IOSR Journal of Mathematics (IOSR-JM)
e-ISSN: 2278-5728, p-ISSN: 2319-765X. Volume 14, Issue 2 Ver. II (Mar. - Apr. 2018), PP 12-17
www.iosrjournals.org

Consistent Three-Dimensional Finite Difference Modeling of Heat Transferwith non-homogenous boundary conditions in Solar Modules

Abbas Abdelaziz Guma[1], Mahmud Salem Mahmud[2]
[1](Department of Mathematics,AlzaiemAlazhariUniversity, Sudan)
[2](Department of Mathematics, AlzaiemAlazhariUniversity, Sudan)

Abstract: *This paper presents consistent and stable numerical solutionsof the three dimensional transient heat transfer problem with non-homogenous boundary conditions. Finite difference schemes; forward time, backward time, and Crank- Nicolson methods have been used to predict the temperature distribution acrossa photovoltaic solar cell with specified dimensions. Numerical predictionsusing the three schemes are compared with each other and tested against the exact solution of the same problem. Results of the numerical and analytical solutions are quite similar with levels of numerical errors kept very negligible. Furthermore, accuracy, consistency, convergence, and stability analysis of the finite difference solutions is investigated and presented in this study.Conditions for consistency, convergence and stability of each algorithm is derived in this paper. The methodology showed to be quite robust for such a time-dependent three-dimensional problem.*
Keywords *–heat equation, finite difference, photovoltaic solar cell.*

Date of Submission: 19-03-2018 Date of acceptance: 03-04-2018

I. Introduction

Photovoltaic solar systems are one of the most promising sources of electrical energy. Because of low efficiency of PV modules which estimates to be in the range between 13% and 20% [1], some practical techniques are used to increase the incoming radiation on the PV cells by using concentrators or using solar tracking. However, these techniques cause the temperature of these cells to riseabove the operating limit. This in turn, lowers the solar cell efficiency and subsequently its maximum power output. Sensitivity of the electrical efficiency to the thermal state of these solar modules motivates research on predictions of temperature distributions in the cell. Computational modeling has been favorable due to the lack of practical measurements. Numerical predictions of temperature distribution and its response to the amount of solar radiation give better insight into the factors controlling the efficiency of this solar system and ways of attaining optimum conditions. In addition, numerical methods are regarded as cheap yet accurate tools for testing newly proposed physical systems at several operating conditions. For example, it has been suggested that the loss in energy conversion would be partially avoided by heat extraction using cooling media.This would be tested numerically by applying a convection boundary condition to the mainly conduction heat equation. Also, computational predictions give much more details on the temperature distributions across the solar cell.

Application of finite difference computations to heat transfer problems varies from large industrial systems to microscale systems. Dai and Nassar [2] implemented the Crank–Nicholsonscheme on thin microscale films of microelectronic devices. They showed that the discrete energy method is unconditionally stable in three dimensional solutions of the heat equation. Notton et al. [3,4] developed a one-dimensional Finite Difference transient model to solve energy balance equations, between different layers of a PV panel. They also studied the effect of various convective heat transfer equations on the accuracy of the results. Shahzada Pamir Aly*et al.* developed a fully transient two-dimensionalfinite difference (FD) thermal model. They developed a completely generic computational code that can apply to any PV configuration. Also they have studied were the effects of including heat transfer from the sides of a PV panel and heat generation in the front glass cover [5]

This study presents numerical simulations of the temperature distribution on the PV module using three-dimensional finite difference algorithms. The temperature distribution inside a polycrystalline PV solar module (SOL-060-01) has been predicted by solving the heat diffusion equation numerically. Three numerical solutions and the exact analytical solution have all been produced at the same initial and boundary conditions for meaningful comparisons.

Table 1: Thermo-Physical Parameters.

Solar PV parameters	Value
The length of PV module, a	1.5m
The width of PV module, b	0.5m
The thickness of PV module, c	5cm
The thermal conductivity, k	0.5 w/m.k
Specific heat, Cp	871 J/kg.k
Density, ρ	2719 kg/m^3
The thermal diffusivity, α	$2.111 \times 10^{-7} m^2/sec$

The lower surface of the PV module is cooled and maintained at 20^oC, and the upper surface is hot as it is exposed to the solar irradiance and maintained at 60^oC. While, all sides of the PV module are thermally insulated. The dimensions and the electrical specifications of the PV module are shown in Table1.

