

講義ノート

計算数理科学

Ver.1.0

2018年4月2日

東北大学大学院情報科学研究科

教授 山本 悟

目 次

1. はじめに
 2. 計算数理学の典型としての数値流体力学
 3. 数値計算とは
 4. 計算数理学入口
 - 4-1 反応方程式
 - 4-2 反応方程式系
 - 4-3 偏微分方程式
 - 4-4 拡散方程式
 - 4-5 偏微分方程式の型の分類
 - 4-6 ラプラス方程式の解析解
 - 4-7 ラプラス方程式の差分解法
 - 4-8 テイラー展開
 - 4-9 境界条件
 - 4-10 ラプラス方程式を手計算
 5. 数理モデルの数値計算法
 - 5-1 ラプラス方程式の反復解法
 - 5-2 三次元ポテンシャル流れのSOR解法
 - 5-3 熱伝導方程式の差分解法
 - 5-4 線形安定理論
 - 5-5 直接法と反復法
 - 5-6 熱伝導方程式の反復解法
 - 5-7 二次元熱伝導方程式のExcel計算
 - 5-8 三次元熱伝導方程式の差分解法
 - 5-9 特性方程式
 6. 数理モデリングと計算
 - 6-1 反応拡散方程式
 - 6-2 反応拡散方程式系
 - 6-3 バクテリアと白血球のバトル
 - 6-4 自己複製パターンの数理モデル
 7. まとめ
- 参考文献

1. はじめに

計算数理学 (Mathematical Modeling and Computation) という名前は、あまり聞きなれないかもしれません。別に私が考えたものではなく、このキーワードで Google 検索すれば、いくつか既存のホームページにこの名前を見つけることができます。

さて、計算数理学は、「計算」と「数理学」からなっています。まず、数理学 (Mathematical Science) ですが、自然科学 (Natural Science) や社会科学 (Social Science) の様々な現象や事象から、数学的手法を用いてそれらをモデル化する、すなわち、数理モデル (Mathematical Model) を構築する学問です。この分野は、主に数学を専攻する研究者により研究されています。計算数理学は、その構築された数理モデルを、さらにコンピュータを用いて数値計算する研究分野です。応用数学や工学を専攻する研究者が得意な分野です。

2. 計算数理学の典型としての数値流体力学

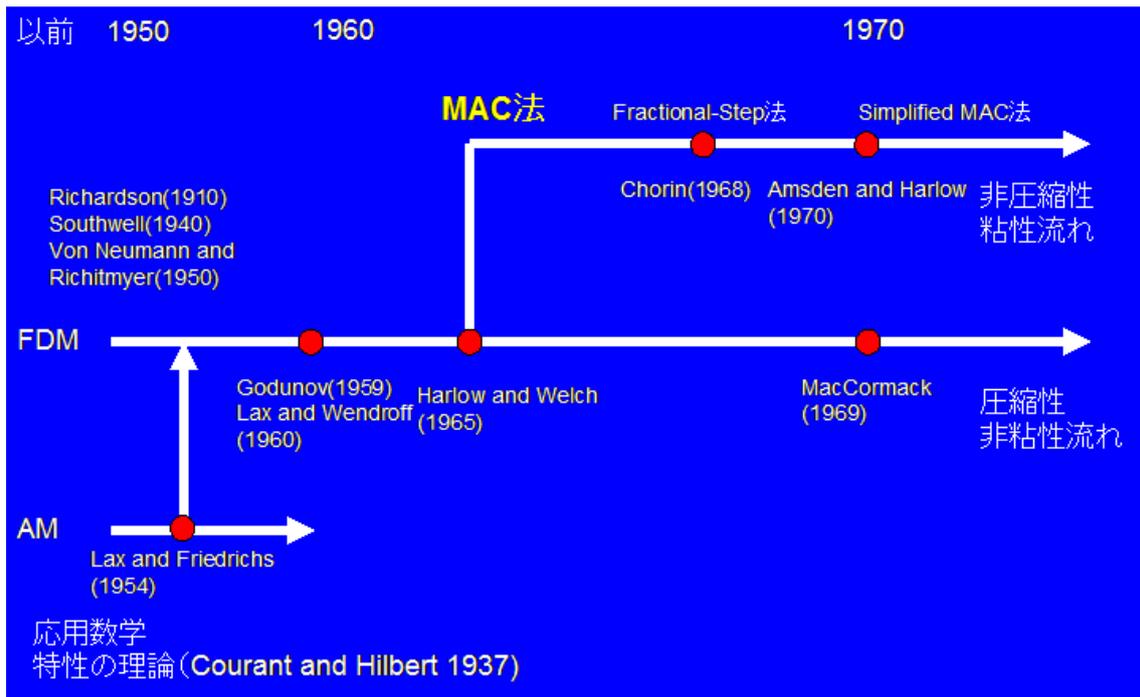
私の研究分野である数値流体力学という学問があります。これは、Computational Fluid Dynamics の日本語訳で、英語の頭文字を取って、通称、CFD と呼ばれます。本来ですと、計算流体力学と訳されるのが自然ですが、最初に訳した日本人の研究者がこのように訳したことから、その後、CFD のことを数値流体力学と呼ぶのが一般的になりました。数値流体力学は、計算数理学の典型的な成功例の一つです。

流体力学とはいろいろな流れを研究する学問で、身の回りにある流れ、たとえば川の流れや空気の流れから、自動車や航空機周りの流れ、そして台風や海洋流などの地球規模の流れなど、様々な流れの力学です。数値流体力学は、そのような流れをコンピュータでシミュレーションする学問です。

流体力学では流れを偏微分方程式 (Partial Differential Equation, 略して PDE) で表現します。その数式をコンピュータで計算します。数値流体力学ではこの流れを支配している数式を、如何に精度良く、そして速く計算するかが研究されてきました。したがって、流体力学の知識のみならず、コンピュータの知識も必要になります。

流体の流れを支配する偏微分方程式は、ナビエ・ストークス方程式 (Navier-Stokes Equations) と呼ばれます。最も完成された典型的数理モデルの一つです。Navier と Stokes の両研究者により構築された偏微分方程式で、数値流体力学ではこの方程式がコンピュータにより解かれます。実はこの方程式は1つの方程式ではなく、複数の連立方程式から成っています。流れの連続、運動、そしてエネルギーを支配する方程式です。なお、運動方程式の数は次元の数だけありますので、三次元では、合計5つの連立方程式となります。

数値流体力学研究は、20世紀に入ってから始まりました。ですから、100年程度の歴史があります。1910年に発表された Richardson の論文[1]がその始まりという方もいます。創世記の代表的な研究を系統図にまとめてみました。



FDMはFinite Difference Methodの略で差分法のことです。AMはApplied Mathematics、応用数学のことです。差分法以外にも有限要素法(Finite Element Method, 略してFEM)、境界要素法(Boundary Element Method, 略してBEM)などもあります。本講義では、差分法を用います。差分法は、応用数学の影響を強く受けています。応用数学者がFDM自体を研究している場合もあります。1950年以前というのは、まともなコンピュータはありません。紙とえんぴつ、もしくは手動の手回し計算機(タイガー計算機)などを用いて研究していた時代です。1946年にやっと、真空管でできたコンピュータENIACが誕生しました。

フォンノイマン(von Neumann)という研究者の名前をご存知でしょうか? コンピュータの原理を提案した偉大な研究者ですが、いろいろなことを研究しました。差分法の基礎になる線形安定性理論(Linear Stability Analysis)も提案しています。

1950年代に入り、コンピュータの進歩とともに数値流体力学の研究は急速に発展します。大きな研究の方向としては、圧縮性流れの数値流体力学と、非圧縮性流れの数値流体力学がそれぞれ独自の計算手法に基づき進展してゆきます。

圧縮性流れ(Compressible Flow)と、非圧縮性流れ(Incompressible Flow)の違いは、空気と水の流れの違いと同じです(厳密には水も圧縮性を持っています)。風船の中に空気と水を入れて風船の外から力を加えると、空気の風船は縮みますが、水の風船は変形しますが縮みません。この縮む現象、すなわち、空気の密度は変化しますが、水の密度はほとんど変化しません。この密度の変化があるかないかは、数値流体力学において、決定的な計算手法の違いになって現れます。MAC法(Marker and Cell Method)と書かれ

た方法は、この後に非圧縮性流れの数値流体力学において標準的な計算手法となっています。本講義では、ナビエ・ストークス方程式を初めとする偏微分方程式に基づく数値モデルを解くための基礎に当たる偏微分方程式ならびに差分法について詳細に説明します。

3. 数値計算とは

数値計算はやった経験がないかもしれませんが、知らないうちに数値計算の恩恵にあずかっています。その最たるものは、ゲーム。アクションゲームやシミュレーションゲームは数値計算のかたまりのようなものです。他にも、自動車、パソコン、スマホ、SNSなど、それらのどこかで必ず数値計算が行われています。さらに、気象や津波予報から株価予測など身の回りに数値計算は欠かせません。

数値計算ではたしかに難しい数式が解かれるのですが、最終的にコンピュータの中ではそれらの数式は細かく分解されて四則演算になります。したがって、コンピュータがやっているのは小学校で勉強する算数レベルの計算です。ただし、その回数は1秒間に数億回から数兆回というものすごい回数です。数値計算は、難しい数式を四則演算に噛み砕いてコンピュータに計算させるための手段です。

日本にはあいにく、数学と工学を橋渡ししてくれる研究教育機関がまだまだ不足しています。数学は数学屋、工学は工学屋がやるのが当たり前という考え方がいまでも根強いからです。でも海外の大学では、数学科のなかに応用数学を専攻する数学者がおり、数値流体力学の基礎を研究している研究者もたくさんいます。彼らは応用先としてのたとえば数値流体力学を念頭においた研究をしています。応用数学では、偏微分方程式を教えます。計算数理学にとって偏微分方程式は必須です。これまで数学で勉強した微積分、代数幾何も極めて重要です。

4. 計算数理学の入口

4-1 反応方程式

e^{-t} を微分すると、 $-e^{-t}$ になります。 $u = e^{-t}$ とすれば、 $u' = -e^{-t}$ ですから、 $u' = -u$ になります。これは微分方程式 (Differential Equation) に相当します。正確には常微分方程式 (Ordinary Differential Equation、略して ODE) です。 $du/dt = -u$ とも書けます。

差分法は、数値流体力学や計算数理学の基礎として今でも広く用いられている計算手法の一つです。

いま、 $du/dt = -u$ を差分法で解いてみましょう。まずは次式を作ります。

$$\frac{(u^{n+1} - u^n)}{\Delta t} = -u^n \quad (4.1)$$

ここで、 n は時間ステップ(Time Step)と呼ばれます。べき乗ではありません。 Δt は n 時間ステップから $n+1$ 時間ステップまでの時間間隔、 u^n は n 時間ステップの u 、そして u^{n+1} は $n+1$ 時間ステップの u です。この式をさらに変形すると、

$$u^{n+1} = u^n - \Delta t u^n \quad (4.2)$$

これは、コンピュータプログラムの代入文のような式で、 n の値を $1, 2, 3, \dots$ と増やしながら繰り返し計算する式です。いま、 $n=0$ で、 $t=0$ を初期値として、 $\Delta t=0.1$ で繰り返し計算すれば、図4.1のような結果が得られます。

n	t	u	e^{-t}
0	0.0	1.0000	1.0000
1	0.1	0.9000	0.9048
2	0.2	0.8100	0.8187
3	0.3	0.7290	0.7408
4	0.4	0.6561	0.6703
5	0.5	0.5905	0.6065
6	0.6	0.5314	0.5488
7	0.7	0.4783	0.4966
8	0.8	0.4305	0.4493
9	0.9	0.3874	0.4066
10	1.0	0.3487	0.3679
11	1.1	0.3138	0.3329
12	1.2	0.2824	0.3012
13	1.3	0.2542	0.2725
14	1.4	0.2288	0.2466
15	1.5	0.2059	0.2231
16	1.6	0.1853	0.2019
17	1.7	0.1668	0.1827
18	1.8	0.1501	0.1653
19	1.9	0.1351	0.1496
20	2.0	0.1216	0.1353

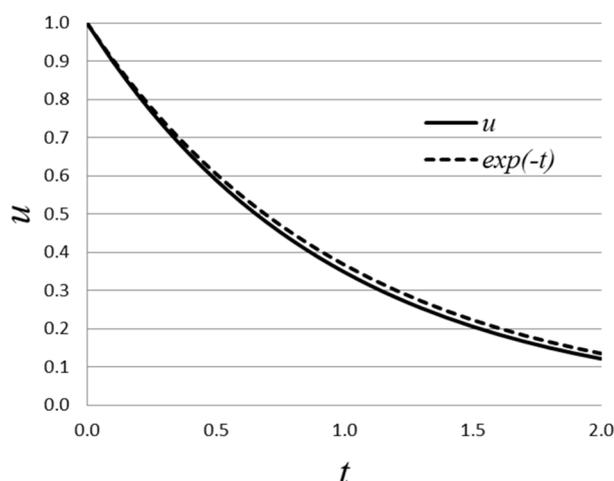


Fig. 4.1 e^{-t} の厳密解と差分法による解の比較.

