SCHOOL OF ELECTRICAL ENGINEERING

UNIVERSITI TEKNOLOGI MALAYSIA

SSCE 2393 NUMERICAL METHODS

2020-2021/1

ASSIGNMENT REPORT

SECTION: 36

GROUP: 8

Lecturer: Dr Hang See Pheng

NAME	MATRIC NO.
KENNETH LO JIN CHAN	A18KE0083
HA HOW UNG	A18KE0064
OOI CHUN WEI	A18KE0249
SOYEB AL SAAD	A18KE0326

INTRODUCTION

Python is a general-purpose programming language supporting object-oriented and structured programming. It is free, simple to use and implement, and well structured, and equally useful for non-numerical as for numerical applications. For these reasons it is sometimes chosen as a language for a first programming course. It is clear that in such a case, a first course on numerical methods prefers using Python as a tool for implementing and testing the algorithms over an alternative such as MATLAB, or *Scilab*, the open-source MATLAB clone.

In this group assignment to solve the given PDE problem we have decided to use the Crank-Nicolson Implicit method. There are a few reasons for this. To begin with, the explicit method of solving PDE might be easier to solve but it has very low accuracy with respect to time. However, the implicit methods overcome problems associated with explicit methods at the expense of somewhat more complicated algorithms. To add on, in implicit methods the spatial derivative is approximated at an advanced time interval. For the question in this assignment for the reasons given above we have deemed it better to use the Crank-Nicolson method over other methods.

Therefore, in our group assignment, we will be using Python to solve the given question using the Crank-Nicolson method.

The Crank Nicolson Implicit Method

In the Crank-Nicolson method, the heat equation

$$c\frac{\partial^2 u}{\partial x^2} = \frac{\partial u}{\partial t} \text{ or } \frac{\partial^2 u(x,t)}{\partial x^2} = \frac{\partial u(x,t)}{\partial t}, \qquad \text{at } \left(x_i, t_{j+\frac{1}{2}}\right)$$

$$c\frac{\partial^2 u\left(x_i, t_{j+\frac{1}{2}}\right)}{\partial x^2} = \frac{\partial u\left(x_i, t_{j+\frac{1}{2}}\right)}{\partial t}$$

$$c\frac{\partial^2 u_{i,j+\frac{1}{2}}}{\partial x^2} = \frac{\partial u_{i,j+\frac{1}{2}}}{\partial t}$$

is approximated (see the Figure 1) by

$$\frac{c}{2}\left(\frac{u_{i+1,j+1} - 2u_{i,j+1} + u_{i-1,j+1}}{h^2} + \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2}\right) = \frac{u_{i,j+1} - u_{i,j}}{2\left(\frac{k}{2}\right)}$$

$$\frac{ck}{h^2}\left(u_{i+1,j+1} - 2u_{i,j+1} + u_{i-1,j+1} + u_{i+1,j} - 2u_{i,j} + u_{i-1,j}\right) = 2(u_{i,j+1} - u_{i,j})$$

and after simplification this equation gives

$$-ru_{i-1,j+1} + (2+2r)u_{i,j+1} - ru_{i+1,j+1} = ru_{i-1,j} + (2+2r)u_{i,j} + ru_{i+1,j}$$

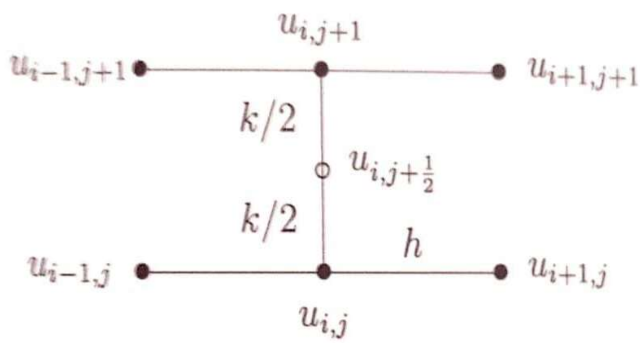


Figure A: Finite-Difference Grid for Crank-Nicolson Method

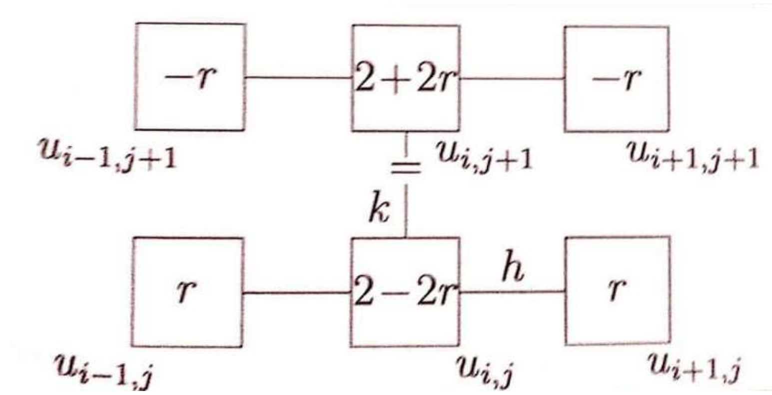


Figure B: Computational Molecule for Crank-Nicolson Method

where $r = \frac{ck}{h^2}$, known as the Crank-Nicolson implicit formula. The computational molecule of Crank-Nicolson is shown in Figure 2 and Figure 3 for r=1.

In general, the left side for Crank-Nicolson implicit formula contains three unknowns and the right-hand side has three known values of u. For j = 0 and I = 1, 2, ..., n-1 the formula generates (n-1) simultaneous equation in (n-1) unknown $\{u_{1,1}, u_{2,1}, \dots, u_{n-1,1}\}$ (at first time step) in terms of known boundary value $\{u_{0,1}, u_{n,1}\}$ and initial value $\{u_{1,0}, u_{2,0}, \dots, u_{n-1,0}\}$.

Similarly, for j =1 and I =1, 2, …, n-1 we obtain another set of unknown values $\{u_{1,2}, u_{2,2}, \dots, u_{n-1,2}\}$ (at second time step) in term of calculated value for j = 0 and known boundary value $\{u_{0,2}, u_{n,2}\}$, and so on.

In this method, the value of $u_{i,j+1}$ is can't be calculated directly from of known $u_{i,j}$ of previous time steps, and hence the method is called an implicit method.

The matrix form of the Crank Nicolson method is

$$Au^{(j+1)} = b^{(j)}, j = 0, 1, 2, \dots$$

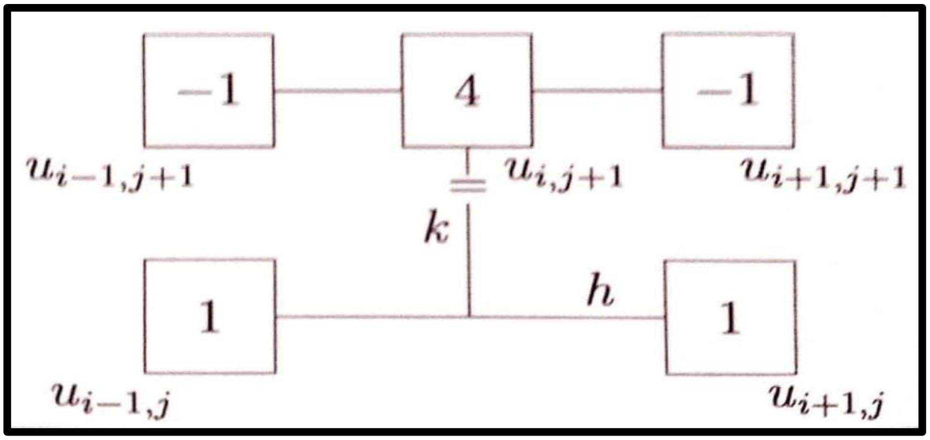


Figure C: Computational Molecule for Crank-Nicolson Method, r =1

Where

$$u^{(j+1)} = \left(u_{1,(j+1)}, u_{2,(j+1)}, \dots, u_{n-1,(j+1)}\right)^T,$$

$$A = \begin{pmatrix} 2(1+r) & -r & 0 & \cdots & 0 \\ -r & 2(1+r) & -r & \ddots & \vdots \\ 0 & -r & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -r \\ 0 & \dots & 0 & -r & 2(1+r) \end{pmatrix}$$

and

$$\mathbf{b} = \begin{pmatrix} 2(1-r)u_{1,j} + ru_{2,j} + r(u_{0,j} + u_{0,j+1}) \\ ru_{1,j} + 2(1-r)u_{2,j} + ru_{3,j} \\ \vdots \\ ru_{n-3,j} + 2(1-r)u_{n-2,j} + ru_{n-1,j} \\ ru_{n-2,j} + 2(1-r)u_{n-1,j} + r(u_{n,j} + u_{n,j+1}) \end{pmatrix}.$$

The resulting tridiagonal system can be solved using Thomas algorithm, Gauss-Seidel method or any other method which could be used to solve the matrix.

