ISSN No:-2456-2165

A Time-Dependent Numerical Statistical Solution of the Partial Differential Heat Diffusion Equation

Dr. Ismail Abbas, Senior Lecturer at MTC, Cairo University

Abstract:- The present article is an extension of the previous article based on the statistical transition matrix B which aimed to study the steady-state solution of the IC-BC distribution in the diffusion PDE. We extend here the same theory to find the spatio-temporal evolution of the energy density vector.

Numerical results of the required solution are presented for 2D and 3D configuration space problems where the precision and convergence speed or computation time are superior to the conventional method of separating variables and numerical finite difference techniques.

I. INTRODUCTION

In a previous article [1], we proposed a numerical statistical method for the steady-state solution of heat diffusion, the Laplace and Poisson PDEs based on the transition marix B.

Moreover, we can find an efficient spatio-temporal statistical solution to heat diffusion, the Poisson and Laplace partial differential equations based on the same techniques [1,2]. All these equations describe the spatiotemoral evolution of the energy density distribution of the function U (x, t) through the PDE,

d U / d t) partial = a . Nabla $^{\wedge 2}$ U + S (x, t) . . . .(1)

with boundary conditions of Dirichlet or Neumann BC and initial conditions IC given by U (x, 0).

a is the thermal diffusivity and S (x, t) is the energy density source / sink term.

In the current article, we use Cartesian coordinates, with the Dirichlet boundary conditions while the extension to spherical or cylindrical coordinates and to Neumann BC is possible and straight forward.

Here the unconventional method proposed to solve Equation 1 is based on a rigorous physical statistical assumption assuming the existence of a transition matrix B for a time step dt which can be formulated mathematically as,

$U_{i,j,k}{}^{(N+1)}$ = B. (b + S) + $U_{i,j,k}{}^{(N)}$ . . . (2)

b is the vector of the boundary conditions BC arranged in the proper order.

U^{N} (x, t) is the spatiotemporal solution of the situation described by heat diffusion equation or Poisson PDE (1). N represents the number of steps dt or number of iterations N.

dt is implicitly or inherently included in matrix multiplication.

Equation 2 is a recurrence formula which leads for a transfer matrix E and D for N consecutive time steps .

Hence, we define the transfer matrix E as the power series of matrix B,

E (N)= B ^ 0 + B + B ^ 2 +. . . . . + B ^ N . . . . (3)
Where B ^ 0 = I

At the limit where N is large enough, the relation,

E infinite= $(I\text{-}B)^{-1}$.. . . . . . . (4) holds

And the transfer matrix D is similarly defined by the power series of matrix B,

D(N) = B + B ^ 2 +. . . . . + B ^ N . . . . (5)
Obviously,
D = E-I. . . . . . . (6)
And it is also evident that for N consecutive time steps dt, the time-dependent solution of the heat diffusion equation 1 is given by,
$U^{N}_{i,j,k}$ = D (N) (b + S). . . . . (7)

For a time interval t = Ndt.

Equation 7 is the numerical statistical replacement of the PDE 1 based on successive application of B-transition matrix Equation 2.

However, Equation (6) is appropriate to study the steady-state solution as used in article [1] while in the current article we deal with the case of the small N applied to a time-dependent transient solution.

The initial conditions IC are assumed to be zero in the current analysis of equations 2 and 7, but they can be simply implemented in equation 7 by adding the term B ^ N .U (x, 0) which tends to conform to BC for a large N as described in reference [2]. Initial conditions are simply avoided in this article so as not to overload the analysis.

II. THEORY

The main assumption of the numerical statistical solution Eq 2 is that there exists a physical statistical transition matrix B such that equation 2 is satisfied for each jump or particular time step dt.

It follows that the time-dependent solution of the heat diffusion Eq 1 at time t = Ndt will be given by,

U(N) = D(N). (b + S). . . . .(7)

Therefore, the starting point or key to the numerical statistical solution is to find the base transition matrix B.

However, the steps of the resolution procedure are explained and matrix B is well defined through conditions i-iv below,

We have to explain the whole procedure of the proposed numerical statistical method which should be done in 3 consecutive precise steps and will be more explained further by 2 illustrative applications in 2D and 3D configuration space.

*First Step

Discretize the 2D or 3D domain in n equally spaced free nodes and find the appropriate stochastic transition matrix B(nxn) satisfying the statistical physical conditions i-iv below, and therefore the hypothesis of equation 2 applies.