厳密解である e^{-t} の値と差分法による数値解の比較からは、いずれの u もゼロに漸近していくのがわかります。もともと、 $u = e^{-t}$ であり、近似的にこの指数関数の解を差分法で求めることができることを示しています。

常微分方程式 $du/dt = -u$ は、数理科学の分野では、反応方程式 (Reaction Equation) と呼ばれます。反応とは、まさに化学反応です。 $du/dt = -u$ は t を十分大きな値にしたときゼロに漸近しますが、 u が何かの物質だとすれば、その物質が十分時間が経った後に別の物質に変化して、その物質自体はなくなってしまったことを模擬しています。その場合に、 t は時間そのものです。

いま、 $du/dt = u$ とすれば、この常微分方程式の一般解の一つは e^t になりますので、 t を十分大きくすると今度は u が爆発的に大きな値になります。ただし、このような化学反応はあまり現実的ではありません。爆発的に増えるものとしては、インフルエンザウイルスの増殖などが考えられます。伝染病の伝播を模擬した数理モデルもこれまでにいろいろ研究されており、この式の形が基本になります。数理科学では、反応方程式の

左辺を、時間微分項（Time-derivative term）、右辺を反応項（Reaction term）もしくは生成項（Source term）と呼びます。

4-2 反応方程式系

2つの反応方程式を同時に解いて、2つの物質の化学反応を模擬してみましょう。いま、物質 u と v があるとします。化学反応により、物質 u が物質 v に変化する数理モデルは、次の2つの反応方程式で簡単に模擬することができます。

$$\begin{aligned} \frac{du}{dt} &= -vu \\ \frac{dv}{dt} &= uv \end{aligned} \tag{4.3}$$

複数以上の反応方程式からなる数理モデルを、反応方程式系（System of Reaction Equations）と呼びます。これらから、差分法により、

$$\begin{aligned} u^{n+1} &= u^n - \Delta t v^n u^n \\ v^{n+1} &= v^n + \Delta t u^n v^n \end{aligned} \tag{4.4}$$

が作られます。

初期値として、 $t=0$ で、99%の物質が u で、残りの1%が v とすれば、 $\Delta t=0.5$ で実際に計算すると図 4.2 のような結果が得られます。

n	t	u	v
0	0.0	0.9900	0.0100
1	0.5	0.9851	0.0150
2	1.0	0.9777	0.0223
3	1.5	0.9668	0.0332
4	2.0	0.9507	0.0493
5	2.5	0.9273	0.0727
6	3.0	0.8936	0.1064
7	3.5	0.8460	0.1540
8	4.0	0.7809	0.2191
9	4.5	0.6954	0.3046
10	5.0	0.5895	0.4105
11	5.5	0.4685	0.5315
12	6.0	0.3440	0.6560
13	6.5	0.2311	0.7689
14	7.0	0.1423	0.8577
15	7.5	0.0813	0.9187
16	8.0	0.0439	0.9561
17	8.5	0.0229	0.9771
18	9.0	0.0117	0.9883
19	9.5	0.0059	0.9941
20	10.0	0.0030	0.9970

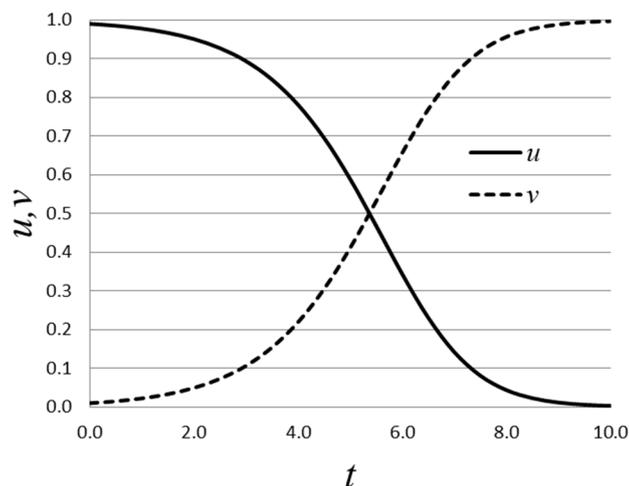


Fig. 4.2 u と v の数値解.

$n=11$ で、 u と v の値がほぼ逆転しているのがわかります。この計算をさらに続けると、 $u=0$ 、 $v=1$ に漸近します。

化学反応を例にして説明しましたが、何でもかまいません。何かが別の何かに変化することは、世の中にたくさんあります。そのような現象や事象が、2つの反応方程式により模擬できるわけです。ただし、もう少し複雑な式になるかもしれません。

次に、3つの反応方程式からなる典型的な数理モデルについて紹介します。冬に流行するインフルエンザは、感染が始まると急激に広がります。同時にある期間を経てその感染者は徐々に減少します。これを模擬した次のような Kermack-McKendrick モデル[2]-[4]、通称 SIR モデルが知られています。

$$\begin{aligned}\frac{dS(t)}{dt} &= -\beta S(t)I(t) \\ \frac{dI(t)}{dt} &= -\gamma I(t) + \beta S(t)I(t) \\ \frac{dR(t)}{dt} &= \gamma I(t)\end{aligned}\tag{4.5}$$

ここで、 $S(t)$ は非感染者の数 (Population under susceptible condition)、 $I(t)$ は感染者の数 (Population under infectious condition)、 $R(t)$ は回復者の数 (Population under recovered condition) を表し、それらの英語名で定義された頭文字をとって SIR モデルと呼ばれます。 β は感染率、 γ は回復率です。全人口は $N(t) = S(t) + I(t) + R(t)$ で与えられます。また、式(4.5)を足し合わせると

$$\frac{dS(t)}{dt} + \frac{dI(t)}{dt} + \frac{dR(t)}{dt} = 0\tag{4.6}$$

となります。式(4.5)を差分法により近似して展開すれば次式のようにになります。

$$\begin{aligned}S(t)^{n+1} &= S(t)^n - \Delta t \beta S(t)^n I(t)^n \\ I(t)^{n+1} &= I(t)^n - \Delta t \gamma I(t)^n + \Delta t \beta S(t)^n I(t)^n \\ R(t)^{n+1} &= R(t)^n + \Delta t \gamma I(t)^n\end{aligned}\tag{4.7}$$

いま、 $\Delta t = 1.0$ 、 $\beta = 0.4$ 、 $\gamma = 0.03$ とし、非感染者の初期値を 999 人： $S(0) = 999$ 、感染者数の初期値を 1 人： $I(0) = 1$ 、回復者の初期値をゼロ： $R(0) = 0$ とし、式(4.7)を解けば Fig.4.3 が得られます。これを見ると、30日くらいでほぼ全員が急激に感染しているのがわかります。その一方で、感染者数は25日くらいでピークを迎えてその後減少に転じており、100日以降では感染者数は全体の1割以下になっています。実際のインフルエンザの流行はこれよりも複雑ですが、ごく簡単な3つの反応方程式を解くだけで、感染者数の最大値が求められるところは興味深いものがあります。

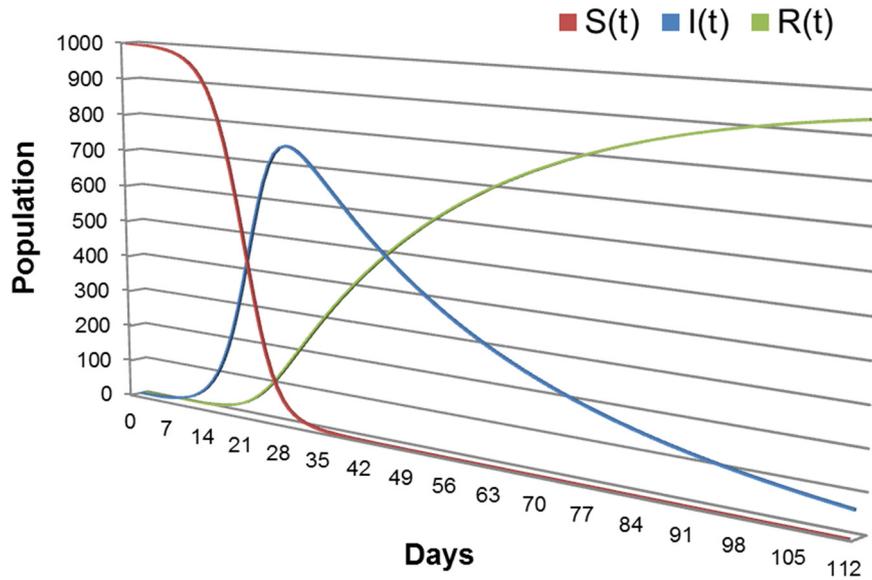


Fig. 4.3 非感染者、感染者、回復者の推移

4-3 偏微分方程式

反応方程式 $du/dt = -u$ の t は独立変数 (Independent Variable) でここでは時間です。一方、独立変数が2つ以上の微分方程式のこと、偏微分方程式、英語で Partial Differential Equation (略して PDE) といいます。たとえば、

$$\frac{\partial u}{\partial t} = \frac{\partial u}{\partial x} \quad (4.8)$$

は最も簡単な偏微分方程式の一つです。ただし、 ∂ は偏微分記号です。 $\partial u / \partial t$ と $\partial u / \partial x$ は偏導関数 (Partial Derivative Function) です。この式は、数値流体力学や計算数理学の分野では、移流方程式 (Convection Equation) に分類されます。

この方程式の最も簡単な解としては、たとえば $u = x + t$ があります。偏微分方程式を微分する際には、微分しようとする独立変数の項のみが微分され、それ以外のものは定数として取り扱われますので、 u を x で偏微分しても t で偏微分してもその値は同じ、すなわちこの場合には $\partial u / \partial t = \partial u / \partial x = 1$ になります。

式(4.8)を実際に解いてみましょう。簡単に説明するため厳密解は、 $u = x + t$ とします。差分法ではとりあえず

$$\frac{u^{n+1}(x) - u^n(x)}{\Delta t} = \frac{u^n(x + \Delta x) - u^n(x - \Delta x)}{2\Delta x} \quad (4.9)$$

と差分近似できます。 $u(x)$ は空間の1次元座標 x における u の値を示します。 $x \pm \Delta x$ は x から $\pm \Delta x$ だけずれたところの座標です。 x における偏導関数 $\partial u / \partial x$ は、2次精度の中

心差分 (Second-order Central Difference) というもので差分近似されています (後述します)。さらに変形しますと、

$$u^{n+1}(x) = u^n(x) + \frac{\Delta t}{2\Delta x} [u^n(x + \Delta x) - u^n(x - \Delta x)] \quad (4.10)$$

となり、この式を $n=1,2,3, \dots$ と繰り返して計算します。 $x=0$ から $x=1$ の領域で $\Delta x=0.1$ とし、 $t=0$ から $\Delta t=0.01$ で計算してみましよう。なお、 $x=0$ と $x=1$ の u に厳密解を与えることにします。まず厳密解の値を示しますと、

n	t	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1	x
1	0.01	0.01	0.11	0.21	0.31	0.41	0.51	0.61	0.71	0.81	0.91	1.01	
2	0.02	0.02	0.12	0.22	0.32	0.42	0.52	0.62	0.72	0.82	0.92	1.02	
3	0.03	0.03	0.13	0.23	0.33	0.43	0.53	0.63	0.73	0.83	0.93	1.03	
4	0.04	0.04	0.14	0.24	0.34	0.44	0.54	0.64	0.74	0.84	0.94	1.04	
5	0.05	0.05	0.15	0.25	0.35	0.45	0.55	0.65	0.75	0.85	0.95	1.05	
6	0.06	0.06	0.16	0.26	0.36	0.46	0.56	0.66	0.76	0.86	0.96	1.06	
7	0.07	0.07	0.17	0.27	0.37	0.47	0.57	0.67	0.77	0.87	0.97	1.07	
8	0.08	0.08	0.18	0.28	0.38	0.48	0.58	0.68	0.78	0.88	0.98	1.08	
9	0.09	0.09	0.19	0.29	0.39	0.49	0.59	0.69	0.79	0.89	0.99	1.09	
10	0.1	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1	1.1	

となります。次に差分法で解きますと、

n	t	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1	x
1	0.01	0.01	0.11	0.21	0.31	0.41	0.51	0.61	0.71	0.81	0.91	1.01	
2	0.02	0.02	0.12	0.22	0.32	0.42	0.52	0.62	0.72	0.82	0.92	1.02	
3	0.03	0.03	0.13	0.23	0.33	0.43	0.53	0.63	0.73	0.83	0.93	1.03	
4	0.04	0.04	0.14	0.24	0.34	0.44	0.54	0.64	0.74	0.84	0.94	1.04	
5	0.05	0.05	0.15	0.25	0.35	0.45	0.55	0.65	0.75	0.85	0.95	1.05	
6	0.06	0.06	0.16	0.26	0.36	0.46	0.56	0.66	0.76	0.86	0.96	1.06	
7	0.07	0.07	0.17	0.27	0.37	0.47	0.57	0.67	0.77	0.87	0.97	1.07	
8	0.08	0.08	0.18	0.28	0.38	0.48	0.58	0.68	0.78	0.88	0.98	1.08	
9	0.09	0.09	0.19	0.29	0.39	0.49	0.59	0.69	0.79	0.89	0.99	1.09	
10	0.1	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1	1.1	

になります。解はまったく同じになりました。

4-4 拡散方程式

拡散 (Diffusion) とは、読んで字のごとくで拡がり散らばることです。物理学ではそれを拡散現象といいます。匂い、流行、うわさなどいろいろなものが拡散します。拡散現象を模擬する数理モデルは拡散方程式 (Diffusion Equation) と呼ばれます。拡散現象は、数学的には2階 (2回微分している) の偏導関数 $\partial^2 u / \partial x^2$ で表現されます。

いま代表的な拡散方程式として、

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} \quad (4.11)$$

があります。これは時間に依存した拡散現象を模擬するための方程式です。拡散の典型例として、熱の伝わり方、すなわち熱伝導 (Heat Conduction) があります。そんなことから、式(4.11)は熱伝導方程式 (Equation of Heat Conduction) とも呼ばれます。正確には右辺に係数がついて、

$$\frac{\partial u}{\partial t} = \kappa \frac{\partial^2 u}{\partial x^2} \quad (4.12)$$

のように定義されます。この係数は、熱伝導係数 (Coefficient of Heat Conduction) です。熱の伝わり方は物質ごとに異なりますので、この係数の値は物質ごとに違った値になります。金属と空気を比較すれば、金属の方がより熱が伝わりやすく、この係数の値は空気に比べて2桁近く大きな値になります。

この方程式を差分法で近似すると、

$$\frac{u^{n+1}(x) - u^n(x)}{\Delta t} = \kappa \frac{u^n(x + \Delta x) - 2u^n(x) + u^n(x - \Delta x)}{(\Delta x)^2} \quad (4.13)$$

が得られます。2階の偏導関数を2次精度の中心差分で近似すると、いま計算しようとする x における $u^n(x)$ の値と、そのとなりの $u^n(x + \Delta x)$ と $u^n(x - \Delta x)$ にそれぞれ-2, 1, 1を掛けて足し合わせ、それを $(\Delta x)^2$ で割った式になります。さらに変形すれば、

$$u^{n+1}(x) = u^n(x) + \frac{\kappa \Delta t}{(\Delta x)^2} [u^n(x + \Delta x) - 2u^n(x) + u^n(x - \Delta x)] \quad (4.14)$$

が得られます (詳しくは後述します)。

いま長さ10で温度10度の任意の物質を考えます。左端 $x=0$ を10度で冷やし、右端 $x=10$ を100度で熱すると、 $x=10$ から熱は内部に徐々に伝わってきます。 $\Delta x = 1.0$, $\Delta t = 0.5$ として、 $n = 10$ まで計算した結果をグラフにしますと、