PROBLEM STATEMENT

When we try to solve with the Partial Differential Equation given and compared with the analytical solution given in Question 2, we try to substitute the analytical solution $u(x,t) = e^{-t} cos\ (\pi \left(x - \frac{1}{2}\right))$ into the PDE

$$\frac{1}{\pi^2}\frac{\partial^2 u}{\partial x^2} = \frac{\partial^2 u}{\partial t^2}$$

LHS:

$$\frac{\partial^2}{\partial x^2}\left(e^{-t}\cos\left(\pi\left(x-\frac{1}{2}\right)\right)\right) = -\pi^2 e^{-t}\cos\left(\pi\left(x-\frac{1}{2}\right)\right)$$

$$\frac{1}{\pi^2}\frac{\partial^2}{\partial x^2}\left(e^{-t}\cos\left(\pi\left(x-\frac{1}{2}\right)\right)\right)$$

$$= \frac{1}{\pi^2}\left(-\pi^2 e^{-t}\cos\left(\pi\left(x-\frac{1}{2}\right)\right)\right)$$

$$= -e^{-t}\cos\left(\pi\left(x-\frac{1}{2}\right)\right)$$

RHS:

$$\frac{\partial^2}{\partial t^2}\left(e^{-t}\cos\left(\pi\left(x-\frac{1}{2}\right)\right)\right)$$

$$\frac{\partial}{\partial t}\left(e^{-t}\cos\left(\pi\left(x-\frac{1}{2}\right)\right)\right)$$

$$= -e^{-t}\cos\left(\pi\left(x-\frac{1}{2}\right)\right)$$

$$\frac{\partial}{\partial t}\left(-e^{-t}\cos\left(\pi\left(x-\frac{1}{2}\right)\right)\right)$$

$$= e^{-t}\cos\left(\pi\left(x-\frac{1}{2}\right)\right)$$

It can be seen that both sides of the equation are not equal when we try to substitute the analytical solution into the PDE. If the RHS only has the first derivative, both sides of the equation will be equal.

So instead, we proposed making some changes to the RHS of the equation to make it into a heat equation so that this question is more solvable.

$$\frac{1}{\pi^2}\frac{\partial^2 u}{\partial x^2} = \frac{\partial u}{\partial t}$$

CODING

The coding of Python files is uploaded in a separate zip file.

RESULT AND DISCUSSION

In this section, we will be including the results from explicit method. This is to compare them side by side in explaining the pros and cons of each method used. The formula below shows how the average error is calculated.

$$Error_{avg} = \frac{\sum \left| u_{calculated}(x,t) - u_{analytical}(x,t) \right|}{\left(\frac{2}{h}+1\right)\left(\frac{2}{k}+1\right)}$$

where

$\frac{2}{h} + 1 =$ number of x values

$\frac{2}{k} + 1 =$ number of t values

The calculations are conducted inside the Python program, and the result is displayed in the console.

**Note: The grid plot, CSV files and surface plots are generated from our Python coding. The x-axis and t-axis are labelled afterwards by ourselves.

```
Calculation result has been exported as result_Q1a_calculated_CrankNic.csv into the current directory.

Analytical solution has been exported as result_Q1a_analytical_CrankNic.csv into the current directory.
```

The CSV files are generated by our Python coding.

The errors calculated are displayed during the presentation of Question 1. Therefore, Question 2 has also been answered in the meantime.

Question 1 (a)