II. The mathematical model

The governing equation and the boundary conditions for the transient heat diffusion inside the solar PV module is derived from the general form of the heat (energy) equation. The governing differential equation for this physical problem (3-D transient diffusion in a solar module) can bewritten as [6,7,8],

$$\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}+\frac{\partial^2 T}{\partial z^2}=\frac{1}{\alpha}\frac{\partial T}{\partial t} \quad (1)$$

$$; \quad T_x(0,y,z,t)=0 \quad ; \quad T_x(a,y,z,t)=0$$

$$; \quad T(x,0,z,t)=20 \quad ; \quad T(x,c,z,t)=60$$

$$; \quad T_z(x,y,0,t)=0 \quad ; \quad T_z(x,y,b,t)=0$$

$$; \quad T(x,y,z,0)=25xyz$$

The exact solution for the non-homogenous system above has been found analytically by using the assumption, $\mathrm{T(x,y,z,t)=u(y)+v(x,y,z,t)}$withseparation of variable technique and the resulting expression is [7,8],

$$\mathrm{T(x,y,z,t)}=\frac{\mathrm{T_2-T_1}}{\mathrm{c}}.\mathrm{y}+\mathrm{T_1}+\sum_{\mathrm{l=1}}^{\infty}\sum_{\mathrm{m=1}}^{\infty}\sum_{\mathrm{n=1}}^{\infty}\frac{30}{\mathrm{n^2 m l^2 \pi^5}}\mathrm{e}^{-\alpha\left(\left(\frac{\mathrm{n}}{a}\right)^2+\left(\frac{\mathrm{m}}{\mathrm{c}}\right)^2+\left(\frac{\mathrm{l}}{b}\right)^2\right)\pi^2 \mathrm{t}}\cos\frac{\mathrm{n\pi}}{\mathrm{a}}\mathrm{x}.\sin\frac{\mathrm{m\pi}}{\mathrm{c}}\mathrm{y}.\cos\frac{\mathrm{l\pi}}{b}\mathrm{z} \quad (2)$$

2.1 Finite difference algorithm

In this method, the partial differential equation is transformed to a set of algebraic equations expressing temperature at discrete points. This numerical formulation is derived by applying the finite difference and Taylor expansion for the function$\mathrm{T(x,y,z,t)}$. Spaces between nodal points can straightforwardly be taken as variable or constant. For simplicity of presented formulas, a derivation with a fixed grid spacing is considered here but the final numerical model is not constrained to this option.As a result,we get a system of linear algebraic equations which couldbe solved with algebraic methods [8,9],

$$\mathrm{T^{l+1}_{i,j,k}-T^{l}_{i,j,k}}=\lambda[\mathrm{T^{l}_{i+1,j,k}-2T^{l}_{i,j,k}+T^{l}_{i-1,j,k}+T^{l}_{i,j+1,k}-2T^{l}_{i,j,k}+T^{l}_{i,j-1,k}+T^{l}_{i,j,k+1}-2T^{l}_{i,j,k}+T^{l}_{i,j,k-1}}] \quad (3)$$

2.1.1 Theta algorithms

By taking a linear combination of right side of equation (3) on twoperiod of time we get,

$$\mathrm{T^{l+1}_{i,j,k}-T^{l}_{i,j,k}}=\lambda\theta\left[\mathrm{T^{l+1}_{i+1,j,k}+T^{l+1}_{i-1,j,k}+T^{l+1}_{i,j+1,k}+T^{l+1}_{i,j-1,k}+T^{l+1}_{i,j,k+1}+T^{l+1}_{i,j,k-1}-6T^{l+1}_{i,j,k}}\right]+\lambda(1-\theta)[\mathrm{T^{l}_{i+1,j,k}+T^{l}_{i-1,j,k}}$$
$$+\mathrm{T^{l}_{i,j+1,k}+T^{l}_{i,j-1,k}+T^{l}_{i,j,k+1}+T^{l}_{i,j,k-1}-6T^{l}_{i,j,k}}] \quad (4)$$

where the parameterθ represents a weighting factor[8,10].

