Optimal boundary control of a linear parabolic evolution system

Y. H. Wu M. Chuedoung P. F. Siew*

(Received 16 July 2001; revised 26 September 2002)

Abstract

We consider the optimal boundary control of a linear parabolic boundary value problem. Firstly, the problem is formulated as an optimization problem with the system state governed by a parabolic partial differential equation. Based on the formulation for the variation of the cost functional, a gradient-type optimization technique utilizing the finite element method is then developed to solve the constrained optimization problem. Finally, a numerical example is given and the results show that the method of solution is robust.

Contents

*Department of Mathematics and Statistics, Curtin University of Technology, GPO Box U1987, W.A. 6845, Australia.
mailto:yhwu@maths.curtin.edu.au

[0]See http://anziamj.austms.org.au/V44/CTAC2001/Wu00 for this article,
 Published 1 April 2003. ISSN 1446-8735

1 Introduction

Many natural and industrial process involve diffusion. Typical examples are the transient transfer of heat and the diffusion of chemicals. A diffusion process is governed by a parabolic type partial differential equation subject to certain initial and boundary conditions [3, 4], and the behavior of the process can be controlled by the condition imposed on the boundary. In this paper, we are concerned with the boundary control of a diffusion process governed by a linear parabolic partial differential equation.

Let T be the system state, t be time, $\boldsymbol{x} = (x_1, x_2)$ be the position vector, Ω be the region under consideration, Γ be the boundary of Ω, $\Sigma = \Gamma \times (0, t_A]$, and $Q = \Omega \times (0, t_A]$. Then the boundary value problem governing the state of a typical linear parabolic evolution system is the BVP:

$$\begin{aligned}
&\frac{\partial T(\boldsymbol{x}, t; u)}{\partial t} - \nabla^2 T(\boldsymbol{x}, t; u) = f(\boldsymbol{x}, t)\,, \quad (\boldsymbol{x}, t) \in Q\,,\\
&T(\boldsymbol{x}, 0) = T^0(\boldsymbol{x})\,, \quad \boldsymbol{x} \in \Omega\,,\\
&\frac{\partial T(\boldsymbol{x}, t)}{\partial n} = u(\boldsymbol{x}, t)\,, \quad (\boldsymbol{x}, t) \in \Sigma\,,
\end{aligned} \tag{1}$$

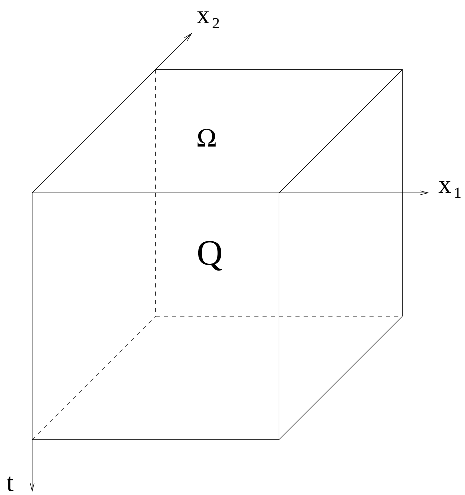

Figure 1: Diagram showing the region Ω and the domain Q .

where $T = T(\boldsymbol{x}, t; u)$ and the inclusion of u is to indicate that the state $T(\boldsymbol{x}, t)$ depends on the boundary control u.

Let $z(\boldsymbol{x}, t)$ be the desired target state of T and U be the admissible space for u. Then the problem of finding the boundary control u to achieve the desired target state is cast in the least square sense by the following constrained optimization problem (COP) [1, 6, 7]:

$$\min_{v \in U} J(v) = \int_{Q} \left[T(\boldsymbol{x}, t; v) - \boldsymbol{z}(\boldsymbol{x}, t)\right]^2 \, d\boldsymbol{x} \, dt \,, \tag{2}$$

subject to T being the solution of the BVP (1).