The statistical transition matrix B = ($B_{i, j}$) itself is well defined by statistical assumptions i-iv.

For 2D and 3D Cartesian coordinates, the inputs B $_{i, j}$ satisfy or are subject to the following conditions:

This means that the statistical transition matrix B = (Bi, j) itself is well defined through the statistical assumptions i-iv.

For 2D and 3D Cartesian coordinates, the inputs B i, j satisfy or are subject to the following conditions:

i- B $_{i, j}$ = 1/4 for 2D or 1/6 for 3D for i adjacent to j .. and B i, j = 0 otherwise.

This means that there is no substantially preferred spatial direction.

ii- B i, i = RO, i.e. the main diagonal consists of constant inputs RO

The transition is a collective identical process.

RO can take any value in the closed interval [0,1] and plays a crucial role in the heat diffusion equation.

We show that the thermal diffusivity a is proportional to 1-RO . Ultimately, Ro = 0 corresponds to super conduction and RO = 1 corresponds to the case of insulators while for Laplace and Poisson PDE, RO = 0

That is to say that by solving Poison and Laplace PDE B is a zero principal diagonal matrix which corresponds to the assumption of a zero residue or of no energy storage after each time step for all the free nodes n.

iii- B $_{i, j}$ = B $_{j, i}$ for all i, j.

The matrix B is symmetrical to conform to the physical principle of detailed equilibrium.

iv- The sum of B $_{i, j}$ = 1 for all rows far from the borders and the sum B $_{i, j}$ <1 for all rows connected to the borders to allow the contribution BC to the energy of the system as an energy density source / sink term. The meaning of condition iv is that the probability of whole space = 1.

Obviously, the statistical matrix B is very different from the Laplacian mathematical matrix A described in Ref [4,6] and from the Markov transition matrix [3,4].

The physical nature of matrix B is clear and briefly explained above through conditions i to iv which support Hypothesis 2.

**Second Step

Define b vector which is the vector of the boundary conditions by arranging BC in the correct order.

Calculate the source / sink term vector in energy density J / m ^ 3 rather than the temperature in degrees Kelvin (for the case of the heat diffusion equation) or the voltage for the Poisson and Laplace PDEs.

***Third Step

Compute the transfer matrix E and D by equations 3 and 5 and therefore find the solution as a function of time of the heat diffusion equation from the formula,

U (N) = D(N). (b + S)…. . . . (7)

where N = 1,2, ... N. That is to say the solution U(x,t) at iteration N or at time = N dt is given by Equation 7.

Notice that for N sufficiently large,

E infinite=$(I\text{-}B)^{-1}$ which is the transfer matrix for the steady state equilibrium solution as N tends towards infinity.

Obviously, all the inputs of the term matrix B ^ N converge to zero when N tends to infinity, since the sum of one or more rows is less than unity, which is a necessary condition for the convergence of the matrices E and D.

Actually, it is not complicated to calculate the matrix E or D. The finite series (3& 5) can be evaluated in a simple separate calculation algorithm by computing the matrix multiplication power series and adding for the required number N. Double precision algorithms are a must in such calculations [5]

The simplicity and the precision of the proposed numerical statistical method are quite surprising, it is enough first to calculate the matrix B and the BC vector b corresponding to the geometry of the energy field and the configuration of BC of the problem then to calculate the matrix D and E by the sum of the power series of B Eq. (3 & 5) or the use of equation (4) in the case where N is large enough.

However, so as not to worry too much about the details of the theory, let's go right into 2D and 3D illustrative applications.

III. APPLICATIONS

A.-2D CONFIGURATION SPACE

Consider the simple case of a rectangular domain with 9 equidistant free nodes, u1, u2, u3, ... u9 and 12 Dirichlet boundary conditions BC1 to BC12 as illustrated in figure 1.

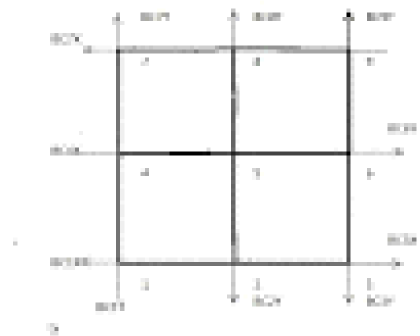

Fig.1 A 2D rectangular domain with 9 equidistant free nodes.