2.1.1.1 Forward difference algorithm

The forward time algorithm is obtained by assigning$\theta=0$ in equation (4),

$$\mathrm{T^{l+1}_{i,j,k}}=(1-6\lambda)\mathrm{T^{l}_{i,j,k}}+\lambda[\mathrm{T^{l}_{i+1,j,k}+T^{l}_{i-1,j,k}+T^{l}_{i,j+1,k}+T^{l}_{i,j-1,k}+T^{l}_{i,j,k+1}+T^{l}_{i,j,k-1}}] \quad (5)$$

Equation (5) can be written in matrix form as follows,

$$\mathrm{T^{l+1}}=(\mathrm{I}+\lambda\mathrm{A})\mathrm{T^{l}} \quad (6)$$

2.1.1.2 Backward difference algorithm

The backward-time algorithm is obtained by assigning$\theta=1$ in equation (4),

$$\mathrm{T^{l}_{i,j,k}}=(1+6\lambda)\mathrm{T^{l+1}_{i,j,k}}-\lambda[\mathrm{T^{l+1}_{i+1,j,k}+T^{l+1}_{i-1,j,k}+T^{l+1}_{i,j+1,k}+T^{l+1}_{i,j-1,k}+T^{l+1}_{i,j,k+1}+T^{l+1}_{i,j,k-1}}] \quad (7)$$

$$or, \mathrm{T^{l+1}_{i,j,k}}=\frac{\lambda}{1+6\lambda}\left[\mathrm{T^{l+1}_{i+1,j,k}+T^{l+1}_{i-1,j,k}+T^{l+1}_{i,j+1,k}+T^{l+1}_{i,j-1,k}+T^{l+1}_{i,j,k+1}+T^{l+1}_{i,j,k-1}}\right]+\frac{1}{1+6\lambda}\mathrm{T^{l}_{i,j,k}} \quad (8)$$

Equation (8) can be written in matrix form as follows,

$$\mathrm{T^{l}}=(\mathrm{I}-\lambda\mathrm{A})\mathrm{T^{l+1}} \quad (9)$$

2.1.1.3 Crank-Nicolson algorithm

The Crank Nicolson algorithm is obtained by assigning$\theta=0.5$ in equation (4),

$$\mathrm{T^{l+1}_{i,j,k}-T^{l}_{i,j,k}}=\frac{1}{2}\lambda[\mathrm{T^{l+1}_{i+1,j,k}+T^{l+1}_{i-1,j,k}+T^{l+1}_{i,j+1,k}+T^{l+1}_{i,j-1,k}+T^{l+1}_{i,j,k+1}+T^{l+1}_{i,j,k-1}-6T^{l+1}_{i,j,k}}]$$

$$+\frac{1}{2}\lambda[T^{l}_{i+1,j,k}+T^{l}_{i-1,j,k}+T^{l}_{i,j+1,k}+T^{l}_{i,j-1,k}+T^{l}_{i,j,k+1}+T^{l}_{i,j,k-1}-6T^{l}_{i,j,k}] \quad (10)$$

$$\text{Or, } T^{l+1}_{i,j,k}=\frac{\lambda}{2+6\lambda}\left[T^{l+1}_{i+1,j,k}+T^{l+1}_{i-1,j,k}+T^{l+1}_{i,j+1,k}+T^{l+1}_{i,j-1,k}+T^{l+1}_{i,j,k+1}+T^{l+1}_{i,j,k-1}\right]$$
$$+\frac{\lambda}{2+6\lambda}[T^{l}_{i+1,j,k}+T^{l}_{i-1,j,k}+T^{l}_{i,j+1,k}+T^{l}_{i,j-1,k}+T^{l}_{i,j,k+1}+T^{l}_{i,j,k-1}]+\frac{1-3\lambda}{1+3\lambda}T^{l}_{i,j,k} \quad (11)$$

Equation (11) can be written in a matrix form as follows,

$$(I-\frac{\lambda}{2}A)T^{l+1}=(I+\frac{\lambda}{2}A)T^{l} \quad (12)$$

III. Numerical Treatment

The finite difference systems of equations derived above are solved with iterative methods. Marching of the solution is performed using iterative time stepping with no internal iteration. The solution is performed using Matlab script which has been developed to compute temperature at all nodal points and thus produces a detailed distribution of temperature in all the domain with the required resolution. A sample of the predicted temperature field in the (y-z) plan is shown in Table (2).

