

The Solution of Heat Equation in One Dimensional using Matlab

Abdel Radi Abdel Rahman Abdel Gadir¹, Nedal Hassan Elbadawy², Ragaa Mohammed Haj Ibrahim³

¹Associate Professor, Department of Mathematics, Faculty of Education,
Omdurman Islamic University, Omdurman, Sudan

²Assistant Professor, Department of Mathematics, College of Science, Tabuk University, Tabuk, Saudi Arabia,

³Department of Mathematics, Faculty of Education, Elzaiem ElAzhary University, Omdurman, Sudan

Abstract : We discuss and explain the solution of elementary problems in solving partial differential equation, the kinds of problems that arise in various fields of sciences and engineering. This study aims to solve the heat equation in One dimensional using the Matlab. We followed the applied mathematical method and found the following results: Solving heat equation using Matlab is best than manual solution in terms of speed and accuracy and possibility of drawing surface and curve for heat equation using Matlab.

Keywords : Solution, Heat Equation, Matlab, Dimension

I. Introduction [2] [4]

This study dealing with solution of heat equation using Matlab. The heat equation representing partial differential equation.

For over one million years fire and it's product heat have contributed significantly to the rise of man. Thus naturally, the study of heat transfer has become very important. At first the physical properties of heat could only be guesses through the senses only after sir Isaac Newton and Gottfried Leibniz develop calculus in the seventeenth century could we mathematically model the physical properties of heat transfer. Today we know three types of heat transfer, conduction, convection and radiation. Although the mathematical formulation of all three types of heat transfer is important.

Matlab is high performance language for technical computing. Matlab which is short for matrix laboratory in corporate numerical computation symbolic computation graphics and programming as the name suggests it is particularly oriented towards matrix computations and it provides both state of the art algorithms and a simple easy to learn interface for manipulating matrices.

II. Heat Equation

i. One Dimensional Heat Equation: [6]

We consider the problem of heat flow in a conducting rod length l , with internal heat generation where the left hand boundary $x = 0$, is insulated and there is a loss l gain of heat energy by convection across the right – hand boundary $x = l$ this problem is modeled by the following boundary values and initial value

$$\frac{\partial u}{\partial t} = K \left(\frac{\partial^2 u}{\partial x^2} + \gamma u \right), \quad 0 < x < l, \quad t > 0$$

$$\frac{\partial u}{\partial x}(0, t) = 0$$

$$Ku(1, t) + \frac{\partial u}{\partial x}(l, t) = 0$$

$$U(x, 0) = F(x)$$

Here we assume that the material properties governing heat generation γ , thermal diffusivity K are all independent of x and t and are all positive constants. We will find the solution to this problem. Analysis the long term behavior of heat flow in the rod since this is a linear homogeneous P.D.E. , with linear and homogenous boundary condition as before we assume separated solution of the form

$$U(x, t) = X(x).T(t) \quad (1)$$

Substituting this into the P.D.E. and separating variables, we obtain

$$\frac{\dot{T}}{KT} - \gamma = \frac{\dot{X}}{X} = -\lambda$$

Where λ is the separation constant, this gives the following problems for X and T

$$\dot{X} + \lambda x = 0, \quad 0 < x < L, \quad \dot{T} + (\lambda - \gamma)KT = 0 \quad t > 0$$

$$\dot{x}(0) = 0$$

$$KX(L) + \dot{x}(l) = 0 \quad (2)$$

The problem for x is a regular Sturm-Liouville problem with $P(x) = 1$, $q(x) = 0$

