Mathematical Modelling and Analysis
Volume 7 Number 1, 2002, pages 71–78

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

N. JUSHCHENKO and V. DENISOV

Klaipėda University

H. Manto 84, Klaipėda, 5800, Lithuania

E-mail: nataly@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 sparce 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 convergation 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]:

$$c_s\left(w\right)\frac{\partial T_s(t,x)}{\partial t}=\frac{\partial}{\partial x}\Big(\lambda_s\left(w\right)\frac{\partial T_s(t,x)}{\partial x}\Big) \tag{1.1}$$

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

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

The initial condition is given at $t = 0$:

$$T_s\left(0,x\right) = \varphi\left(x\right). \tag{1.3}$$

In (1.1) – (1.3) $T_s(t,x)$ is the soil temperature profile, $c_s(w)$ – volumetric soil heat capacity, $\lambda_s\left(w\right)$ – 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 $\lambda_s\left(w\right)$ can be approximated by the following equations [5]:

$$c_s(w) = c_w w + c_{ss}\rho_s, \tag{1.4}$$

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

where c_w is the averaged soil water heat capacity, c_{ss} – averaged heat capacity of soil aggregates, ρ_s – soil bulk density.

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

$$\min_{\boldsymbol{\beta}} Q(\boldsymbol{\beta}) = \left\{\left(\mathbf{T} - \widehat{\mathbf{T}}(\boldsymbol{\beta})\right)^\top \boldsymbol{\Gamma}\left(\mathbf{T} - \widehat{\mathbf{T}}(\boldsymbol{\beta})\right)\right\}, \tag{1.6}$$

where $\widehat{\mathbf{T}}(\boldsymbol{\beta})$ – calculated soil temperature profile, $\mathbf{T}$ – observed soil temperature profile and $\boldsymbol{\Gamma} = \mathrm{diag}\left(\gamma_i\right)$ – weight coefficients matrix.

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

Let the computational space A be covered by a grid $\overline{\omega}_t \times \overline{\omega}_x$:

$$\overline{\omega}_t = \{t_j = j\tau, \quad j = 0,1,\ldots,N, \ t_N = T\},$$
$$\overline{\omega}_x = \{x_i = x_{i-1} + h_i, \quad i = 1,\ldots,M, \ x_0 = 0, \ x_M = l\}.$$

We denote the discrete approximations of the function $T_s(t,x)$ by $T_i^j = T(t_j, x_i)$, $\psi(t)$ by $\psi_0^j = \psi(t_j)$ and $\varphi\left(x\right)$ by $\varphi_i^0 = \varphi\left(x_i\right)$.

Initial-boundary value problem (1.1) – (1.3) is solved numerically by approximating it with the implicit finite-difference scheme

$$
\begin{aligned}
& c_{si}\frac{T_i^{j+1}-T_i^j}{\tau}=\frac{1}{h_{i+\frac{1}{2}}}\left(\overline{\lambda}^{j-1}_{i+\frac{1}{2}}\Big(\frac{T^j_{i+1}-T^j_i}{h_{i+1}}\Big)-\overline{\lambda}^{j-1}_{i-\frac{1}{2}}\Big(\frac{T^j_i-T^j_{i-1}}{h_i}\Big)\right),\\
& j=0,1,\dots,N-1,\ i=1,2,\dots,M-1, \qquad (2.1)\\
& T_0^{j+1}=\psi_0^{j+1},\quad T_M^{j+1}=const,\ j=0,1,\dots,N-1,\\
& T_i^0=\varphi_i^0,\quad i=0,1,\dots,M,
\end{aligned}
$$

or in a matrix form:

$$\mathbf{C}^j\mathbf{T}^{j+1}=\mathbf{D}^j\mathbf{T}^j,\quad j=0,1,\dots,N-1, \qquad (2.2)$$

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

$$\mathbf{T}^j=(T_0^j,T_2^j,\dots,T_M^j),\quad \mathbf{D}^j=\mathrm{diag}(-h_{i+\frac{1}{2}}h_{i+1}h_ic_{si}),$$

$$h_{i+\frac{1}{2}}=\frac{h_i+h_{i+1}}{2},\quad \overline{\lambda}^j_{i+\frac{1}{2}}=\lambda\left(\overline{w}^j_{i+\frac{1}{2}}\right),\quad \overline{w}^j_{i+\frac{1}{2}}=\frac{w^j_{i+1}h_{i+1}+w^j_ih_i}{h_i+h_{i+1}}$$