Theorem 1 *The cost functional $J(v)$ in (2) can be expressed as*

$$J(v) = a(v, v) - 2I(v) + k \,, \tag{3}$$

where $a(v,v)$ and $I(v)$ are respectively the continuous bilinear and linear functionals defined by

$$\begin{aligned}
a(u,v) &= \int_Q [T(\boldsymbol{x},t;u) - T(\boldsymbol{x},t;0)]\,[T(\boldsymbol{x},t;v) - T(\boldsymbol{x},t;0)]\; d\boldsymbol{x}\,dt\,,\\
I(v) &= -\int_Q [T(\boldsymbol{x},t;v) - T(\boldsymbol{x},t;0)]\,[T(\boldsymbol{x},t;0) - z(\boldsymbol{x},t)]\; d\boldsymbol{x}\,dt\,,\\
k &= \int_Q [T(\boldsymbol{x},t;0) - z(\boldsymbol{x},t)]^2\; d\boldsymbol{x}\,dt\,. \qquad (4)
\end{aligned}$$

Proof: by substituting (4) into (3). ♠

Various attempts have been made to solve this type of constrained optimization problem [8, 12, 5, 11]. However, there does not appear to be any efficient numerical algorithm available. In this paper, we present and test an efficient gradient type numerical technique for the solution of the problem based on previous work in the field [7, 8, 12, 5, 11] and utilizing the finite element method for the associated direct boundary value problems.

2 Variation of the cost functional

Consider the cost functional $J(u)$ as defined in (3). Let h be the variation of u, then the corresponding increment of the functional $J(u)$ is

$$\Delta J(u) = J(u+h) - J(u)\,.$$

From Theorem 1, we have

$$\begin{aligned}
\Delta J(u) &= a(u+h,u+h) - 2I(u+h) - a(u,u) + 2I(u)\\
&= 2a(u,h) - 2I(h) + a(h,h)\,. \qquad (5)
\end{aligned}$$

Neglecting the higher order term of h, the variation of the functional $J(v)$, which is the principal linear part of the functional increment, is

$$\delta J(u) = 2a(u,h) - 2I(h)\,.$$

Let $h = \frac{1}{2}v$, we have

$$\delta J(u) = a(u,v) - I(v)\,. \tag{6}$$

Theorem 2 *The variation of the cost functional can be determined by*

$$\delta J(u) = \int_\Sigma p(u)v\,d\boldsymbol{x}\,dt\,,$$

where $p(u)$, denoting $p(\boldsymbol{x},t;u)$, is defined by the following adjoint initial boundary value problem

$$\begin{aligned}
&\frac{-\partial p(\boldsymbol{x},t;u)}{\partial t} - \nabla^2 p(\boldsymbol{x},t;u) = T(\boldsymbol{x},t;u) - z(\boldsymbol{x},t)\,, \quad (\boldsymbol{x},t)\in Q\,,\\
&\frac{\partial p}{\partial n} = 0\,, \quad (\boldsymbol{x},t)\in\Sigma\,,\\
&p(\boldsymbol{x},t_A) = 0\,, \quad (\boldsymbol{x},t)\in\Omega\,.
\end{aligned} \tag{7}$$

Proof: For simplicity in notation, we denote $T(\boldsymbol{x},t;w)$ by $T(w)$ throughout the proof. Now, by the definition of a and I as given in (4), we have from (6)

$$\delta J(u) = \int_Q [T(u) - z]\,[T(v) - T(0)]\,d\boldsymbol{x}\,dt\,, \tag{8}$$

which, on using $(7)_1$ (the first equation of (7)), becomes

$$\delta J(u) = \int_Q \left\{\frac{-\partial p(u)}{\partial t} - \nabla^2 p(u)\right\}[T(v) - T(0)]\,d\boldsymbol{x}\,dt\,. \tag{9}$$

Using Green's Theorem and making use of equations $(1)_1$, $(1)_3$ and $(7)_2$, we have

$$\begin{aligned} & -\int_Q \nabla^2 p(u)\,[T(v)-T(0)]\;d\boldsymbol{x}\,dt \\ & = \int_Q p(u)\left[\frac{\partial T(0)}{\partial t}-\frac{\partial T(v)}{\partial t}\right]\,d\boldsymbol{x}\,dt+\int_\Sigma p(u)v\,d\boldsymbol{x}\,dt\,. \end{aligned} \tag{10}$$