Table 2: Sample of the results and errors.

No	x	y	Z	The exact value	Time step 55					
					Forward	error	Backward	error	Crank-Nicolson	Error
1	0.015	0.03333	0.0125	46.7858	46.6652	0.1206	46.6645	0.1213	46.6648	0.121
2	0.015	0.03333	0.0375	46.7678	46.6652	0.1026	46.6646	0.1032	46.6648	0.103
3	0.015	0.03333	0.0625	46.7479	46.6652	0.0827	46.6646	0.0833	46.6649	0.083
4	0.015	0.03333	0.0875	46.7326	46.6652	0.0674	46.6646	0.068	46.6649	0.0677
5	0.015	0.03333	0.1125	46.7171	46.6652	0.0519	46.6646	0.0525	46.6649	0.0522
6	0.015	0.03333	0.1375	46.7013	46.6652	0.0361	46.6646	0.0367	46.6649	0.0364
7	0.015	0.03333	0.1625	46.6881	46.6652	0.0229	46.6646	0.0235	46.6649	0.0232
8	0.015	0.03333	0.1875	46.6757	46.6652	0.0105	46.6646	0.0111	46.6649	0.0108
9	0.015	0.03333	0.2125	46.6632	46.6652	0.002	46.6646	0.0014	46.6649	0.0017
10	0.015	0.03333	0.2375	46.6526	46.6652	0.0126	46.6646	0.012	46.6649	0.0123
11	0.015	0.03333	0.2625	46.6432	46.6652	0.022	46.6646	0.0214	46.6649	0.0217
12	0.015	0.03333	0.2875	46.6338	46.6652	0.0314	46.6646	0.0308	46.6649	0.0311
13	0.015	0.03333	0.3125	46.6259	46.6652	0.0393	46.6646	0.0387	46.6649	0.039
14	0.015	0.03333	0.3375	46.6195	46.6652	0.0457	46.6646	0.0451	46.6649	0.0454
15	0.015	0.03333	0.3625	46.6133	46.6652	0.0519	46.6646	0.0513	46.6649	0.0516
16	0.015	0.03333	0.3875	46.6082	46.6652	0.057	46.6646	0.0564	46.6649	0.0567
17	0.015	0.03333	0.4125	46.6047	46.6652	0.0605	46.6646	0.0599	46.6649	0.0602
18	0.015	0.03333	0.4375	46.6016	46.6652	0.0636	46.6646	0.063	46.6649	0.0633
19	0.015	0.03333	0.4625	46.5993	46.6652	0.0659	46.6646	0.0653	46.6649	0.0656
20	0.015	0.03333	0.4875	46.5987	46.6652	0.0665	46.6646	0.0659	46.6649	0.0662

IV. Consistency analysis

The consistency of the three finite difference algorithms for the current partial differential equation is satisfied with the condition that the limit of the truncation error approaches zero [9,10],

$$\lim_{(h,\Delta t)\to(0,0)} T.E=0$$

where $h=\Delta x=\Delta y=\Delta z$.

$$T.E=[\frac{\lambda^2}{2}-\lambda\theta_1(\frac{1}{12}+\lambda)-\frac{\lambda}{12}(1-\theta_1)]h^4\frac{\partial^4 T}{\partial x^4}+[\frac{\lambda^3}{6}-\lambda\theta_1(\frac{1}{360}+\frac{\lambda}{12}+\frac{\lambda^2}{2})-\frac{\lambda}{360}(1-\theta_1)]h^6\frac{\partial^6 T}{\partial x^6}$$
$$+[\frac{\lambda^2}{2}-\lambda\theta_2(\frac{1}{12}+\lambda)-\frac{\lambda}{12}(1-\theta_2)]h^4\frac{\partial^4 T}{\partial y^4}+[\frac{\lambda^3}{6}-\lambda\theta_2(\frac{1}{360}+\frac{\lambda}{12}+\frac{\lambda^2}{2})-\frac{\lambda}{360}(1-\theta_2)]h^6\frac{\partial^6 T}{\partial y^6}$$
$$+[\frac{\lambda^2}{2}-\lambda\theta_3(\frac{1}{12}+\lambda)-\frac{\lambda}{12}(1-\theta_3)]h^4\frac{\partial^4 T}{\partial z^4}+[\frac{\lambda^3}{6}-\lambda\theta_3(\frac{1}{360}+\frac{\lambda}{12}+\frac{\lambda^2}{2})-\frac{\lambda}{360}(1-\theta_3)]h^6\frac{\partial^6 T}{\partial z^6} \quad (13)$$