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

$$
\begin{aligned}
& c^j_{i,i-1}=\tau h_{i+1}\overline{\lambda}^j_{i-\frac{1}{2}},\quad c^j_{i,i+1}=\tau h_i\overline{\lambda}^j_{i+\frac{1}{2}}, \qquad (2.3)\\
& c^j_{i,i}=-(c^j_{i,i-1}+c^j_{i,i+1}+h_{i+\frac{1}{2}}h_{i+1}h_ic_{si}),\quad 0<i<M,
\end{aligned}
$$

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:

$$\mathbf{GT}=\mathbf{X} \qquad (2.4)$$

or, in an explicit form:

$$
\begin{pmatrix}
\mathbf{C}^0 & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \mathbf{0}\\
-\mathbf{D}^1 & \mathbf{C}^1 & \mathbf{0} & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \mathbf{0}\\
\dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots\\
\mathbf{0} & \dots & \dots & \dots & \mathbf{0} & -\mathbf{D}^j & \mathbf{C}^j & \mathbf{0} & \dots & \dots & \mathbf{0}\\
\dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots\\
\mathbf{0} & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \mathbf{0} & -\mathbf{D}^{L-1} & \mathbf{C}^{L-1}
\end{pmatrix}
$$

$$
\times\begin{pmatrix}\mathbf{T}^1\\ \mathbf{T}^2\\ \dots\\ \mathbf{T}^{j+1}\\ \dots\\ \mathbf{T}^L\end{pmatrix}=\begin{pmatrix}\mathbf{D}^0\mathbf{T}^0\\ \mathbf{0}\\ \dots\\ \mathbf{0}\\ \dots\\ \mathbf{0}\end{pmatrix}.
$$

A solution of a whole direct problem therefore is given by

$$\mathbf{T} = \mathbf{G}^{-1}\mathbf{X}. \tag{2.5}$$

3. NUMERICAL SOLUTION OF IDENTIFICATION PROBLEM

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

$$\frac{\partial Q\left(\boldsymbol{\beta}\right)}{\partial \boldsymbol{\beta}} = 0 \tag{3.1}$$

or in an explicit form

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

which is nonlinear with respect to $\boldsymbol{\beta}$. So it is advisable to use gradient optimization methods for minimization of $Q\left(\boldsymbol{\beta}\right)$. 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]:

$$\boldsymbol{\beta}^{l+1} = \boldsymbol{\beta}^{l} + \left(\mathbf{P}^{\top}\boldsymbol{\Gamma}\mathbf{P} + \mu\mathbf{R}\right)^{-1}\mathbf{P}^{\top}\boldsymbol{\Gamma}\left(\mathbf{T} - \widehat{\mathbf{T}}(\boldsymbol{\beta}^{l})\right), \tag{3.2}$$

where l – iteration number, μ – regularization step parameter,

$$\begin{aligned}\mathbf{P} &= \frac{\partial \widehat{\mathbf{T}}\left(\boldsymbol{\beta}\right)}{\partial \boldsymbol{\beta}}\Big|_{\boldsymbol{\beta}=\boldsymbol{\beta}^{l}} = \frac{\partial\left(\mathbf{G}^{-1}\left(\boldsymbol{\beta}_{l}\right)\mathbf{X}\left(\boldsymbol{\beta}_{l}\right)\right)}{\partial \boldsymbol{\beta}} \\ &= \left(-\mathbf{G}^{-1}\frac{\partial \mathbf{G}\left(\boldsymbol{\beta}_{l}\right)}{\partial \boldsymbol{\beta}}\mathbf{G}^{-1}\mathbf{X} + \mathbf{G}^{-1}\frac{\partial \mathbf{X}(\boldsymbol{\beta}_{l})}{\partial \boldsymbol{\beta}}\right),\end{aligned}$$