For $0 \leq x \leq L$ and $\alpha_1 = 0, \beta_1 = 1, \alpha_2 = K, \beta_2 = 1$

Since $\alpha_1\beta_1 = 0$ and $\alpha_1\beta_1 \geq 0$ according to $\lambda = \mu^2$ if $\mu = 0$ the equation $\dot{x} = 0$ has general solution

$$X(x) = ax + b \quad (3)$$

and since $\dot{x}(0) = a = 0$ the solution becomes $X(x) = 0$

since $KX(e) + \dot{X}(e) = 0$ then $Kb = 0$, so that $b = 0$, and there are no nontrivial solutions thus $\lambda = 0$ is not an Eigen Value if $\mu \neq 0$ the equation becomes

$$\dot{x} + \mu^2 X = 0 \quad (4)$$

and has general solution : $X(x) = a \cos \mu x + b \sin \mu x$

$$\dot{X}(x) = -a\mu \sin \mu x + b\mu \cos \mu x$$

Applying the boundary condition the left hand boundary $x = 0$ we have $\dot{x}(0) = 0$ and $b = 0$, the solution is now $X(x) = a \cos \mu x$ with

$$\dot{x}(x) = -a\mu \sin \mu x$$

applying the boundary condition the right hand boundary $x = l$ we have $Xx(l) + \dot{x}(l) = 0$ which implies that

$$aK \cos \mu l - a\mu \cos \mu l = 0$$

That is

$$\tan \mu l = \frac{K}{\mu} \quad (5)$$

Since $\tan \mu l$ is a periodic function of μ with singularities and odd multiples of $\pi/2l$ and $\sin K/\mu$ is a decreasing function of μ , we have an infinite sequence of solutions and the Eigen values are

$$\lambda n = \mu^2 n;$$

Where $\tan \mu l = k/\mu_n$ the corresponding Eigen function are $X_n(x) = \cos \mu_n x$ and the corresponding solutions to the time equations are intersection of $y = \tan \mu l$ and $y = k/\mu$ for $n \geq 1$ the norm of the eigen function relative to the appropriate inner product

$$\begin{aligned} \|X_n\|^2 &= \langle x_n, x_n \rangle = \int_0^l x_n^2(x) dx = \int_0^l \cos^2 \mu_n dx \\ &= \frac{1}{2} \left(l + \frac{\sin 2\mu_n l}{2\mu_n} \right) \end{aligned} \quad (6)$$

Applying the principle of superposition we obtain

$$u(x, t) = \sum_{n=1}^{\infty} C_n X_n(x). T_n(t) = \sum_{n=1}^{\infty} C_n e^{-(\mu_n^2 - \gamma)k \cos \mu_n x} \quad (7)$$

and from the initial condition $u(x, 0) = F(x)$ we get $f(x) = \sum_{n=1}^{\infty} C_n \cos \mu_n x$.

Thus the problem is solved provided that we can find expansion of F above in interims of the eigen functions if $f \in (0, l)$ the expansion exists with coefficients given by:

$$C_n = \frac{\langle f, x_n \rangle}{\|X_n\|^2} = \frac{4\mu_n}{2\mu_n l + \sin 2\mu_n l} \cdot \int_0^l F(x) \cos \mu_n x$$

We can interpret the long term behavior of the results as follows. The eigen values are the positive solution to the transect indented equation

$$\tan \mu l = \frac{k}{\mu}$$

and hence the eigen value μ_n is the n intersection point of the graphs of

$$y = \tan \mu l \quad \text{or} \quad y = K/\mu$$

There fore

$$\frac{(n-1)\pi}{l} \leq \mu_n \leq \frac{(2n-1)\pi}{2l}$$

For $n \geq 1$ and we have

$$0 \leq \mu_1 \leq \frac{\pi}{2l}, \quad \frac{\pi}{l} \leq \mu_2 \leq \frac{3\pi}{2l}, \dots, \frac{(n-1)\pi}{l} \leq \mu_n \leq \frac{(2n-1)\pi}{2l}, \dots$$

Clearly μ_n depends on k , so we can write

$$\frac{(n-1)\pi}{l} \leq \mu_n(k) \leq \frac{(2n-1)\pi}{2l}$$

For $n \geq 1$ we consider two cases:

Case (i): perfect in solution if $k \rightarrow 0$ we have

$$\lim_{k \rightarrow 0} \mu_n(K) = \frac{(n-1)\pi}{l} \quad (8)$$

and in particular $\lim_{k \rightarrow 0} \mu_1(k) = 0$ other quantities simplify as follows:

$$\begin{aligned} \|X_n\|^2 &= \frac{\frac{1}{2}(l + \sin 2\mu_n l)}{2\mu_n} \\ &= \frac{\frac{1}{2}(l + \sin 2(n-1)\pi)}{2(n-1)\pi/L}; \quad \begin{cases} 1/2 & n \neq 1 \\ l & 1 \end{cases} \end{aligned} \quad (9)$$

$$C_n = \frac{2}{l} \int_0^l F(x) \cos\left((n-1)\frac{\pi}{l}\right) dx$$

$$u(x, t) = e^{\gamma kt} \left[c_1 + \sum_{n=2}^{\infty} C_n e^{\mu_n^2 k + \cos \mu_n x} \right] \quad (10)$$

Taking the limit as $t \rightarrow \infty$, and assuming that $c_1 > 0$ and we get

$$\lim_{t \rightarrow \infty} u(x, t) = +\infty$$

remember that $\gamma > 0$ represents internal heat generation as $k \rightarrow 0$ the boundary condition become

$$\frac{\partial u}{\partial x}(0, t) = 0, \quad \frac{\partial u}{\partial x}(l, t) = 0$$

Which are homogeneous Neumann conditions there fore, with perfect in solution no heat can escape thus the temperature must continue to rise indefinitely.

Case (ii): imperfect in solution if $k > 0$, in this case the solution is

$$u(x, t) = c_1 e^{(\gamma - \mu_1^2)kt} \cos \mu_1 x + \sum_{n=2}^{\infty} C_n e^{(\gamma - \mu_n^2)kt} \cos \mu_n x$$

and letting $t \rightarrow \infty$, we get

$$\lim_{t \rightarrow \infty} u(x, t) = \begin{cases} +\infty & \text{for } \gamma > \mu_1^2 \\ c_1 \cos \mu_1 x & \text{for } \gamma = \mu_1^2 \\ 0 & \text{for } \gamma < \mu_1^2 \end{cases}$$

With imperfect in solution there is heat loss what is saying is that if the internal heat generation is sufficiently large, that is if $\gamma > \mu_1^2$, heat is generated at a rate faster than it can escapes through the boundary the before the temperature rises in definitely on the other hand if internal heat generation is very low heat escapes faster that it is generated internally and the temperature eventually goes to zero.

But if the rate of internal heat generation is just right that is, if $\gamma = \mu_1^2$.

There is a balance between internal heat loss through the boundary and an equilibrium temperature distribution is reached. The critical value of internal heat generation is precisely the value of the smallest eigen value λ_1 .

III. Matlab

Definition(3.1): [1]

Matlab is one of the most widely used computational tools in sciences and engineering. There are three good reasons to learn a computational mathematics tools.

i. Matlab Partial Differential Equations Solver: [3]

Partial differential equation function that defines the components of the P.D.E. it computes the terms c, f and s and has the form

$$[c, f, s] = Pde\ fun(x, t, u, \partial u \partial x) \quad (11)$$

where x and t are scalars, and u and $\partial u \partial x$ are vectors that approximate the solution U and it's partial derivative with respect to x , c, f and s are column vectors c stores the diagonal elements of the matrix c .

ii. The Code of Partial Differential Equation in Matlab: [3]

Once you rewrite the P.D.E. in the form shown in (16) and identify the terms, you can code the P.D.E. in a function that pdepe can use the function must be of the form

$$[c, f, s] = pde\ fun(x, t, u) \partial u \partial x \quad (12)$$

The code below computes c, f and s .

iii. Heat Equation:[5]

We would like to solve the heat equation

$$U_t = U_{xx}, \quad U(t, 0) = 0, \quad U(t, L) = 1 \quad (0, x) = 2x$$

Matlab specifies such parabolic P.D.E. in the form

$$(x, t, u, u_r) u_t = x - m \frac{\partial}{\partial x} (x_m b(x, t, u, u_x) + s(x, r, u, u_x)) \quad (13)$$

With boundary conditions

$$P(xL, t, u) = q(xl, t).b(xL, t, u, u_x) = 0$$

$$P(x_r, t, u) + q(x_r, t).b(x_r, t, u, u_x) = 0$$

where xl represents the left endpoint of the boundary and x_r represents the right endpoint of the boundary and initial condition.