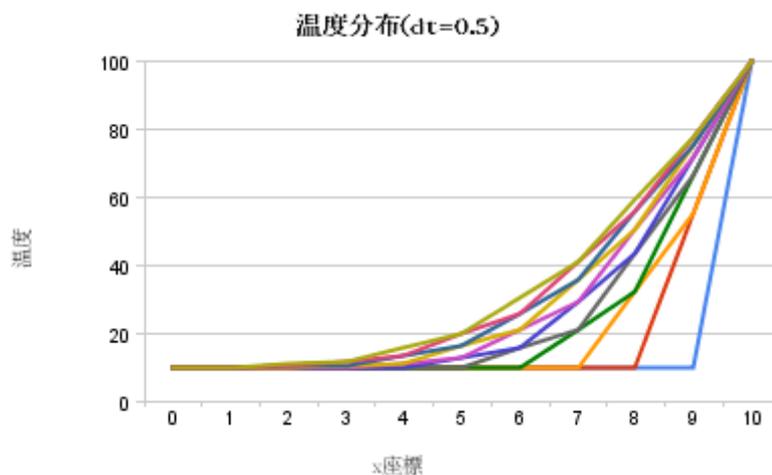


Fig. 4.4 熱伝導方程式の数値解 ($\Delta t = 0.5$)

が得られました。物質の内部に熱が伝わり、温度が徐々に上がっていく様子が示されています。もっと計算を繰り返すと、最終的には $x=0$ の 10 度と、 $x=10$ の 100 度を直線でつないだような温度分布になります。

ところで、同じ計算を $\Delta t = 0.6$ で再度計算してみると次のような結果が得られました。

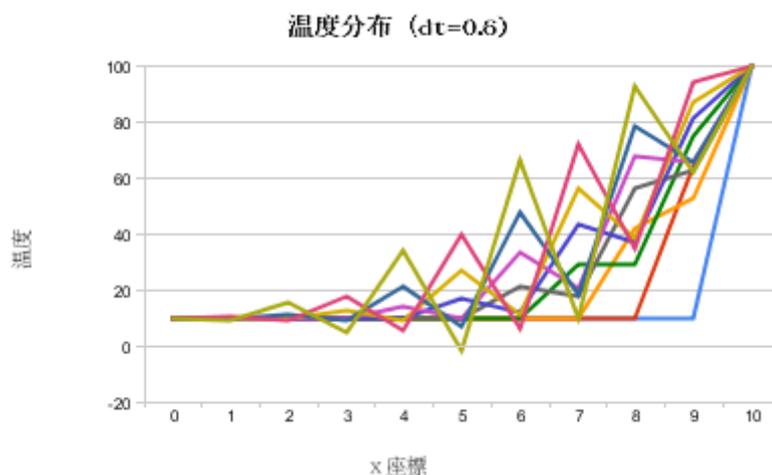


Fig. 4.5 熱伝導方程式の数値解 ($\Delta t = 0.6$)

温度が波打っていますが、現実的にこのようなことは起こりません。これは数値振動 (Numerical Oscillation) と呼ばれる数値計算特有の解の振動です。非物理的な値ですのおかした計算をしていることになります。移流方程式のところでも紹介したように、線形安定性理論から、 $\Delta t = 0.5$ では安定、 $\Delta t = 0.6$ では不安定になります。たった 0.1 の違いですが、数値計算の世界では決定的な差となって現れます。

4-5 偏微分方程式の型の分類

これまでに、偏微分方程式として移流方程式ならびに拡散方程式を紹介しました。これらの偏微分方程式はその特徴から、大きく3種類に分類できます。

いま、次の二次元一般形で表された2階の偏微分方程式を考えます。

$$A \frac{\partial^2 u}{\partial x^2} + 2B \frac{\partial^2 u}{\partial x \partial y} + C \frac{\partial^2 u}{\partial y^2} = f \left(x, y, u, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y} \right) \quad (4.15)$$

ここで、 A, B ならびに C は高々 x, y の関数で、右辺の f は高々1階の偏導関数からなる関数とします。

$D = B^2 - AC$ としますと、 D の符号に応じて、

$D > 0$ なら、双曲型 (Hyperbolic)

$D = 0$ なら、放物型 (Parabolic)

$D < 0$ なら、楕円型 (Elliptic)

という型に分類されます。なお、型の分類の詳細は少し複雑な話ですので、ここでは省略します。

それぞれの型には典型的な式の形があり、

双曲型では、

$$\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} = 0 \quad (4.16)$$

放物型では、

$$\frac{\partial^2 u}{\partial x^2} = \frac{\partial u}{\partial y} \quad (4.17)$$

楕円型では、

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \quad (4.18)$$

などです。これらの式は標準形と呼ばれます。

双曲型の式は、計算数理学では波動方程式 (Wave Equation) といいます。放物型の式はよく見ますと、 y が t になれば、熱伝導方程式と同じであることがわかります。そして、楕円型の式は、ラプラス方程式 (Laplace Equation) といいます。また、この式の右辺がゼロでない場合には、ポアソン方程式 (Poisson Equation) といいます。これらの式はいずれも、物理現象を模擬する上でたいへん有用な式です。

4-6 ラプラス方程式の解析解

楕円型方程式の標準形であるラプラス方程式の解析解を求めてみましょう。

いま、次の図で示した正方形領域について、各辺にそれぞれ境界条件 (Boundary Condition) を与えます。

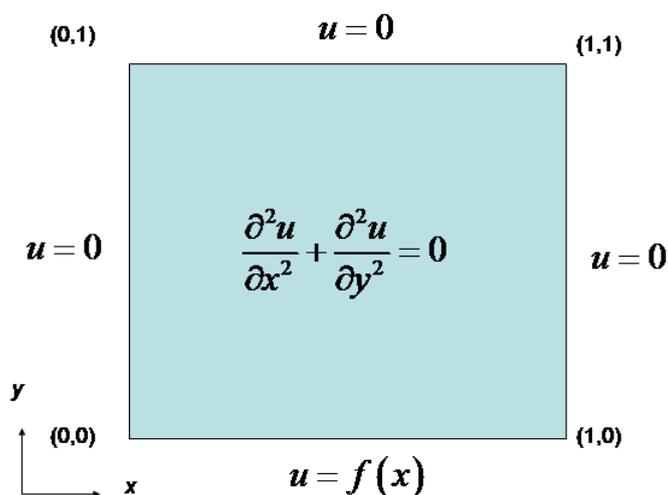


Fig. 4.6 ラプラス方程式の境界条件

この問題は、典型的な偏微分方程式の計算問題で、変数分離法と呼ばれる方法により解析解を求めることができます。計算の詳細についてはここでは省略しますが、解析解は次のように得られます。

$$u(x, y) = 2 \sum_{n=1}^{\infty} \frac{\sinh[n\pi(y-1)]}{\sinh(-n\pi)} \sin(n\pi x) \int_0^1 f(x) \sin(n\pi z) dz \quad (4.19)$$

ただし、 $\sinh(y) = (e^y - e^{-y})/2$ です。かなり複雑な式になっていますが、図 4.6 に示した領域内の任意の点の解が求まりますので、厳密解です。

このように、ラプラス方程式の解析解は極めて限られた場合には紙とえんぴつで求めることができます。ただし、仮に上記の図で示した境界条件（境界に与えた値）の内、 $u = 0$ のひとつでも 0 以外の値にしますと、途端に解析解を求めるのが難しくなります。もしくは、もはや求めること自体できなくなります。そのときは、コンピュータにより数値計算するしかありません。

4-7 ラプラス方程式の差分法

ラプラス方程式は二次元で次式になります。

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \quad (4.20)$$

これを中心差分で近似すると、

$$\frac{u(x + \Delta x, y) - 2u(x, y) + u(x - \Delta x, y)}{(\Delta x)^2} + \frac{u(x, y + \Delta y) - 2u(x, y) + u(x, y - \Delta y)}{(\Delta y)^2} = 0 \quad (4.21)$$

が得られます。簡単のため、 $\Delta x = \Delta y$ にして、この式をさらに変形すると、

$$u(x, y) = \frac{1}{4} [u(x + \Delta x, y) + u(x - \Delta x, y) + u(x, y + \Delta y) + u(x, y - \Delta y)] \quad (4.22)$$

が得られます。これは、自分自身の値を隣接する4点を足して4で割って求めるという単純な計算式です。ただし、具体的には次式を繰り返し計算します。

$$u(x, y)^{n+1} = \frac{1}{4} \left[u(x + \Delta x, y)^n + u(x - \Delta x, y)^n + u(x, y + \Delta y)^n + u(x, y - \Delta y)^n \right] \quad (4.23)$$

いま、ラプラス方程式の解析解で紹介した問題の底辺の境界条件を $f(x)=1$ として実際に Excel で計算しますと次のような計算結果を得ることができます。

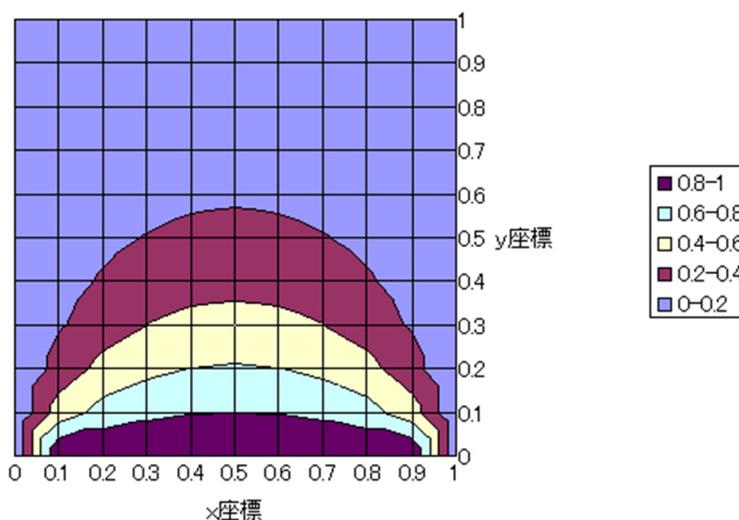


Fig. 4.7 二次元ラプラス方程式の数値解

境界条件が底辺角の点で不連続のため解が多少波打っていますが、本来は連続的な境界条件を与える必要があります。Excel による計算方法については後述します。

4-8 テイラー展開

x 座標上の任意の点における未知変数 $u(x)$ の微分 du/dx はもともと

$$\frac{du}{dx} = \lim_{\Delta x \rightarrow 0} \frac{u(x + \Delta x) - u(x)}{\Delta x} \quad (4.24)$$

で定義されます。ただし、未知変数は図 4.8 のように設定されます

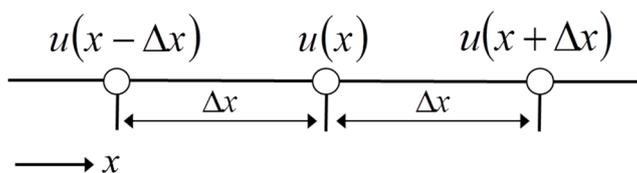


Fig. 4.8 未知変数 $u(x)$ の定義

式(4.24)は、テイラー展開 (Taylor's Expansion) により漸化式に置き換えることができま

す。まず、 $u(x+\Delta x)$ をテイラー展開すると次式ようになります。

$$u(x+\Delta x) = u(x) + \Delta x \frac{du(x)}{dx} + \frac{(\Delta x)^2}{2!} \frac{d^2u(x)}{dx^2} + \dots + \frac{(\Delta x)^{n-1}}{(n-1)!} \frac{d^{n-1}u(x)}{dx^{n-1}} + \frac{(\Delta x)^n}{n!} \frac{d^n u(\xi)}{dx^n}$$
(4.25)

ただし、 $x < \xi < x + \Delta x$. この式から、 du/dx を逆算すれば、

$$\frac{du(x)}{dx} = \frac{u(x+\Delta x) - u(x)}{\Delta x} - \frac{\Delta x}{2!} \frac{d^2u(x)}{dx^2} - \dots - \frac{(\Delta x)^{n-2}}{(n-1)!} \frac{d^{n-1}u(x)}{dx^{n-1}} - \frac{(\Delta x)^{n-1}}{n!} \frac{d^n u(\xi)}{dx^n}$$
(4.26)

が得られます。この式の右辺には、第1項に差分式があり、第2項以降には高階（微分回数が多）の微分項が付加されています。たとえば、第2項以降をすべて無視した場合には、打ち切り誤差 (Truncation Error) が1次精度で $O(\Delta x)$ となり次式に帰着します。

$$\frac{du(x)}{dx} = \frac{u(x+\Delta x) - u(x)}{\Delta x} + O(\Delta x)$$
(4.27)

$O(\Delta x)$ は実際には値はありません。これは1次精度差分近似式 (First-order Finite-difference Approximation) になります。同様に、 $u(x-\Delta x)$ についても式(4.26)に基づき、簡単にテイラー展開することができます。

$$u(x-\Delta x) = u(x) + (-\Delta x) \frac{du(x)}{dx} + \frac{(-\Delta x)^2}{2!} \frac{d^2u(x)}{dx^2} + \dots + \frac{(-\Delta x)^{n-1}}{(n-1)!} \frac{d^{n-1}u(x)}{dx^{n-1}} + \frac{(-\Delta x)^n}{n!} \frac{d^n u(\xi)}{dx^n}$$
(4.28)

式(4.25) の $u(x+\Delta x)$ と式(4.28)の $u(x-\Delta x)$ を用いると、次のように2次精度の差分近似式を導出することもできます。

$$\begin{aligned} \frac{du(x)}{dx} &= \frac{u(x+\Delta x) - u(x-\Delta x)}{2\Delta x} - \frac{(\Delta x)^2}{12} \frac{d^3u}{dx^3} + \dots \\ &= \frac{u(x+\Delta x) - u(x-\Delta x)}{2\Delta x} + O[(\Delta x)^2] \end{aligned}$$
(4.29)

この式は、2次精度中心差分近似式 (Second-order Central-difference Approximation) になります。

さて、ここで式を簡略化して表記するため、表記方法を変更したいと思います。差分法では、空間を有限個の点に置き換えて、その点上のみで計算されます。またこの点のことを格子点 (Grid Point) と呼びます。図 4.9 では、格子点に格子番号 j を割り当てて、その j 格子点上の未知変数を u_j と定義します。