$\mathbf{P}$ is the matrix of derivatives of solution with regard to parameters, $\mathbf{R} = \mathrm{diag}(\mathbf{P}^{\top}\boldsymbol{\Gamma}\mathbf{P})$ or $\mathbf{R} = \mathbf{I}$, $\boldsymbol{\Gamma}$ 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]:

$$\mathbf{P}_{k}^{j} = \left(\mathbf{C}^{j}\right)^{-1}\left[\mathbf{D}^{j}\mathbf{P}_{k}^{j-1} + \mathbf{F}^{j}\mathbf{T}^{j-1} + \mathbf{B}^{j}\mathbf{T}^{j}\right], \tag{3.3}$$

where

$$\mathbf{V}_{k} = \frac{\partial \mathbf{G}\left(\boldsymbol{\beta}\right)}{\partial \boldsymbol{\beta}_{k}}\widehat{\mathbf{T}}\left(\boldsymbol{\beta}\right), \ \mathbf{B}^{j} = \frac{\partial \mathbf{C}^{j}}{\partial \boldsymbol{\beta}_{k}}, \ \mathbf{F}^{j} = \frac{\partial \mathbf{D}^{j}}{\partial \boldsymbol{\beta}_{k}}, \ k = 1, ..., m, \ j = 1, ..., N.$$

Having the formula (3.3) for recurrent calculation of matrix $\mathbf{P}$ elements at j time step of a direct problem it is easy to see that the values of elements of Hessian matrix approximation

$$\mathbf{H} = \mathbf{P}^{\top}\mathbf{\Gamma}\mathbf{P} = \begin{pmatrix} h_{11} & h_{12} & \dots & h_{1m} \\ \dots & \dots & \dots & \dots \\ h_{m1} & h_{m2} & \dots & h_{mm} \end{pmatrix}$$

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

$$h_{rq} = \sum_{j=1}^{L} p_{rj}\gamma_r p_{jq}, \quad h_{rq}^{j+1} = h_{rq}^{j} + p_r^j \gamma_r^j p_q^j, \quad r, q = 1, ..., m. \tag{3.4}$$

4. IDENTIFICATION ALGORITHM AND ITS SOFTWARE IMPLEMENTATION

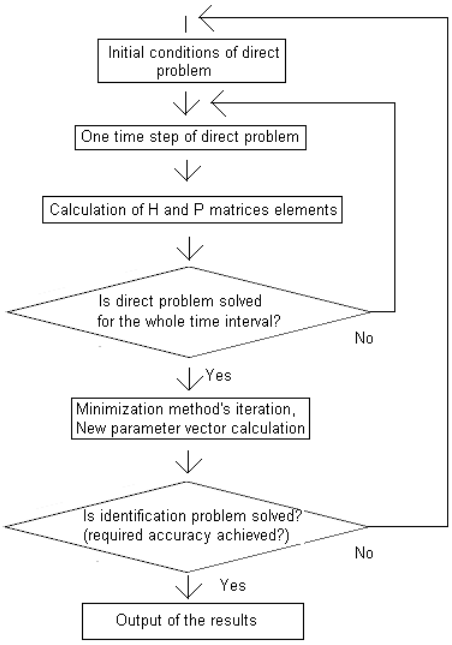
Initial conditions of direct problem
One time step of direct problem
Calculation of H and P matrices elements
Is direct problem solved for the whole time interval?
No
Yes
Minimization method's iteration, New parameter vector calculation
Is identification problem solved? (required accuracy achieved?)
No
Yes
Output of the results

Figure 1. Scheme of the identification algorithm.

A method described above consists of the repetition of the following main steps (Fig. 1):

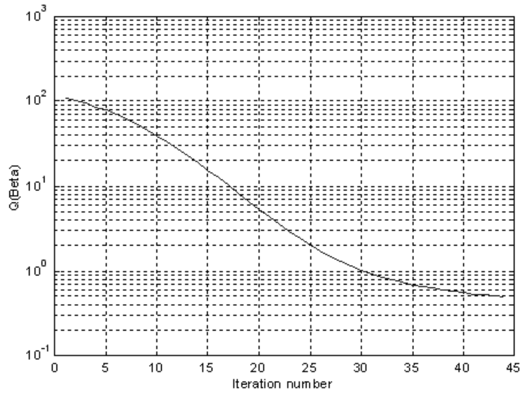

Figure 2. Example of algorithm convergation process. $\boldsymbol{\beta}_{44} = [0.04; -255.6; 11.9; 5.2; 0.05]$. $Q_0 = 113$, $Q_{44} = 0.49$.

