ELSEVIER

Advances in Engineering Software 31 (2000) 275–285

ADVANCES IN ENGINEERING SOFTWARE

www.elsevier.com/locate/advengsoft

New algorithm for extreme temperature measurements

N. Damean*

University of Twente, EL/MI, P.O. Box 217, 7500 AE, Enschede, The Netherlands

Received 5 September 1995; received in revised form 3 October 1996; accepted 26 May 1999

Abstract

A new algorithm for measurement of extreme temperature is presented. This algorithm reduces the measurement of the unknown temperature to the solving of an optimal control problem, using a numerical computer. Based on this method, a new device for extreme temperature measurements is projected. It consists of a hardware part that includes some standard temperature sensors and it also has a software section.

The principal component of the device is a rod. The variation in the temperature, which is produced near one end of the rod, is determined using some temperature measurements at the other end of the rod and the new algorithm described here.

The mathematical model of the device and the algorithm are explained in detail. At the same time, some possible practical implementations and a collection of simulations are presented.

Keywords: Gradient algorithm; Local variation method; Numerical method; Heat equation; Temperature measurement

Nomenclature

C_1	thermal exchange coefficient between rod and measurement medium [m^{-1}]
d	diameter of the rod [mm]
h	convection heat transfer coefficient [$W\,m^{-2}\,K^{-1}$]
k_r	thermal conductivity of the rod [$W\,m^{-1}\,K^{-1}$]
L	length of the rod [mm]
L_1	length of the part of the rod on which the temperature sensors are not placed [mm]
m	number of the division parts of T
N	number of the temperature sensors
n	number of the division parts of L
T	duration of contact between the hot end and the point where the temperature is measured [s]
t	time [s]
U	unknown temperature to be measured [K]
$[M_1,M_2]$	temperature range including the unknown temperature U [K]
x	distance along the rod relating to the hot end [mm]
$Y(t,x)$	temperature of the rod at the moment t in the point x [K]
$Y^m(t,x)$	temperature at the moment t of a point x at the cold end of the rod which is measured by the temperature sensors [K]
α	thermal diffusivity [$m^2\,s^{-1}$]
β	volumetric thermal expansion coefficient of the measurement medium [K^{-1}]
ν	kinematic viscosity of the measurement medium [$m^2\,s^{-1}$]
Δt	time step [s]
Δx	space step [m]
subscripts	
i	associated with a space position
j	associated with a time position
superscripts	
k	iteration number of the numerical algorithm

1. Introduction

It is known that for temperature measurement some kinds of transducers are used. They are divided in two main classes: modulating transducers and self-generating transducers. Thermistors, resistance-temperature detectors, semiconductor temperature transducers are in the first class. Thermocouples are important elements in the second class [1,2].

All these usual transducers have in common that they can only measure usual temperatures, in the range 200–2000 K with good accuracy. Pyrometric methods are used for measurement of high temperature out of this range. Unfortunately, some integration effects appear using these

* Tel.: + 31-53-489-2778; fax: + 31-53-489-1067.
E-mail address: n.damean@el.utwente.nl (N. Damean).

0965-9978/00/$ - see front matter
PII: S0965-9978(99)00029-0

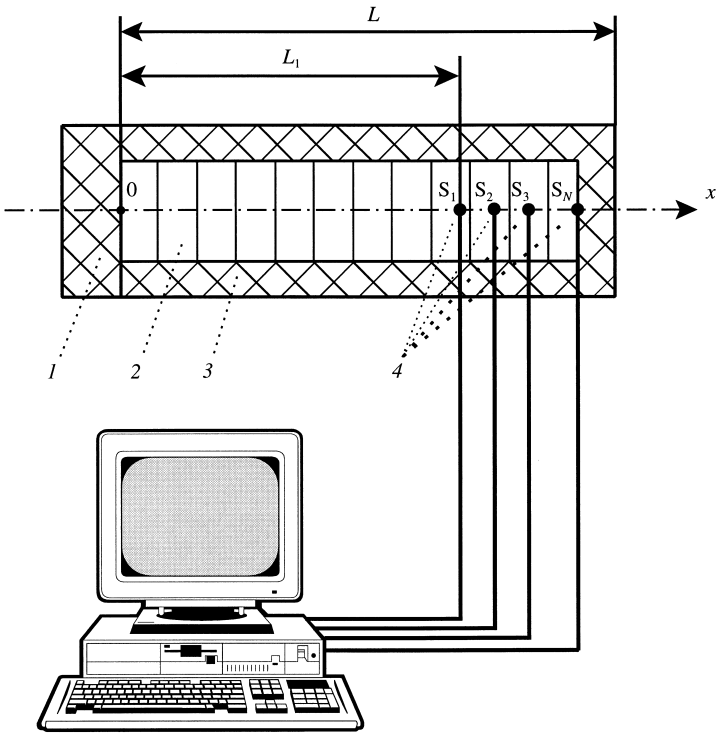

Fig. 1. The scheme of the device: (*1*) detachable lid; (*2*) rod; (*3*) thermic insulating layer; (*4*) common temperature sensors.

methods, so that practical results are disturbed by a lot of errors. Some cryogenic temperature sensors are developed in the last decade. However, the technical feasibility of these sensors and the achievement of a good metrological performances may be adversely affected by a broad working range [3,4]. Thus it is necessary to develop new devices and methods for extreme, i.e. very high and very low, temperature measurements.

In this work, the author proposes a new method to measure extreme temperatures using standard sensors in the usual range, where their behaviour is known with great accuracy. One possible way to obtain this extension of range is to use an adequate software tool.

Therefore, the author consider the problem in which the unknown temperature is determined through the formulation of the measurement problem as an optimal control one. The last problem is solved using a numerical algorithm.

The principal component of the device for which a new numerical algorithm is developed is a long cylindrical rod with a diameter as small as possible. It is covered by a thermic, insulating layer.

One end of the rod, called the “hot end”, is in contact with the point in which the temperature is measured. This end is covered with a detachable lid made of a thermic insulating material.