Further, by using the product rule for differentiation and noting equation $(7)_3$ and utilizing $T(\boldsymbol{x},0;v)=T(\boldsymbol{x},0;0)$, we have

$$-\int_Q \frac{\partial p(u)}{\partial t}[T(v)-T(0)]\,d\boldsymbol{x}\,dt=\int_Q p(u)\left[\frac{\partial T(v)}{\partial t}-\frac{\partial T(0)}{\partial t}\right]\,d\boldsymbol{x}\,dt\,. \tag{11}$$

Substituting (10) and (11) into (9), we have

$$\delta J(u)=\int_\Sigma p(u)v\,d\boldsymbol{x}\,dt\,. \tag{12}$$

♠

3 Numerical algorithm

The solution of the COP problem (2) is difficult, as it involves the determination of the boundary control $u(\boldsymbol{x},t)$ at an infinite number of points on Σ. Thus, in order to find the numerical solution of the problem, we approximate the problem by a finite dimensional optimization problem. For this purpose, we firstly discretize Σ as N equal-sized subregions $\Delta\Sigma_i$ with $\Sigma=\bigcup_{i=1}^N\Delta\Sigma_i$ and then approximate $u(\boldsymbol{x},t)$ as a piecewise continuous function, namely

$$u(\boldsymbol{x},t)=u_i\,,\quad \forall(\boldsymbol{x},t)\in\Delta\Sigma_i\,,\quad i=1,2,3,\ldots,N\,.$$

Thus the problem becomes to find $\boldsymbol{u} = [u_1, u_2, u_3, \ldots, u_N]^T$ such that

$$J(\boldsymbol{u}) = \inf_{v_i \in U} J(\boldsymbol{v}) .$$

As the cost functional now depends on $u_1, u_2, u_3, \ldots, u_N$, we define the gradient of the cost functional by

$$\boldsymbol{G} = \left[\frac{\partial J}{\partial u_1}, \frac{\partial J}{\partial u_2}, \ldots, \frac{\partial J}{\partial u_N},\right]^T ,$$

where

$$\begin{aligned}\frac{\partial J}{\partial u_i} &= \frac{1}{\Delta u_i}\left[J(u_1, u_2, \ldots, u_i + \Delta u_i, \ldots, u_N)\right.\\ &\qquad \left. - J(u_1, u_2, \ldots, u_i, \ldots, u_N)\right] .\end{aligned}$$

Using (12) with

$$\frac{1}{2} v(\boldsymbol{x}, t) = h(\boldsymbol{x}, t) = \begin{cases} \Delta u_i\,, & \text{on } \Delta\Sigma_i\,; \\ 0\,, & \text{otherwise}\,; \end{cases}$$

we have by using the one-point quadrature rule

$$\frac{\partial J}{\partial u_i} \approx \frac{1}{\Delta u_i}\left[2\,|\Delta\Sigma_i|\,p_i \Delta u_i\right] = 2\,|\Delta\Sigma_i|\,p_i\,,$$

where p_i is the solution of the adjoint system (7) corresponding to $\Delta\Sigma_i$ and $|\Delta\Sigma_i|$ is the area of the subregion $\Delta\Sigma_i$. With the gradient obtained, the gradient type algorithm of Table 1 determines the optimal value of $\boldsymbol{u}$ based on the Fletcher-Reeves method [10, 2, 9].

4 Solution of the state system

To determine the gradient of the cost functional, we need to solve the state system for $T(\boldsymbol{x}, t; u)$ and then the adjoint system for $p(\boldsymbol{x}, t; u)$.

