A NEW METHOD FOR SOLVING A CLASS OF HEAT CONDUCTION EQUATIONS

by

Yi TIAN, Zai-Zai YAN* [*] *and Zhi-Min HONG

College of Sciences, Inner Mongolia University of Technology, Hohhot, China

Original scientific paper
DOI: 10.2298/TSCI1504205T

A numerical method for solving a class of heat conduction equations with variable coefficients in one dimensional space is demonstrated. This method combines the Crank-Nicolson and Monte Carlo methods. Using Crank-Nicolson method, the governing equations are discretized into a large sparse system of linear algebraic equations, which are solved by Monte Carlo method. To illustrate the usefulness of this technique, we apply it to two problems. Numerical results show the performance of the present work.

Key words: *heat conduction equation, Crank-Nicolson method, Monte Carlo method*

Introduction

In this paper, we consider a problem of identifying an unknown function $u(x,t)$ in the heat conduction equation [1]:

$$u_t = a(t)u_{xx}, \quad 0 < x < 1, \quad 0 < t \le T \tag{1}$$

with an initial condition $u(x,0) = \rho(x)$, $0 < x < 1$, and the boundary conditions $u(0,t) = \varphi(t)$, $u(1,t) = \psi(t)$, and $t > 0$, where $a(t)$, $\rho(x)$, $\varphi(t)$, and $\psi(t)$ are known functions.

Certain types of physical problems can be modeled by eq. (1). For example, let an aluminum rod of length 10 cm be initially at the uniform temperature 0 °C. Suppose that at time $t = 0$, the end $x = 0$ is heated to 100 °C, while the end $x = 10$ is heated to 50 °C and both are thereafter maintained at those temperatures. One needs to find the temperature distribution $u(x, t)$ in the rod at any time t. This problem can be modeled by the equations:

$$\begin{cases} u_t = a(t)u_{xx}, & 0 < x < 10, \quad t > 0 \\ u(x,0) = 0 \ \text{°C}, & 0 < x < 10 \\ u(0,t) = 100 \ \text{°C}, & \qquad\qquad\quad t > 0 \\ u(10,t) = 50 \ \text{°C}, & \qquad\qquad\quad t > 0 \end{cases}$$

where $a(t) = 0.835$ cm^2/s, that is the thermal coefficient of aluminum.

Theorem 1. The solution of the problem (1) exists and that is unique if $\rho(x)$, $\varphi(t)$, and $\psi(t)$ are continuous functions and $a(t) > 0$ [2].

[*] Corresponding author; e-mail: zz.yan@163.com

The Crank-Nicolson method is employed to discretize the problem (1), a large sparse system of linear algebraic equations is obtained. There are many numerical algorithms to solve large systems of linear algebraic equations $AX = b$, where $A \in R^{n\times n}$ and $X, b \in R^n$. They are typically classified as one of iterative, Monte Carlo and direct methods. The iterative methods are preferred for sparse *A* with relatively small *n* and high precision. When *n* is large and the required precision is relatively low, Monte Carlo methods are preferred [3]. Direct methods are favorable for dense A with relatively small *n*. In this paper, we shall use Monte Carlo method to solve *AX* = *b*.

Monte Carlo algorithms have many advantages. For one thing, these algorithms are parallel, and have high efficiency [4], for another thing, these algorithms are preferable for solving large sparse systems of linear algebraic equations, such as those arising from approximations of PDE [1, 5, 6]. Many numerical methods in which used Crank-Nicolson scheme to discretize PDF, such as diffusion equation, non-linear parabolic equation, *etc.*, have successfully solved these equations [7-9].

Crank-Nicolson method for discretizing

The domain $[0, 1]\times[0, T]$ is divided into an $n \times N$ mesh with the spatial-step size *h* = 1/*n* in the x-direction and the time-step size *τ* = *T*/*N*, respectively. Denote $x_i = ih$, $t_k = k\tau$ ($0 \le i \le n$, $0 \le k \le N$) and $u_i^k \approx u(ih, k\tau)$. We define the difference operators as:

$$\delta_t u_i^{k+\frac{1}{2}} = \frac{1}{\tau}(u_i^{k+1} - u_i^k),\ \delta_x u_i^k = \frac{1}{h}(u_{i+1/2}^k - u_{i-1/2}^k),\ \delta_x^2 u_i^k = \frac{1}{h^2}(u_{i+1}^k - 2u_i^k + u_{i-1}^k),\ u_i^{k+\frac{1}{2}} = \frac{1}{2}(u_i^k + u_i^{k+1})$$