The other end of the rod is termed the “cold end”. In its vicinity, on a small distance along a generating line of the rod, a number N of standard temperature sensors, e.g. thermocouples, are placed equidistantly. They are stuck directly on the rod, under the thermic insulating layer. Temperature sensors are coupled with a computer.

The scheme of this device is shown in Fig. 1.

Its operation is explained in the assumption that the unknown temperature U is higher than the temperature of the environment, and is constant during the time of measurement T.

The device is introduced into the medium of measurement so its hot end is in the measurement point and the lid is removed for a short time, T. During this period, temperature versus time variations are measured with the N sensors near the cold end. In fact, the hot end is excited by a step function having value U as amplitude, and its influence over the vicinity of the cold end is observed. The unknown temperature U is computed from the temperatures measured by the N sensors.

From the above description it is noticed that the temperature of the cold end is much less than the measured temperature U. Therefore, it can be considered that this device performs the function of a temperature attenuator. Obviously, the device works in a similar manner when the temperature to be measured is lower than the temperature of the environment. In this case it can be considered that it performs the function of a temperature amplifier.

2. Theoretical approach

The phenomenon described above is modeled by a partial differential equation of parabolic type, named the heat

equation [5]

$$Y_t(t,x) - \alpha Y_{xx}(t,x) = 0, \qquad (t,x) \in (0,T) \times (0,L) \tag{1}$$

with the boundary conditions of Newton type

$$Y_x(t,0) = C_1(Y(t,0) - U), \qquad t \in [0,T], \tag{2}$$

$$Y_x(t,L) = 0, \qquad t \in [0,T] \tag{3}$$

and the initial condition

$$Y(0,x) = Y_0(x), \qquad x \in [0,L]. \tag{4}$$

The existence and regularity for the problem (1)–(4) are well known [6].

As is noticed, unknown temperature U appears in condition (2) in a boundary condition of Newton type.

The problem is to compute U, using the mathematical model and the set of measured temperatures in the vicinity of the cold end of the rod, during time of measurement T.

This problem is considered in terms of an optimal control one, denoted by (P), with the following typical elements [7]:

(i) equations of state are given by (1)–(4);
(ii) the class of admissible controls is $[M_1, M_2]$, where M_1, M_2 are chosen from the beginning by the physical limitations on the problem, as being the extreme values where the unknown temperature U can be situated;
(iii) the cost function to be minimized is

$$\Phi(u) = \int_0^T \int_{L_1}^L [Y(t,x) - Y^{\mathrm{m}}(t,x)]^2 \, \mathrm{d}x \, \mathrm{d}t \tag{5}$$

for $u \in [M_1, M_2]$. Quantity $Y(t,x)$ is the solution of the above problem (1)–(4), which corresponds to u; $Y^{\mathrm{m}}(t,x)$ is temperature of a point near the cold end that is measured by the common temperature sensors.

The optimal control problem (P) is now formulated as (P).

Find an admissible control $U^* \in [M_1, M_2]$ that minimizes the cost function Φ.

Concerning the enunciated optimal problem (P), it is proved that it admits only one solution [8]. Thus, there exists a pair $\{U^*, Y^*(t,x)\}$, such that $U^* \in [M_1, M_2]$ is the optimal control of problem (P) and $Y^*(t,x)$ is the optimal state that corresponds to U^*. This is equivalent to:

$$\Phi(U^*) = \int_0^T \int_{L_1}^L [Y^*(t,x) - Y^{\mathrm{m}}(t,x)]^2 \, \mathrm{d}x \, \mathrm{d}t = \min_{u \in [M_1, M_2]} \Phi(u).$$

The development of an descent algorithm [9] is detailed in the following. For that, the necessary conditions for optimalization have to be established. The following theorem is already proved [10].

Theorem. *If the pair* $\{U^*, Y^*(t,x)\}$ *is an optimal one for the problem* (1)–(5)*, then a size* $A(t,x)$ *is a solution of the problem:*

$$A_t(t,x) + \alpha A_{xx}(t,x) = Y^*(t,x) - Y^{\mathrm{m}}(t,x), \quad (t,x) \in (0,T) \times [L_1, L), \tag{6}$$

$$A_t(t,x) + \alpha A_{xx}(t,x) = 0, \qquad (t,x) \in (0,T) \times (0,L_1),$$

$$A_x(t,0) = C_1 A(t,0), \qquad t \in [0,T], \tag{7}$$

$$A_x(t,L) = 0, \qquad t \in [0,T], \tag{8}$$

$$A(T,x) = 0, \qquad x \in [0,L] \tag{9}$$

and it happens that

$$U^* = \begin{cases} M_2 & \text{if } \int_0^T A(t,0) \, \mathrm{d}t > 0 \\ M_1 & \text{if } \int_0^T A(t,0) \, \mathrm{d}t \le 0 \end{cases}. \tag{10}$$

Relations (6)–(10) make the necessary conditions for optimalization.

The results presented in this section show that entire problem has only one solution in accordance with physical reality. At the same time, they suggest the possibility for numerical implementation that is developed in the following section.

3. Numerical approach

For the numerical calculation of the temperature U, which is an optimal control problem, as seen before, one type of gradient algorithm is used [9,11].

Using formula (10) as the main one, the following iterative algorithm is developed (k being iteration number). In view of relations (1)–(10) one considers the following iterative algorithm:

Step 0: Choose $u^{(0)} \in [M_1, M_2]$; set $k := 0$.

The algorithm is initialized with an admissible control and a counter of the iterative algorithm sets to zero.