Table 1: Numerical Algorithm

1. Choose an initial boundary control $\boldsymbol{u}^0$. If $\boldsymbol{G}(\boldsymbol{u}^0) = 0$, $\boldsymbol{u}^0$ is the solution of the problem.
2. Set the first searching direction $\boldsymbol{S}^0 = -\boldsymbol{G}(\boldsymbol{u}^0)$.
3. Set $\boldsymbol{u}^1 = \boldsymbol{u}^0 + \alpha^0 \boldsymbol{S}^0$, with α^0 being the optimal step length in the searching direction $\boldsymbol{S}^0$. Set $i = 1$ and go to step 4.
4. Find $\boldsymbol{G}(\boldsymbol{u}^i)$ by solving the state and adjoint systems and then set $\boldsymbol{S}^i = -\boldsymbol{G}(\boldsymbol{u}^i) + \beta^i \boldsymbol{S}^{i-1}$, with $\beta^i = [\boldsymbol{G}(\boldsymbol{u}^i), \boldsymbol{G}(\boldsymbol{u}^i)]/[\boldsymbol{G}(\boldsymbol{u}^{i-1}), \boldsymbol{G}(\boldsymbol{u}^{i-1})]$.
5. Compute the optimum step length α^i in the searching direction $\boldsymbol{S}^i$ and update $\boldsymbol{u}$ by $\boldsymbol{u}^{i+1} = \boldsymbol{u}^i + \alpha^i \boldsymbol{S}^i$.
6. Test the optimality of $\boldsymbol{u}^{i+1}$. If $\boldsymbol{u}^{i+1}$ is optimum, stop the process. Otherwise, set $i = i + 1$ and go to step 4.

Both systems are parabolic type initial boundary value problems. Various numerical methods, such as the finite element method and the finite difference method can be used to solve these problems. In the present work, the finite element method [13] is used for the solution. To keep details to a minimum, in the following, we briefly describe only the numerical technique for the solution of the state system. The adjoint system is solved similarly.

To find the numerical solution of the state system (1), we firstly multiply both sides of equation $(1)_1$ by an arbitrary function Φ and integrate over the domain Ω to yield

$$\int_\Omega \Phi \left(\frac{\partial T}{\partial t} - \nabla^2 T \right) d\Omega = \int_\Omega \Phi \left[f(\boldsymbol{x}, t) \right] d\Omega \,. \tag{13}$$

Noting the boundary condition and using integration by parts and the divergence theorem, we have

$$\int_\Omega \Phi \frac{\partial T}{\partial t} \, d\Omega + \int_\Omega \frac{\partial \Phi}{\partial x_j} \frac{\partial T}{\partial x_j} d\Omega = \int_\Omega \Phi \left[f(\boldsymbol{x}, t) \right] d\Omega + \int_\Gamma \Phi u \, d\Gamma \,. \tag{14}$$

To solve the initial boundary value problem, the domain Ω is divided into a finite number of simple shaped regions Ω_e $(e = 1, 2, \ldots, E)$ called elements. Consequently, the boundary Γ of the domain Ω is divided into a number of boundary segments Γ_b $(b = 1, 2, \ldots, B)$. Within each element, the coordinate dependent variables T and Φ are interpolated by functions of compatible order, in terms of values to be determined at a set of nodal points. Denoting $\boldsymbol{T}_e$ and $\boldsymbol{\Phi}_e$ as the column vectors of the element nodal point values of T and Φ respectively, and $N(\boldsymbol{x})$ as the interpolation function, then T and Φ within each element are

$$T = N^T \boldsymbol{T}_e \,, \quad \Phi = \boldsymbol{\Phi}_e^T N \,. \tag{15}$$

Substituting (15) into (14), we have

$$\sum_{e=1}^{E} \boldsymbol{\Phi}_e^T \left\{ c_e \frac{\partial \boldsymbol{T}_e}{\partial t} + k_e \boldsymbol{T}_e \right\} = \sum_{e=1}^{E} \boldsymbol{\Phi}_e^T \boldsymbol{f}_e + \sum_{b=1}^{B} \boldsymbol{\Phi}_b^T \boldsymbol{f}_b \,, \tag{16}$$

where

$$c_e = \int_{\Omega_e} N N^T \, d\Omega \,, \qquad k_e = \int_{\Omega_e} \frac{\partial N}{\partial x_j} \frac{\partial N^T}{\partial x_j} \, d\Omega \,,$$
$$\boldsymbol{f}_e = \int_{\Omega_e} f(\boldsymbol{x}, t) N \, d\Omega \,, \qquad \boldsymbol{f}_b = \int_{\Gamma_b} u N \, d\Gamma \,. \tag{17}$$