The 12 boundary conditions in figure 1 can be reduced to 9 BC for the 9 free nodes as follows,
BC1 = BC1X + BC1Y
BC2 = BC2X + BC2Y
. . . .
BC9 = BC9X + BC9Y

The inputs of matrix $B_{i,\ j}$ 9x9 are constructed according to figure 1 and the statistical base described by conditions i-iv, and are given by,

B = 1-RO 1/4-RO / 4 0.0000 1/4-RO / 4 0.0000 0.0000 0.0000 0.0000 0.0000
2- 1/4-RO/4 RO 1/4-RO/4 0.0000 1/4-RO/4 .0000 0.0000 0.0000 0.0000
3- 0.00000 1/4-RO/4 RO 0.0000 0.0000 1/4-RO/4 0.0000 0.0000 0.0000
4- 1/4-RO/4 0.0000 0.0000 RO 1/4-RO/4 0.0000 1/4RO/4 0.0000 0.0000

5- 0.0000 1/4-RO/4 0.0000 1/4-RO/4 RO 1/4-RO/4 0.0000 1/4-RO/4 0.0000

6-0.0000 0.0000 1/4-RO/4 0.0000 1/4-RO/4 RO 0.0000 0.0000 1/4-RO/4

7-0.000 0.0000 0.0000 1/4-RO/4 0.0000 0.0000 RO 1/4RO/4 0.0000

8- 0.0000 0.0000 0.0000 0.0000 1/4-RO/4 0.0000 0.1500 RO 1/4-RO/4

9- 0.0000 0.0000 0.0000 0.0000 0.0000 1/4-RO/4 0.0000 1/4-RO/4 RO

If the 2D rectangle in figure 1 is placed inside a larger rectangle of uniform temperature of 100 degrees for all 4 sides. then the boundary conditions vector b for the 9 free nodes is given by,
b = (50,25,50,25,0,25,50,25,50). . . . (8)

We solve the heat diffusion equation of Figure 1 with a zero source term, S = 0, for two cases (a) and (b).
Case (a) RO=0 :

Using equation 7 and placing RO = 0 in matrix B, and considering the vector b given by equation 8, the time evolution of the resulting temperature T as a function of time is shown in Table I.

Table I, spatiotemporal evolution of temperature for 9 free nodes in figure 1.(t = n dt time step)

(1dt)	50.0000000	25.0000000	50.0000000	25.0000000	0.00000000	25.0000000	50.0000000	25.0000000	50.0000000
(2dt)	62.5000000	50.0000000	62.5000000	50.0000000	25.0000000	50.0000000	62.5000000	50.0000000	62.5000000
(3dt)	75.0000000	62.5000000	75.0000000	62.5000000	50.0000000	62.5000000	75.0000000	62.5000000	75.0000000
(4dt)	81.2500000	75.0000000	81.2500000	75.0000000	62.5000000	75.0000000	81.2500000	75.0000000	81.2500000
(5dt)	87.5000000	81.2500000	87.5000000	81.2500000	75.0000000	81.2500000	87.5000000	81.2500000	87.5000000
(6dt)	90.6250000	87.5000000	90.6250000	87.5000000	81.2500000	87.5000000	90.6250000	87.5000000	90.6250000
(7dt)	93.7500000	90.6250000	93.7500000	90.6250000	87.5000000	90.6250000	93.7500000	90.6250000	93.7500000
(8dt)	95.3125000	93.7500000	95.3125000	93.7500000	90.6250000	93.7500000	95.3125000	93.7500000	95.3125000

To elucidate the efficiency of the new computational technique, we apply it to Table I.
We explain how to get U (x, t) in step 5 from step 4 using the transition matrix B and the boundary conditions b as follows,
Vector U^{k+1}, = B $^{\wedge k}$ (b+S)+ (U^k) .... (9)

Equation 9 calculates the 9 spatio-temporal operations of the 4dt vector in a single operation, for example to calculate the 5dt vector from 4dt that we apply,

Vector 5dt = Vector 4dt + B^4 .b
Since,

U(4dt) = [81.25, 75, 81.25, 75, 62.5, 75, 81.25, 75, 81.25] ^ T.
And matrix B^4 (9x9)=