Step 1: Compute $Y^{(k)}(t,x)$, that is, the state system described by Eqs. (1)–(4)

$$Y_t^{(k)}(t,x) - \alpha Y_{xx}^{(k)}(t,x) = 0, \qquad (t,x) \in (0,T) \times (0,L), \tag{11}$$

$$Y_x^{(k)}(t,0) = C_1[Y^{(k)}(t,0) - u^{(k)}], \qquad t \in [0,T], \tag{12}$$

$$Y_x^{(k)}(t,L) = 0, \qquad t \in [0,T], \tag{13}$$

$$Y^{(k)}(0,x) = Y_0(x), \qquad x \in [0,L]. \tag{14}$$

Step 2: Compute $A^{(k)}(t,x)$, that is, the adjoint state system

described by Eqs. (6)–(9)

$$A_t^{(k)}(t,x) + \alpha A_{xx}^{(k)}(t,x) = Y^{(k)}(t,x) - Y^{\mathrm{m}}(t,x),$$

$$(t,x) \in (0,T) \times [L_1, L),$$

$$A_t^{(k)}(t,x) + \alpha A_{xx}^{(k)}(t,x) = 0, \qquad (t,x) \in (0,T) \times (0,L_1),$$

$$A_x^{(k)}(t,L) = 0, \qquad t \in [0,T],$$

$$A^{(k)}(T,x) = 0, \qquad x \in [0,L].$$

Step 3: Use relation (10) to compute the optimal control at iteration k, denoted as $u^{*(k)}$, from

$$u^{*(k)} = \begin{cases} M_2 & \text{if } \int_0^T A^{(k)}(t,0)\,\mathrm{d}t > 0 \\ M_1 & \text{if } \int_0^T A^{(k)}(t,0)\,\mathrm{d}t \le 0 \end{cases}.$$

An approximation of the optimal control problem (P) is derived.

Step 4: Compute $\delta^{(k)} \in [0,1]$ as a solution to the minimization process

$$\min\{\Phi(\delta u^{(k)} + (1-\delta)u^{*(k)});\ \delta \in [0,1]\}$$

where Φ is the cost function of problem (1)–(4) given by Eq. (5). It is verified if the optimal control derived in Step 3 minimizes the cost function.

Step 5: Set

$$u^{(k+1)} := \delta^{(k)} u^{(k)} + (1-\delta^{(k)})u^{*(k)}.$$

A new admissible control $u^{(k+1)}$ placed between $u^{(k)}$ and $u^{*(k)}$ is computed.

Step 6: (the "stopping criterion")

IF $|u^{(k+1)} - u^{(k)}| \le \varepsilon_{\mathrm{s}}$

THEN $U^* = u^{(k+1)}$

STOP

ELSE $k := k + 1$

GO TO Step 1

The "stopping criterion" in Step 6 can also be as follows.

Step 6′:

IF $|\Phi(u^{(k+1)}) - \Phi(u^{(k)})| \le \varepsilon_{\mathrm{s}}'$

THEN $U^* = u^{(k+1)}$

STOP

ELSE $k := k + 1$

GO TO Step 1

The symbols ε_{s} and $\varepsilon_{\mathrm{s}}'$ in Step 6 and Step 6′ denote two small, positive real values chosen in accordance with practical requirements.

As is observed from the practical implementation of this algorithm, there are difficulties in trying to find a good "guess" of $u^{(0)}$ in Step 0. For avoiding it, in Step 0 a local variation procedure is used, namely, the "azimuth mark" method (AMM) [12].Two "stopping criterions" that could be used for this procedure are:

IF $|r(i+1) - r(i)| \le \varepsilon_0$

THEN STOP AMM

ELSE $i := i + 1$

CONTINUE AMM

and

IF $i = i_0$

THEN STOP AMM

ELSE $i := i + 1$

CONTINUE AMM

where $r(i)$ is the radius of the azimuth mark (AM) at the i-th iteration of this procedure, and ε_0 and i_0 are natural and real numbers, respectively. They are specified in the initialization stage of the algorithm.

Steps 1 and 2 are solved using some implicit finite-difference approximations, namely, the O'Brian et al. formula and the Crank–Nicolson formula [13]. Boundary conditions impose the need to use a "false" boundary for practical implementation of these formulas.

Step 3 does not cause any serious problems from an applicative point of view.

Concerning Step 4, at the beginning it used a carefully supervised loop with the form (nn $\times\ \delta \in \mathbb{N}^*$)

$\Delta\delta := 1/\mathrm{nn}$

DO FOR $l = 0, 1, \ldots \mathrm{nn}$

$\delta := l\Delta\delta$

Compute $\Phi[\delta u^{(k)} + (1-\delta)u^{*(k)}]$ using Eq. (5)

END DO

This procedure, with constant-length steps to solve a minimization problem, could be improved (from a computational time point of view) using variable-length steps. Afterwards, it has been used a carefully supervised loop of the form (mm $\in \mathbb{N}^*$)

DO FOR $l = 0, 1, \ldots \mathrm{mm} \times \delta$

$$\delta := \sum_{kk=0}^{l} (1/2)^{kk} - 1$$

Compute $\Phi[\delta u^{(k)} + (1-\delta)u^{*(k)}]$ using Eq. (5)

END DO

Between $nn \times \delta$ and $mm \times \delta$ there is the following relation:

$$2^{mm} = nn \tag{15}$$

obtained from the condition to have the same smallest length step in the two loops. It means that the second loop has a smaller number of steps than the first. Generally, the second procedure is faster, but the accuracy is lower than the first.

The double integrals from Step 4, as well as Step 5 and Step 6 (Step 6′), do not lead to potential problems from a numerical point of view.

4. Implementation of the algorithm

In the following, it is referred only to Step 1 and Step 2, because the others are implemented without difficulties, as is seen in the previous paragraph.

Steps 1 and 2 are solved using some implicit finite difference approximations; the grids in effect with equidistant knots are:

$$0 = t_1 < t_2 < \ldots < t_{m+1} = T$$

$$x_0 < 0 = x_1 < x_2 < \ldots < x_{n+1} = L$$

The existence of a "false" boundary, denoted by x_0, is noticed in the previous relation.

In the following, it is presented a detailed algorithm using the O'Brian et al.'s formula.

Implementation of the algorithm and results when the Crank–Nicolson formula is used are similar [13].