Let's consider eq. (1) at point $(x_i, t_{k+1/2})$,

$$\frac{\partial u}{\partial t}(x_i, t_{k+1/2}) = a(t_{k+1/2})\frac{\partial^2 u}{\partial x^2}(x_i, t_{k+1/2}) \tag{2}$$

By Taylor's expansion:

$$\frac{\partial^2 u}{\partial x^2}(x_i, t_{k+1}) = \frac{\partial^2 u}{\partial x^2}(x_i, t_{k+1/2}) + \frac{\partial^3 u}{\partial x^2 \partial t}(x_i, t_{k+1/2})(t_{k+1} - t_{k+1/2}) +$$

$$+\frac{1}{2\,!}\frac{\partial^4 u}{\partial x^2 \partial t^2}(x_i, \xi_{ik})(t_{k+1} - t_{k+1/2})^2, \qquad t_{k+1/2} < \xi_{ik} < t_{k+1} \tag{3}$$

$$\frac{\partial^2 u}{\partial x^2}(x_i, t_k) = \frac{\partial^2 u}{\partial x^2}(x_i, t_{k+1/2}) + \frac{\partial^3 u}{\partial x^2 \partial t}(x_i, t_{k+1/2})(t_k - t_{k+1/2}) +$$

$$+\frac{1}{2\,!}\frac{\partial^4 u}{\partial x^2 \partial t^2}(x_i, \xi_{ik}^{'})(t_k - t_{k+1/2})^2, \qquad t_k < \xi_{ik}^{'} < t_{k+1/2} \tag{4}$$

Hence

$$\frac{\partial^2 u}{\partial x^2}(x_i, t_{k+1/2}) = \frac{1}{2}\left[\frac{\partial^2 u}{\partial x^2}(x_i, t_k) + \frac{\partial^2 u}{\partial x^2}(x_i, t_{k+1})\right] - \frac{\tau^2}{8}\frac{\partial^4 u}{\partial x^2 \partial t^2}(x_i, \zeta_{ik}) \tag{5}$$

where $1 \le i \le n-1, k \ge 0$, and $t_k < \zeta_{ik} < t_{k+1}$. Similarly, we have:

$$\frac{\partial u}{\partial t}(x_i, t_{k+1/2}) = \delta_t u_i^{k+\frac{1}{2}} - \frac{\tau^2}{24}\frac{\partial^3 u}{\partial t^3}(x_i, \eta_{ik}), \qquad t_k < \eta_{ik} < t_{k+1} \tag{6}$$

$$\frac{\partial^2 u}{\partial x^2}(x_i, t_k) = \delta_x^2 u_i^k - \frac{h^2}{12}\frac{\partial^4 u}{\partial x^4}(\xi_{ik}, t_k), \qquad t_k < \xi_{ik} < t_{k+1} \tag{7}$$

Based on eqs. (5), (6), and (7), eq. (2) can be transformed:

$$\delta_t u_i^{k+\frac{1}{2}} - a(t_{k+1/2})\delta_x^2 u_i^{k+\frac{1}{2}} = R_{ik} \tag{8}$$

where R_{ik} is higher order term. Omit R_{ik} from eq. (8), and consider initial condition and boundary conditions at point (x_i, t_k), we have:

$$\delta_t u_i^{k+\frac{1}{2}} = a(t_{k+1/2})\delta_x^2 u_i^{k+\frac{1}{2}}, \qquad 1 \le i \le n-1, \qquad k \ge 0 \tag{9}$$

with $u_i^0 = \rho(x_i)$, $0 \le i \le n,$ and $u_0^k = \varphi(t_k), u_n^k = \psi(t_k), k \ge 0.$