1-	3.9062500000000000E-2	0.0000000000000000	3.1250000000000000E-2	0.0000000000000000	6.2500000000000000E-2	0.0000000000000000	3.1250000000000000E-2	0.0000000000000000	2.3437500000000000E-2
2-	0.0000000000000000	7.0312500000000000E-2	0.0000000000000000	6.2500000000000000E-2	0.0000000000000000	6.2500000000000000E-2	0.0000000000000000	5.4687500000000000E-2	0.0000000000000000
3-	3.1250000000000000E-2	0.0000000000000000	3.9062500000000000E-2	0.0000000000000000	6.2500000000000000E-2	0.0000000000000000	2.3437500000000000E-2	0.0000000000000000	3.1250000000000000E-2
4-	0.0000000000000000	6.2500000000000000E-2	0.0000000000000000	7.0312500000000000E-2	0.0000000000000000	5.4687500000000000E-2	0.0000000000000000	6.2500000000000000E-2	0.0000000000000000
5-	6.2500000000000000E-2	0.0000000000000000	6.2500000000000000E-2	0.0000000000000000	0.12500000000000000	0.0000000000000000	6.2500000000000000E-2	0.0000000000000000	6.2500000000000000E-2
6-	0.0000000000000000	6.2500000000000000E-2	0.0000000000000000	5.4687500000000000E-2	0.0000000000000000	7.0312500000000000E-2	0.0000000000000000	6.2500000000000000E-2	0.0000000000000000
7-	3.1250000000000000E-2	0.0000000000000000	2.3437500000000000E-2	0.0000000000000000	6.2500000000000000E-2	0.0000000000000000	3.9062500000000000E-2	0.0000000000000000	3.1250000000000000E-002
8-	0.0000000000000000	5.4687500000000000E-2	0.0000000000000000	6.2500000000000000E-2	0.0000000000000000	6.2500000000000000E-2	0.0000000000000000	7.0312500000000000E-2	0.0000000000000000
9 -	2.3437500000000000E-2	0.0000000000000000	3.1250000000000000E-2	0.0000000000000000	6.2500000000000000E-2	0.0000000000000000	3.1250000000000000E-2	0.0000000000000000	3.9062500000000000E-2

And boundary Conditions b vector=
(50,25,50,25,0,25,50,25,50) T

The formula 9 precisely computes vector 5dt, ie.It results in,
U (5dt)= (87,500 81,2500 87,500 81,2500 75,000 81,2500 87,500 81,2500 87,500) ^ T
Which equals U(6dt).

On the other hand, the classical method, finite difference techniques, evaluates $Nabla^2U^k_{i,j}$ individually 9 times for the 9 free nodes and applies,
$U^{k+1}_{i,j}= U^k_{i,j} + a .Nabla^2U^k_{i,j} .dt$. . . . .(10)
instead of equation 9.

where $Nabla^2U^k_{i,j} =(U^k_{I+1,j}+U^k_{i-1,j}+U^k_{i,J+1}+U^k_{i,j-1} -4 U^k_{i,j})/ h^2$
for 2D, 3 point finite difference scheme.

Hence, In order to solve the heat PDE 1 by the FDM, we have to evaluate the equation (10) point by point 9 times for the 9 free nodes while in the statistical transition matrix B, the equation 9 replaces the 9 operations with a single operation.

It should be mentioned that the above example clearly explains how the new method works and why it is much faster and more accurate than conventional finite difference techniques.

Case b:
Similar to case (a) but with RO = 0.2
Here, the change in temperature T as a function of space and time is calculated using Equation 7 and is shown in Table II.

Table II, spatio-temporal evolution of the temperature T on 9 free nodes of figure 1, (t = n dt time step)

(1dt)	40.0000000	20.0000000	40.0000000	20.0000000	0.00000000	20.0000000	40.0000000	20.0000000	40.0000000
(2dt)	56.0000000	40.0000000	56.0000000	40.0000000	16.0000000	40.0000000	56.0000000	40.0000000	56.0000000
(3dt)	67.1999969	53.5999985	67.1999969	53.5999985	35.2000008	53.5999985	67.1999969	53.6000023	67.2000046
(4dt)	74.8800049	64.6400070	74.8800049	64.6400070	49.9199982	64.6399994	74.8799973	64.6399994	74.8800049
(5dt)	80.8320084	72.8639984	80.8320007	72.8639984	61.6959991	72.8639984	80.8320084	72.8639984	80.8320007
(6dt)	85.3119965	79.2448044	85.3119965	79.2448044	70.6304016	79.2448044	85.3119965	79.2448044	85.3120041
(7dt)	88.7603149	84.0998306	88.7603226	84.0998383	77.5219269	84.0998383	88.7603226	84.0998383	88.7603226
(8dt)	91.3920059	87.8284912	91.3920059	87.8284836	82.7842636	87.8284836	91.3920059	87.8284836	91.3919983