Denote

$$y(j,i) = Y^{(k)}(t_j, x_i), \qquad j = 1,2,\ldots,m+1,\ i = 0,1,\ldots n+1, \tag{16}$$

$$u = u^{(k)}.$$

The following usual discretizations for derivatives are performed:

$$Y_t^{(k)}(t,x)\big|_{j,i} \approx \frac{y(j,i) - y(j-1,i)}{\Delta t}, \tag{17}$$

$$j = 1,2,\ldots m+1,\ i = 0,1,\ldots n+1,$$

$$Y_{xx}^{(k)}(t,x)\big|_{j,i} \approx \frac{y(j,i-1) - 2y(j,i) + y(j,i+1)}{\Delta x^2}, \tag{18}$$

$$j = 1,2,\ldots m+1,\ i = 1,2,\ldots n,$$

$$Y_x^{(k)}(t,0)\big|_{j,1} \approx \frac{y(j,i) - y(j,0)}{\Delta x}, \qquad j = 1,2,\ldots m+1, \tag{19}$$

$$Y_x^{(k)}(t,L)\big|_{j,n+1} \approx \frac{y(j,n+i) - y(j,n)}{\Delta x}, \qquad j = 1,2,\ldots m+1, \tag{20}$$

where

$$\Delta t = \frac{T}{m},\ \Delta x = \frac{L}{n},\ C = \frac{\Delta t}{\Delta x^2}. \tag{21}$$

Using the discretizations (16)–(21), the value of $Y^{(k)}(t,x)$ is obtained in Step 1 from Eqs. (11)–(14) as follows.

$$\begin{bmatrix} \alpha C C_1 \Delta x + \alpha C + 1 & -\alpha C & 0 & 0 & \ldots & 0 & 0 & 0 \\ -\alpha C & 1+2\alpha C & -\alpha C & 0 & \ldots & 0 & 0 & 0 \\ 0 & -\alpha C & 1+2\alpha C & -\alpha C & \ldots & 0 & 0 & 0 \\ & & & & \ldots & & & \\ & & & & \ldots & & & \\ & & & & \ldots & & & \\ 0 & 0 & 0 & 0 & \ldots & -\alpha C & 1+2\alpha C & -\alpha C \\ 0 & 0 & 0 & 0 & \ldots & 0 & -\alpha C & 1+2\alpha C \end{bmatrix} \begin{bmatrix} y(j,1) \\ y(j,2) \\ y(j,3) \\ \\ \\ \\ y(j,n-1) \\ y(j,n) \end{bmatrix} = \begin{bmatrix} y(j-1,1) + \alpha C C_1 \Delta x u \\ y(j-1,2) \\ y(j-1,3) \\ \\ \\ \\ y(j-1,n-1) \\ y(j-1,n) \end{bmatrix}$$

- The quantities $y(j,i)$, $j = 2,3,\ldots,m+1,\ i = 1,2,\ldots,n$, are computed by solving the following system of linear equations:
- The quantities $y(j,n+1)$ are computed like this:

  $$y(j,n+1) = y(j,n), \qquad j = 2,3,\ldots m+1.$$

- The quantities $y(1,i)$ are computed like this:

  $$y(1,i) = Y_0(x_i), \qquad i = 1,2,\ldots n+1.$$

  In a similar manner as Step 1, Step 2 is developed. Now denote

  $$a(j,i) = A^{(k)}(t_j, x_i),$$

  $$j = 1,2,\ldots m+1,\ i = 0,1,\ldots n+1.$$

  The value of $A^{(k)}(t,x)$ is obtained as follows.
- The quantities $a(j,i)$, $j = 1,2,\ldots,m,\ i = 2,3,\ldots,n+1$,

are computed solving the following system of linear equations:

$$\begin{bmatrix} -\alpha C/1 + C_1\Delta x + 1 + 2\alpha C & -\alpha C & 0 & 0 & \dots & 0 & 0 & 0 \\ -\alpha C & 1 + 2\alpha C & -\alpha C & 0 & \dots & 0 & 0 & 0 \\ 0 & -\alpha C & 1 + 2\alpha C & -\alpha C & \dots & 0 & 0 & 0 \\ & & & & \dots & & & \\ & & & & \dots & & & \\ & & & & \dots & & & \\ 0 & 0 & 0 & 0 & \dots & -\alpha C & 1 + 2\alpha C & -\alpha C \\ 0 & 0 & 0 & 0 & \dots & 0 & -\alpha C & 1 + 2\alpha C \end{bmatrix} \begin{bmatrix} a(j,2) \\ a(j,3) \\ a(j,4) \\ \\ \\ \\ a(j,n)_z \\ a(j,n+1) \end{bmatrix} = \begin{bmatrix} a(j+1,2) - \Delta t\, w(j,2) \\ a(j+1,3) - \Delta t\, w(j,3) \\ a(j+1,4) - \Delta t\, w(j,4) \\ \\ \\ \\ a(j+1,n) - \Delta t\, w(j,n) \\ a(j+1,n+1) - \Delta t\, w(j,n+1) \end{bmatrix},$$

where

$$w(j,i) = \begin{cases} Y^{(k)}(t_j,x_i) - Y^{\mathrm{m}}(t_j,x_i) & (t_j,x_i) \in (0,T)\times[L_1,L) \\ 0 & (t_j,x_i) \in (0,T)\times(0,L_1) \end{cases}.$$

- The quantities $a(j,1)$ are computed like this:

  $a(j,1) = a(j,2)/(1 + C_1\Delta x), \qquad j = 1,2,\dots m+1.$

- The quantities $a(m+1,i)$ are computed like this:

  $a(m+1,i) = 0, \qquad i = 2,3,\dots n+1.$

  The matrix of the both systems is a band one (the bandwidth is 3), therefore the both systems are solved by a special Gauss routine.