Using a standard finite element assembling procedure, equation (16) is represented in matrix form

$$\boldsymbol{\Phi}^T \left\{ C \left(\frac{\partial \boldsymbol{T}}{\partial t} \right) + K \boldsymbol{T} \right\} = \boldsymbol{\Phi}^T \boldsymbol{F} \,. \tag{18}$$

Further, due to the arbitrary nature of Φ, we have from equation (18)

$$C \left(\frac{\partial \boldsymbol{T}}{\partial t} \right) + K \boldsymbol{T} = \boldsymbol{F} \,, \tag{19}$$

which constitutes a system of N first-order ordinary differential equations with N unknown values of $\boldsymbol{T}$ and is solved by using a standard time stepping scheme.

5 Numerical results

To test the numerical algorithm developed, consider the following example. Find $u(\boldsymbol{x}, t)$ such that

$$J(u) = \int_Q \left[T(\boldsymbol{x}, t; u) - \boldsymbol{z}(\boldsymbol{x}, t) \right]^2 \, d\boldsymbol{x} \, dt \,, \tag{20}$$

is minimized subject to $T(\boldsymbol{x}, t; u)$ being governed by

$$\begin{aligned} \frac{\partial T(\boldsymbol{x}, t; u)}{\partial t} - \nabla^2 \left[T(\boldsymbol{x}, t; u) \right] &= f(\boldsymbol{x}, t) \,, \quad \text{in } Q \\ T(\boldsymbol{x}, 0) &= T^0(\boldsymbol{x}) \,, \quad \text{in } \Omega \\ \frac{\partial T(\boldsymbol{x}, t)}{\partial n} &= u(\boldsymbol{x}, t) \,, \quad \text{on } \Sigma \,, \end{aligned} \tag{21}$$

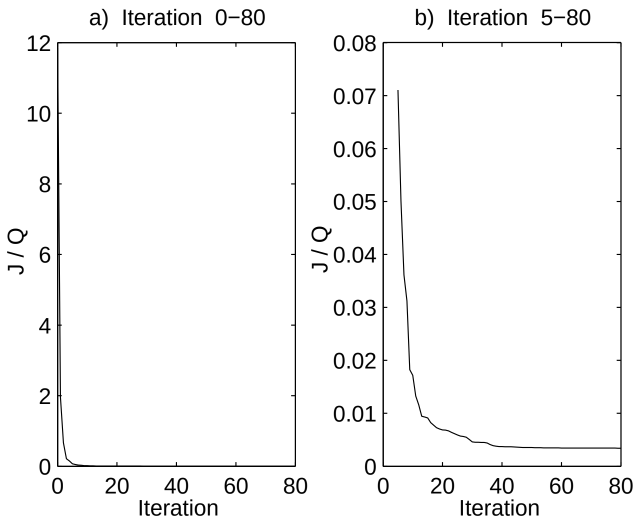

Figure 2: Value of the scaled cost functional (J/Q) versus iteration.

where $z(\boldsymbol{x}, t)$ is the target system state given by

$$z(\boldsymbol{x}, t) = 50e^t + \left(x_1^2 + x_2^2 + 0.5x_1^2 x_2^2\right) t\,.$$

Ω is the square region $[0,4]\times[0,4]$, $Q = \Omega\times(0, t_A]$, Γ is the boundary of Ω, Σ denotes $\Gamma \times (0, t_A]$, $t_A = 0.5$, $T^0(\boldsymbol{x}) = 50$ and the function

$$f(\boldsymbol{x}, t) = 50e^t + (1-t)(x_1^2 + x_2^2) + \frac{x_1^2 x_2^2}{2} - 4t\,.$$

For this particular problem, the exact solution for u is

$$u(\boldsymbol{x}, t)_{\text{exact}} \quad = \quad \begin{cases} 8t + 4x_2^2 t\,, & \text{on } x_1 = 4\,; \\ 8t + 4x_1^2 t\,, & \text{on } x_2 = 4\,; \\ 0\,, & \text{on } x_1 = 0 \text{ and } x_2 = 0\,. \end{cases}$$

To validate the numerical algorithm, we use it to solve the problem and then compare the numerical results with the exact solution.