4.1. Consistency analysis of the forward time algorithm

The truncation error for the forward-time algorithm can be found from equation (13) by substituting $\theta_1=\theta_2=\theta_3=0$, $\lambda=\frac{\alpha.\Delta t}{h^2}$, and by neglecting the term of the sixth derivative we get,

$$T.E = \alpha.\Delta t[\frac{\alpha.\Delta t}{2} - \frac{h^2}{12}]\frac{\partial^4 T}{\partial x^4} + \alpha.\Delta t[\frac{\alpha.\Delta t}{2} - \frac{h^2}{12}]\frac{\partial^4 T}{\partial y^4} + \alpha.\Delta t[\frac{\alpha.\Delta t}{2} - \frac{h^2}{12}]\frac{\partial^4 T}{\partial z^4} \quad (14)$$

From the equation above it is obvious that the forward algorithm is first-order in time, second-order in space, and is consistent with the heat equation.

4.2. Consistency analysis of the backward time algorithm

The truncation error for the backward-time algorithm can be found from equation (13) by substituting $\theta_1 = \theta_2 = \theta_3 = 1$,$\lambda = \frac{\alpha.\Delta t}{h^2}$, and by neglecting the term of the sixth derivative we get,

$$T.E = -\alpha.\Delta t[\frac{\alpha.\Delta t}{2} + \frac{h^2}{12}]\frac{\partial^4 T}{\partial x^4} - \alpha.\Delta t[\frac{\alpha.\Delta t}{2} + \frac{h^2}{12}]\frac{\partial^4 T}{\partial y^4} - \alpha.\Delta t[\frac{\alpha.\Delta t}{2} + \frac{h^2}{12}]\frac{\partial^4 T}{\partial z^4} \quad (15)$$

From the equation above, it is obvious that the backward algorithm is first-order in time,second-order in space, and is consistent with the heat equation.

4.3. Consistency analysis of the Crank-Nicolson algorithm

The truncation error for the Crank-Nicolson time algorithm can be found from equation (13) by substituting $\theta_1 = \theta_2 = \theta_3 = 0.5, = \frac{\alpha.\Delta t}{h^2}$, and by neglecting the term of the sixth derivative we get,

$$T.E = -\frac{\alpha.\Delta t}{12}[h^2\frac{\partial^4 T}{\partial x^4} + \alpha^2.(\Delta t)^2\frac{\partial^3 T}{\partial t^3}] - \frac{\alpha.\Delta t}{12}[h^2\frac{\partial^4 T}{\partial y^4} + \alpha^2.(\Delta t)^2\frac{\partial^3 T}{\partial t^3}] - \frac{\alpha.\Delta t}{12}[h^2\frac{\partial^4 T}{\partial z^4} + \alpha^2.(\Delta t)^2\frac{\partial^3 T}{\partial t^3}] \quad (16)$$

From the equation above, it is obvious that the Crank-Nicolson algorithm is second-order in both time and space. Also, it is consistent with the heat equation.

V. Convergence

Convergence of the finite difference algorithms with the partial differential equation is reached when limit of the difference between the exact solution U and the numerical solution V approaches to zero [9,10],

$$\lim_{(h,\Delta t)\to(0,0)}(U - V) = 0 \quad (17)$$

VI. Stability

The finite difference algorithm is stable if the resulting errorsdecay from one time step to the next one.To Study the stability of the finite difference algorithm we apply Von-Numman method for numerical stability and we put$T_{i,j,k}^{l} = E^l e^{gh\alpha i} e^{gh\beta j} e^{gh\gamma k}$; $g = \sqrt{-1}$ in the truncation error equation (13). The equation for the magnification coefficient E is derived as follows [8,11],

$$E = \frac{1 - 4\lambda[(1-\theta_1)\sin^2\frac{h\alpha}{2} + (1-\theta_2)\sin^2\frac{h\beta}{2} + (1-\theta_3)\sin^2\frac{h\gamma}{2}]}{1 + 4\lambda[\theta_1\sin^2\frac{h\alpha}{2} + \theta_2\sin^2\frac{h\beta}{2} + \theta_3\sin^2\frac{h\gamma}{2}]} \quad (18)$$