According to eq. (9) , thus further the equation is expressed as:

$$-\frac{a(t_{k+1/2})r}{2}u_{i-1}^{k+1} + \left[1 + a(t_{k+1/2})r\right]u_i^{k+1} - \frac{a(t_{k+1/2})r}{2}u_{i+1}^{k+1} =$$

$$= \frac{a(t_{k+1/2})r}{2}u_{i-1}^k + \left[1 - a(t_{k+1/2})r\right]u_i^k + \frac{a(t_{k+1/2})r}{2}u_{i+1}^k \tag{10}$$

where $r = \tau/h^2$, $1 \le i \le n-1$, and $k \ge 0.$

Equation (10) can be written as the matrix form:

$$AU = b \tag{11}$$

where
$A = a(t_{k+1/2})rJ + I$, $B = -a(t_{k+1/2})rJ + I$, and

$$b = B(u_1^k, u_2^k, \cdots, u_{n-1}^k)^t + a(t_{k+1/2})r\left[\frac{\varphi(t_k) + \varphi(t_{k+1})}{2}, 0, \cdots, 0, \frac{\psi(t_k) + \psi(t_{k+1})}{2}\right]^t$$

$$J = \begin{pmatrix} 1 & -1/2 & & & \\ -1/2 & 1 & -1/2 & & \\ & \ddots & \ddots & \ddots & \\ & & -1/2 & 1 & -1/2 \\ & & & -1/2 & 1 \end{pmatrix}, \quad I = \begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & \\ & & & & 1 \end{pmatrix}, \quad U^t = \left(u_1^{k+1}, u_2^{k+1}, \cdots, u_{n-1}^{k+1}\right)$$

Theorem 2. The Crank-Nicolson scheme [9] is consistent with the parabolic eq. (1).

Proof. According to eqs. (2)-(8), we use Crank-Nicolson method to convert eq. (1) into eq. (8). Taking the focus on R_{ik}, we have that $R_{ik} = O(\tau^2) + O(h^2)$. Now as $h \to 0$ and $\tau \to 0$, then $R_{ik} \to 0$. Consequently, eq. (9) is consistent with the parabolic eq. (1).

Theorem 3. The Crank-Nicolson scheme [9] is unconditionally stable.

Proof. According to the Fourier series method and Von Neumann's method, we assume the error function $E_{p,q} = \mathrm{e}^{i\beta ph}\xi^q$, where $\xi = \mathrm{e}^{\alpha v}$, and α is a complex constant. The error will not increase as t increases provided that $|\xi|\le 1$. Substituting $E_{p,q}$ into eq. (9), we have:

$$-\mathrm{e}^{ih(-1+p)\beta} r\xi^{1+q} a(t_{k+1/2}) - \mathrm{e}^{ih(1+p)\beta} r\xi^{1+q} a(t_{k+1/2}) + 2\mathrm{e}^{ihp\beta}\xi^{1+q}[1+ra(t_{k+1/2})] =$$

$$= \mathrm{e}^{ih(-1+p)\beta} r\xi^{q} a(t_{k+1/2}) + \mathrm{e}^{ih(1+p)\beta} r\xi^{q} a(t_{k+1/2}) + 2\mathrm{e}^{ihp\beta}\xi^{q}[1-ra(t_{k+1/2})]$$

Divided by $\mathrm{e}^{i\beta ph}\xi^q$ leads to:

$$\xi = \frac{1-ra(t_{k+1/2})+ra(t_{k+1/2})\cos(h\beta)}{1+ra(t_{k+1/2})-ra(t_{k+1/2})\cos(h\beta)}$$

For $r>0$ and $a(t_{k+1/2})>0$, the condition $|\xi| \le 1$ is true.

Lemma 1. Given that we have a well-posed linear initial value problem and a linear finite difference approximation, and that the consistency condition exists, therefore stability is the necessary and sufficient condition for convergence [2].

Theorem 4. If $\mathcal{U}$ is the exact solution of the problem (1) and u is the exact solution of the finite difference eq. (9), then u converges to $\mathcal{U}$ as h and τ tend to zero.

Proof. From theorems 2 and 3 and lemma 1, we can obtain theorem 4.