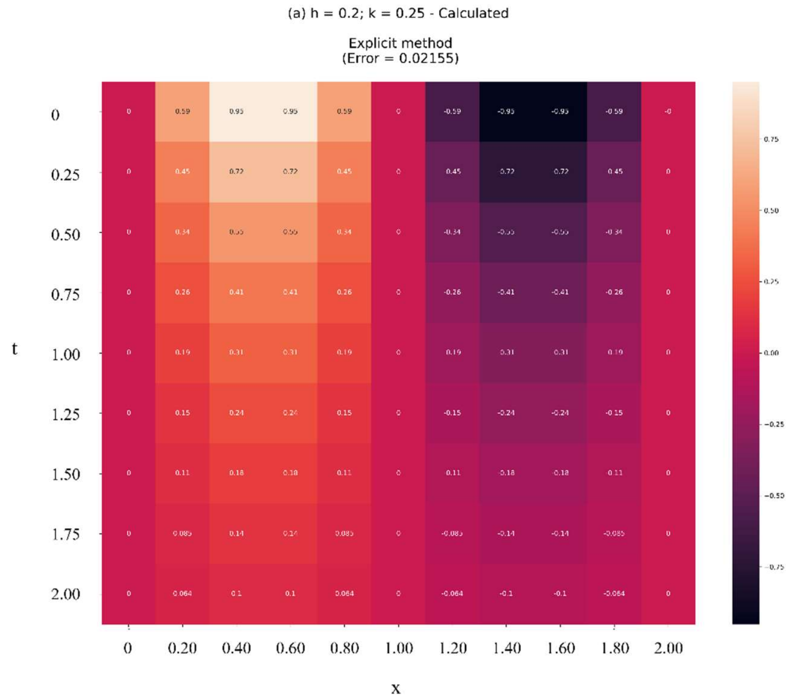


Figure 1A.1: Grid plot of calculated solution with explicit method

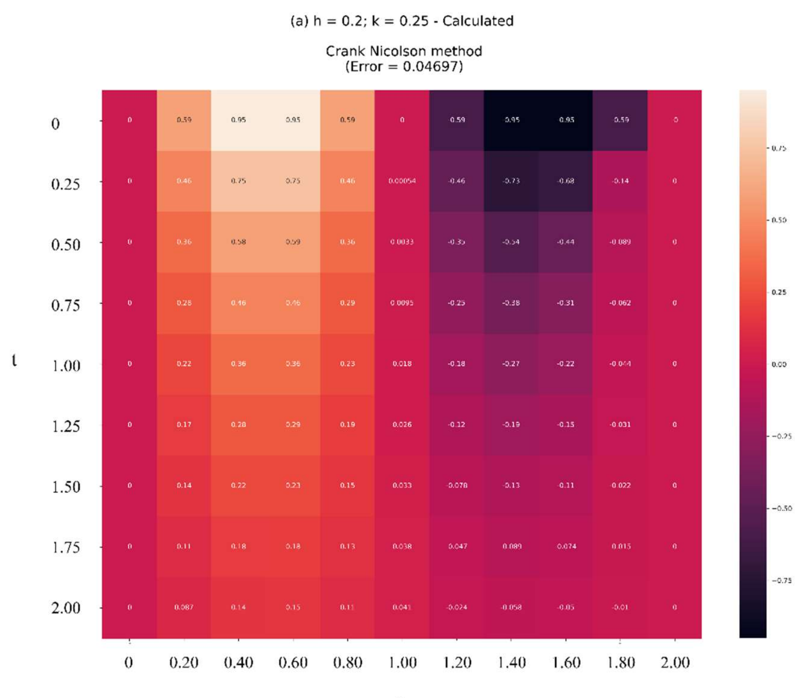


Figure 1A.2: Grid plot of calculated solution with Crank Nicolson method

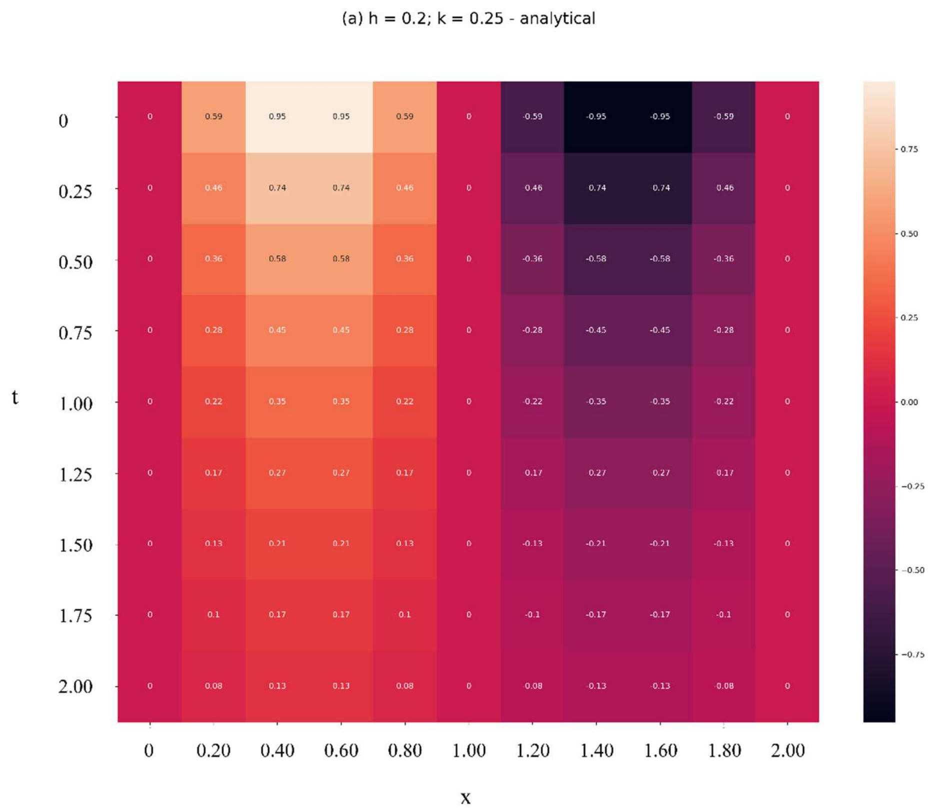


Figure 1A.3: Grid plot of analytical solution

	x	0	0.2	0.4	0.6	0.8	1	1.2	1.4	1.6	1.8	2
t	Q1 (a)	0	1	2	3	4	5	6	7	8	9	10
0	0	0	0.58779	0.95106	0.95106	0.58779	0	-0.58779	-0.95106	-0.95106	-0.58779	0
0.25	1	0	0.44561	0.72102	0.72102	0.44561	0	-0.44561	-0.72102	-0.72102	-0.44561	0
0.5	2	0	0.33783	0.54661	0.54661	0.33783	0	-0.33783	-0.54661	-0.54661	-0.33783	0
0.75	3	0	0.25611	0.4144	0.4144	0.25611	0	-0.25611	-0.4144	-0.4144	-0.25611	0
1	4	0	0.19416	0.31416	0.31416	0.19416	0	-0.19416	-0.31416	-0.31416	-0.19416	0
1.25	5	0	0.1472	0.23817	0.23817	0.1472	0	-0.1472	-0.23817	-0.23817	-0.1472	0
1.5	6	0	0.11159	0.18056	0.18056	0.11159	0	-0.11159	-0.18056	-0.18056	-0.11159	0
1.75	7	0	0.0846	0.13688	0.13688	0.0846	0	-0.0846	-0.13688	-0.13688	-0.0846	0
2	8	0	0.06413	0.10377	0.10377	0.06413	0	-0.06413	-0.10377	-0.10377	-0.06413	0

Figure 1A.4: Calculated data with explicit method

	x	0	0.2	0.4	0.6	0.8	1	1.2	1.4	1.6	1.8	2
t	Q1 (a)	0	1	2	3	4	5	6	7	8	9	10
0	0	0	0.58779	0.95106	0.95106	0.58779	0	-0.58779	-0.95106	-0.95106	-0.58779	0
0.25	1	0	0.46095	0.74584	0.74586	0.46106	0.00054	-0.45829	-0.73265	-0.68049	-0.13707	0
0.5	2	0	0.36149	0.58494	0.58509	0.3623	0.0033	-0.34865	-0.53879	-0.44317	-0.08927	0
0.75	3	0	0.28353	0.4589	0.45949	0.28635	0.00954	-0.25459	-0.38348	-0.30763	-0.06197	0
1	4	0	0.22249	0.36038	0.36189	0.22887	0.018	-0.17886	-0.27176	-0.21658	-0.04363	0
1.25	5	0	0.17481	0.28364	0.28647	0.18566	0.02622	-0.12126	-0.1908	-0.15273	-0.03076	0
1.5	6	0	0.13768	0.22409	0.2284	0.15295	0.03284	-0.07834	-0.13189	-0.10699	-0.02155	0
1.75	7	0	0.10887	0.178	0.18368	0.12788	0.03757	-0.04675	-0.08893	-0.0739	-0.01489	0
2	8	0	0.08656	0.14238	0.14916	0.10839	0.04057	-0.02376	-0.05763	-0.04984	-0.01004	0

Figure 1A.5: Calculated data with Crank-Nicolson method

	x	0	0.2	0.4	0.6	0.8	1	1.2	1.4	1.6	1.8	2
t	Analytical	0	1	2	3	4	5	6	7	8	9	10
0	0	0	0.58779	0.95106	0.95106	0.58779	0	-0.58779	-0.95106	-0.95106	-0.58779	0
0.25	1	0	0.45777	0.74068	0.74068	0.45777	0	-0.45777	-0.74068	-0.74068	-0.45777	0
0.5	2	0	0.35651	0.57684	0.57684	0.35651	0	-0.35651	-0.57684	-0.57684	-0.35651	0
0.75	3	0	0.27765	0.44925	0.44925	0.27765	0	-0.27765	-0.44925	-0.44925	-0.27765	0
1	4	0	0.21623	0.34987	0.34987	0.21623	0	-0.21623	-0.34987	-0.34987	-0.21623	0
1.25	5	0	0.1684	0.27248	0.27248	0.1684	0	-0.1684	-0.27248	-0.27248	-0.1684	0
1.5	6	0	0.13115	0.21221	0.21221	0.13115	0	-0.13115	-0.21221	-0.21221	-0.13115	0
1.75	7	0	0.10214	0.16527	0.16527	0.10214	0	-0.10214	-0.16527	-0.16527	-0.10214	0
2	8	0	0.07955	0.12871	0.12871	0.07955	0	-0.07955	-0.12871	-0.12871	-0.07955	0

Figure 1A.6: Analytical data

From the comparisons above, we see that both methods yield almost similar results with small errors, and the difference between their errors is approximately 0.02542. Bear in mind that the error calculated is the average difference between calculated and analytical data.

Since two methods do not depict a significant difference, there is not much that we can conclude at this stage.

Question 1 (b)