The stability condition is satisfied using $|E| \le 1$,

$$\left|\frac{1 - 4\lambda[(1-\theta_1)\sin^2\frac{h\alpha}{2} + (1-\theta_2)\sin^2\frac{h\beta}{2} + (1-\theta_3)\sin^2\frac{h\gamma}{2}]}{1 + 4\lambda[\theta_1\sin^2\frac{h\alpha}{2} + \theta_2\sin^2\frac{h\beta}{2} + \theta_3\sin^2\frac{h\gamma}{2}]}\right| \le 1 \quad (19)$$

6.1. The stability for the forward-time algorithm

By putting $\theta_1 = \theta_2 = \theta_3 = 0$ into equation above we conclude that the algorithm is conditionally stable,

$$\lambda \le \frac{1}{2[\sin^2\frac{h\alpha}{2} + \sin^2\frac{h\beta}{2} + \sin^2\frac{h\gamma}{2}]} \Rightarrow \lambda \le \frac{1}{6} \quad (20)$$

6.2. The stability for the backward-time algorithm

By putting $\theta_1 = \theta_2 = \theta_3 = 1$ into equation above we conclude that the algorithm is unconditionally stable.

6.3. The stability for the crank-Nicolson algorithm

By putting $\theta_1 = \theta_2 = \theta_3 = 0.5$ into equation above we conclude that the algorithm is unconditionally stable.

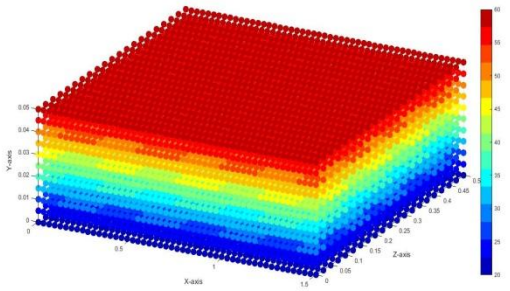
VII. Results

Three-dimensional contours for temperature inside the PV module are for the exact and the three finite difference solutions as shown in fig(1). Figure(2) shows two-dimensional temperature distribution on y-z plan. It is obvious from figures (1) and (2) that predictions of the three numerical algorithms are identical. On the other hand, deviations of these finite-difference computations from the exact solution are not negligible especially at the point's lies between 0.01 to 0.04 in y-direction. Yet, the accuracy is still very good. From the results analysis that has been done, the finite difference algorithm shows a faster convergence to the solution

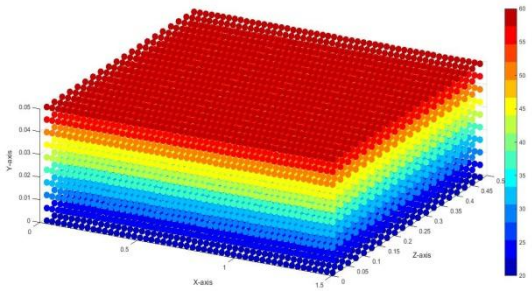
than Crank Nicolson and the backward algorithms. In addition, the forward time algorithm is stable under the condition $\Delta t \leq 55.5$.

VIII. Conclusion

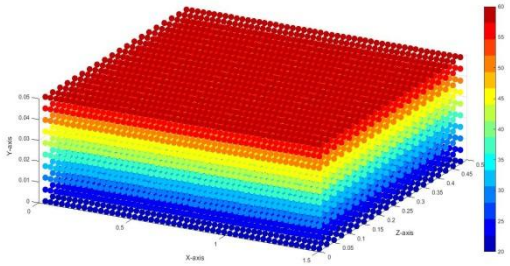
A three-dimensional finite difference model for heat transfer was derived and implemented in MATLAB to predict temperature distributions inside the PV panel SOL-060-01. The forward-time, the backward-time, and Crank-Nicolson algorithms were used to solve temperature equation for the PV panel. The results from numerical solutions were very similar and quite comparable to the exact solution as can be seen in figures (1,2). The accuracy, the consistency, the convergence, and the stability of the solution of the three