Similar to case (a), we compare the results obtained by Equation 9 to the classical conventional method .In the classical conventional FDM . The change for a single time step dt in the solution is classically evaluated as,

$U^{k+1}_{i,j}=U^k_{i,j}+a.Nabla^2 U^k_{i,j} . dt$ . . . . .(10)

where $Nabla^2 U^k_{i,j} =(U^k_{I+1,j}+U^k_{I-1,j}+U^k_{i,J+1}+U^k_{i,j-1} -4 U^k_{i,j})/4$ for 2D scheme.

Once again, equation (10) is evaluated point by point for the 9 free nodes while in the statistical matrix B equation 9 works in one operation,

Vector U^{k+1}=B^k (b+S) +U^k .... (9)

Similar to case a, Eq 9 calculates the 9 operations of Eq 10 in a single multiplication operation.

For example, calculate Vector 6dt from 5dt (k=5) that we apply,
Vector 6dt=Vector 5dt +B^5 .b/ (1-RO) . . . .(11)

ISSN No:-2456-2165

Matrix B^5 for RO=0.2 is given by:

1- 2.2720001692771961E-002 2.8480002121925418E-002 1.6000001192092932E-002 2.8480002121925415E-002 3.2000002384185863E-002 1.9200001430511517E-002 1.6000001192092932E-002 1.9200001430511517E-002 9.6000007152557583E-003
2- 2.8480002121925412E-002 3.8720002884864886E-002 2.8480002121925412E-002 3.2000002384185863E-002 4.7680003552436928E-002 3.2000002384185863E-002 1.9200001430511517E-002 2.5600001907348690E-002 1.9200001430511517E-002
3- 1.6000001192092932E-002 2.8480002121925418E-002 2.2720001692771961E-002 1.9200001430511517E-002 3.2000002384185856E-002 2.8480002121925415E-002 9.6000007152557583E-003 1.9200001430511517E-002 1.6000001192092932E-002
4- 2.8480002121925415E-002 3.2000002384185863E-002 1.9200001430511517E-002 3.8720002884864893E-002 4.7680003552436928E-002 2.5600001907348690E-002 2.8480002121925412E-002 3.2000002384185863E-002 1.9200001430511517E-002
5- 3.2000002384185856E-002 4.7680003552436928E-002 3.2000002384185856E-002 4.7680003552436928E-002 6.4320004792213559E-002 4.7680003552436928E-002 3.2000002384185856E-002 4.7680003552436928E-002 3.2000002384185856E-002
6- 1.9200001430511513E-002 3.2000002384185856E-002 2.8480002121925415E-002 2.5600001907348686E-002 4.7680003552436928E-002 3.8720002884864886E-002 1.9200001430511517E-002 3.2000002384185856E-002 2.8480002121925415E-002
7- 1.6000001192092932E-002 1.9200001430511517E-002 9.6000007152557583E-003 2.8480002121925418E-002 3.2000002384185863E-002 1.9200001430511517E-002 2.2720001692771961E-002 2.8480002121925415E-002 1.6000001192092932E-002
8- 1.9200001430511517E-002 2.5600001907348690E-002 1.9200001430511517E-002 3.2000002384185863E-002 4.7680003552436928E-002 3.2000002384185863E-002 2.8480002121925415E-002 3.8720002884864893E-002 2.8480002121925415E-002
9- 9.6000007152557583E-003 1.9200001430511517E-002 1.6000001192092932E-002 1.9200001430511517E-002 3.2000002384185863E-002 2.8480002121925412E-002 1.6000001192092932E-002 2.8480002121925412E-002 2.2720001692771961E-002

And the computation for Vector U(6dt) goes as:
Vector (6dt)=Vector (5dt) +B^5 .b .(1-RO) . . .(11)

Note that 1-RO = 0.8 which is proportional to the thermal diffusivity a.
where,
Vector,u(5dt)=[81.25,75,81.25,75,62.5,75,81.25,75,81.25]^T.

