Application of evolutionary algorithms in identification of solidification parameters

E. Majchrzak a,b,*, J. Mendakiewicz a, M. Paruch a

a Department for Strength of Materials and Computational Mechanics, Silesian University of Technology, ul. Konarskiego 18a, 44-100 Gliwice, Poland
b Institute of Mathematics and Computer Science, Częstochowa University of Technology, ul. Dąbrowskiego 73, 42-200 Częstochowa, Poland

Corresponding author: E-mail address: ewa.majchrzak@polsl.pl

Received 10.04.2007; published in revised form 01.08.2007

1. Introduction

The thermal processes proceeding in the system casting-mould-environment are described by the system of partial differential equations supplemented by the adequate boundary, initial, physical and geometrical conditions. The solidification process in domain of casting is described using one domain approach [1, 2, 3, 4, 5]. If the parameters appearing in governing equations are known then the direct problem is considered, while if part of them is unknown then
the inverse problem should be formulated [6, 7, 8, 9, 10]. In order to solve the inverse problem the additional information concerning the time-dependent temperature field at the set of points from the domain considered must be used.

In the paper the possibilities of evolutionary algorithms [11, 12, 13, 14, 15] application in identification of solidification parameters are discussed. In the final part of the paper the results of computations are shown.

2. Formulation of the problem

A casting-mould-environment system is considered. A transient temperature field in casting sub-domain determines the following energy equation

\[ x \in \Omega: \quad C(T) \frac{\partial T(x, t)}{\partial t} = \nabla \left[ \lambda(T) \nabla T(x, t) \right] \]  

(1)

where \( C(T) \) is the substitute thermal capacity [1, 4, 9], \( \lambda(T) \) is the thermal conductivity, \( T \) is the temperature, \( x \) are the spatial co-ordinates and \( t \) is the time.

The substitute thermal capacity for metallic alloy is defined as follows [1]

\[ C(T) = \begin{cases} c_L, & T > T_L \\ \frac{c_L + c_S}{2} - Q \frac{dS(T)}{dT}, & T \leq T_L \\ c_S, & T < T_S \end{cases} \]  

(2)

where the temperatures \( T_L, T_S \) correspond to the beginning and the end of the solidification process, respectively, \( c_L, c_S \) are the constant volumetric specific heats of liquid and solid state, \( Q \) is the volumetric latent heat, \( S(T) \) is the solid state fraction at the point considered.

In the case of cast iron solidification the following approximation of substitute thermal capacity can be taken into account (Figure 1) [1, 2].

![Fig. 1. Substitute thermal capacity of cast iron](image)

\[ p_1 = c_L, \quad T > T_L \]
\[ p_2 = \frac{c_L + c_S}{2} + Q_{\text{aus}} \frac{T_L + T_E}{T_E - T_L}, \quad T_L < T \leq T_L \]
\[ C(T) = \begin{cases} p_1 = c_L, & T > T_L \\ \frac{c_L + c_S}{2} + Q_{\text{aus}} \frac{T_L + T_E}{T_E - T_L}, & T_L < T \leq T_L \\ \frac{c_L + c_S}{2} + Q_{\text{euc}} \frac{T_S - T_L}{T_E - T_L}, & T_L < T \leq T_E \\ p_3 = c_S, & T < T_S \end{cases} \]  

(3)

where \( T_C, T_S \) correspond to the border temperatures, \( Q_{\text{aus}} = Q_{\text{aus1}} + Q_{\text{aus2}}, Q_{\text{euc}} \) are the latent heats connected with the austenite and eutectic phases evolution, at the same time \( Q = Q_{\text{aus}} + Q_{\text{euc}} \).

The thermal conductivity of cast iron can be assumed as follows

\[ \lambda(T) = \begin{cases} \lambda_L, & T > T_L \\ \frac{\lambda_L}{2} + \lambda_S, & T_L < T \leq T_L \\ \lambda_S, & T < T_L \end{cases} \]  

(4)

where \( \lambda_L, \lambda_S \) are the constant thermal conductivities of liquid and solid state, respectively.

A temperature field in mould sub-domain describes the equation of the form

\[ x \in \Omega_m: \quad c_v \frac{\partial T_m(x, t)}{\partial t} = \lambda_m \nabla^2 T_m(x, t) \]  

(5)

where \( p_h = \lambda_m \) is the thermal conductivity, \( p_v = c_v \) is the volumetric specific heat of mould.

On the contact surface between casting and mould the continuity condition

\[ x \in \Gamma_c: \quad \begin{cases} -\lambda_m \nabla T(x, t) = -\lambda_m n \cdot \nabla T_m(x, t) \\ T(x, t) = T_m(x, t) \end{cases} \]  

(6)

is assumed.

On the outer surface of the system the no-flux condition can be accepted, namely

\[ x \in \Gamma_o: \quad q_n(x, t) = -\lambda_m n \cdot \nabla T_m(x, t) \]  

(7)

For the moment \( t = 0 \) the initial temperature distribution is given

\[ T(x, 0) = T_h(x), \quad T_m(x, 0) = T_m(0) \]  

(8)

3. Evolutionary algorithm

As was mentioned before, if the parameters appearing in governing equations are known then the direct problem is considered, while if part of them is unknown then the inverse problem should be formulated. In order to identify the unknown solidification parameters, e.g. \( p_h, p_v \) the additional information connected with the course of the process analyzed is necessary.