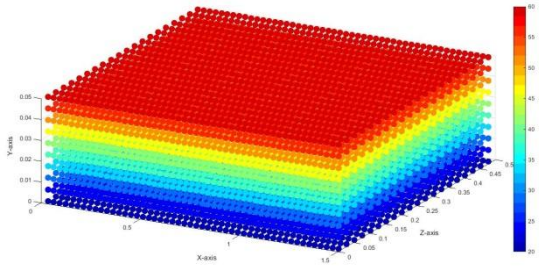
Temperature distribution for the exact solution.



Temperature distribution for the forward algorithm.

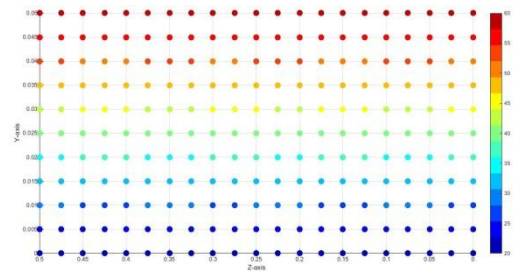


Temperature distribution for the backward algorithm.

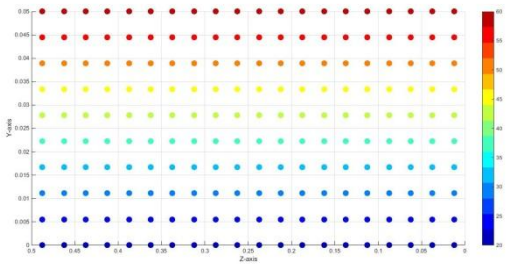


Temperature distribution for the crank-Nicolson.

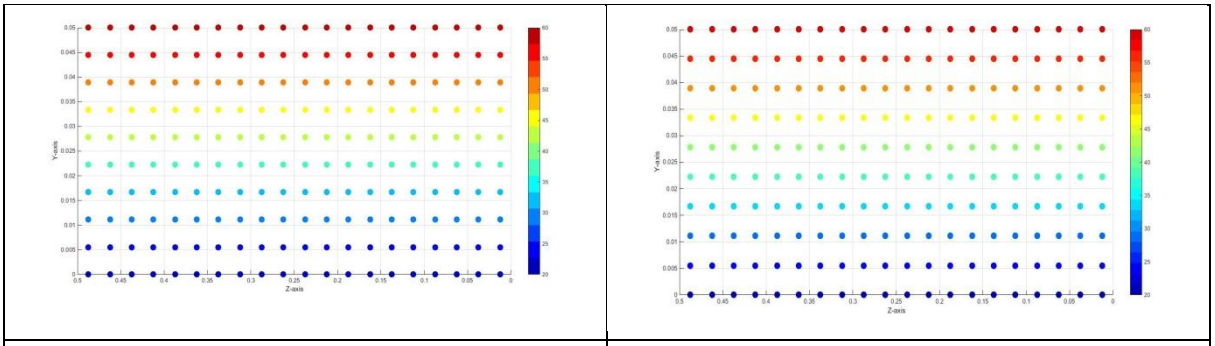
Figure 1: 3-D temperature distribution for the exact solution & the finite difference algorithms.



Temperature distribution for the exact solution.



Temperature distribution for the forward algorithm.



Temperature distribution for the backward algorithm. | Temperature distribution for the crank-Nicolson.

Figure 2: 2-Dtemperature distributions on y-z plane.

algorithms were discussed in this paper. It has been shown that these three finite difference schemes give consistent solution for the partial differential heat equation. The forward-time is shown to be conditionally stable while the backward-time and the crank-Nicolson schemes are unconditionally stable. The properties of these schemes are summarized in Table (3).

Table 3: Numerical comparison between numerical results of the finite difference algorithms.

