots & h_{mm}
\end{pmatrix}
$$

are accumulated during the direct problem solution as well [3]:

$$
h_{rq} = \sum_{j=1}^{L} p_r^j \gamma_{r}^j p_{qj}, \quad h_{rq}^{j+1} = h_{rq}^{j} + p_r^j \gamma_{r}^j p_{qj}, \quad r, q = 1, \ldots, m. \quad (3.4)
$$

4. IDENTIFICATION ALGORITHM AND ITS SOFTWARE IMPLEMENTATION

![Diagram]

**Figure 1.** Scheme of the identification algorithm.

A method described above consists of the repetition of the following main steps (Fig. 1):
1. Solve finite-difference scheme with a parallel calculation of the elements of matrix $P$ and $H$;

2. Make an iteration of a modified Marquard optimization process to calculate new parameter values;

3. If the required accuracy of obtained parameter vector values is achieved, the iterative process ends, else the algorithm returns to the first step.

A mathematical package Matlab [7] is chosen for software implementation of the proposed method. There are several reasons for this choice. The algorithm deals with a number of large dimension sparse (diagonal) matrices during calculations. The Matlab package apart a great number of standard matrix functions provides a set of functions for dealing with sparse matrices, that saves computer memory and decreases calculation time [4]. The second reason is that it is easy to provide a user interaction with a program and to control the iterative process during calculations. The method described above was numerically investigated using the data obtained from the field experiments conducted at the Lithuanian Institute of Agriculture, the initial, boundary and measured values of soil temperatures from different fields were placed to the input files. A typical identification exercise proceeds as follows. Parameters $\gamma_i$, $\beta$, and required accuracy $\varepsilon$ were defined at the beginning of calculation. The initial parameter values $\beta_0$ were chosen depending on soil type [6], and the initial value of the regularization parameter $\gamma = 40$ was chosen empirically, from computer experiments using recommendations in [8]. Soil temperature dynamic at soil surface is approximated by function [5]

$$\psi(t) = T_{av} + T_a \cos(\omega(t - t_m)),$$

where $T_{av}$ is a daily temperature average, $T_a$ - amplitude of daily temperature,
$t_m$ – the time of daily temperature maximum, $\omega = \frac{2\pi}{24}$.

Typical soil moisture profile fluctuations are described as follows:

$$w(t, x) = w_0 + (b_1 + b_2 \sin(2\pi t))e^{-0.1x^2},$$

where $w_0$ is the soil moisture profile minimal value at $t = 0$.

In the case of heterogeneous measurements the elements of matrix $\Gamma$ can be chosen as inversely proportional to variances of measured values:

$$\Gamma = \sigma_i^{-1} I, \quad i = 1, \ldots, n.$$

When the measurements covariance matrix is known then the weights can be chosen as inversely proportional to the covariances. Finally, it is also possible to recalculate the weights values on each iteration by replacing them by their new estimates [2].

A typical set of initial parameters values for a chosen soil type is as follows:

$$\beta_0 = (0.3; -468; 11.2; 4.36; 0.02), \quad \varepsilon = 0.5, \quad \gamma = 40.$$

A particular example of the convergence of proposed algorithm as well as the final parameter values are presented in Fig. 2. Fig. 3 shows identification results in terms of calculated and measured soil profiles. To make
modelling results more visual the graph covers only the 3 days period (72h). The calculated dynamics of soil temperature is shown at three typical depths: 20cm, 50cm, 100cm. Identified model demonstrates a typical daily temperature course of soil temperatures at different depth and calculated values are in a good agreement with measurements.

REFERENCES


Baigtinių skirtumų metodo agroekologinio modelio parametrų identifikacijai programinė realizacija

N. Jushchenko, V. Denisov