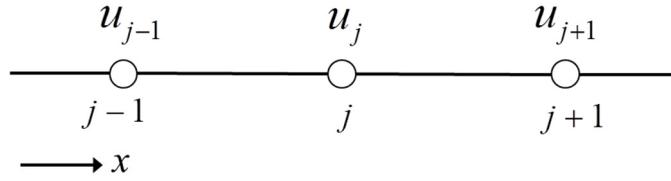


Fig. 4.9 格子点と未知変数の定義

du/dx の j 格子点における 1 次精度差分近似は、その差分点の取り方により、前進差分 (Forward Difference) もしくは後退差分 (Backward Difference) で次のように定義されます。

$$\left(\frac{du}{dx}\right)_j = \frac{u_{j+1} - u_j}{\Delta x}, \quad \left(\frac{du}{dx}\right)_j = \frac{u_j - u_{j-1}}{\Delta x} \quad (4.30)$$

また、同様に 2 次精度中心差分近似は、次のように再定義されます。

$$\left(\frac{du}{dx}\right)_j = \frac{u_{j+1} - u_{j-1}}{2\Delta x} \quad (4.31)$$

さらに、2 階微分 d^2u/dx^2 は 2 次精度中心差分で次のように近似されます。

$$\left(\frac{d^2u}{dx^2}\right)_j = \frac{u_{j+1} - 2u_j + u_{j-1}}{(\Delta x)^2} \quad (4.32)$$

以下にはこの式を簡単に導出してみます。もともと

$$\left(\frac{d^2u}{dx^2}\right)_j = \left[\frac{d}{dx}\left(\frac{du}{dx}\right)\right]_j \quad (4.33)$$

ですので、まず du/dx を先に差分します。便宜上、図 4.10 のように仮の中間格子点 $j \pm 1/2$ を定義して、その格子点上の値を $(du/dx)_{j \pm 1/2}$ とします。

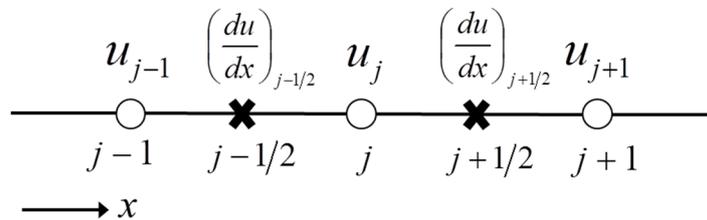


Fig. 4.10 中間格子点 $j \pm 1/2$ の定義

したがって、式(4.33)は 2 次精度中心差分でまずは次式のように近似されます。

$$\left(\frac{d^2u}{dx^2}\right)_j = \frac{\left(\frac{du}{dx}\right)_{j+1/2} - \left(\frac{du}{dx}\right)_{j-1/2}}{\Delta x} \quad (4.34)$$

一方、 $(du/dx)_{j\pm 1/2}$ は中間格子点 $j\pm 1/2$ で次のように 2 次精度中心差分近似します。

$$\left(\frac{du}{dx}\right)_{j+1/2} = \frac{u_{j+1} - u_j}{\Delta x}, \quad \left(\frac{du}{dx}\right)_{j-1/2} = \frac{u_j - u_{j-1}}{\Delta x} \quad (4.35)$$

式(4.35)を式(4.34)に代入すれば、式(4.32)が導出されます。

4-9 境界条件

ラプラス方程式の解析解のところでも示したように、楕円型方程式の差分計算には、境界条件が欠かせません。そのため、楕円型方程式の問題は境界値問題 (Boundary Value Problem) と呼ばれます。未知変数である u そのものを規定する境界条件のことを、ディリクレ境界条件 (Dirichlet's Boundary Condition)、もしくは第 1 種境界条件と呼びます。これらの点の値はすでに求まっていますから、改めて計算する必要はありません。一方、未知変数 u の 1 階微分を規定する境界条件のことを、ノイマン境界条件 (Neumann's Boundary Condition)、もしくは第二種境界条件と呼びます。ノイマンとは、先に紹介した von Neumann です。 u 自体の値は未知ですので計算により求める必要があります。たとえば、図 4.11 の格子点 $j=1$ は境界上にあります。ここにノイマン境界条件 $(\partial u/\partial x)_1 = 0$ が与えられている場合、 u_1 は未知です。したがって、この点においても、差分近似式を解かなければなりません。ところが、式(4.32)の 2 次精度の中心差分では u_{j-1} が図 4.11 では定義できないため、そのまま使えません。これを解決するためには、内側の格子点のみを用いた差分式を別途導出する必要があります。

いま、図 4.11 のように、中間格子点 $1+1/2$ で仮に $(du/dx)_{1+1/2}$ を定義すれば、境界上における $(d^2u/dx^2)_1$ は次式のように片側から差分近似できます。

$$\left(\frac{d^2u}{dx^2}\right)_1 = \frac{\left(\frac{du}{dx}\right)_{1+1/2} - \left(\frac{du}{dx}\right)_1}{\Delta x/2} \quad (4.36)$$

ここで、

$$\left(\frac{du}{dx}\right)_{1+1/2} = \frac{u_2 - u_1}{\Delta x}, \quad \left(\frac{du}{dx}\right)_1 = 0 \quad (4.37)$$

ですから、式(4.36)は最終的に次式のような差分近似式として導出されます。

$$\left(\frac{d^2u}{dx^2}\right)_1 = \frac{2(u_2 - u_1)}{(\Delta x)^2} \quad (4.38)$$

ただし、式(4.36)は1次精度であるため、ここで紹介した境界条件は、1次精度の差分により計算されます。

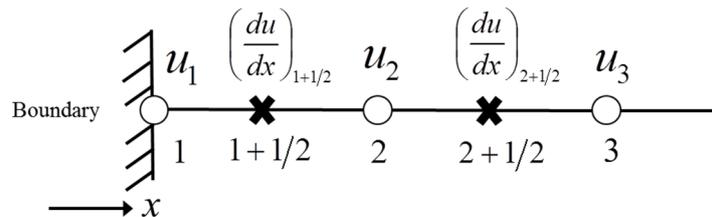


Fig. 4.11 境界およびその近傍格子点の定義

4-10 例題：ラプラス方程式を手計算

ラプラス方程式の解析解ならびに差分法について紹介してきました。ラプラス方程式はたいへん汎用性のある方程式で、多くの物理現象を模擬することができます。

ここでは、差分法による数値計算がより身近に感じるように、手計算で解く方法を紹介します。

まず具体的な問題を設定します。図 4.12 にあるような格子点がたった9点の問題を考えます。

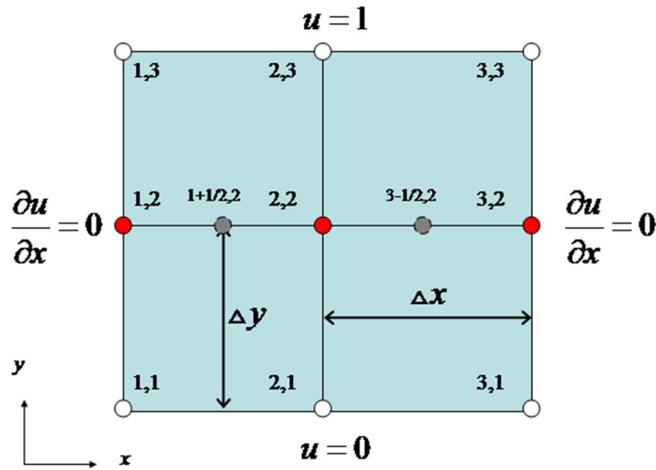


Fig. 4.12 ラプラス方程式問題の計算格子と境界条件

たった9点の問題でも立派なラプラス方程式の差分計算ができます。計算格子点にはそれぞれ番号*i, j*をつけています。ただし、ここでは*i=1,2,3, j=1,2,3*です。

境界条件として、

上辺にある(1,3), (2,3), (3,3)には、 $u = 1$

下辺にある(1,1), (2,1), (3,1)には、 $u = 0$

を与えます。一方、左辺と右辺にある(1,2)ならびに(3,2)には、 $\partial u / \partial x = 0$ を与えます。

次に、差分近似式を導出します。この問題で、 u が未知な計算格子点は(1,2), (2,2), (3,2)の3点です。いま、式(4.20)で示したラプラス方程式を、次のように2次精度差分近似します。

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{(\Delta x)^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{(\Delta y)^2} = 0$$

さらに簡単にするため、 $\Delta x = \Delta y = 1$ と仮定すれば、格子点(2,2)における差分近似式は、

$$u_{3,2} - 2u_{2,2} + u_{1,2} + u_{2,3} - 2u_{2,2} + u_{2,1} = 0$$

が求まります。同じ項で整理すれば、

$$u_{3,2} + u_{1,2} + u_{2,3} + u_{2,1} - 4u_{2,2} = 0$$

一方、格子点(1,2), (3,2)についてはノイマン境界条件が与えられた境界上の格子点になるため、片側差分近似しなければなりません。格子点(1,2)における d^2u/dx^2 の片側差分近似式は、まず1階の偏導関数を用いて次式のように定義されます。

$$\left(\frac{d^2u}{dx^2}\right)_{1,2} = \frac{\left(\frac{du}{dx}\right)_{1+1/2,2} - \left(\frac{du}{dx}\right)_{1,2}}{\Delta x/2}$$

ここで、 $\Delta x=1$ でしたので、

$$\left(\frac{d^2u}{dx^2}\right)_{1,2} = 2\left[\left(\frac{du}{dx}\right)_{1+1/2,2} - \left(\frac{du}{dx}\right)_{1,2}\right]$$

になります。さらに、もともと境界条件として、 $(\partial u/\partial x)_{1,2} = 0$ でしたので、

$$\left(\frac{d^2u}{dx^2}\right)_{1,2} = 2\left(\frac{du}{dx}\right)_{1+1/2,2}$$

になります。 $(\partial u/\partial x)_{1+1/2,2} = (u_{2,2} - u_{1,2})/\Delta x = u_{2,2} - u_{1,2}$ ですから、結局

$$\left(\frac{d^2u}{dx^2}\right)_{1,2} = 2(u_{2,2} - u_{1,2})$$

と片側差分近似式が導出されます。

同様に、格子点(3,2)では、

$$\left(\frac{d^2u}{dx^2}\right)_{3,2} = 2\left[\left(\frac{du}{dx}\right)_{3,2} - \left(\frac{du}{dx}\right)_{3-1/2,2}\right]$$

となります。境界条件として $(\partial u/\partial x)_{3,2} = 0$ なので、結局

$$\left(\frac{d^2u}{dx^2}\right)_{3,2} = -2(u_{3,2} - u_{2,2})$$

が得られます。

これより、格子点(1,2)における差分近似式は

$$2(u_{2,2} - u_{1,2}) + u_{1,3} - 2u_{1,2} + u_{1,1} = 0$$

となり、同じ項をまとめれば、

$$-4u_{1,2} + u_{1,3} + u_{1,1} + 2u_{2,2} = 0$$

が得られます。

同様に、格子点(3,2)における差分近似式は

$$-2(u_{3,2} - u_{2,2}) + u_{3,3} - 2u_{3,2} + u_{3,1} = 0$$

で、これを整理すれば、

$$-4u_{3,2} + u_{3,3} + u_{3,1} + 2u_{2,2} = 0$$

となります。

格子点(1,2), (2,2), (3,2) における差分近似式はそれぞれ、

$$-4u_{1,2} + u_{1,3} + u_{1,1} + 2u_{2,2} = 0$$

$$u_{3,2} + u_{1,2} + u_{2,3} + u_{2,1} - 4u_{2,2} = 0$$

$$-4u_{3,2} + u_{3,3} + u_{3,1} + 2u_{2,2} = 0$$

と求められました。

ところで、格子点(1,3),(2,3),(3,3)では、 $u = 1$ 、格子点(1,1),(2,1),(3,1)では、 $u = 0$ から、

$$-4u_{1,2} + 1 + 0 + 2u_{2,2} = 0$$

$$u_{3,2} + u_{1,2} + 1 + 0 - 4u_{2,2} = 0$$

$$-4u_{3,2} + 1 + 0 + 2u_{2,2} = 0$$

結局、これら 3つの式からなる連立 1 次方程式を解けば、

$$u_{1,2} = u_{2,2} = u_{3,2} = 0.5$$

となり、ラプラス方程式を手計算で解くことができました。ラプラス方程式の差分法による数値計算は、それをひも解けば、連立 1 次方程式の計算に最終的には帰着されます。ここでの問題はたった 9 点の差分計算でしたが、考え方は点の数が 1 億点になってもまったく同じです。ただし、1 億点では手計算できませんので、代わりにコンピュータに計算を任せることとなります。実は、物理的なセンスがある人は、この問題は解く前に答えがわかります。いま、無限に広がった一様な厚さのある金属平板を想像してみてください。上面を温度 1°C 、下面を温度 0°C としたならば、金属平板の厚さ方向中心部の温度は何度になるのでしょうか？ よほど不均一な物質でない限り中間の温度になるはずです。すなわち、 0.5 度です。ノイマン境界条件として与えた $\partial u / \partial x = 0$ は、 x 方向に u の勾配がないということですので、 x 方向に u は同じ値になります。

5 数理モデルの数値計算法

5-1 ラプラス方程式の反復解法

ラプラス方程式の差分近似式を、任意の計算格子点 i, j に対して改めて定義すれば、

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{(\Delta x)^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{(\Delta y)^2} = 0 \quad (5.1)$$

と記述することができます。ここで、 $\Delta x = \Delta y$ とすれば、結局、

$$u_{i,j} = \frac{1}{4}(u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1}) \quad (5.2)$$

になります。

計算格子点 i, j における $u_{i,j}$ は、近接する4つの計算格子点における $u_{i+1,j}$, $u_{i-1,j}$, $u_{i,j+1}$ ならびに $u_{i,j-1}$ を足して4で割ることにより求めるという単純な四則演算式です。

この式にさらに、反復回数(Iteration Number)として n を当てはめた式は、

$$u_{i,j}^{n+1} = \frac{1}{4}(u_{i+1,j}^n + u_{i-1,j}^n + u_{i,j+1}^n + u_{i,j-1}^n) \quad (5.3)$$

と定義されます。この式は、 n の値を 1, 2, 3, ... と増やして反復計算することにより、 $u_{i,j}^{n+1}$ の収束解を求める式で、このような反復計算する方法のことを反復法

(Relaxation Method)と呼びます。この反復式はそれを考えた研究者の名前で、ヤコビ法 (Jacobi Method)とも呼ばれ、極めて単純な式ですが立派な反復法の一つです。ヤコビ