Therefore, calculations using matrix B of equation 11 yield the vector 6dt, i.e.
U(6dt)=(85.312 79.2448 85.312 79.24480 70.630 79.2448 85.312 79.2448 85.312) T

B. 3D CONFIGURATION SPACE, 27 FREE NODES

Let us now consider the more complicated case of a Rectanguloid domain with 27 equidistant free nodes, u1, u2, u3, ... u27 and 52 Dirichlet boundary conditions shown in Fig.2.

ISSN No:-2456-2165

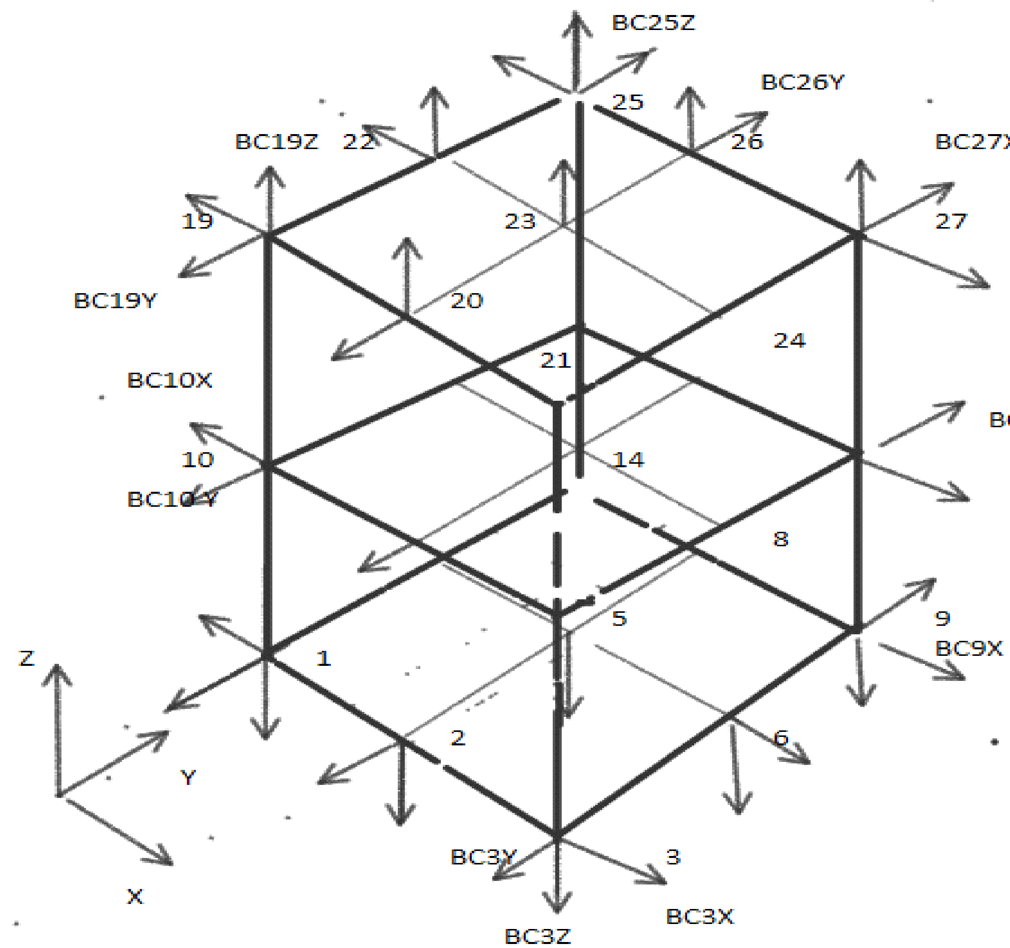

Fig. 2. 3 D Rectanguloid with 27 equidistant free nodes and 52 BC

In the 3 D with 27 free nodes of figure 2, the conditions of dirichlet 52 BC can be reduced to 27 BC through the relation,
BC1=BC1X+BC1Y +BC1Z
BC2=BC2X+BC2Y +BC2Z
. . . .