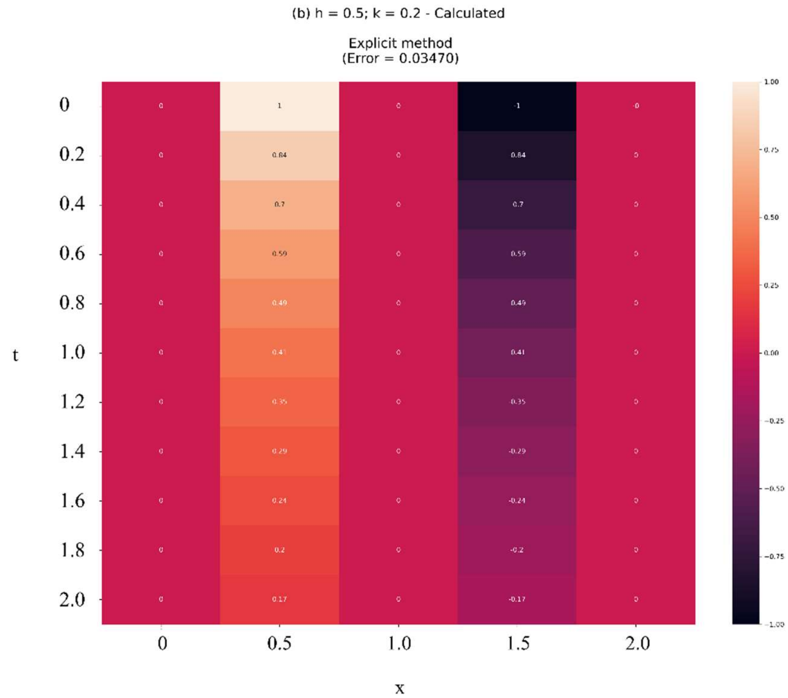


Figure 1B.1: Grid plot of calculated solution with explicit method

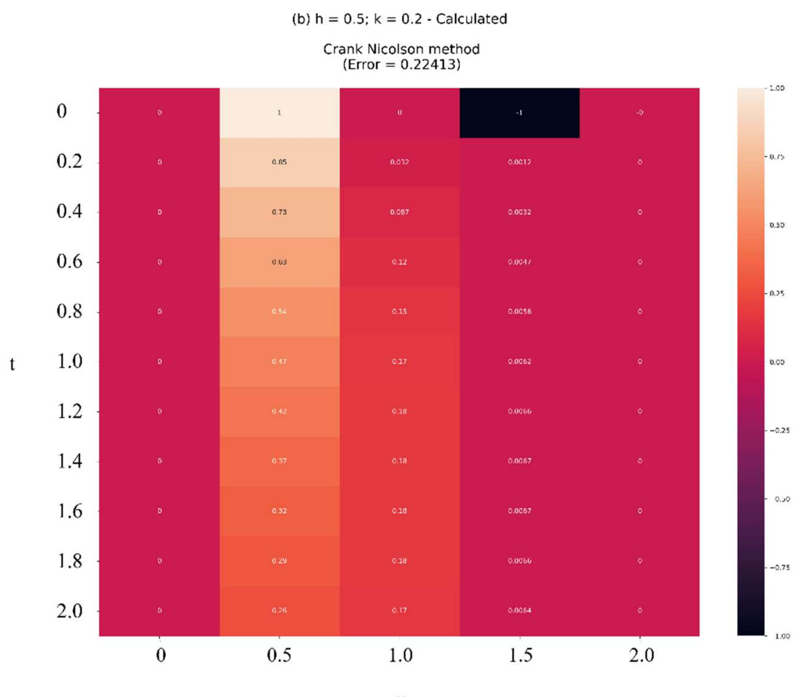


Figure 1B.2: Grid plot of calculated solution with Crank Nicolson method

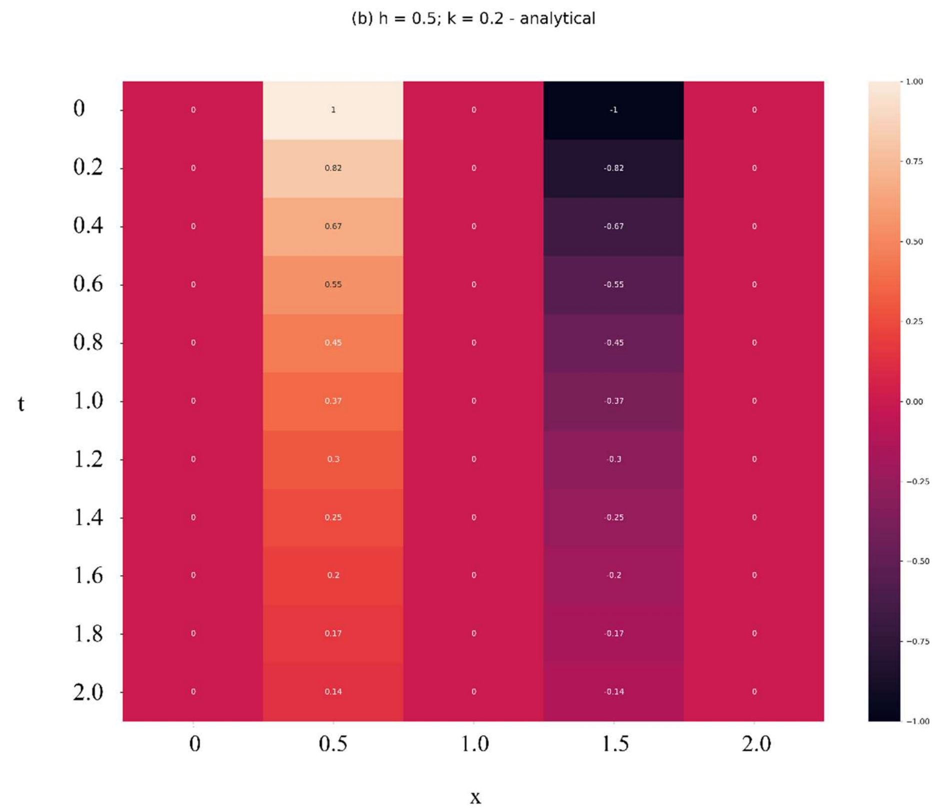


Figure 1B.3: Grid plot of analytical solution

	x	0	0.5	1	1.5	2
t	Q1 (b)	0	1	2	3	4
0	0	0	1	0	-1	0
0.2	1	0	0.83789	0	-0.83789	0
0.4	2	0	0.70206	0	-0.70206	0
0.6	3	0	0.58825	0	-0.58825	0
0.8	4	0	0.49289	0	-0.49289	0
1	5	0	0.41299	0	-0.41299	0
1.2	6	0	0.34604	0	-0.34604	0
1.4	7	0	0.28994	0	-0.28994	0
1.6	8	0	0.24294	0	-0.24294	0
1.8	9	0	0.20356	0	-0.20356	0
2	10	0	0.17056	0	-0.17056	0

Figure 1B.4: Calculated data with explicit method

	x	0	0.5	1	1.5	2
t	Q1 (b)	0	1	2	3	4
0	0	0	1	0	-1	0
0.2	1	0	0.85124	0.03196	0.0012	0
0.4	2	0	0.72803	0.08654	0.00325	0
0.6	3	0	0.62677	0.12465	0.00468	0
0.8	4	0	0.54308	0.1502	0.00564	0
1	5	0	0.4735	0.16623	0.00624	0
1.2	6	0	0.41529	0.1751	0.00657	0
1.4	7	0	0.36628	0.17864	0.00671	0
1.6	8	0	0.32473	0.17826	0.00669	0
1.8	9	0	0.28928	0.17504	0.00657	0
2	10	0	0.25883	0.16982	0.00638	0

Figure 1B.5: Calculated data with Crank-Nicolson method

	x	0	0.5	1	1.5	2
t	Analytical	0	1	2	3	4
0	0	0	1	0	-1	0
0.2	1	0	0.81873	0	-0.81873	0
0.4	2	0	0.67032	0	-0.67032	0
0.6	3	0	0.54881	0	-0.54881	0
0.8	4	0	0.44933	0	-0.44933	0
1	5	0	0.36788	0	-0.36788	0
1.2	6	0	0.30119	0	-0.30119	0
1.4	7	0	0.2466	0	-0.2466	0
1.6	8	0	0.2019	0	-0.2019	0
1.8	9	0	0.1653	0	-0.1653	0
2	10	0	0.13534	0	-0.13534	0

Figure 1B.6: Analytical data

From the data shown above, we can clearly observe that in this case, explicit method is more accurate than Crank-Nicolson method. Hence, we can deduce that explicit method is more suitable when the interval is larger.

Question 1 (c)