	NumericalProperty	Forward	Bacward	Crank − Nicolson
1	Algorithm	$\theta = 0$	$\theta = 1$	$\theta = \frac{1}{2}$
2	Truncation Error	$T.E = \alpha.\Delta t[\frac{\alpha.\Delta t}{2} - \frac{h^2}{12}]\frac{\partial^4 T}{\partial x^4} + \alpha.\Delta t[\frac{\alpha.\Delta t}{2} - \frac{h^2}{12}]\frac{\partial^4 T}{\partial y^4} + \alpha.\Delta t[\frac{\alpha.\Delta t}{2} - \frac{h^2}{12}]\frac{\partial^4 T}{\partial z^4}$	$T.E = -\alpha.\Delta t[\frac{\alpha.\Delta t}{2} + \frac{h^2}{12}]\frac{\partial^4 T}{\partial x^4} - \alpha.\Delta t[\frac{\alpha.\Delta t}{2} + \frac{h^2}{12}]\frac{\partial^4 T}{\partial y^4} - \alpha.\Delta t[\frac{\alpha.\Delta t}{2} + \frac{h^2}{12}]\frac{\partial^4 T}{\partial z^4}$	$T.E = -\frac{\alpha.\Delta t}{12}[h^2\frac{\partial^4 T}{\partial x^4} + \alpha^2.(\Delta t)^2\frac{\partial^3 T}{\partial t^3}] - \frac{\alpha.\Delta t}{12}[h^2\frac{\partial^4 T}{\partial y^4} + \alpha^2.(\Delta t)^2\frac{\partial^3 T}{\partial t^3}] - \frac{\alpha.\Delta t}{12}[h^2\frac{\partial^4 T}{\partial z^4} + \alpha^2.(\Delta t)^2\frac{\partial^3 T}{\partial t^3}]$
3	Accuracy	$O(h^2 + \Delta t)$	$O(h^2 + \Delta t)$	$O(h^2 + (\Delta t)^2)$
4	Consistency	Consistent	Consistent	Consistent
5	Convergence	Convergent	Convergent	Convergent
6	Stability	Conditionally stable $\lambda \leq \frac{1}{6}$	Unconditionally stable	Unconditionally stable
7	Time Step size	55	55	55
8	Number of time steps	179	267	223

References

[1]. Fraunhofer Institute for Solar Energy Systems ISE, Photovoltaics Report, 2016.(https://www.ise.fraunhofer.de/en/downloads-englisch/pdf-files-english/photovoltaics-report-slides.pdf).

[2]. W.Dai, R.Nassar, An unconditionally stable finite difference scheme for solving a 3D heat transport equation in a sub-microscale thin film, *Computational and Applied Mathematics* 145 (2002) 247–260.

[3]. G. Notton, C. Cristofari, M. Mattei, P. Poggi, Modelling of a double-glass photovoltaicmodule using finite differences, Appl. Therm. Eng. 25 (2005) 2854–2877,http://dx.doi.org/10.1016/j.applthermaleng.2005.02.008.

[4]. M. Mattei, G. Notton, C. Cristofari, M. Muselli, P. Poggi, Calculation of thePolycrystalline PV Module Temperature Using a Simple Method of Energy Balance,31, 2006, pp. 553–567. http://dx.doi.org/10.1016/j.renene.2005.03.010.

[5]. Two-dimensional finite difference-based model for coupled irradiation andheat transfer in photovoltaic modules, Shahzada Pamir Aly, Said Ahzi, Nicolas Barth, Benjamin W. Figgis

[6]. E.P. Incropera, D.P. Dewitt, *Fundamentals of Heat and Mass Transfer*, Wiley, New York, 1990

[7]. Dean G. Duffy, *Advanced Engineering Mathematics*, CRC press LLC, 1th-1998.

[8]. M. NecatiOzisik, *Heat Conduction*,John wiley& Sons, 2th-1993.

[9]. Daniel R. Lynch, *Numerical Partial Differential Equations for Environmental Scientists and Engineers*, Springer, 1th-2005.

[10]. K. W. Morton and D. F. Mayers, *Numerical Solution of Partial Differential Equations*, Cambridge University Press, 2th-2005.

[11]. PaulDuchateau and David W. Zachmann, *Theory and Problems of Partial Differential Equations*, Mc Graw-Hill,1th-1988.

Abbas Abdelaziz Guma "Consistent Three-Dimensional Finite Difference Modeling of Heat Transferwith non-homogenous boundary conditions in Solar Modules." IOSR Journal of Mathematics (IOSR-JM) 14.2 (2018): 12-17.