5. Numerical results

For numerical tests, it is considered that the rod is made from tungsten, and the measurement medium is melting AISI 304 steel. The thermophysical properties of these materials used in the numerical simulations are those corresponding to a temperature of 1800 K, namely

- for tungsten (the rod): $k_{\mathrm{r}} = 174\ \mathrm{W\,m^{-1}\,K^{-1}}$, $\alpha = 63.3\times10^{-6}\ \mathrm{m^2\,s^{-1}}$;
- for melting AISI 304 steel (the measurement medium): $\alpha = 3.95\times10^{-6}\ \mathrm{m^2\,s^{-1}}$, $\beta = 5.85\times10^{-5}\ \mathrm{K^{-1}}$, $\nu = 1\times10^{-7}\ \mathrm{m^2\,s^{-1}}$.

The geometrical characteristics of the rod are: $d = 3$, $L = 30$ and $L_1 = 20$ mm. The thermal exchanged coefficient is defined as usual:

$$C_1 = \frac{h}{k_{\mathbf{r}}}$$

Using the Morgan's formula for a long horizontal cylinder in the case of the external free convection flows [5,14], the average Nusselt number is $\mathrm{Nu} = 6.75$. Finally, the thermal exchanged coefficient is gotten as $C_1 = 2.25\times10^{3}\ \mathrm{m^{-1}}$.

At the same time for the numerical discretizations, it is considered that $n = m = 144$.

The initial distribution of the temperature in the rod is

$Y_0(x) = 300$ K, $\qquad x \in [0,L]$.

The following simulations are achieved.

5.1. Computation of temperature variation of rod versus time

The first two steps of the above algorithm, with some minor modifications are used. Results are displayed in Fig. 2 when the temperature of hot end is $U = 3650$ K.

Curve $c1$ corresponds to the hot end, curve $c4$ corresponds to the cold end and curves $c2$ and $c3$ correspond to points placed equidistantly on the rod, between ends.

It is observed that the temperature in the vicinity of the cold end is smaller than the temperature close to the hot end, as is expected from the physical point of view. At the same

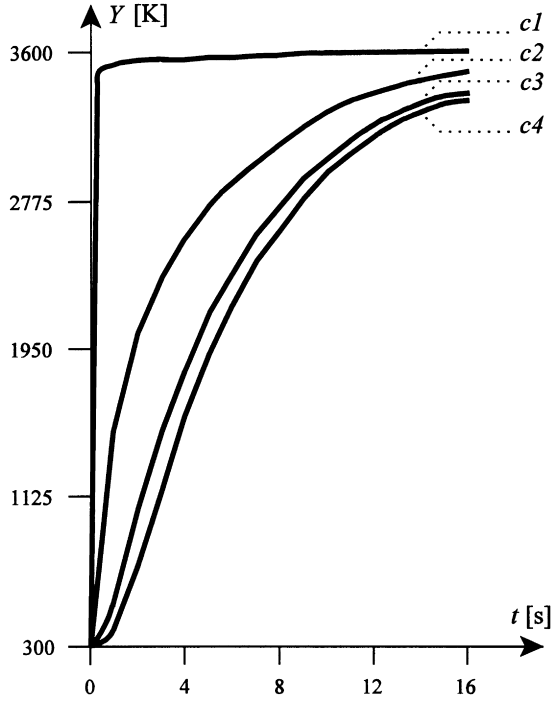

Fig. 2. Temperature variations in the rod.

Table 1
Computational results for theoretical running

U [K]	No. of iterations of AM	Φ_0	e_0 (%)	Type step	No. of iterations of principal algorithm	Φ_s	e_s (%)
50	–	–	–	Constant	2	0	0.92×10^{-4}
				Variable	5	0.92×10^{-8}	0.11
	8	0.28×10^{-4}	6.25	Constant	2	0.14×10^{-8}	0.42×10^{-1}
				Variable	2	0.14×10^{-8}	0.42×10^{-1}
350	–	–	–	Constant	2	0.37×10^{-9}	0.26×10^{-2}
				Variable	5	0.81×10^{-9}	0.46×10^{-2}
	7	0.11×10^{-3}	1.79	Constant	2	0.63×10^{-7}	0.42×10^{-1}
				Variable	3	0.10×10^{-6}	0.53×10^{-1}
650	–	–	–	Constant	2	0.28×10^{-7}	0.15×10^{-1}
				Variable	6	0.52×10^{-7}	0.21×10^{-1}
	7	0.11×10^{-3}	0.96	Constant	2	0.59×10^{-7}	0.23×10^{-1}
				Variable	3	0.97×10^{-7}	0.28×10^{-1}
950	–	–	–	Constant	2	0.67×10^{-8}	0.51×10^{-2}
				Variable	4	0.77×10^{-7}	0.17×10^{-1}
	8	0.28×10^{-4}	0.33	Constant	2	0.10×10^{-8}	0.22×10^{-2}
				Variable	2	0.10×10^{-8}	0.22×10^{-2}
1250	–	–	–	Constant	2	0	0
				Variable	7	0.65×10^{-9}	0.13×10^{-2}
	7	0	0	Constant	0	0	0
				Variable	0	0	0
1550	–	–	–	Constant	2	0.64×10^{-8}	0.30×10^{-2}
				Variable	5	0.14×10^{-8}	0.14×10^{-2}
	8	0.28×10^{-4}	0.20	Constant	2	0.14×10^{-8}	0.14×10^{-2}
				Variable	2	0.14×10^{-8}	0.14×10^{-2}
1850	–	–	–	Constant	2	0.16×10^{-8}	0.13×10^{-2}
				Variable	4	0.49×10^{-8}	0.23×10^{-2}
	7	0.11×10^{-3}	0.34	Constant	2	0.61×10^{-7}	0.79×10^{-2}
				Variable	3	0.96×10^{-7}	0.10×10^{-1}
2150	–	–	–	Constant	2	0.14×10^{-8}	0.11×10^{-2}
				Variable	4	0.48×10^{-8}	0.20×10^{-2}
	8	0.11×10^{-3}	0.29	Constant	2	0.61×10^{-7}	0.68×10^{-2}
				Variable	3	0.96×10^{-7}	0.86×10^{-2}
2450	–	–	–	Constant	2	0.62×10^{-8}	0.19×10^{-2}
				Variable	5	0.15×10^{-8}	0.91×10^{-3}
	8	0.28×10^{-4}	0.13	Constant	2	0.12×10^{-8}	0.87×10^{-3}
				Variable	2	0.12×10^{-8}	0.87×10^{-3}
2750	–	–	–	Constant	2	0	0
				Variable	6	0.22×10^{-10}	0.15×10^{-3}
	7	0	0	Constant	0	0	0
				Variable	0	0	0
3050	–	–	–	Constant	2	0.67×10^{-8}	0.16×10^{-2}
				Variable	4	0.78×10^{-7}	0.54×10^{-2}
	8	0.28×10^{-4}	0.10	Constant	2	0.12×10^{-8}	0.70×10^{-3}
				Variable	2	0.12×10^{-8}	0.70×10^{-3}
3350	–	–	–	Constant	2	0.27×10^{-7}	0.29×10^{-2}
				Variable	6	0.50×10^{-7}	0.40×10^{-2}
	7	0.11×10^{-3}	0.19	Constant	2	0.63×10^{-7}	0.44×10^{-2}
				Variable	3	0.97×10^{-7}	0.55×10^{-2}
3650	–	–	–	Constant	2	0.20×10^{-9}	0.25×10^{-3}
				Variable	5	0.83×10^{-9}	0.44×10^{-3}
	7	0.11×10^{-3}	0.17	Constant	2	0.61×10^{-7}	0.40×10^{-2}
				Variable	3	0.97×10^{-7}	0.51×10^{-2}
3950	–	–	–	Constant	2	0	0
				Variable	5	0.11×10^{-7}	0.15×10^{-2}
	8	0.28×10^{-4}	0.079	Constant	2	0.90×10^{-9}	0.54×10^{-3}
				Variable	2	0.90×10^{-9}	0.54×10^{-3}