法は、 $u_{i,j}^{n+1}$ を求めるために周りの4点から計算します。 $n+1$ 反復を計算するとき、 n 反復の値は既知ですので、任意の i, j 点における計算はすべて独立に、いわゆる並列計算が可能です。すべての計算格子点に CPU を割り当てて計算すれば、1反復計算は1回の計算で済んでしまいます。ただし、ここでは並列計算は考えないことにします。

ヤコビ法は、式(5.3)をひたすら繰り返して、最終的に $n+1$ 反復と n 反復の値が同じになれば、解が求まったと判断されます。しかしながら、十分な反復回数が必要であることが知られています。この反復回数を減らすことができる反復法として、ガウス・ザイデル法 (Gauss-Seidel Method)があります。ガウス・ザイデル法では、図 5.1 に示すような、ハイパーライン (Hyper Line)と呼ばれる掃引により計算します。すなわち、図中のハイパーラインに付いた大きな矢印の方向に向かって計算します。すると、ハイパーラインが通過した点での値は更新されますから、 $n+1$ 反復の値になります。

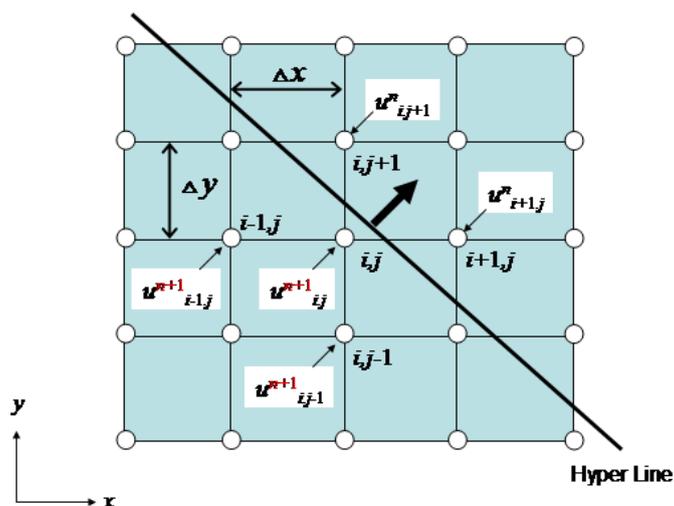


Fig. 5.1 ガウス・ザイデル法による計算方法

これを式で記述すれば、

$$u_{i,j}^{n+1} = \frac{1}{4} (u_{i+1,j}^n + u_{i-1,j}^{n+1} + u_{i,j+1}^n + u_{i,j-1}^{n+1}) \quad (5.4)$$

となります。すでに更新された点の値を有効活用することにより、反復回数を約半分に減らすことができます。

ガウス・ザイデル法の反復回数はさらに短縮することができます。いま、ガウス・ザイデル法の $u_{i,j}^{n+1}$ を、 $(u_{i,j}^{n+1})_{GS}$ とおいてその計算式を導出すれば、

$$u_{i,j}^{n+1} = \omega (u_{i,j}^{n+1})_{GS} + (1-\omega) u_{i,j}^n \quad (5.5)$$

と定義されます。この式は、 n 反復における $u_{i,j}^n$ とガウス・ザイデル法で求めた値を

緩和係数 ω による線形結合により $u_{i,j}^{n+1}$ を求めるという形になっています。線形結合において ω は通常、 $0 < \omega < 1$ なのですが、ここでの ω は $1 < \omega$ になります。それゆえ、このことを過緩和係数(Over-relaxation Parameter)と呼びます。 $1 < \omega$ にするという事は、ガウス・ザイデル法の値を過大評価することを意味します。理論的に $\omega < 2$ ですが、経験的に ω は 1.5 近傍の値をとります。仮に $\omega = 1.5$ とすれば、この方法はヤコビ法に比べて反復回数が 3 分の 1 程度で済むこととなります。この方法のことを、SOR 法 (Successive Over-relaxation Method)[5] と呼びます。したがって、ラプラス方程式の差分法には SOR 法が最も広く用いられています。

5-2 三次元ポテンシャル流れのSOR解法

三次元のポテンシャル流れをSOR法で解いてみます。ポテンシャル流れとは、流れの速度成分 u , v , w が、ポテンシャル ϕ の微分で与えられると仮定した流れで次のように定義されます。

$$u = \frac{\partial \phi}{\partial x}, \quad v = \frac{\partial \phi}{\partial y}, \quad w = \frac{\partial \phi}{\partial z} \quad (5.6)$$

非圧縮性流れを支配する方程式には、連続の式、運動方程式がありますが、このうち、連続の式は次式で定義されます。

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \quad (5.7)$$

この式にポテンシャルの微分で定義された速度成分を代入すれば、

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} + \frac{\partial^2 \phi}{\partial z^2} = 0 \quad (5.8)$$

となり、三次元ラプラス方程式に帰着します。ところで、三次元非圧縮性流れの渦度はベクトルで

$$\left(\frac{\partial w}{\partial y} - \frac{\partial v}{\partial z}, \quad \frac{\partial u}{\partial z} - \frac{\partial w}{\partial x}, \quad \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right)$$

ですが、ここにポテンシャルの微分で定義された速度を代入すると、すべての成分がゼロになります。すなわち、ポテンシャル流れは渦なしを仮定した流れです。ポテンシャル流れを支配する三次元ラプラス方程式を差分近似すると

$$\frac{\phi_{i+1,j,k} - 2\phi_{i,j,k} + \phi_{i-1,j,k}}{(\Delta x)^2} + \frac{\phi_{i,j+1,k} - 2\phi_{i,j,k} + \phi_{i,j-1,k}}{(\Delta y)^2} + \frac{\phi_{i,j,k+1} - 2\phi_{i,j,k} + \phi_{i,j,k-1}}{(\Delta z)^2} = 0 \quad (5.9)$$

になります。 z 方向に新たに計算格子点 k を用います。さらに、 $\phi_{i,j,k}$ の式に変形すれば、

$$\phi_{i,j,k} = \frac{1}{L} \left[\frac{\phi_{i+1,j,k} + \phi_{i-1,j,k}}{(\Delta x)^2} + \frac{\phi_{i,j+1,k} + \phi_{i,j-1,k}}{(\Delta y)^2} + \frac{\phi_{i,j,k+1} + \phi_{i,j,k-1}}{(\Delta z)^2} \right] \quad (5.10)$$

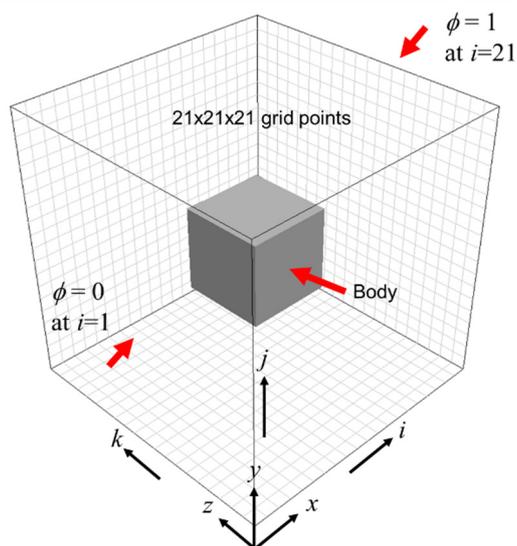
になります。ただし、 $L = 2/(\Delta x)^2 + 2/(\Delta y)^2 + 2/(\Delta z)^2$ 。これにSOR法を適用すれば、

$$\phi_{i,j,k}^{n+1} = (1-\omega)\phi_{i,j,k}^n + \frac{\omega}{L} \left[\frac{\phi_{i+1,j,k}^n + \phi_{i-1,j,k}^{n+1}}{(\Delta x)^2} + \frac{\phi_{i,j+1,k}^n + \phi_{i,j-1,k}^{n+1}}{(\Delta y)^2} + \frac{\phi_{i,j,k+1}^n + \phi_{i,j,k-1}^{n+1}}{(\Delta z)^2} \right] \quad (5.11)$$

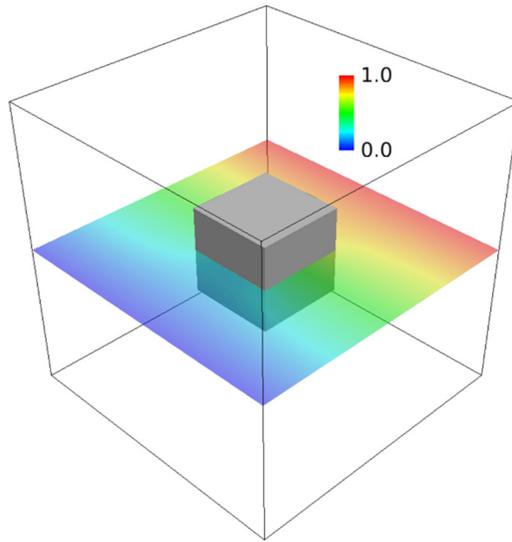
が得られます。この式を反復計算することにより、三次元ポテンシャル流れを計算することができます。

図 5.2(a)に示すように、立方体周りのポテンシャル流れを具体的に計算してみます。格子点数は $21 \times 21 \times 21$ とし、立方体はその表面がちょうど、 $i = j = k = 8$ 、ならびに $i = j = k = 14$ に位置するように設定します。境界条件は、入口境界 $i = 1$ 面で $\phi = 0$ 、出口境界 $i = 21$ 面で $\phi = 1$ を与え、それ以外の外部境界および立方体表面には、法線方向に $\partial\phi/\partial n = 0$ となるようノイマン境界条件を与えます。なお、格子間隔は $\Delta x = \Delta y = \Delta z = 0.1$ で、過緩和係数は1.5とします。SOR法で反復計算しますと、図 5.2(b)に示すようなポテンシャル ϕ が求められます。さらにこのポテンシャルを各座標方向に1階偏微分したものが、それぞれの方向の流速になりますから、 ϕ の値を単純に差分近似すれば、図 5.2(c)に示すような流速ベクトルを求めることができます。なお、ここで紹介した三次元ポテンシャル流れの計算プログラムは Fortran で作成され、研究室ホームページに掲載してあります。

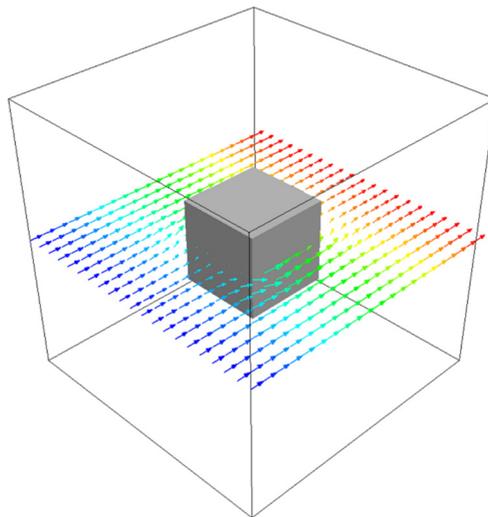
<http://www.caero.mech.tohoku.ac.jp/publicData/Numerical/index.html>



(a) 計算格子と境界条件



(b) 計算により得られたポテンシャル分布



(c) 計算により得られた速度分布

Fig. 5.2 三次元ポテンシャル流れの数値解

5-3 熱伝導方程式の差分解法

1次元熱伝導方程式を改めて次式のように定義します。

$$\frac{\partial u}{\partial t} = \kappa \frac{\partial^2 u}{\partial x^2} \quad (5.12)$$

これを差分近似すれば、

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \kappa \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{(\Delta x)^2} \quad (5.13)$$

となり、さらに、

$$u_j^{n+1} = u_j^n + \ell (u_{j+1}^n - 2u_j^n + u_{j-1}^n) \quad (5.14)$$

と変形できます。

先に、この式を表計算ソフトで解き、計算結果をグラフで可視化しましたが、そのときに Δt をある値より大きくすると、数値振動が発生することを示しました。ところで、この式の n は反復回数を意味しますが、熱伝導方程式の時間微分項に基づく反復であり、いま時間微分項は1次精度で近似されているものの、この反復は時間1次精度の範囲内で時間依存した解を求めていることとなります。そのため、この n は時間ステップ (Time Step) といいます。またこのように時間微分項を使って反復計算する方法を、時間進行法 (Time-marching Method) といいます。 $n+1$ 時間ステップの値を、 n 時間ステップの値のみから計算する方法のことを、陽解法 (Explicit Method) といいますが、線形安定性理論から陽解法には安定限界があることが証明されており、式(5.14)の場合には、 $\ell = \kappa \Delta t / (\Delta x)^2$ が 0.5 を超えると安定限界を超えて数値振動が発生することが知られています (線形安定性については後述します)。

熱伝導方程式の安定限界を緩和する方法として、クランク・ニコルソン法 (Crank-Nicolson Method)[6]が知られています。この方法では、熱伝導方程式を次式のように差分近似します。

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{\kappa}{2} \left(\frac{u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}}{(\Delta x)^2} + \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{(\Delta x)^2} \right) \quad (5.15)$$

これは、2階偏導関数 $\partial^2 u / \partial x^2$ を、 n 時間ステップと $n+1$ 時間ステップで差分近似して、それらを平均するという式です。ところで、このように $n+1$ 時間ステップの値を計算するために、 $n+1$ 時間ステップの値が必要にある方法のことを、陰解法 (Implicit Method) といいます。たとえば、ガウス・ザイデル法は陰解法になります。陰解法であるクランク・ニコルソン法を用いることにより、熱伝導方程式は理論的に無条件安定に計算することができ Δt をいくらでも大きく取れるということになりますが、実際には境界条

件や多次元化などによる制約から、 Δt を無限大にできるというわけではありません。

5-4 線形安定性理論

1次元熱伝導方程式を陽解法で解くとき、 Δt をある値より大きくすると数値振動が発生しますが、その理由をフォンノイマン (von Neumann)の線形安定性理論から簡単に説明します。フォンノイマンの線形安定性理論では、 n 時間ステップにおける格子点 j の解 u_j^n が、有限の振幅を持った任意の位相の三角関数成分で与えられると仮定します。

すなわち、

$$u_j^n = G^n \exp(ji\theta) \quad (5.16)$$

と定義されます。ただし、 $G=G(\theta)$ は振幅で、増幅係数(Amplitude Factor)と呼ばれます。また、 G に付いている n はべき乗です。 $\theta=\pi/s$ ($s=\pm 1, \pm 2, \pm 3, \pm 4, \dots$)。 i は虚数単位(-1の平方根)。これを差分近似式(5.14)に代入すると、

$$G^{n+1}e^{ji\theta} = G^n e^{ji\theta} + \ell(G^n e^{(j+1)i\theta} - 2G^n e^{ji\theta} + G^n e^{(j-1)i\theta}) \quad (5.17)$$