The solution of the linear system of algebraic equation

Consider the following Jacobi over-relaxation iterative method for solving eq. (10):

$$u_i^{(k)} = (1-\gamma)u_i^{(k-1)} + \frac{\gamma}{a_{ii}}\left\{ b_i - \sum_{j=1}^{i-1} a_{ij}u_j^{(k-1)} - \sum_{j=i+1}^{n-1} a_{ij}u_j^{(k-1)} \right\} \tag{12}$$

where $i=1, 2, \cdots, n-1$ and $\gamma\in(0,1]$. Equation (12) can be written as:

$$U^{(k)} = LU^{(k-1)} + f, \qquad k=1, 2, \cdots \tag{13}$$

where $U^{(k)} = (u_1^{(k)}, u_2^{(k)}, \cdots, u_{n-1}^{(k)})^T$, $L = I - DA$, $f = Db$, and $D = \mathrm{diag}(\gamma/a_{11},\cdots,\gamma/a_{n-1,n-1})$. We convert eq. (11) to $U = LU + f$. Let $U^{(0)} = 0$, from eq. (13), we obtain:

$$U^{(k)} = (I + L + \cdots + L^{k-1})f = \sum_{m=0}^{k-1} L^m f, \quad k=1,2,\cdots \tag{14}$$

The eigenvalues of matrix L are:

$$\lambda_z = 1-\gamma + \frac{r\gamma a(t_{k+1/2})}{1+ra(t_{k+1/2})}\cos\frac{z\pi}{n}, \qquad z=1,\cdots n-1 \tag{15}$$

Then the spectral radius of matrix L meets $\rho(L)<1$, this property is a necessary and sufficient condition for convergence, *i. e.* as $k\to\infty$.

Monte Carlo method to solve linear system of algebraic equations

Consider a Markov chain $x_0 \to x_1 \to \ldots \to x_k \to \ldots$with state space $\{1, 2,\cdots, n-1\}$ and transition matrix $P = \{p_{ij}\}$, $i, j = 1,\cdots, n-1$. Let:

$$P(x_0 = i) = p_i, \quad P(x_{n-1} = j \mid x_{n-2} = i) = p_{ij} \tag{16}$$

The weight function W_m for the Markov chain is defined in the form:

$$W_0 = 1, \quad W_m = W_{m-1} \frac{l_{x_{m-1}x_m}}{p_{x_{m-1}x_m}}, \quad m = 1, 2, \cdots \tag{17}$$

Random variable associated with the sample path $x_0 \to x_1 \to \ldots \to x_k \to$ is defined as:

$$\Gamma_k[H] = \frac{H_{x_0}}{p_{x_0}} \sum_{m=0}^{k} W_m f(b_{x_m}) \tag{18}$$

where k is an integer number and $H^t = (h_1, ..., h_{n-1})$ is a given vector.

Theorem 5. The mathematical expectation of the random variable $\Gamma_k[H]$ is equal to the inner product $\langle H, U^{(k)} \rangle$, *i. e.*, $E(\Gamma_k[H]) = \langle H, U^{(k)} \rangle$.

To estimate $\langle H, U^{(k)} \rangle = h_1 u_1^{(k)} + ... + h_{n-1} u_{n-1}^{(k)}$, where $U^{(k)}$ is the k^{th} iterative solution of $U = LU + f$, we simulate M random paths with the length k:

$$x_0^{(s)} \to x_1^{(s)} \to x_2^{(s)} \to \cdots \to x_k^{(s)}, \quad s = 1, 2, \cdots, M \tag{19}$$

and evaluate the sample mean $\langle H, U^{(k)} \rangle \approx 1/M \; \Sigma_{s=1}^{M} \Gamma_k^{(s)}[H]$. Let H^t = (0, …, 0, 1, 0, …, 0), we get that $\langle H, U^{(k)} \rangle = u_j^{(k)}, j = 1, ..., n-1$. Therefore, $\Gamma_k[H]$ is an unbiased estimator of the $u_j^{(k)}$.

Discussion of the numerical result

In this section, we perform numerical tests for the proposed algorithm.

Example 1. Consider the problem $u_t = a(t)u_{xx}$, $0 < x < 1$, $t > 0$ with $u(x, 0) = 2\mathrm{e}^x$, $0 < x < 1$; $u(0, t) = 1 + (1 + 2t^3)(1 + t^3)^{-1}$ and $u(1, t) = \mathrm{e}[1 + (1 + 2t^3)(1 + t^3)^{-1}]$, $t > 0$, where $a(t) = (3t^2)(2 + 5t^3 + 3t^6)^{-1}$. The exact solution is $u(x, t) = \mathrm{e}^x[1 + (1 + 2t^3)(1 + t^3)^{-1}]$.

The results obtained for $u(x,t)$ are presented in tab. 1.