We assume that the values \( T_h(x) \) at the selected set of points \( x \) from the domain considered for times \( t' \) are known, namely.
\[ T_{i,f}' = T_i(x_i,t') \text{, } i = 1,2,\ldots,M, \text{ } f = 1,2,\ldots,F \]  

In order to solve the inverse problem, the least squares criterion is applied [5, 6, 9]

\[ S = \frac{1}{M F} \sum_{i=1}^{M} \sum_{j=1}^{F} (T_{i,j}' - T_{i,j})^2 \]  

where \( T_{i,j}' = T_i(x_i,t') \) is the calculated temperature at the point \( x_i \) for time \( t' \) for arbitrary assumed values of unknown parameters.

Evolutionary algorithm can be very useful in solving of such inverse problems. The algorithm minimizes the fitness function (functional (10)) with respect to parameters \( p_k \) [11, 12, 15]. A chromosome (vector) characterizes the solution

\[ p = [p_1, p_2, \ldots, p_i, \ldots, p_N] \]  

where \( p_k \) are the genes containing information about the solidification parameters.

The genes are the real numbers on which constrains are imposed in the form

\[ p_i^k \leq p_i \leq p_i^k, \text{ } k = 1, 2, \ldots, K \]  

The evolutionary algorithm starts with an initial population. This population consists of \( N \) chromosomes \( p_n^N \), \( n = 1, 2, \ldots, N \), generated in random way. Every gene is taken from the feasible domain. For the assumed values of \( p_n^N \), \( n = 1, 2, \ldots, N \), the direct problems described by equations (1), (5), (6), (7), (8) are solved. The next stage is an evaluation of the fitness function (10) for every chromosome \( p_n^N \) and the selection is employed. The selection is performed in the form of ranking selection or the tournament selection [11, 12, 13] and the evolutionary operators: mutation and crossover are applied. In this way the next population is created. The process is repeated until the chromosome, for which the value of the fitness function is zero, has been found or after the achieving the assumed number of populations.

In evolutionary computations the following genetic operators are applied [11, 12, 15]:
- uniform mutation operator which changes the genes values in chromosome by choosing the new ones in random way,
- nonuniform mutation operator which changes the genes values in chromosome using the Gauss distribution,
- arithmetic crossover operator which creates new chromosome with genes which are the linear combination of two randomly chosen chromosomes.

**4. Results of computations**

The casting-mould system shown in Figure 2 has been considered. At first, the direct problem has been solved. The following input data have been introduced: \( \lambda_c = 20 \text{ [W/(mK)]} \), \( \lambda_s = 40 \text{ [W/(mK)]} \), \( \lambda_m = 1 \text{ [W/(mK)]} \), \( c_c = 5.88 \text{ [MJ/(m^3K)]} \), \( c_s = 5.4 \text{ [MJ/(m^3K)]} \), \( p_2 = 24.384 \text{ [MJ/(m^3K)]} \), \( p_3 = 11.32 \text{ [MJ/(m^3K)]} \), \( p_s = 34.75 \text{ [MJ/(m^3K)]} \) (c.f. equation (3)) \( c_m = 1.75 \text{ [MJ/(m^3K)]} \), pouring temperature \( T_0 = 1300^\circ\text{C} \), liquidus temperature \( T_f = 1250^\circ\text{C} \), eutectic phase evolution, at the same time \( Q_{eu} = 1250^\circ\text{C} \), austenite temperature \( T_d = 1200^\circ\text{C} \), final part of the paper the results are collected. The part of the results obtained is presented in Tables 2 and 3. The first problem concerns the identification of substitute thermal capacity coefficients (c.f. equation (3)) using the heating curves from casting domain (Figure 3). In the second example the
mould parameters \( p_8 = \lambda_{se} \), \( p_9 = c_{in} \) (c.f. equation (5)) have been identified using the cooling curves from mould domain (Figure 4).

Fig. 4. Cooling curves at the points 4, 5, 6, 7, 8

Summing up, the evolutionary algorithm is a good identification tool and always the solution of inverse problem is obtained.

Table 1. Evolutionary algorithm parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of generations</td>
<td>200</td>
</tr>
<tr>
<td>Number of chromosomes in each population</td>
<td>50</td>
</tr>
<tr>
<td>Probability of uniform mutation</td>
<td>30%</td>
</tr>
<tr>
<td>Probability of nonuniform mutation</td>
<td>40%</td>
</tr>
<tr>
<td>Probability of arithmetic crossover</td>
<td>30%</td>
</tr>
<tr>
<td>Probability of cloning</td>
<td>10%</td>
</tr>
</tbody>
</table>

Table 2. Result of computations using the EA – example 1

<table>
<thead>
<tr>
<th>Design variable</th>
<th>Exact value [MJ/(m²K)]</th>
<th>Found value</th>
<th>Error %</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p_1 )</td>
<td>5.88</td>
<td>5.7509</td>
<td>2.19</td>
</tr>
<tr>
<td>( p_2 )</td>
<td>24.384</td>
<td>24.7149</td>
<td>1.36</td>
</tr>
<tr>
<td>( p_3 )</td>
<td>11.32</td>
<td>11.1012</td>
<td>1.93</td>
</tr>
<tr>
<td>( p_4 )</td>
<td>34.75</td>
<td>34.7077</td>
<td>0.12</td>
</tr>
<tr>
<td>( p_5 )</td>
<td>5.4</td>
<td>5.8073</td>
<td>7.54</td>
</tr>
</tbody>
</table>

Table 3. Result of computations using the EA – example 2

<table>
<thead>
<tr>
<th>Design variable</th>
<th>Exact value [W/(mK)]</th>
<th>Found value</th>
<th>Error %</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p_8 )</td>
<td>1</td>
<td>0.9996</td>
<td>0.04</td>
</tr>
<tr>
<td>( p_9 )</td>
<td>1.75</td>
<td>1.7497</td>
<td>0.02</td>
</tr>
</tbody>
</table>

References