となり、整理すれば、

$$G = 1 - 2\ell(1 - \cos\theta) \quad (5.18)$$

が得られます。フォンノイマンの線形安定性理論では、 G の絶対値が1より小さければ線形安定となります。したがって、 $\ell \leq 0.5$ であれば、上記熱伝導方程式の陽解法は線形安定となります。この値が0.5を超えてしまうと数値振動が発生してしまいます。

熱伝導方程式

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

にクランク・ニコルソン法を適用した場合の線形安定限界を同様に求めてみます。

クランク・ニコルソン法により差分近似された式は次のように求められます。

$$-\ell u_{j-1}^{n+1} + 2(1+\ell)u_j^{n+1} - \ell u_{j+1}^{n+1} = \ell u_{j-1}^n + 2(1-\ell)u_j^n + \ell u_{j+1}^n \quad (5.19)$$

$u_j^{n+1} = G^n \exp(ji\theta)$ を代入して整理すれば、

$$G = \frac{1 - \ell(1 - \cos\theta)}{1 + \ell(1 - \cos\theta)} \quad (5.20)$$

になります。この式は常に1以下の値になることがわかります。すなわち、クランク・ニコルソン法は線形安定性の制約を受けず無条件安定になります。

列を上三角形、下三角形領域に分割して計算する LU 分解法 (Lower-upper Decomposition Method) などがあります。

たしかに、一次元の熱伝導方程式の場合には直接法で解くことができますが、二次元以上の熱伝導方程式になると、行列自体がたいへん複雑になるため、直接法では計算できません。さらに計算格子点の数が多くなると、直接法は多くの記憶領域を必要とするため効率的ではありません。したがって一般的には、熱伝導方程式はクランク・ニコルソン法を適用して、反復法により計算します。

・ 5 - 6 熱伝導方程式の反復解法

1次元熱伝導方程式にクランク・ニコルソン法を適用した式は、

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{\kappa}{2} \left(\frac{u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}}{(\Delta x)^2} + \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{(\Delta x)^2} \right) \quad (5.23)$$

でした。 $\ell = \kappa \Delta t / (\Delta x)^2$ として、さらに変形すると、

$$u_j^{n+1} = \ell (u_{j-1}^{n+1} - 2u_j^{n+1} + u_{j+1}^{n+1}) / 2 + \alpha_j \quad (5.24)$$

ただし、

$$\alpha_j = u_j^n + \ell (u_{j-1}^n - 2u_j^n + u_{j+1}^n) / 2$$

ラプラス方程式の反復計算に用いた反復回数 n における解は、収束解が得られるまでには意味のないものでしたが、熱伝導方程式における時間ステップ n により求めた解は実時間の解を与えています。熱伝導方程式に反復法を適用するためには、時間ステップ n は別に、反復法のための反復ステップが必要になります。

反復法の反復ステップを m として、上記式にまずヤコビ法を適用すると、

$$(u_j^{n+1})^{m+1} = \frac{\ell}{2(1+\ell)} \left[(u_{j-1}^{n+1})^m + (u_{j+1}^{n+1})^m \right] + \frac{\alpha_j}{1+\ell} \quad (5.25)$$

が導出されます。 α_j は既知量ですから、反復法の計算からははずれます。 u_j^{n+1} の値を求めるために、 m を 1, 2, 3, ... と増やして計算し、 m と $m+1$ の値が同じになったとき、 u_j^{n+1} が求まるという式です。

ラプラス方程式の反復法と同じく、ヤコビ法はガウス・ザイデル法に改良することができます。ガウス・ザイデル法では、すでに計算済みの値を有効活用するというものでしたから、すなわち、

$$(u_j^{n+1})^{m+1} = \frac{\ell}{2(1+\ell)} \left[(u_{j-1}^{n+1})^{m+1} + (u_{j+1}^{n+1})^m \right] + \frac{\alpha_j}{1+\ell} \quad (5.26)$$

と導出できます。

さらに、ガウス・ザイデル法を SOR 法にすることもできます。SOR 法は、ガウス・ザイデル法の値を過大評価するというものでしたので、その式は、

$$(u_j^{n+1})^{m+1} = \omega \left\{ \frac{\ell}{2(1+\ell)} \left[(u_{j-1}^{n+1})^{m+1} + (u_{j+1}^{n+1})^m \right] + \frac{\alpha_j}{1+\ell} \right\} + (1-\omega)(u_j^{n+1})^m \quad (5.27)$$

となります。

一次元のみならず、二次元、三次元の熱伝導方程式は、クランク・ニコルソン法により差分近似して、SOR 法による反復計算により解を求めるのが一般的です。

5-7 二次元熱伝導方程式の Excel 計算

二次元熱伝導方程式を Excel を用いて解く方法について紹介します。一次元熱伝導方程式については、すでに Excel を用いて計算した例を紹介しました。一次元の場合には、画面上で Excel を手動で操作しながら計算することができますが、二次元の場合には無理かもしれません。

そこで、簡単なマクロプログラムを追加するだけで計算できる方法について紹介します。

まず、二次元熱伝導方程式を次のように定義します。

$$\frac{\partial u}{\partial t} = \kappa \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right)$$

これを陽解法で差分近似すると、

$$\frac{u_{i,j}^{n+1} - u_{i,j}^n}{\Delta t} = \kappa \left(\frac{u_{i+1,j}^n - 2u_{i,j}^n + u_{i-1,j}^n}{(\Delta x)^2} + \frac{u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n}{(\Delta y)^2} \right)$$

簡単にするため、 $\Delta x = \Delta y$ 、 $\ell = \kappa \Delta t / (\Delta x)^2$ とおけば、

$$u_{i,j}^{n+1} = u_{i,j}^n + \ell (u_{i+1,j}^n + u_{i-1,j}^n + u_{i,j+1}^n + u_{i,j-1}^n - 4u_{i,j}^n) \quad (5.28)$$

と比較的簡単な式を導出することができます。

具体的な熱伝導問題として、図 5.3 に示す境界条件を設定します。

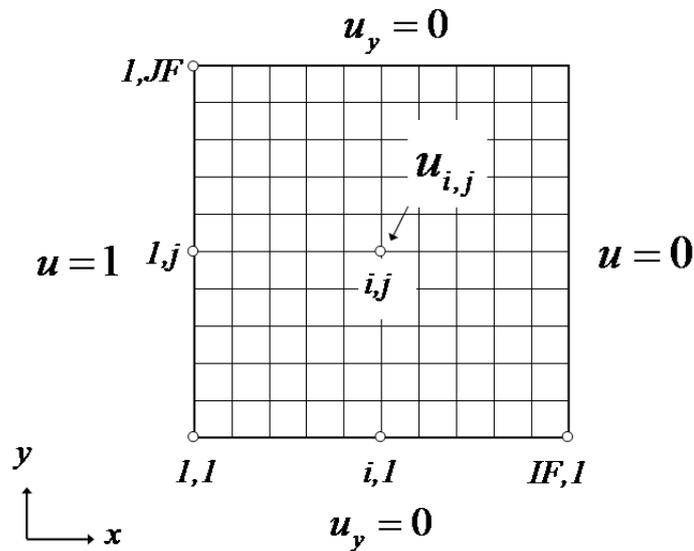


Fig.5.3 計算格子と境界条件

ここでは最も単純な陽解法による方法を紹介していきます。まず、今回用意した Excel シートを図 5.4 に示します。

シートは大きく分けて、初期値を与える「領域 1」、時間ステップ n の計算結果を保管する「領域 2」、時間ステップ $n+1$ の計算ならびに計算結果を保管する「領域 3」、そして計算結果を可視化する「領域 4」の 4 つの領域からなります。領域 1 では 4 つの変数、 $\Delta t = 0.25$ 、 $\Delta x = 0.1$ 、 $\kappa = 0.01$ 、そして初期時間、 $t = 0$ を規定しています。次に、領域 2 には、計算格子の各点における時間ステップ n の値が表示されます。ただし、ここでは計算格子点数を、 $IF = 11$ 、 $JF = 11$ としました。計算初期値として、セル番号 C7 から C 17 まで 1 を代入し、その他のセルにはすべて 0 が代入されています。領域 3 が時間ステップ $n+1$ の計算をしている中枢部分であり、境界以外の内点のセルで、差分近似式が解かれます。たとえば、セル番号 D22 をクリックすると、

$$fx = D8 + \$B\$4 * \$D\$4 * (E8 + C8 + D9 + D7 - D8 * 4) / \$C\$4 / \$C\$4$$

という演算式が現れます。Excel でセルの行番号と列番号にそれぞれ \$ マークが付いた場合には、固定されたセル番号を参照しますので、\$B\$4, \$C\$4, \$D\$4 はそれぞれ領域 1 の Δt , Δx , κ の値である 0.25, 0.1, 0.01 を参照します。また、セル番号 D8, E8, C8, D9, D7 は、それぞれ、 $u_{i,j}^n$, $u_{i+1,j}^n$, $u_{i-1,j}^n$, $u_{i,j+1}^n$, $u_{i,j-1}^n$ の値が保管されている領域 2 のセル番号に相当します。したがって、たとえばセル番号 D22 においては、格子点 $i = 2, j = 2$ における値が計算されます。

一方、境界ではノイマン境界条件が与えられている上辺と底辺に相当する格子点において計算が必要になります。ここでは、計算する差分近似された式は陽解法ですので、時間ステップ $n+1$ で計算された 1 つ内点の値を用いて外挿します。たとえば、上辺境界

に位置するセル番号 D21 の計算式は、 $f_x = D22$ と置きます。

境界と内点の計算式を用いた計算がすべて終了した段階で、領域 3 には時間ステップ $n+1$ の計算結果が保管されます。

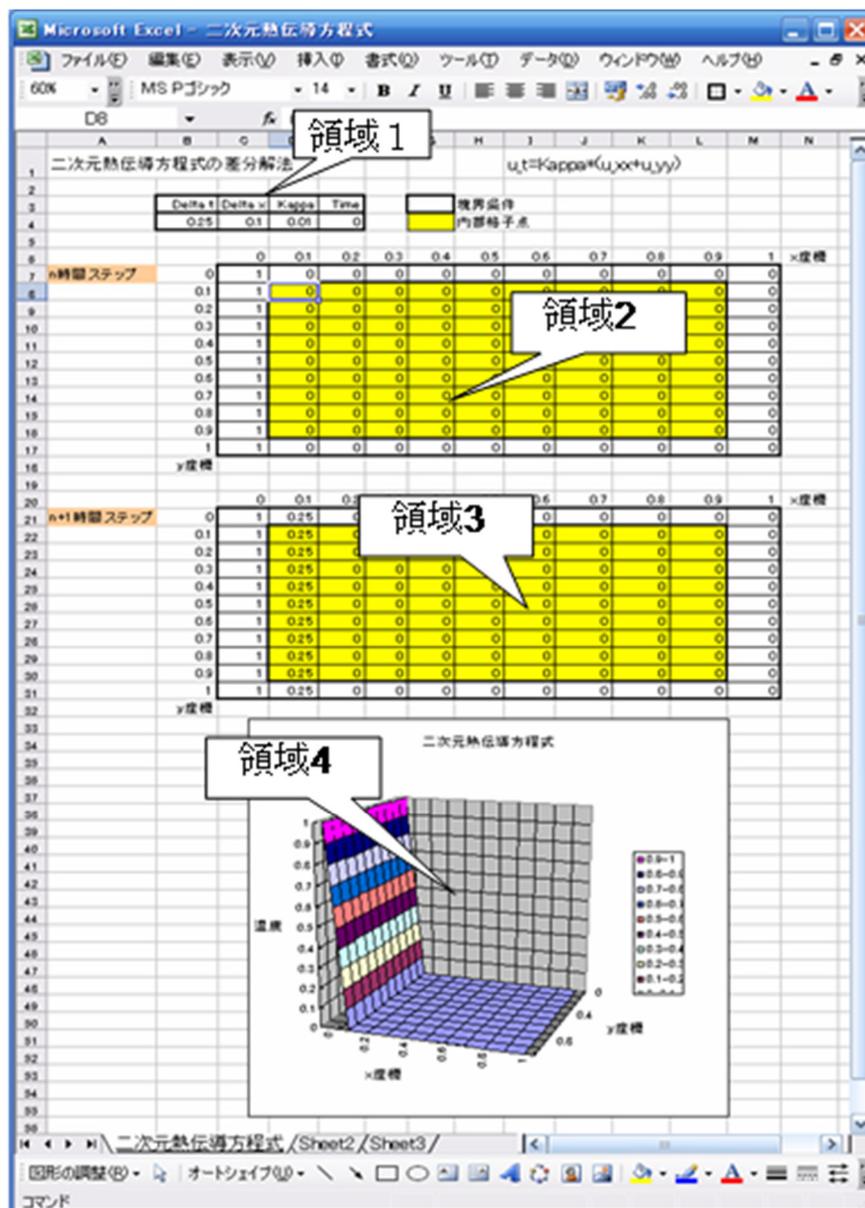


Fig.5.4 Excel シートと初期値設定

ここで紹介する方法で使用する簡単なマクロプログラムを以下に示します。

```

Sub 二次元熱伝導方程式()
  For i = 1 To 40
    Range("C21:M31").Copy
    Range("C7:M17").PasteSpecial (xlPasteValues)
    Range("E4").Value = Range("E4").Value + Range("B4").Value
  Next i
End Sub

```

このプログラムでは、領域3から領域2へのコピーと、時間ステップの更新が行われます。Excelのツールバーにある、ツール/マクロ/Visual Basic Editorを立ち上げると図5.5のようにウィンドウが開きますので、左ボックスに表示された、Sheet1をダブルクリックして開き、その中にプログラムをコピーします。