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 sparce (diagonal) matrices during calculations. The Matlab package apart a great number of standard matrix functions provides a set of functions for dealing with sparce 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 γ_i, $\boldsymbol{\beta}$, and required accuracy ε were defined at the beginning of calculation. The initial parameter values $\boldsymbol{\beta}_0$ 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,

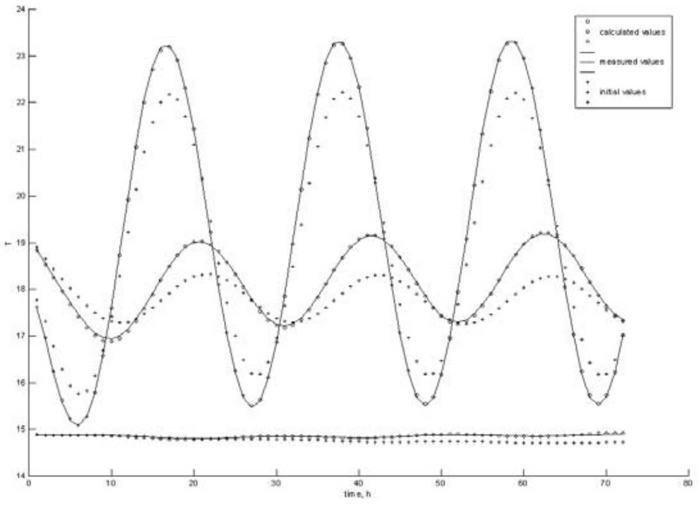

Figure 3. Calculated and measured soil temperature profiles at three soil depths: 20cm, 50cm, 100cm.

t_m – the time of daily temperature maximum, $\omega = \dfrac{2\pi}{24}$.

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 $\mathbf{\Gamma}$ can be chosen as inversely proportional to variances of measured values:

$$\mathbf{\Gamma} = \sigma_{\widehat{\mathbf{T}}_i^j} \mathbf{I}, \quad i = 1, ..., n.$$

When the measurements covariance matrix is known then the weights can be chosen as inversely proportional to the covariations. 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:

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

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

[1] V. Denisov, A. Adomkus, D. Lukianienė and I. Lamsodienė. Simulation system of crop growth and development. *Biologija*, **3**, 52 – 57, 1998.

[2] N. Draper and H. Smith. *Applied regression analysis. Second edition.* New York, John Wiley & Sons, 1981.

[3] N. Jushchenko and V. Denisov. Finite-difference methods for solving inverse problems in agroecological modeling using field experiments data. *Finite difference schemes: theory and applications. Proceedings of the Conference FDS2000*, **6**(1), 117 – 123, 2000.

[4] G.R. Lindfield and J.E. Penny. Using Matlab for sparce matrices. *Int. J. Math. Educ. Sci. Technol.*, **28**(3), 427 – 436, 1997.

[5] S. Nerpin and A. Chudnovsky. *Energy and mass exchange in a soil-plant-atmosphere system.* Hidrometeoizdat, Leningrad, 1975. (in Russian)

[6] R.A. Poluektov. *The dynamic models of agroecological systems.* Hidrometeoizdat, Leningrad, 1991. (in Russian)

[7] V.G. Potemkin. *The MatLab. User's manual.* Moscow, DIALOG-MIFI, 1997. (in Russian)

[8] W. Press, S. Teukolsky, W. Vettering and B. Flannery. *Numerical recepies in FORTRAN. Second edition.* Cambridge University Press, 1992.

[9] I. Supit, A.A. Hooijer and van Diepen C. A. (eds). *EUR 15959 – System description of the WOFOST 6.0 crop simulation model implemented in CGMS.* Luxembourg, EC, 1994.

Baigtinių skirtumų metodo agroekologinio modelio parametrų identifikacijai programinė realizacija

N. Juščenko, V. Denisov

Straipsnyje nagrinėjami autorių pasiūlytos baigtinių skirtumų schemos, skirtos agroekologinių modelių parametrų identifikacijai, algoritminės ir programinės realizacijos ypatumai. Metodo taikymas iliustruojamas sprendžiant šilumos pernešimo dirvožemyje parametrinės identifikacijos uždotį. Pateikta algoritmo schema ir pagrįstas programinių priemonių pasirinkimas algoritmo realizacijai. Uždavinio pradinės ir kraštinės sąlygos bei identifikacijos kriterijus formuluojami naudojant duomenų bazėje sukauptus Lietuvos žemdirbystės instituto lauko bandymų duomenis. Pateikti atvirkštinio ir tiesioginio uždavinių sprendimo rezultatai.