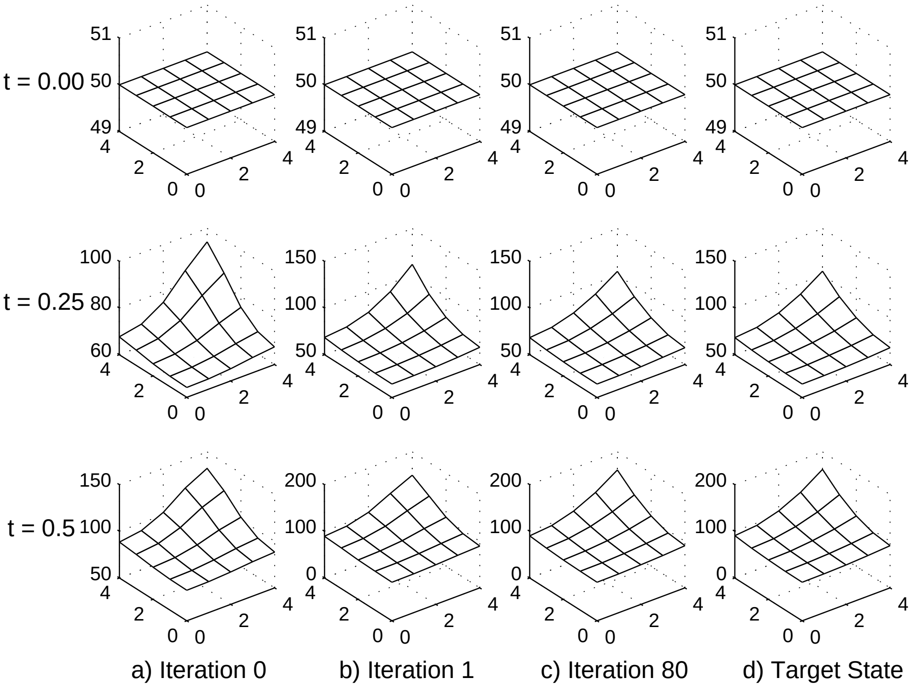

Figure 3: Comparison between computed states and the target state .

Figure 2 shows the variation of the value of the scaled cost functional (J/Q) in the iteration process. Figure 3 shows the computed system state $T(\boldsymbol{x}, t_i, \boldsymbol{u})$ at various stages of the iteration process against the target state. It is noted that, the system state converges to the target state. Figure 4 shows the variation of the computed boundary control $\boldsymbol{u}$ during the iteration process at various time steps. The results show that the numerical algorithm is robust.