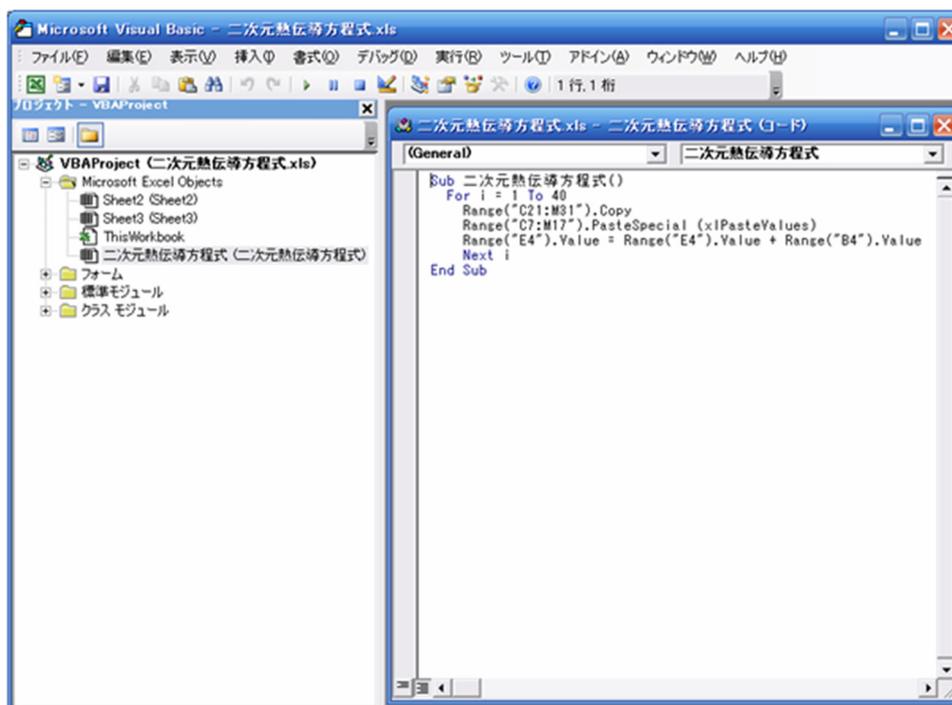


Fig. 5.5 Excel シートとマクロプログラム

最終的に、ツール/マクロ/マクロで、作成済みのマクロプログラム名を選択して、実行ボタンを押せばマクロプログラムが実行されて計算が開始します。

マクロプログラムを実行した結果を次に示します。このマクロプログラムでは、40時間ステップ計算するように設定されており、その間、領域3の二次元セル領域C21:M31の値を、領域2の二次元セル領域C7:M17へのコピーを繰り返し、かつセル番号E4にあるTimeにセル番号B4にあるDelta tの値を加算します。Timeの値は最終的

には、 $0.25 \times 40 = 10.0$ となります。時間ステップの数を増やしたい場合には、繰り返し回数の数値を変更するか、繰り返しマクロプログラムを実行します。

40 時間ステップ計算したときの計算結果は次のようになりました。

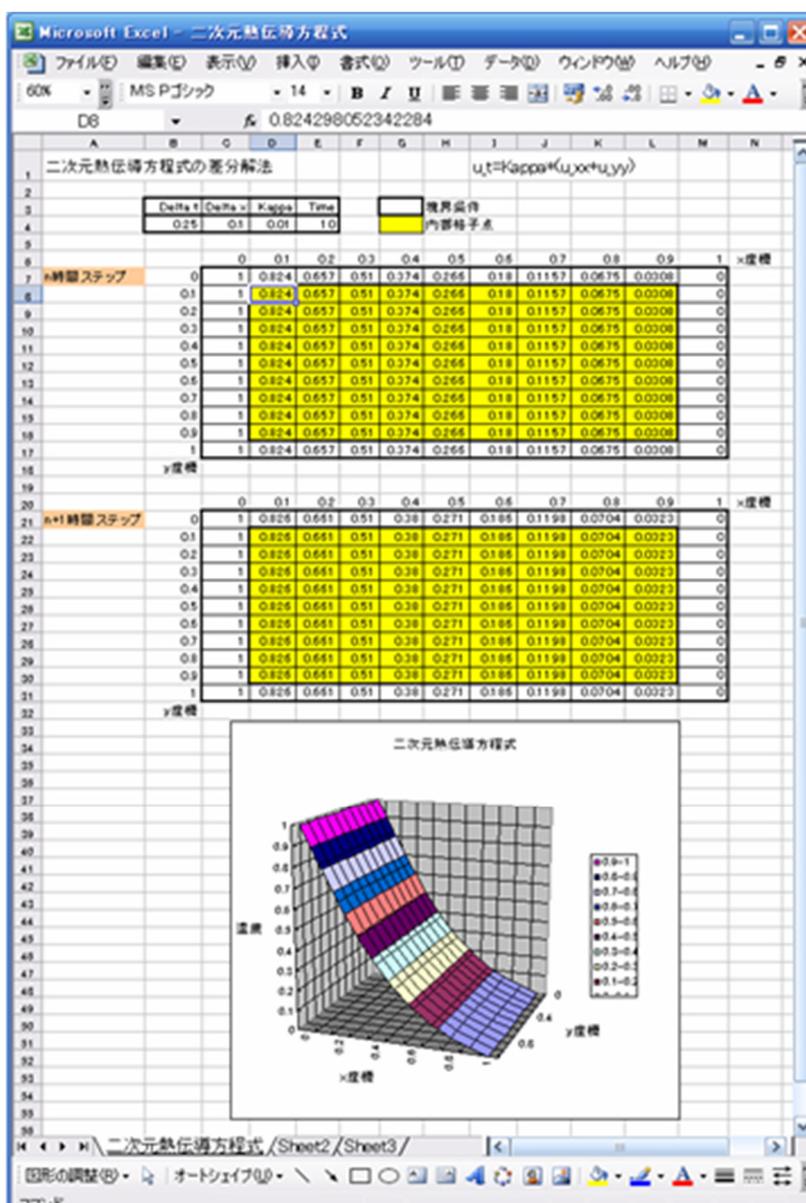


Fig. 5.6 Excel シートと計算結果(40 時間ステップ)

領域 1 の Time の値が 10 に更新され、領域 2, 3 のセルにも計算結果が表示されています。さらに、計算された結果は、領域 4 に可視化されます。

さらに、Time が 40 になるまで繰り返し計算した場合の結果をも示します。80 時間ステップでほぼ収束解（定常解）が得られました。

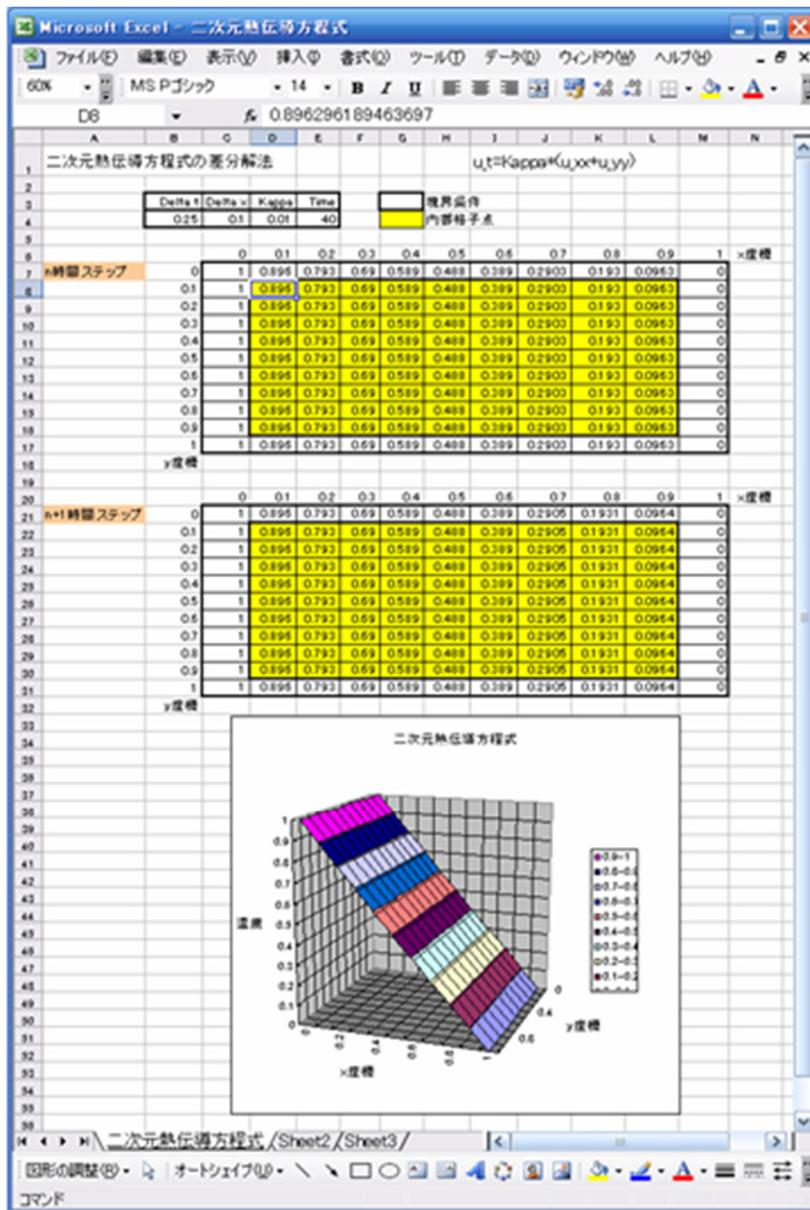


Fig. 5.7 Excel シートと計算結果(80 時間ステップ)

ここで紹介した Excel のプログラムは、下記ホームページに置いてあります。

<http://www.caero.mech.tohoku.ac.jp/publicData/Modeling/index.html>

5-8 三次元熱伝導方程式の差分解析法

二次元熱伝導方程式を Excel で解く方法を紹介しましたが、三次元になるとさすがに Excel で計算するのは無理です。

ここでは、三次元熱伝導方程式をクランク・ニコルソン法により差分近似して、SOR 法で解く、Fortran プログラムを紹介します。

まず、三次元熱伝導方程式を次式で定義します。

$$\frac{\partial T}{\partial t} = \frac{\partial}{\partial x} \left(\kappa_x \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(\kappa_y \frac{\partial T}{\partial y} \right) + \frac{\partial}{\partial z} \left(\kappa_z \frac{\partial T}{\partial z} \right) \quad (5.29)$$

T は温度です。 κ_x , κ_y , κ_z は x , y , z 方向の熱伝導係数です。一般的には同じ値です。これをクランク・ニコルソン法で差分近似すると、

$$\begin{aligned} & \frac{T_{i,j,k}^{n+1} - T_{i,j,k}^n}{\Delta t} \\ &= \frac{\kappa_{xr} (T_{i+1,j,k}^{n+1} - T_{i,j,k}^{n+1}) - \kappa_{xl} (T_{i,j,k}^{n+1} - T_{i-1,j,k}^{n+1})}{2(\Delta x)^2} + \frac{\kappa_{yr} (T_{i,j+1,k}^n - T_{i,j,k}^n) - \kappa_{yl} (T_{i,j,k}^n - T_{i,j-1,k}^n)}{2(\Delta x)^2} \\ &+ \frac{\kappa_{yr} (T_{i,j+1,k}^{n+1} - T_{i,j,k}^{n+1}) - \kappa_{yl} (T_{i,j,k}^{n+1} - T_{i,j-1,k}^{n+1})}{2(\Delta y)^2} + \frac{\kappa_{yr} (T_{i,j+1,k}^n - T_{i,j,k}^n) - \kappa_{yl} (T_{i,j,k}^n - T_{i,j-1,k}^n)}{2(\Delta y)^2} \\ &+ \frac{\kappa_{zr} (T_{i,j,k+1}^{n+1} - T_{i,j,k}^{n+1}) - \kappa_{zl} (T_{i,j,k}^{n+1} - T_{i,j,k-1}^{n+1})}{2(\Delta z)^2} + \frac{\kappa_{zr} (T_{i,j,k+1}^n - T_{i,j,k}^n) - \kappa_{zl} (T_{i,j,k}^n - T_{i,j,k-1}^n)}{2(\Delta z)^2} \end{aligned} \quad (5.30)$$

とたいへん長い式になります。ここで、 i , j , k は x , y , z 方向の格子点番号です。また、添え字 r , l がついた熱伝導係数は、たとえば x 方向では、 $\kappa_{xr} = [(\kappa_x)_{i+1,j,k} + (\kappa_x)_{i,j,k}] / 2$, $\kappa_{xl} = [(\kappa_x)_{i,j,k} + (\kappa_x)_{i-1,j,k}] / 2$ のように計算します。

クランク・ニコルソン法で差分近似された式に SOR 法を適用すると、

$$\begin{aligned} & (T_{i,j,k}^{n+1})^{m+1} = (T_{i,j,k}^{n+1})^m \\ &+ \omega \left[\frac{l_x}{2L} \left\{ \kappa_{xr} (T_{i+1,j,k}^{n+1})^m + \kappa_{xl} (T_{i-1,j,k}^{n+1})^{m+1} \right\} + \frac{l_y}{2L} \left\{ \kappa_{yr} (T_{i,j+1,k}^{n+1})^m + \kappa_{yl} (T_{i,j-1,k}^{n+1})^{m+1} \right\} \right. \\ &\left. + \frac{l_z}{2L} \left\{ \kappa_{zr} (T_{i,j,k+1}^{n+1})^m + \kappa_{zl} (T_{i,j,k-1}^{n+1})^{m+1} \right\} - (T_{i,j,k}^{n+1})^m + \frac{1}{L} \alpha_{i,j,k} \right] \end{aligned} \quad (5.31)$$

ここで、

$$\alpha_{i,j,k} = T_{i,j,k}^n + \frac{l_x}{2} \left\{ \kappa_{xr} (T_{i+1,j,k}^n - T_{i,j,k}^n) - \kappa_{xl} (T_{i,j,k}^n - T_{i-1,j,k}^n) \right\} \\ + \frac{l_y}{2} \left\{ \kappa_{yr} (T_{i,j+1,k}^n - T_{i,j,k}^n) - \kappa_{yl} (T_{i,j,k}^n - T_{i,j-1,k}^n) \right\} \\ + \frac{l_z}{2} \left\{ \kappa_{zr} (T_{i,j,k+1}^n - T_{i,j,k}^n) - \kappa_{zl} (T_{i,j,k}^n - T_{i,j,k-1}^n) \right\}$$

$$l_x = \Delta t / (\Delta x)^2, \quad l_y = \Delta t / (\Delta y)^2, \quad l_z = \Delta t / (\Delta z)^2$$