time, the temperature of the hot end can be considered in a stationary state after 8 s, but the other temperatures are in transitory state after a long period of time.

This simulation can be used for choosing a value of the time T. This value must be selected not too big because the temperature U must be constant during the measurement process and also because the temperature of the cold end must be not too big. At the same time, T must be not too small, because it is preferable that the temperature sensors to measure significant temperature variations of the cold end.

5.2. Numerical simulations of algorithm

Some numerical results are analysed in the following, concerning the entire algorithm presented above. For that purpose, it is necessary to specify a value for a number of sensors denoted by N.

At the beginning, this algorithm is tested using a number of sensors corresponding to some working conditions that are nearly ideal. Therefore, the case when the temperature is known in every discretization point near the cold end is analysed. This case is named "theoretical running". After that, in "real running" case, consideration is given to a realistic number of temperature sensors, and numerical results are compared with the previous results.

The values of other variables used in the algorithm are $M_1 = 0$ K, $M_2 = 4000$ K, $T = 1$ s.

Note. Therefore the class of admissible controls is $[0, 4000]$ K. This covers all industrial applications. The values of M_1 and M_2 could be adjusted for a given application according to real working conditions.

A stopping criterion as shown in Step 6 is used.

5.2.1. Theoretical running

In fact, the target of this simulation is to choose the best variant from those presented before. The problem is to decide about using constant- or variable-length steps in Step 4 and about using AMM or not for initialization. Thus, it is necessary to choose the best from four variants of this algorithm.

To compare these simulations from some points of view, a number of steps must be used in Step 4. The smallest length step of them is equal in all variants, and this value is less than a half from constant ε_s in the stopping criterion defined by Step 6.

In these conditions, it is expected that simulation error, defined as:

$$e_s(\%) = \frac{|U - U^*|}{U} \times 100$$

is uniform in all four variants.

In the previous relation, U^* means the final value of U computed with all the algorithm Steps 0–6. Using Eq. (15), this condition is displayed as

$$\frac{M_2 - M_1}{\mathrm{NC}} = \frac{M_2 - M_1}{2^{\mathrm{NV}}} = \frac{2\varepsilon_0}{\mathrm{NCM}} = \frac{2\varepsilon_0}{2^{\mathrm{NVM}}} < \frac{\varepsilon_s}{2}, \qquad (22)$$

where NC is the number of constant-length steps when AMM is not used; NV the number of variable-length steps when AMM is not used; NCM the number of constant-length steps when AMM is used; NVM the number of variable-length steps when AMM is used.

To compare these simulations, it is necessary to use an ideal number of temperature sensors, as is mentioned above. This number is $N = 49$ in this present simulation.

A weak value for ε_s in Step 6 is used at this stage because a long calculus time is expected for some of these variants. Thus, $\varepsilon_s = 0.5$ is utilized.

According to Eq. (22), the other values are: NC = 16 384; NV = 14; NCM = 128; NVM = 7; $\varepsilon_0 = 15.625$.

These simulations are carried out for 14 values of unknown temperature U.

By analogy with simulation error, initial simulation error (when AMM is used) is defined as

$$e_0(\%) = \frac{|U - U_0|}{U} 100$$

In this relation, U_0 means the value of $u^{(0)}$ computed in Step 0, using AMM.

The corresponding values of cost function Φ relating to U_0 and U^* are denoted with Φ_0 and Φ_s, respectively.

Results are presented in Table 1.

Following Table 1, it is observed that Φ_s and e_s are nearly similar in all variants for a certain temperature.

At the same time, the number of iterations of the principal algorithm can be greater when variable-length steps are used because the accuracy is lower in this case. This phenomenon appears especially when AMM is not used.

Sometimes, some null values for Φ_s and e_s are obtained. These can be explained through using of a "favourable"-length step.

These behaviours are forecasted from the beginning. According to expectation, the shortest computation time is obtained when AMM with variable-length steps is used. In the same situation, the number of steps used in Step 4 is minimum and the number of iterations of the principal algorithm is nearly minimum.

Thus, only this variant of the algorithm, which is fast convergent and can work in real time, is used in the following.

5.2.2. Real running

A realistic number of sensors, namely $N = 3$, equidistantly placed in the vicinity of cold end, is considered in the following.