BC27=BC27X+BC27Y +BC27Z
If we place the rectangle inside a greater of uniform temperature T = 100 units on its six faces, the vector of boundary conditions vector b of the 27 elements is given by,

b = (50.0000000 33.3333321 50.0000000 33.3333321 16.6666660 33.3333321 50.0000000 33.3333321 50.0000000 33.3333321 16.6666660 33.3333321 16.6666660 0.00000000 16.6666660 33.3333321 16.6666660 33.3333321 50.0000000 33.3333321 50.0000000 33.3333321 16.6666660 33.3333321 50.0000000 33.3333321 50.0000000)T

The spatio-temporal evolution of the solution vector (U1, U2, ..., U27) for two consecutive time steps 4dt and 5dt for RO = 0 is presented in table III.

ISSN No:-2456-2165

Table III

U(x,t) vector for t=4dt ,RO=0

92.181071938329723	89.060358466048939	92.181071938329708	89.060358466048939	84.379288602410753
89.060358466048953	92.181071938329708	89.060358466048953	92.181071938329723	89.060358466048939
84.379288602410739	89.060358466048982	84.379288602410739	78.137864086240228	84.379288602410739
89.060358466048967	84.379288602410739	89.060358466048967	92.181071938329737	89.060358466048939
92.181071938329723	89.060358466048967	84.379288602410739	89.060358466048939	92.181071938329723
89.060358466048953	92.181071938329723			

U(x,t) vector for t=5dt ,RO=0

94.530180692096479	92.186787771716723	94.530180692096479	92.186787771716737	89.063217075925607
92.186787771716737	94.530180692096508	92.186787771716752	94.530180692096479	92.186787771716723
89.063217075925607	92.186787771716737	89.063217075925607	84.379291210113848	89.063217075925607
92.186787771716752	89.063217075925607	92.186787771716723	94.530180692096494	92.186787771716723
94.530180692096479	92.186787771716723	89.063217075925607	92.186787771716723	94.530180692096479
92.186787771716737	94.530180692096479			

Again, note that using double precision algorithm to find the B-Matrix solution is a must in such cases [5].

Similar to cases a and b, the solution vector for U (x, t) at t = 5dt is obtained from the one-operation formula 9 with k = 4,

U (x, 5dt) = B ^ 4 .b + U (x, 4dt)

with precise results.

This one-step procedure replaces the classical FDM finite difference method of computing Nabla ^ 2 dt 27 times for the 27 free nodes for each time step dt, and adding the results to the vector solution at kdt.

The proposed technique performs the function of a high speed computer but with a software improvement rather than a hardware improvement.

IV . B-Transition matrix in 4D space

The resolution procedure shows that the matrix B in the IC-BC problem works in a collective statistical behavior to find all the temporal transitions of the solution vector in a real connected 4D space x, t rather than 3D +tinclassical treatment[4] U(x,t+dt)= U(x,t)+a Nabla^2 x .dt

However, in the proposed method, the matrix B ^ N is applied to the vector of boundary conditions (b) and produces an inherent operator Nabla ^ 2 acting across the boundaries of the system and does without Nabla ^ 2 itself.

V. CONCLUSION

The theory and the numerical results show that the complete spatiotemporal numerical solution of the boundary value problem in the partial differential equation of heat diffusion,Laplace and Poisson comes from a SINGLE statistical transition matrix B.

We present the B, E and D transfer matrix and explain their remarkable physical and statistical properties as well as the principles underlying their derivation.

The numerical results of the temporal evolution of the thermal energy density, or voltage, are provided in two cases: 2D and 3D configurations showing the stability, precision and speed of the proposed non-classical transition matrix method.

The numerical results validate the fact that the thermal diffusion phenomenon is an x-t transition phenomenon where the separation of the variables x, t works collectively with the boundary conditions of the system.

REFERENCES

[1]. Abbas,IJISRT,Nov.2020, A numerical statistical solution for Laplace and Poisson partial differential equations.

[2]. I.Abbas,Researchgate,Aug 2020,In the numerical solution of the heat equation, how to simultaneously solve the boundary conditions and the initial conditions.

[3]. I.Abbas,IJISRT,Dec 2020. HOW to transform markov matrix into B-Matrix and vice versa.

[4]. John Mathews,Numerical Methods,Printice-hall inc,1992

[5]. I.abbas etal, IEEE, July 1996, Pseudospark simulation.

[6]. Mona Rahmani ,January 2019, Numerical methods for solving the heat equation, the wave equation and Laplace s equation (Finite difference methods)