$u(0, x) = F(x)$ (observe that the same function appears) typically for clarity each set of functions will be specified in a separate M – file that is, the function c , b and s associated with the equations should be specified in one M – file condition in a second M – file (a gain keep in mind that b is the same and only needs to be specified once) and finally the initial function $F(x)$ in a third the command PDE will combine these M – file and return a solution to the problem in our example we have $C[x, t, u, u_x] = 1$, $b(x, t, u, u_x) = u_x s(x, t, u, u_x) = 0$ which we specify in the function M-file .

Function $(c, b, s) = eqn1(x, t, u, \partial u \partial x)$

%FQNI. Matlab function M – file that specifies %PDE in time and one space dimension , $c = 1, b = \partial u \partial x; s = 0$

For our boundary conditions, we have

$$P(0, t, u) = u, \quad q(0, t) = 0, \quad p(1, t, u) = u - 1, \quad q(1, t) = 0$$

Which we specify in the function M-file function $[P1, q1, pr, qr] = bc(x_1, u_1, x_r, u_r, t)$

%BC₁: Matlab function M –file that specifies boundary condition % for a PDE in time and one space dimension

$$P_1 = U_1; q_1 = 0; Pr = U_r = 1, q_r = 0.$$

IV. Solution of Heat Equation Using Matlab

$$\text{Function } [P1, q1, Pr, qr] = bc [X1, u1, Xr, Ur, t] \quad (14)$$

$$P1 = u1; q1 = 0; Pr = U_{r-1}; qr = 0$$

Which we specify in the function M –file:

function [pl,ql,pr,qr] = bc1(xl,ul,xr,ur,t)

%BC1: MATLAB function M-file that specifies boundary conditions
%for a PDE in time and one space dimension.

pl = ul;

ql = 0;

pr = ur-1;

qr = 0;

solution:

>> bc1(x, t, u, DuDx)

ans =

t

.....

$$\text{Function } [c, b, s] = eqn1(x, t, u, du, dx) \quad (15)$$

$c = 1, b = du \, dx; \, s = 0$; the code in Matlab , M- file

function [c,b,s] = eqn1(x,t,u,DuDx)

%EQN1: MATLAB function M-file that specifies
%a PDE in time and one space dimension.

c = 1;

b = DuDx;

s = 0;

solution

```
>> syms x t u DuDx
>> eqn1(x, t, u, DuDx)
ans =
1
```

function value = initial1(x)

**% initial1: MATLAB function M-file that specifies the initial condition
%for a PDE in time and one space dimension.**

value = 2*x/(1+x^2);

finally ready to solve the P.D.E. with pdepe. In the following script M-file, that by choose a grid of x and t values, solve the P.D.E. and create a surface plot of its solution (given in “Figure 1”).

function initial2

m = 0;

**%NOTE: m=0 specifies no symmetry in the problem. Taking
%m=1 specifies cylindrical symmetry, while m=2 specifies
%spherical symmetry.**

%

%Define the solution mesh

x = linspace(0,1,20);

t = linspace(0,2,10);

%Solve the PDE

sol = pdepe(m,@eqn1,@initial1,@bc1,x,t);

% Extract the first solution component as u.

u = sol(:,1);

%Plot solution

surf(x,t,u);

title('Surface plot of solution.');

xlabel('Distance x');

ylabel('Time t');

% A solution profile can also be illuminating.

figure

plot(x,u(end,:))

title('Solution at t = 2')

xlabel('Distance x')

ylabel('u(x,2)')

solution:

and find it useful to plot solution *profiles*, for which t is fixed, and u is plotted against x . The solution $u(t,x)$ is stored as a matrix indexed by the vector indices of t and x .