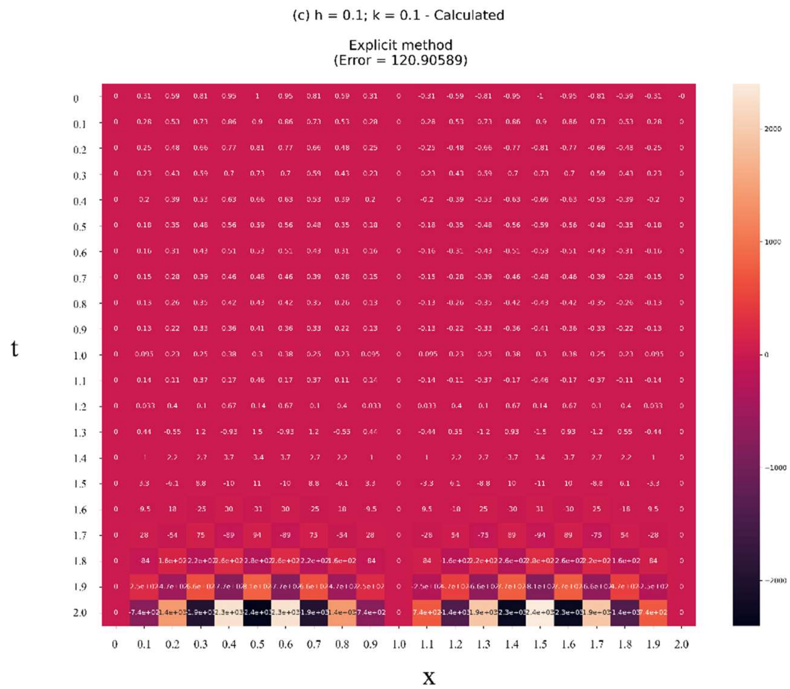


Figure 1C.1: Grid plot of calculated solution with explicit method

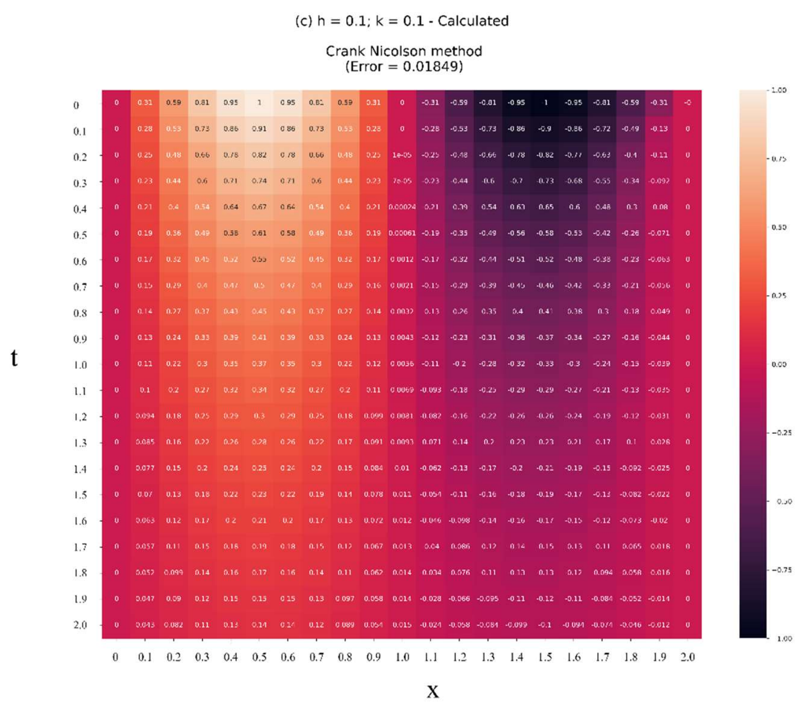


Figure 1C.2: Grid plot of calculated solution with Crank Nicolson method

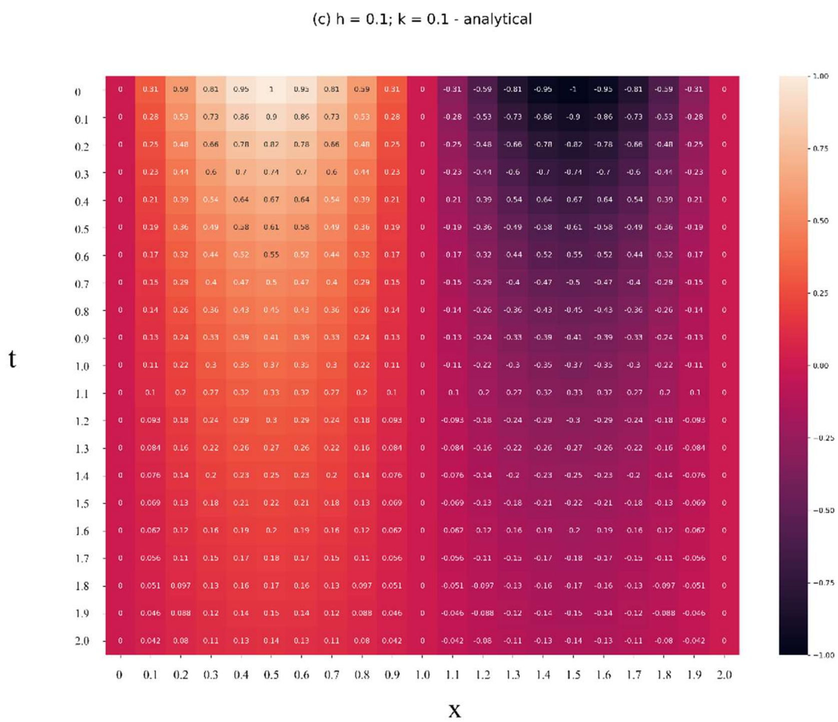


Figure 1C.3: Grid plot of analytical solution

	x	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2
t	Q1 (c)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
0	0	0	0.30902	0.58779	0.80902	0.95106	1	0.95106	0.80902	0.58779	0.30902	0	-0.30902	-0.58779	-0.80902	-0.95106	-1	-0.95106	-0.80902	-0.58779	-0.30902	0
0.1	1	0	0.27837	0.52949	0.72878	0.85673	0.90083	0.85673	0.72878	0.52949	0.27837	0	-0.27837	-0.52949	-0.72878	-0.85673	-0.90083	-0.85673	-0.72878	-0.52949	-0.27837	0
0.2	2	0	0.25076	0.47698	0.6565	0.77177	0.81146	0.77177	0.6565	0.47698	0.25076	0	-0.25076	-0.47698	-0.6565	-0.77177	-0.81146	-0.77177	-0.6565	-0.47698	-0.25076	0
0.3	3	0	0.2259	0.42966	0.5914	0.69519	0.73103	0.69519	0.5914	0.42966	0.2259	0	-0.2259	-0.42966	-0.5914	-0.69519	-0.73103	-0.69519	-0.5914	-0.42966	-0.2259	0
0.4	4	0	0.20347	0.38708	0.53268	0.62634	0.6584	0.62634	0.53268	0.38708	0.20347	0	-0.20347	-0.38708	-0.53268	-0.62634	-0.6584	-0.62634	-0.53268	-0.38708	-0.20347	0
0.5	5	0	0.18335	0.34857	0.48005	0.56393	0.59343	0.56393	0.48005	0.34857	0.18335	0	-0.18335	-0.34857	-0.48005	-0.56393	-0.59343	-0.56393	-0.48005	-0.34857	-0.18335	0
0.6	6	0	0.16498	0.31438	0.43182	0.50883	0.53365	0.50883	0.43182	0.31438	0.16498	0	-0.16498	-0.31438	-0.43182	-0.50883	-0.53365	-0.50883	-0.43182	-0.31438	-0.16498	0
0.7	7	0	0.14919	0.282	0.39086	0.45595	0.48335	0.45595	0.39086	0.282	0.14919	0	-0.14919	-0.282	-0.39086	-0.45595	-0.48335	-0.45595	-0.39086	-0.282	-0.14919	0
0.8	8	0	0.13259	0.25773	0.34651	0.41776	0.42783	0.41776	0.34651	0.25773	0.13259	0	-0.13259	-0.25773	-0.34651	-0.41776	-0.42783	-0.41776	-0.34651	-0.25773	-0.13259	0
0.9	9	0	0.12504	0.22089	0.32875	0.35577	0.40742	0.35577	0.32875	0.22089	0.12504	0	-0.12504	-0.22089	-0.32875	-0.35577	-0.40742	-0.35577	-0.32875	-0.22089	-0.12504	0
1	10	0	0.09546	0.23306	0.24684	0.38073	0.30276	0.38073	0.24684	0.23306	0.09546	0	-0.09546	-0.23306	-0.24684	-0.38073	-0.30276	-0.38073	-0.24684	-0.23306	-0.09546	0
1.1	11	0	0.13816	0.1076	0.36854	0.16607	0.46076	0.16607	0.36854	0.1076	0.13816	0	-0.13816	-0.1076	-0.36854	-0.16607	-0.46076	-0.16607	-0.36854	-0.1076	-0.13816	0
1.2	12	0	-0.03279	0.40295	-0.10099	0.6698	-0.13641	0.6698	-0.10099	0.40295	-0.03279	0	0.03279	-0.40295	0.10099	-0.6698	0.13641	-0.6698	0.10099	-0.40295	0.03279	0
1.3	13	0	0.44193	-0.54914	1.19058	-0.92804	1.49731	-0.92804	1.19058	-0.54914	0.44193	0	-0.44193	0.54914	-1.19058	0.92804	-1.49731	0.92804	-1.19058	0.54914	-0.44193	0
1.4	14	0	-1.01	2.21773	-2.71874	3.67596	-3.41748	3.67596	-2.71874	2.21773	-1.01	0	1.01	-2.21773	2.71874	-3.67596	3.41748	-3.67596	2.71874	-2.21773	1.01	0
1.5	15	0	3.28372	-6.05433	8.76214	-9.99038	10.95683	-9.99038	8.76214	-6.05433	3.28372	0	-3.28372	6.05433	-8.76214	9.99038	-10.9568	9.99038	-8.76214	6.05433	-3.28372	0
1.6	16	0	-9.50481	18.41932	-25.2504	30.23386	-31.4911	30.23386	-25.2504	18.41932	-9.50481	0	9.50481	-18.4193	25.25036	-30.2339	31.49109	-30.2339	25.25036	-18.4193	9.50481	0
1.7	17	0	28.41864	-54.1204	75.21355	-88.5239	93.58981	-88.5239	75.21355	-54.1204	28.41864	0	-28.4186	54.12038	-75.2136	88.52386	-93.5898	88.52386	-75.2136	54.12038	-28.4186	0
1.8	18	0	-84.005	160.5518	-221.73	261.8966	-275.45	261.8966	-221.73	160.5518	-84.005	0	84.00497	-160.552	221.7298	-261.897	275.4496	-261.897	221.7298	-160.552	84.00497	0
1.9	19	0	248.8977	-474.568	655.6184	-772.565	813.4414	-772.565	655.6184	-474.568	248.8977	0	-248.898	474.5683	-655.618	772.5649	-813.441	772.5649	-655.618	474.5683	-248.898	0
2	20	0	-736.313	1403.574	-1936.55	2281.448	-2400.48	2281.448	-1936.55	1403.574	-736.313	0	736.3126	-1403.57	1936.552	-2281.45	2400.479	-2281.45	1936.552	-1403.57	736.3126	0