Example 2. Consider $u_t = a(t)u_{xx}$, $0 < x < 10$, $t > 0$ with $u(x, 0) = 0$ $(0 < x < 10)$, $u(0, t) = 100$, and $u(10, t) = 50$ $(t > 0)$, where $a(t) = 0.835$ [10].

Its exact solution $u(2, 10) = 64.8018$. The obtained results are presented in tab. 2.

Table 1. Result for u with γ = 0.98, h = 0.1, τ = 0.005, k = 3, and M = 5000

i	$u_{1,k}$		$u_{5,k}$		$u_{9,k}$	
	Numerical	Exact	Numerical	Exact	Numerical	Exact
10	2.210484	2.210480	3.297645	3.297649	4.919524	4.919514
30	2.213945	2.214059	3.303562	3.302988	4.927930	4.927479
50	2.227507	2.227344	3.323506	3.322807	4.957563	4.957046
70	2.255782	2.255778	3.364918	3.365225	5.020593	5.020326
80	2.276725	2.276818	3.396312	3.396614	5.067114	5.067152
100	2.333022	2.333139	3.481497	3.480634	5.193743	5.192495

Table 2. Result for $u(x, t)$ at point (2, 10) with γ = 0.98 and h = 2

τ	N = 500	N = 1000
10	79.72852194	80.00578982
5	64.84440263	64.83113599
2	64.75862352	64.86922777
1	64.74613185	64.74322734
0.5	64.65431986	64.75843972
0.2	64.81215905	64.76997201
0.1	64.78087397	64.81168721

Conclusions

This paper reports a new method to solve a heat conduction equation. We first discretize governing equations by Crank-Nicolson method, and obtain a large sparse system of linear algebraic equations $AX = b$, then use Jacobi over-relaxation iterative method and Monte Carlo method to solve the algebraic equations. The numerical results show that the proposed numerical method is efficient and accurate to estimate the temperature distribution $u(x, t)$ in eq. (1).

Acknowledgment

The work is supported by National Natural Science Foundation of China (Grant No 11161031), the Joint Specialized Research Fund for the Doctoral Program of Higher Education, Inner Mongolia Education Department (20131514110005) and Natural Science Foundation of Inner Mongolia (2013 MS0108).

References

[1] Farnoosh, R., *et al.*, Monte Carlo Method via a Numerical Algorithm to Solve a Parabolic Problem, *Appl. Math. Comput*, *190* (2007), 2, pp. 1593-1601
[2] Cannon, J. R., *The One-Dimensional Heat Equation,* Addison-Wesley, Menlo Park, Cal., USA, 1984
[3] Dimov, I. T., *et al.*, A New Iterative Monte Carlo Approach for Inverse Matrix Problem, *Journal of Computational and Applied Mathematics*, *92* (1998), 1, pp. 15-35
[4] Dimov, I. T., *et al.*, Monte Carlo Algorithms: Performance Analysis for Some Computer Architectures, *Journal of Computational and Applied Mathematics*, *48* (1993), 3, pp. 253-277
[5] Farnoosh, R., *et al.*, Monte Carlo Simulation via a Numerical Algorithm for Solving a Nonlinear Inverse Problem, *Commun. Nonlinear. Sci. Numer. Simulat.*, *15* (2010), 9, pp. 2436-2444
[6] Yan, Z. Z., Hong Z., Using the Monte Carlo Method to Solve Integral Equations Using a Modified Control Variate, *Applied Mathematics and Computation*, *242* (2014), Sept., pp. 764-777
[7] Shidfar, A., *et al.*, A Numerical Method for Solving of a Nonlinear Inverse Diffusion Problem, *Computers and Mathematics with Applications*, *52* (2006), 6-7, pp. 1021-1030
[8] Shidfar, A., *et al.*, A Numerical Solution Technique for a One-Dimensional Inverse Nonlinear Parabolic Problem, *Applied Mathematica and Computation*, *184* (2007), 2, pp. 308-315
[9] Hu, B., *et al.*, Crank-Nicolson Finite Difference Scheme for the Rosenau-Burgers Equation, *Applied Mathematica and Computation*, *204* (2008), 1, pp. 311-316
[10] Yu, Y. H., *et al.*, *Engineering Numerical Analysis,* Tsinghua University Press, Beijing, 2010

Paper submitted: January 16, 2014
Paper revised: May 25, 2015
Paper accepted: April 15, 2015