For example, $u(1,5)$ returns the value of u at the point $(t(1)x(5))$. We can plot u initially (at $t = 0$) with the command $plot(x,u(1,:))$ (see “Figure 2”).

Finally, a quick way to create a movie of the profile’s evolution in time is with the following Matlab sequence.

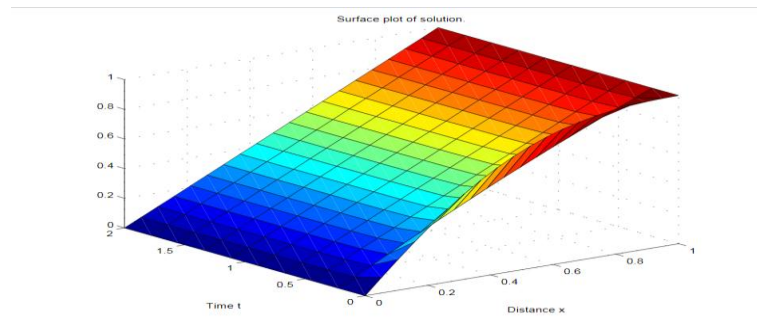
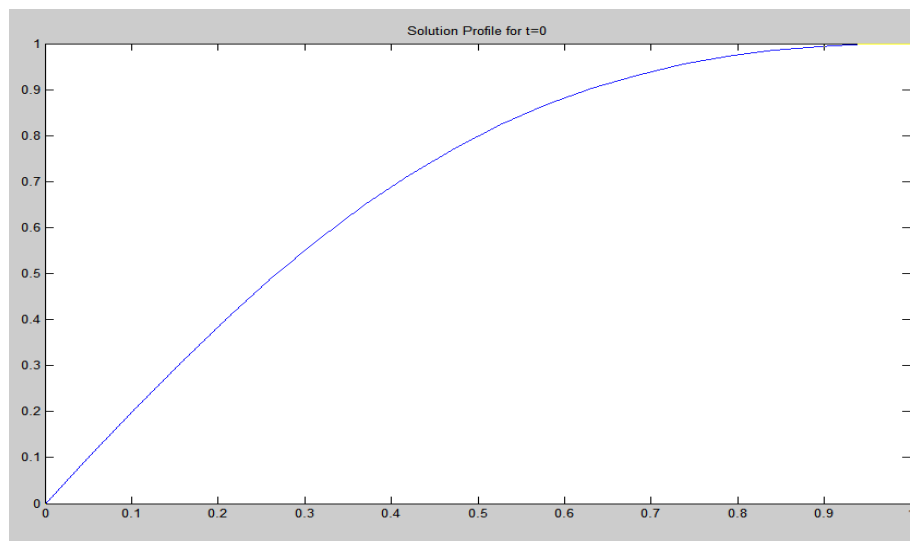


Figure1 "The Surface of Heat Equation in One Dimensiona"



"Figure2" The Curve of Heat Equation in One Dimensional

V. Results

Solving Heat Equation using Matlab is best than manual solution in terms of speed and accuracy, sketch possibility the curve and surface of heat equation using Matlab.

VI. Conclusion

Finally we say that the heat equation has a solution by matlab and it is very important to solve it using matlab.

References

- [1] David Mc. Mohon, Matlab Demystified, Mc Graw. Hill companies, 2007.
- [2] Mark. Cocken bach, Matlab titorial to accompany, partial differential equations analytical and numerical methods, 2nd edition, siam 2010.
- [3] Math works, Matlab the language of technical computing fourth printing, 2001.
- [4] Michael k. keane, partial differential equations, hall uppers saddle river new jersey, untied states.
- [5] P. Howard, partial differential equation in Matlab, 2010.
- [6] T. Hillen . I .E. leonard. H van roessel, partial differential equations theory and completely solved problems, first edition, john wiley & sons, Canada, 2012.