Figure 1C.4: Calculated data with explicit method

	x	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2
t	Q1 (c)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
0	0	0	0.30902	0.58779	0.80902	0.95106	1	0.95106	0.80902	0.58779	0.30902	0	-0.30902	-0.58779	-0.80902	-0.95106	-1	-0.95106	-0.80902	-0.58779	-0.30902	0
0.1	1	0	0.27982	0.53225	0.73257	0.86119	0.90551	0.86119	0.73257	0.53225	0.27982	0	-0.27981	-0.53223	-0.73252	-0.86098	-0.90472	-0.85829	-0.72182	-0.49242	-0.13229	0
0.2	2	0	0.25338	0.48195	0.66335	0.77981	0.81994	0.77981	0.66335	0.48195	0.25338	1.00E-05	-0.25333	-0.48181	-0.6629	-0.77839	-0.81556	-0.76691	-0.62792	-0.39651	-0.10652	0
0.3	3	0	0.22943	0.43641	0.60066	0.70612	0.74246	0.70612	0.60067	0.43642	0.22946	7.00E-05	-0.22923	-0.43581	-0.59899	-0.70163	-0.73107	-0.67954	-0.54589	-0.34275	-0.09208	0
0.4	4	0	0.20775	0.39517	0.5439	0.6394	0.6723	0.6394	0.54392	0.3952	0.20784	0.00024	-0.20712	-0.39354	-0.53992	-0.63025	-0.65287	-0.60208	-0.48053	-0.29946	-0.08045	0
0.5	5	0	0.18812	0.35783	0.49251	0.57898	0.60878	0.57899	0.49254	0.35792	0.18837	0.00061	-0.18667	-0.35453	-0.48541	-0.56474	-0.58248	-0.5347	-0.42467	-0.26391	-0.0709	0
0.6	6	0	0.17035	0.32402	0.44597	0.52427	0.55126	0.5243	0.44606	0.32425	0.17089	0.00123	-0.16769	-0.31858	-0.43548	-0.50537	-0.51966	-0.47546	-0.37659	-0.23353	-0.06274	0
0.7	7	0	0.15425	0.2934	0.40383	0.47474	0.49919	0.47481	0.40403	0.29386	0.15526	0.00209	-0.1501	-0.28562	-0.39006	-0.45186	-0.46363	-0.42328	-0.3346	-0.20723	-0.05567	0
0.8	8	0	0.13967	0.26568	0.36568	0.4299	0.45207	0.43004	0.36605	0.26648	0.14131	0.00315	-0.13387	-0.25554	-0.34893	-0.40376	-0.41368	-0.3771	-0.29772	-0.18422	-0.04949	0
0.9	9	0	0.12648	0.24058	0.33114	0.38932	0.40943	0.38956	0.33176	0.24184	0.12887	0.00434	-0.11897	-0.22821	-0.31178	-0.36059	-0.36911	-0.33615	-0.26515	-0.16397	-0.04405	0
1	10	0	0.11453	0.21786	0.29988	0.35259	0.37086	0.35298	0.30081	0.21966	0.11779	0.0056	-0.10534	-0.20344	-0.27831	-0.32188	-0.32933	-0.29973	-0.23629	-0.14607	-0.03924	0
1.1	11	0	0.10372	0.1973	0.27159	0.31936	0.33599	0.31992	0.27289	0.19972	0.1079	0.00686	-0.09292	-0.18105	-0.24819	-0.28718	-0.29379	-0.26731	-0.21066	-0.13019	-0.03497	0
1.2	12	0	0.09393	0.17869	0.246	0.28931	0.30446	0.29007	0.24771	0.18177	0.09906	0.0081	-0.08166	-0.16084	-0.22111	-0.25608	-0.26203	-0.23839	-0.18784	-0.11607	-0.03118	0
1.3	13	0	0.08508	0.16185	0.22285	0.26214	0.27597	0.26311	0.22499	0.16562	0.09114	0.00928	-0.07146	-0.14263	-0.19678	-0.22821	-0.23363	-0.21258	-0.1675	-0.1035	-0.02781	0
1.4	14	0	0.07707	0.14663	0.20191	0.23758	0.25023	0.23877	0.20451	0.15108	0.08404	0.01038	-0.06225	-0.12625	-0.17494	-0.20323	-0.20822	-0.18953	-0.14936	-0.09229	-0.02479	0
1.5	15	0	0.06982	0.13286	0.18299	0.21538	0.22698	0.2168	0.18604	0.13798	0.07766	0.01138	-0.05395	-0.11152	-0.15533	-0.18085	-0.18548	-0.16892	-0.13315	-0.08229	-0.02211	0
1.6	16	0	0.06328	0.12041	0.16588	0.19533	0.20598	0.19696	0.16937	0.12617	0.07192	0.01229	-0.04649	-0.09829	-0.13775	-0.16079	-0.16513	-0.15049	-0.11867	-0.07335	-0.01971	0
1.7	17	0	0.05735	0.10916	0.15043	0.17721	0.18701	0.17906	0.15433	0.11552	0.06674	0.0131	-0.03979	-0.08642	-0.12198	-0.14282	-0.14691	-0.134	-0.10572	-0.06536	-0.01756	0
1.8	18	0	0.052	0.09899	0.13646	0.16084	0.16989	0.16289	0.14076	0.10591	0.06205	0.01381	-0.03378	-0.07578	-0.10784	-0.12673	-0.13059	-0.11924	-0.09413	-0.05822	-0.01564	0
1.9	19	0	0.04717	0.08981	0.12384	0.14606	0.15442	0.1483	0.1285	0.09722	0.05781	0.01442	-0.02841	-0.06624	-0.09518	-0.1123	-0.11598	-0.10603	-0.08376	-0.05182	-0.01392	0
2	20	0	0.0428	0.08151	0.11245	0.1327	0.14045	0.13511	0.11743	0.08937	0.05396	0.01494	-0.02361	-0.0577	-0.08383	-0.09939	-0.10289	-0.0942	-0.07448	-0.0461	-0.01238	0

Figure 1C.5: Calculated data with Crank-Nicolson method

	x	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2
t	Analytical	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
0	0	0	0.30902	0.58779	0.80902	0.95106	1	0.95106	0.80902	0.58779	0.30902	0	-0.30902	-0.58779	-0.80902	-0.95106	-1	-0.95106	-0.80902	-0.58779	-0.30902	0
0.1	1	0	0.27961	0.53185	0.73203	0.86055	0.90484	0.86055	0.73203	0.53185	0.27961	0	-0.27961	-0.53185	-0.73203	-0.86055	-0.90484	-0.86055	-0.73203	-0.53185	-0.27961	0
0.2	2	0	0.253	0.48124	0.66237	0.77866	0.81873	0.77866	0.66237	0.48124	0.253	0	-0.253	-0.48124	-0.66237	-0.77866	-0.81873	-0.77866	-0.66237	-0.48124	-0.253	0
0.3	3	0	0.22893	0.43544	0.59933	0.70456	0.74082	0.70456	0.59933	0.43544	0.22893	0	-0.22893	-0.43544	-0.59933	-0.70456	-0.74082	-0.70456	-0.59933	-0.43544	-0.22893	0
0.4	4	0	0.20714	0.394	0.5423	0.63751	0.67032	0.63751	0.5423	0.394	0.20714	0	-0.20714	-0.394	-0.5423	-0.63751	-0.67032	-0.63751	-0.5423	-0.394	-0.20714	0
0.5	5	0	0.18743	0.35651	0.49069	0.57684	0.60653	0.57684	0.49069	0.35651	0.18743	0	-0.18743	-0.35651	-0.49069	-0.57684	-0.60653	-0.57684	-0.49069	-0.35651	-0.18743	0
0.6	6	0	0.16959	0.32258	0.444	0.52195	0.54881	0.52195	0.444	0.32258	0.16959	0	-0.16959	-0.32258	-0.444	-0.52195	-0.54881	-0.52195	-0.444	-0.32258	-0.16959	0
0.7	7	0	0.15345	0.29189	0.40175	0.47228	0.49659	0.47228	0.40175	0.29189	0.15345	0	-0.15345	-0.29189	-0.40175	-0.47228	-0.49659	-0.47228	-0.40175	-0.29189	-0.15345	0
0.8	8	0	0.13885	0.26411	0.36351	0.42734	0.44933	0.42734	0.36351	0.26411	0.13885	0	-0.13885	-0.26411	-0.36351	-0.42734	-0.44933	-0.42734	-0.36351	-0.26411	-0.13885	0
0.9	9	0	0.12564	0.23898	0.32892	0.38667	0.40657	0.38667	0.32892	0.23898	0.12564	0	-0.12564	-0.23898	-0.32892	-0.38667	-0.40657	-0.38667	-0.32892	-0.23898	-0.12564	0
1	10	0	0.11368	0.21623	0.29762	0.34987	0.36788	0.34987	0.29762	0.21623	0.11368	0	-0.11368	-0.21623	-0.29762	-0.34987	-0.36788	-0.34987	-0.29762	-0.21623	-0.11368	0
1.1	11	0	0.10286	0.19566	0.2693	0.31658	0.33287	0.31658	0.2693	0.19566	0.10286	0	-0.10286	-0.19566	-0.2693	-0.31658	-0.33287	-0.31658	-0.2693	-0.19566	-0.10286	0
1.2	12	0	0.09307	0.17704	0.24367	0.28645	0.30119	0.28645	0.24367	0.17704	0.09307	0	-0.09307	-0.17704	-0.24367	-0.28645	-0.30119	-0.28645	-0.24367	-0.17704	-0.09307	0
1.3	13	0	0.08422	0.16019	0.22048	0.25919	0.27253	0.25919	0.22048	0.16019	0.08422	0	-0.08422	-0.16019	-0.22048	-0.25919	-0.27253	-0.25919	-0.22048	-0.16019	-0.08422	0
1.4	14	0	0.0762	0.14495	0.1995	0.23453	0.2466	0.23453	0.1995	0.14495	0.0762	0	-0.0762	-0.14495	-0.1995	-0.23453	-0.2466	-0.23453	-0.1995	-0.14495	-0.0762	0
1.5	15	0	0.06895	0.13115	0.18052	0.21221	0.22313	0.21221	0.18052	0.13115	0.06895	0	-0.06895	-0.13115	-0.18052	-0.21221	-0.22313	-0.21221	-0.18052	-0.13115	-0.06895	0
1.6	16	0	0.06239	0.11867	0.16334	0.19201	0.2019	0.19201	0.16334	0.11867	0.06239	0	-0.06239	-0.11867	-0.16334	-0.19201	-0.2019	-0.19201	-0.16334	-0.11867	-0.06239	0
1.7	17	0	0.05645	0.10738	0.14779	0.17374	0.18268	0.17374	0.14779	0.10738	0.05645	0	-0.05645	-0.10738	-0.14779	-0.17374	-0.18268	-0.17374	-0.14779	-0.10738	-0.05645	0
1.8	18	0	0.05108	0.09716	0.13373	0.15721	0.1653	0.15721	0.13373	0.09716	0.05108	0	-0.05108	-0.09716	-0.13373	-0.15721	-0.1653	-0.15721	-0.13373	-0.09716	-0.05108	0
1.9	19	0	0.04622	0.08791	0.121	0.14225	0.14957	0.14225	0.121	0.08791	0.04622	0	-0.04622	-0.08791	-0.121	-0.14225	-0.14957	-0.14225	-0.121	-0.08791	-0.04622	0
2	20	0	0.04182	0.07955	0.10949	0.12871	0.13534	0.12871	0.10949	0.07955	0.04182	0	-0.04182	-0.07955	-0.10949	-0.12871	-0.13534	-0.12871	-0.10949	-0.07955	-0.04182	0

Figure 1C.6: Analytical data

From the grid plots attached, we can easily observe that both methods yield completely different results and hence, completely different plots. The error of the results obtained from explicit method is a whopping 120.90589, whereas that from Crank Nicolson method is only 0.01849, which is highly accurate.