Author suggests two variants for real running of the algorithm:

- only data obtained from temperature sensors are used;

Table 2
Computational results for real running

U [K]	Φ_s			e_s (%)		
	Theoretical running	Real running		Theoretical running	Real running	
		Data from sensors	Data from sensors and interpolation		Data from sensors	Data from sensors and interpolation
50	0	0	0.178×10^{-3}	0.404×10^{-3}	0.404×10^{-3}	17.230
350	0.138×10^{-11}	0.264×10^{-11}	0.715×10^{-5}	0.270×10^{-3}	0.270×10^{-3}	0.483
650	0.294×10^{-11}	0.446×10^{-11}	0.350×10^{-3}	0.150×10^{-3}	0.150×10^{-3}	1.878
950	0.150×10^{-12}	0.485×10^{-12}	0.121×10^{-2}	0.642×10^{-4}	0.642×10^{-4}	2.370
1250	0	0	0.258×10^{-2}	0	0	2.634
1550	0.148×10^{-11}	0.199×10^{-11}	0.446×10^{-2}	0.158×10^{-4}	0.158×10^{-4}	2.796
1850	0.325×10^{-11}	0.364×10^{-11}	0.686×10^{-2}	0.726×10^{-4}	0.726×10^{-4}	2.868
2150	0.113×10^{-11}	0.139×10^{-11}	0.977×10^{-2}	0.568×10^{-4}	0.568×10^{-4}	2.980
2450	0.230×10^{-10}	0.278×10^{-10}	0.132×10^{-1}	0.130×10^{-3}	0.130×10^{-03}	2.997
2750	0	0	0.171×10^{-1}	0	0	3.036
3050	0.170×10^{-10}	0.203×10^{-10}	0.216×10^{-1}	0.104×10^{-3}	0.104×10^{-3}	3.125
3350	0.976×10^{-12}	0.128×10^{-11}	0.266×10^{-1}	0.292×10^{-4}	0.292×10^{-4}	3.078
3650	0.709×10^{-12}	0.808×10^{-12}	0.320×10^{-1}	0.268×10^{-4}	0.268×10^{-4}	3.168
3950	$0.880.10^{-11}$	0.120×10^{-10}	0.380×10^{-1}	0.804×10^{-4}	0.804×10^{-4}	3.201

- data obtained from temperature sensors and interpolation data for points between sensors are used; interpolation with spline functions is used.

A variant of the algorithm with AMM and variable-length steps is considered. In this case, a strong value $\varepsilon_s = 0.001$ is used for the stopping criterion. The other values of constants are: NV = 16 and $\varepsilon_0 = 15.625$, according to relation (22).

These simulations are carried out for 14 values of unknown temperature U. Results are displayed in Table 2.

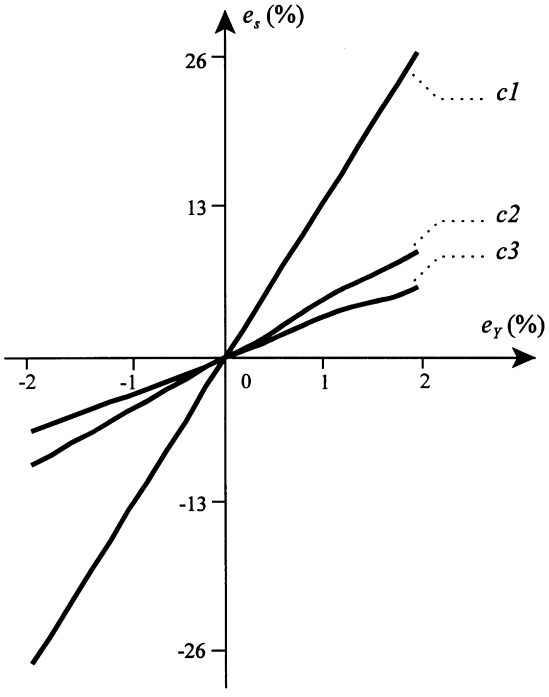

Fig. 3. Plot of e_s (%) versus e_Y (%).

Examining Table 2, it is seen that the best way is to use only the data obtained from temperature sensors, without interpolation.

For this variant, simulation error e_s is the same as in theoretical running. Values of Φ_s are a little greater for the first variant of real running in comparison with the theoretical running, as it is expected. Values of Φ_s and e_s are very large when interpolation is used, in comparison with the other situations.

Therefore, in the future, only the first variant of real running is used.

5.3. Numerical simulations of algorithm when disturbed data are used

From a practical point of view, it is possible that some errors appear that are caused by technical realization of the rod, inexact measuring of time, and so on.

Using above algorithm, in real running, the simulation error is computed versus perturbation error, for three temperatures $U = 350$, 2150 and 3650 K. These three curves are denoted by $c1$, $c2$ and $c3$, respectively.

The following perturbations are carried out.

- Perturbation of temperature measured by sensors in the vicinity of the cold end.

Consideration is given to a relative error e_Y of temperature measurements made by all sensors, in the range 0–2%.

The dependencies of simulation error versus relative error of temperature measurement are displayed in Fig. 3.

- Inexact placements of the sensors on the rod.

Consideration is given to a deviation Δl of all sensors in the range of 0–1.25 mm, toward the hot end.

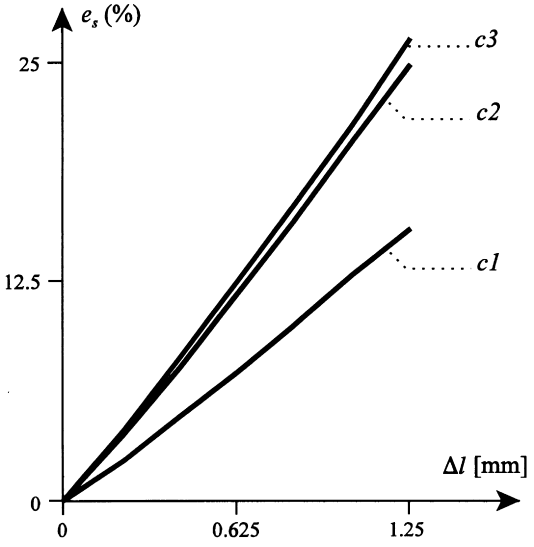

Fig. 4. Plot of e_s (%) versus Δl (mm).