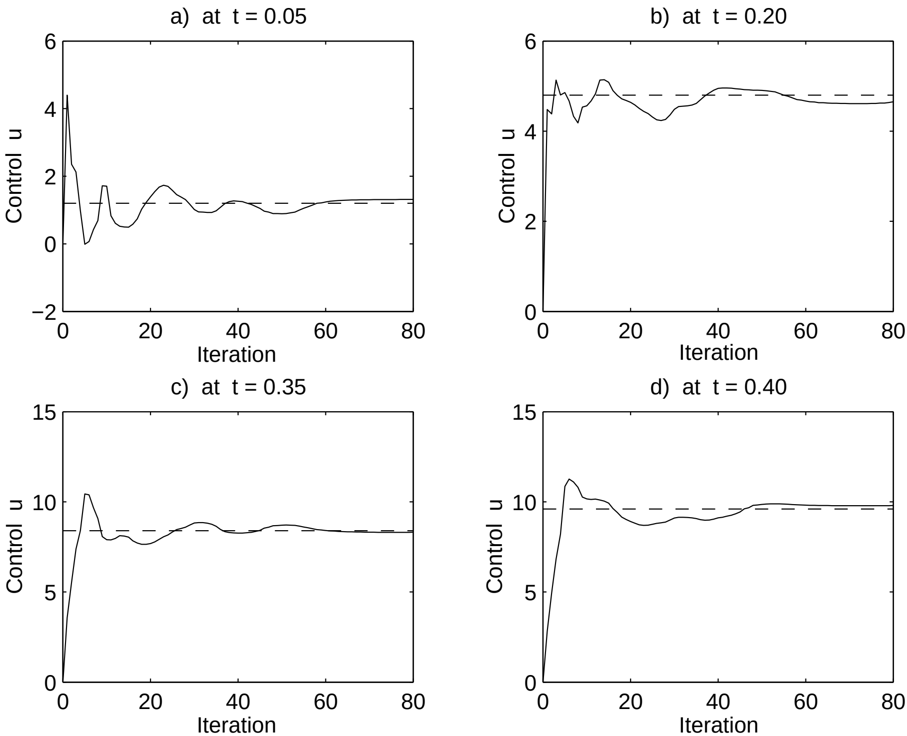

Figure 4: Comparison between the computed boundary control at the typical point $(2,4)$ and the exact solution: —— Computed result; - - - Exact result.

References

[1] Engl, H. W. and Langthaler, T. (1978), "On an inverse problem for a nonlinear heat equation connected with continuous casting of steel", *Optimal control of differential equations,* Birkhäuser Verlag, Boston. C838

[2] Gabay, D. (1982), "Reduced quasi-Newton methods with feasibility improvement for nonlinearly constrained optimization", *Mathematical Programming Study,* Vol. 16, pp. 18–44. C842

[3] Hill, J. M and Wu, Y. H. (1994), "On a nonlinear Stefan problem arising in continuous casting of steel", *Acta machanica,* Vol. 107, pp. 183–198. C837

[4] Latinen E., and Neittaanmaki, P. (1998), "On numerical simulation of the continuous casting process", *J. Eng. Math,* Vol. 22, pp. 164–168. C837

[5] Murray W., Pricet. J.P. (1995), "A sequential quadratic programming algorithm using an incomplete solution of the subproblem", *Numer. SIAM J. Optim.,* Vol. 5, pp. 590–640. C839

[6] Neittaanmäki, P. (1978), "On the control of the secondary cooling in the continuous casting process", *Optimal control of differential equations,* Birkhäuser Verlag, Boston C838

[7] Pawlow, I;, Shindo Y. and Satawa, T. (1985), "Numerical solution of multidimension two - phase Stefan problem", *Numer. Funct. Anal. Optimiz.,* Vol. 8, pp. 55–84. C838, C839

[8] Phillipson, G.A. (1971), *Identification of Distributed Systems,* American Elsevier Publishing Company Inc., New York. C839

[9] Rao, S. S. (1996), *Engineering optimization — Theory and Practice,* John Wiley & Sons, New York. C842

[10] Shanno D. F. (1983), "Recent advances in gradient based unconstrained optimization techniques for large problem", *ASME J. of Mechanisms, Transmission, and Automation in Design,* Vol. 105, pp. 155–159. C842

[11] Spellucci, P.(1998), "An SQP method for general nonlinear programs using only equality constrained subproblems", *Mathematical Programming,* Vol. 82, pp. 413–448. C839

[12] Tröltzech, T. (1989), "On the semigroup approach for the optimal control of semilinear parabolic equations including distributed and boundary control", *Zeitschr. F. Analysis and ihre Anwedungen,* Vol. 8, pp. 431–443. C839

[13] Wu, Y. H., Hill, J. M. and Flint, P. (1994), "A novel finite element method for heat transfer in the continuous caster", *J. Austral. Math. Soc.,* Ser. B, Vol. 35, pp. 263–288. C844