In this case, both the values of h and k are relatively small. The grid plot obtained through explicit method shows that from $t = 1.1$ and onwards, the frequency of oscillation of data form $x = 0$ to $x = 2$ starts to increase. Plus, as the value of t increases, the amplitude of oscillation increases exponentially.

This phenomenon can be explained by the nature of the two methods. Explicit method is more accurate with bigger step sizes because its algorithms are less computationally demanding. When the step sizes are big, however, this method becomes unstable, as shown in the results obtained. On the other hand, Crank Nicolson method runs more computational algorithms than the explicit method. So, being more computationally intensive, Crank Nicolson method is a more stable and hence, reliable scheme to solve the heat equation given when the step sizes are smaller.

In the upcoming discussion, we include the 3-dimensional surface plots of both methods and the analytical plot for more comparisons from a different approach.

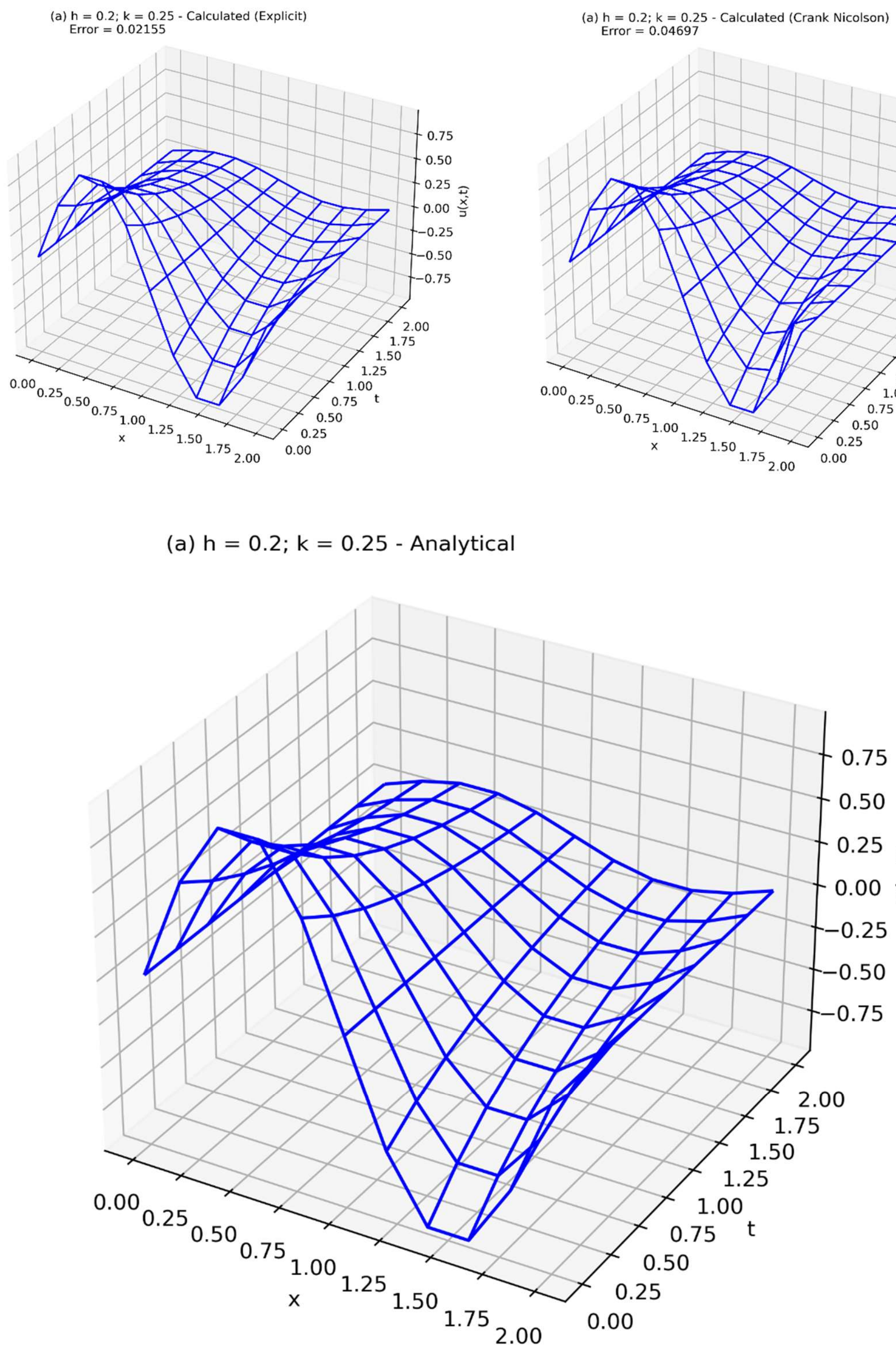


Figure 2A: Surface plots of calculated and numerical results for ***Q1(a)***

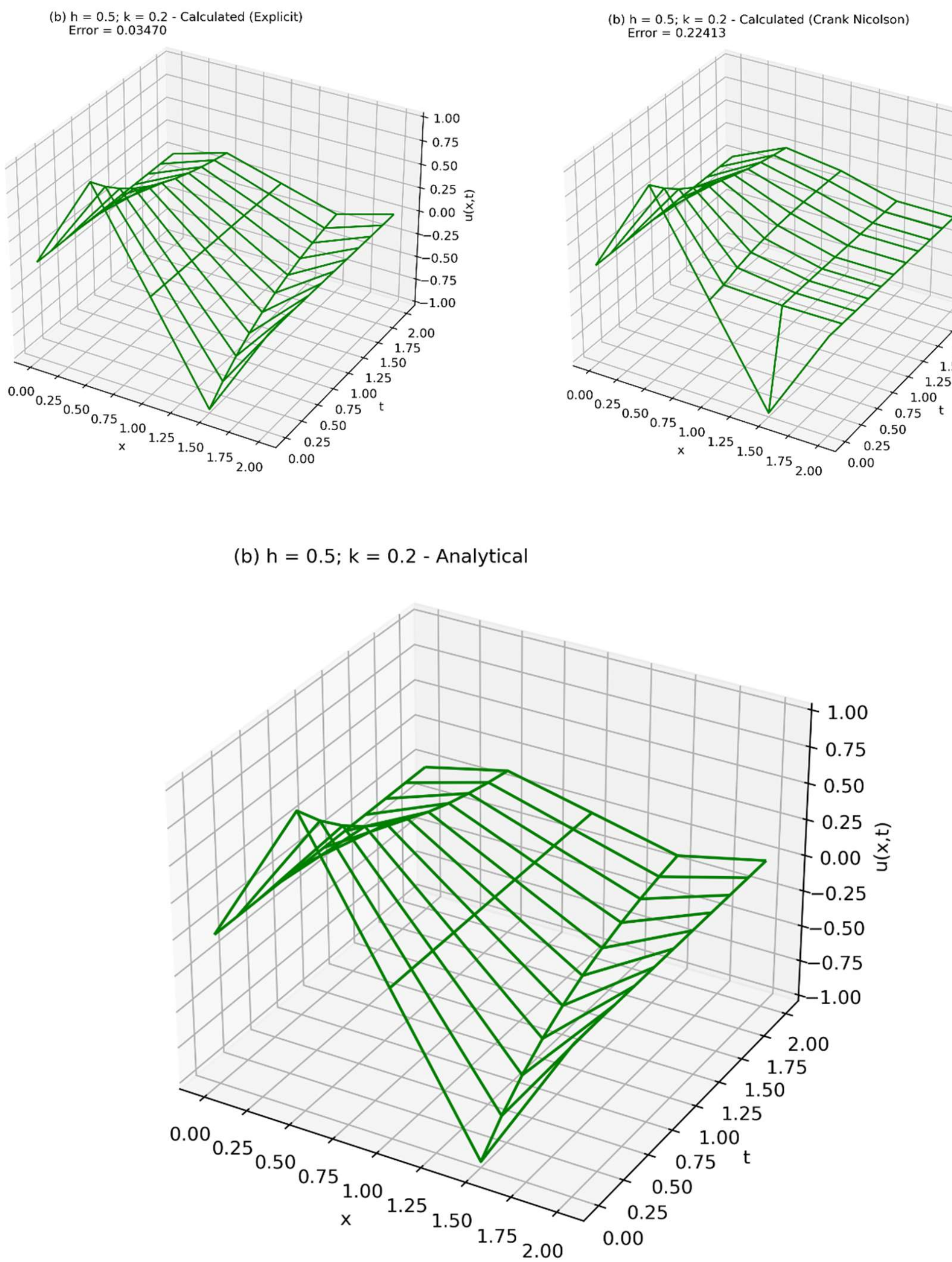


*Figure 2B: Surface plots of calculated and numerical results for **Q1(b)***

The figures above show that the surface generated through Crank Nicolson method strays more from the analytical plot due to the larger step size.