The dependencies of simulation error versus deviation of sensors are displayed in Fig. 4.

- Inexact measurement of time.

Consideration is given to an error of time measurement Δt in the range of 0–0.1 s (delay).

The dependencies of simulation error versus time delay are displayed in Fig. 5.

Examining Figs. 3–5, note the nearly linear dependencies between perturbation factors and simulation error in all cases. These errors are large at the end of the perturbation range.

According with these simulations, it is necessary to construct the measuring system carefully, so that these errors are as small as possible.

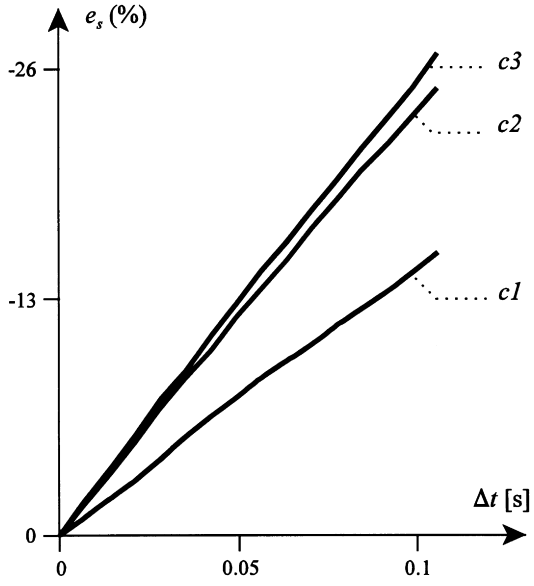

Fig. 5. Plot of e_s (%) versus Δt (s).

6. Conclusions

A new numerical algorithm and a device based on it for extreme temperature measurements are presented.

The mathematical model of the device and new numerical algorithm are exposed in detail. At the same time, some possibilities of practical implementation are analysed and the best variant from them is chosen. Behaviour of algorithm in real conditions using disturbed data is shown in the end.

Generally, the temperature sensors work in the stationary state. The system described can be considered as a temperature sensor operating in the transient state. It uses a small beginning part of the transient state of the heat transfer in the rod.

The proposed device can measure any values of temperature in difficult places. In fact, it works as an attenuator of high temperatures and as an amplifier of low temperatures.

Some observations can be noted.

- It is not necessary that temperature sensors be placed equidistantly. This type of placement is chosen to make the numerical implementation easier. The distances between sensors could be arbitrary, but knowing them exactly is necessary.
- It is not necessary that temperature sensors be placed in the vicinity of the cold end. This type of placement is chosen to make the explanation of the theoretical part easier. Depending on the real working conditions, they could be placed on another part of the rod (e.g., in the middle of it). However, knowing these placements precisely is necessary, otherwise an important error could appear, as shown in Fig. 4.
- It is not necessary that time be considered from 0 to T in the cost function (5). From a practical point of view, the measuring of very small variations of temperature must be avoided, because important errors can occur. Therefore it is possible to use a subrange $[T_1,T] \subset [0,T]$ in the cost function without any other modifications of the algorithm.
- It is considered that the thermophysical properties of the materials implied in the device structure do not depend on temperature, to make the numerical implementation easier. This hypothesis is not true in the real conditions, because there are some slow dependencies between the thermophysical properties and temperature. In the general case, an appropriate form of the nonlinear heat equation might be considered and the entire problem must be reformulated accordingly. Because of the very short measurement time, there are expected relatively small temperature variations in the rod, so that the use of the nonlinear heat equation can be avoided. In this case, only some minor modifications in the algorithm presented herein are necessary to take these dependencies into account, considering different values for these properties in the different temperature ranges when the

device works in certain conditions. Therefore, accordingly to these temperature dependencies, the values of the thermophysical properties have to be adjusted in each iteration of the algorithm presented herein.

References

[1] Helfrick AD, Cooper WD. Modern electronic instrumentation and measurement techniques, New York: Prentice Hall, 1991.

[2] Jones LD, Chin AF. Electronic instruments and measurements, New York: Prentice Hall, 1991.

[3] Pavese F. Cryogenics. In: Scholz J, Ricolfi T, editors. Sensors, 4. Weinheim: VCH, 1990.

[4] OMEGA. The temperature handbook, 1993.

[5] Incropera FP, DeWitt DP. Fundamentals of heat and mass transfer, New York: Wiley, 1996.

[6] Barbu V. Optimal control of variational inequalities, London: Pitman, 1984.

[7] Knowles G. An introduction to applied optimal control, New York: Academic Press, 1981.

[8] Damean N. A new method and a new device for temperature measurement using one-dimensional heat equation. In: Kosuke I, editor. Proceedings of 14th Annual International Computers in Engineering Conference, Sept. 11–14, Minneapolis, 1994, p. 763–6.

[9] Kirsch A. Some remarks concerning the computation of bang-bang controls. In: Hoffmann KH, Krabs W, editors. Optimal control of partial differential equations, Basel: Birkhäuser, 1984. p. 144–56.

[10] Damean N, Anita S. A problem of measurement of temperature solving using the one-dimensional heat equation. International Journal of Heat and Mass Transfer 1993;36:1043–8.

[11] Arnáutu V. On approximation of the inverse one-phase Stefan problem. International Series of Numerical Mathematics 1991;99:69–81.

[12] Sibony M. Sur l'approximation d'équations et inéquations aux dérivées partielles non linéaires de type monotone. Journal of Mathematical Analysis and Applications 1971;34:502–64.

[13] Ames WD. Numerical methods for partial differential equations, New York: Academic Press, 1992.

[14] Bejan A. Heat transfer, New York: Wiley, 1996.