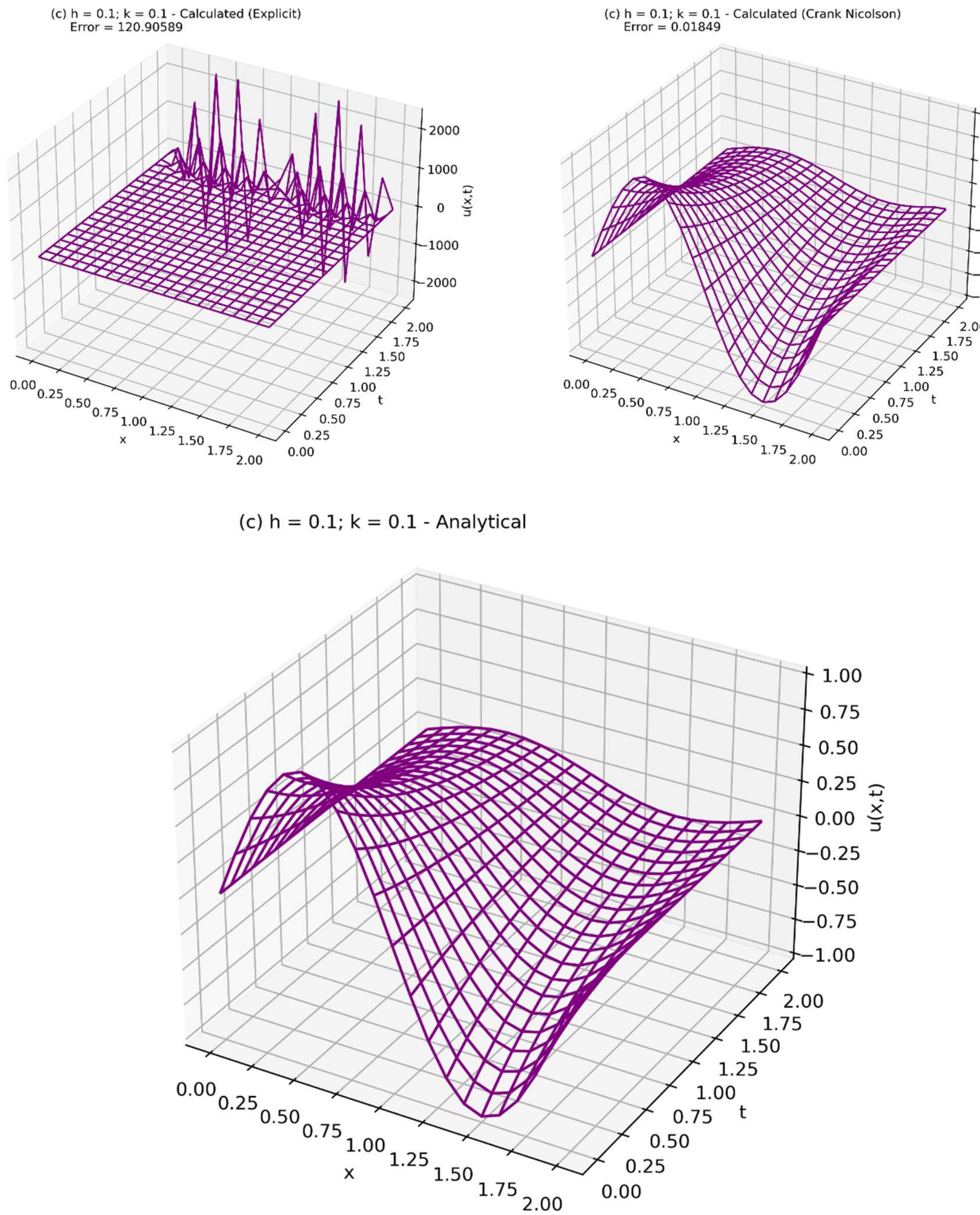
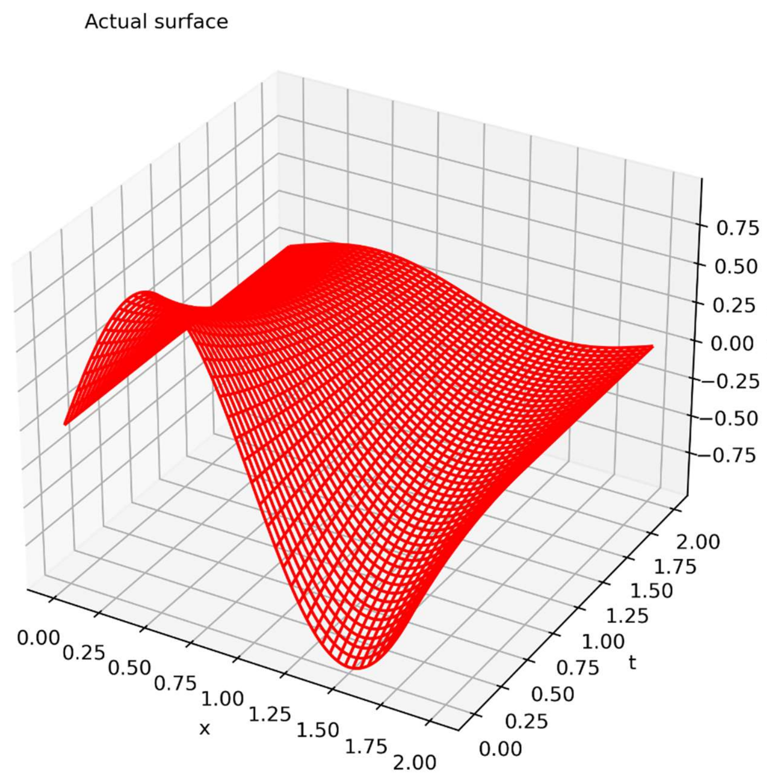


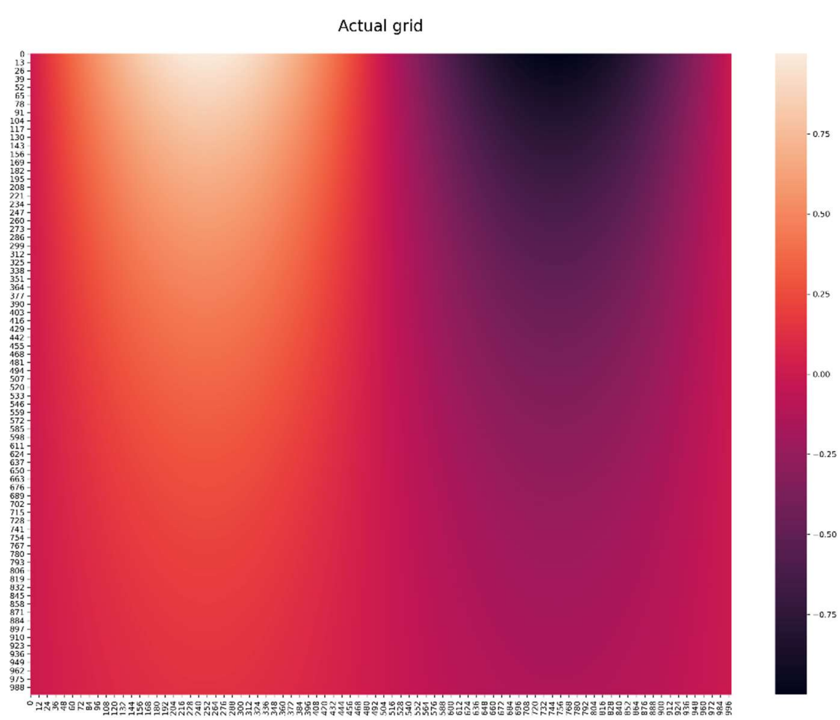
Figure 2C: Surface plots of calculated and numerical results for ***Q1(c)***

In this page, we clearly see that the surface generated through explicit method went haywire, whereas the surface plotted from Crank Nicolson scheme abides closely to the analytical surface.

Extra: The diagrams below represent the actual surfaces of the 3-dimensional equation

$$u(x,t) = e^{-t}cos\left[\pi\left(x - \frac{1}{2}\right)\right]$$





Generated with actual_plots.py

CONCLUSION

From the data collected and the plots generated from our Python program, we learnt that explicit method is suitable for when the step size is larger, while Crank Nicolson method is more suitable for when the step size is smaller.

From all the 3 combinations of h and k, the combination of $h = 0.1$ and $k = 0.1$ in Question 1(c) yield better approximation, with the actual plots in the previous page as references. This is due to the fact that the error of part C is the smallest compared to all the 3 combinations. To conclude, it can be said that when solving the heat equation using the Crank Nicolson Method, the smaller the step size for x and t, the greater the accuracy and better approximation to the analytical solution u(x, t).