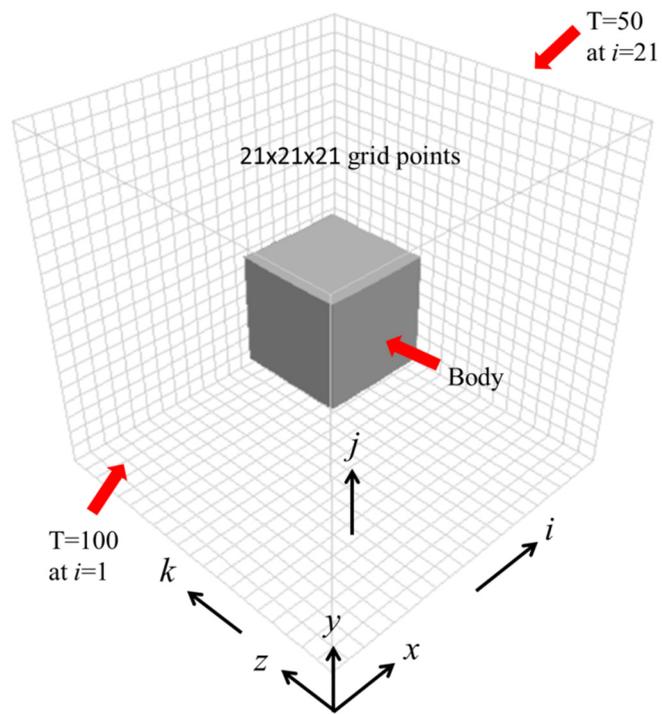
$$L = 1 + \frac{1}{2} l_x (k_{xr} + k_{xl}) + \frac{1}{2} l_y (k_{yr} + k_{yl}) + \frac{1}{2} l_z (k_{zr} + k_{zl})$$

となり、SOR法で反復計算しながら $T_{i,j,k}^{n+1}$ を求める陰解法の差分計算式ができます。

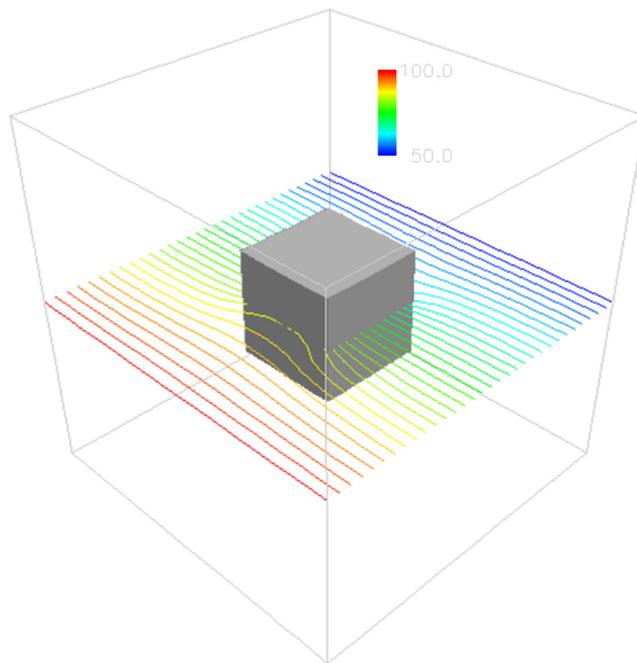
計算例として、図 5.8(a) に示すように立方体内外の熱伝導を計算する Fortran プログラムを紹介します。格子点数は $21 \times 21 \times 21$ とし、立方体はその表面がちょうど、 $i = j = k = 8$ 、ならびに $i = j = k = 14$ に位置するように設定します。立方体内外の熱伝導率はそれぞれ 5.0, 1.0 とし、境界条件は、入口境界 $i = 1$ 面で $T = 100^\circ\text{C}$ 、出口境界 $i = 21$ 面で $T = 50^\circ\text{C}$ を与え、それ以外の外部境界および立方体表面には、法線方向に $\partial\phi/\partial n = 0$ となるようノイマン境界条件を与えます。なお、格子間隔は $\Delta x = \Delta y = \Delta z = 0.1$ で、過緩和係数は 1.5 とします。計算により得られた温度分布を、図 5.8(b) に示します。立方体の周りに 50°C から 100°C に変化する温度場が得られていることがわかります。

これら境界条件・初期条件に設定した Fortran プログラムは、下記のページに置いてあります。

<http://www.caero.mech.tohoku.ac.jp/publicData/Numerical/index.html>



(a) 計算格子と境界条件



(b) 温度分布

Fig. 5.8 立方体周りの熱伝導問題

5-9 特性方程式

2階偏微分方程式は、楕円型、放物型、双曲型の3種類の型に分類されることを説明しました。そのときは、 $D = B^2 - AC$ としますと、 D の符号に応じて、

$D > 0$ なら、双曲型 (Hyperbolic)

$D = 0$ なら、放物型 (Parabolic)

$D < 0$ なら、楕円型 (Elliptic)

となることを紹介しました。ここでは、その理由を特性の理論(Theory of Characteristics)に基づき補足説明します。

二つの独立変数からなる2階の偏微分方程式を一般形で次式のように定義します。

$$A \frac{\partial^2 u}{\partial x^2} + 2B \frac{\partial^2 u}{\partial x \partial y} + C \frac{\partial^2 u}{\partial y^2} = f \left(x, y, u, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y} \right) \quad (5.32)$$

いま、偏導関数を、

$$\frac{\partial u}{\partial x} = p, \quad \frac{\partial u}{\partial y} = q, \quad \frac{\partial^2 u}{\partial x^2} = r, \quad \frac{\partial^2 u}{\partial x \partial y} = s, \quad \frac{\partial^2 u}{\partial y^2} = t \quad (5.33)$$

とにおいて、上式に代入します。すると、

$$Ar + 2Bs + Ct = f \quad (5.34)$$

になります。

ところで、 p, q の全微分は、

$$\begin{aligned} dp &= \frac{\partial p}{\partial x} dx + \frac{\partial p}{\partial y} dy = r dx + s dy \\ dq &= \frac{\partial q}{\partial x} dx + \frac{\partial q}{\partial y} dy = s dx + t dy \end{aligned} \quad (5.35)$$

で定義されますので、これらから、 r と t を逆算して上式に代入すると、

$$A \frac{dp - s dy}{dx} + 2Bs + C \frac{dq - s dx}{dy} = f \quad (5.36)$$

が得られます。これをさらに s で整理すると次式が得られます。

$$s \left\{ A \left(\frac{dy}{dx} \right)^2 - 2B \left(\frac{dy}{dx} \right) + C \right\} - \left\{ A \frac{dp}{dx} \frac{dy}{dx} + C \frac{dq}{dx} - f \frac{dy}{dx} \right\} = 0 \quad (5.37)$$

この式がすべての s に対して常に成り立つためには、

$$A \left(\frac{dy}{dx} \right)^2 - 2B \frac{dy}{dx} + C = 0 \quad (5.38)$$

でかつ、

$$A \frac{dp}{dx} \frac{dy}{dx} + C \frac{dq}{dx} - f \frac{dy}{dx} = 0 \quad (5.39)$$

でなければなりません。ここで式(5.38)は、特性方程式 (Characteristics Equation) になります。特性方程式は、 dy/dx の2次方程式になっていますので、根 (root) は、

$$\frac{dy}{dx} = \frac{B + \sqrt{B^2 - AC}}{A} \quad (5.40)$$

ならびに、

$$\frac{dy}{dx} = \frac{B - \sqrt{B^2 - AC}}{A} \quad (5.41)$$

になります。

実は、式(5.40), (5.41)の平方根の中の値が、 $D = B^2 - AC$ に相当します。したがって、特性方程式(5.38)の根は、 $D > 0$ なら二つの実根 (Real root)、 $D = 0$ なら重根 (Duplicate root)、 $D < 0$ なら二つの複素根 (Imaginary root) になります。

まとめますと、二つの独立変数からなる2階偏微分方程式は、その特性方程式が、

二つの実根を持つ場合は、双曲型

重根を持つ場合は、放物型

二つの複素根を持つ場合は、楕円型

に分類されます。

特性方程式の根は、係数 A, B, C が定数なら、同じく定数になります。

一方、式(5.39)は、 $\lambda = dy/dx$ とおけば次式のように書けます。

$$A\lambda \frac{dp}{dx} + C \frac{dq}{dx} - f\lambda = 0 \quad (5.42)$$

独立変数が x だけの式となり、この式は常微分方程式に帰着されます。結局、偏微分方程式が常微分方程式に簡単化されたことになります。

特性方程式の根をわかりやすく説明するために、図 5.9 に図示してみます。独立変数 (x, y) からなる2階偏微分方程式は、 xy 空間上で図示することができます。初期条件として、 $y = 0$ ($y > 0$) とすれば、 $y = 0$ 上の1本の直線が初期値になります。この直線上の任意の点から、 $y > 0$ 方向に解は伝播します。いま、 $(x_1, 0)$, $(x_2, 0)$ の2点を考えます。そこから、 dy/dx が一定値になる方向に直線をひくことができます。仮にその直線の dy/dx が特性方程式の根であるとすれば、図中のようにそれぞれの点から2本ずつ直線を引くことができます。このような直線のことを、特性曲線 (Characteristic Curve) といいます。直線なのに曲線というのは変ですが、ここでの説明はあくまで、1次精度を仮定していますので、2次精度になれば直線は曲線になります。

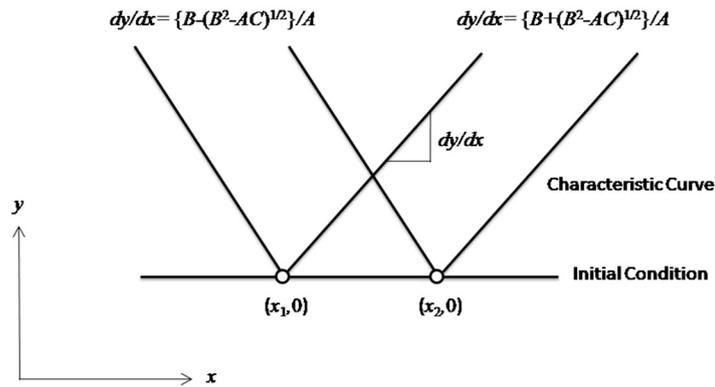


Fig. 5.9 特性曲線のその勾配

図 5.10 に、初期値上の 2 点をそれぞれ L, M として、それぞれから伸びている特性方程式の根を勾配に持つ直線を示します。たとえば、 L 点から伝播した解を示す 2 つの直線の内、 $dy/dx = \{B + \sqrt{B^2 - AC}\}/A$ の勾配を、 λ_L^+ とします。これは、特性曲線の勾配に相当します。このような勾配を持つ特性曲線は、初期値のあらゆる点から伸びており、全空間を埋め尽くしています。

ここからが重要ですが、そのような勾配をもった特性曲線上では、先に示した常微分方程式が成り立ちます。たとえば、 λ_L^+ を勾配に持つ特性曲線上では、

$$A\lambda_L^+ \frac{dp}{dx} + C \frac{dq}{dx} - f\lambda_L^+ = 0 \quad (5.43)$$

が成り立ちます。ここで紹介している偏微分方程式は、その特性方程式の根を勾配に持つ特性曲線上で成り立つ常微分方程式を解くことで解を求めることができます。偏微分方程式は、事実上、数値計算でしか解くことができませんが、常微分方程式でしたらば、紙とえんぴつで解くこともできるわけです。

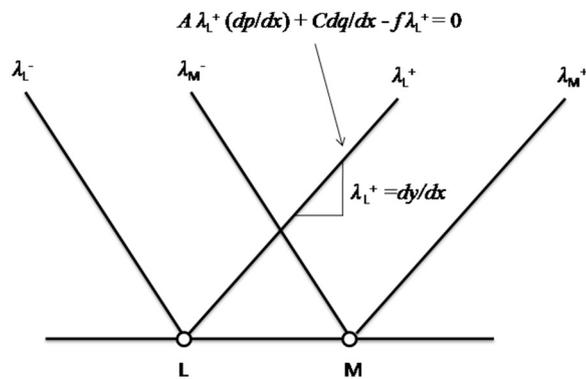


Fig. 5.10 特性曲線上で成り立つ常微分方程式

6 数理モデリングとその計算

6-1 反応拡散方程式

反応方程式と熱伝導方程式についてはすでに説明しましたが、これらを組み合わせた方程式が反応拡散方程式 (Reaction-diffusion Equation) です。すなわち、

$$\frac{\partial u}{\partial t} = \kappa \frac{\partial^2 u}{\partial x^2} + au \quad (6.1)$$

の形をしている偏微分方程式です。ただし、 κ は拡散係数 (Diffusion coefficient)、 a は反応係数 (Reaction coefficient) です。反応項 au は、熱伝導方程式の差分近似式に簡単に付加して計算することができます。たとえばランク・ニコルソン法とヤコビ法により差分近似された反応拡散方程式である式(5.25)に対して次式のように付加されます。

$$(u_j^{n+1})^{m+1} = \frac{\ell}{2(1+\ell)} \left[(u_{j-1}^{n+1})^m + (u_{j+1}^{n+1})^m \right] + \frac{\alpha_j}{1+\ell} + \Delta t a u_j^n \quad (6.2)$$

典型的な反応拡散方程式としては、たとえば、

$$\frac{\partial u}{\partial t} = \varepsilon^2 \frac{\partial^2 u}{\partial x^2} + u(u-a)(1-u) \quad (6.3)$$

のように反応項が三次関数になっているものがあります。これは、Nagumo 方程式[7]と呼ばれ、神経系(neuro system)の信号伝播(signal transmission)を模擬する反応拡散方程式系のよく知られている数理モデルです。また、

$$\frac{\partial u}{\partial t} = \varepsilon^2 \frac{\partial^2 u}{\partial x^2} + au(1-u) \quad (6.4)$$

は、Fisher 方程式[8]と呼ばれ、生態系 (Ecological system) の増殖・拡散 (Growth and diffusion) を模擬する数理モデルです。特に、 $\varepsilon = 0$ の場合は、ロジスティック方程式 (Logistic Equation) と呼ばれ、人口予測に用いられます。他にも数多くの反応拡散方程式に基づく数理モデルが提案されています。反応拡散方程式は、様々な物理現象や社会事象を単純化して模擬する上で、便利な方程式です。

いま、Fisher 方程式を解いてみましょう。 $0 \leq x \leq 1$, $\varepsilon = 0.001$, $a = -1$, $u(x,0) = 0.01 + 0.1 \sin \pi x$ とおき、 $\Delta t = 0.1$ で時間を進行させます。その結果、 $t = 3.0$ のときに次のような結果が得られます。

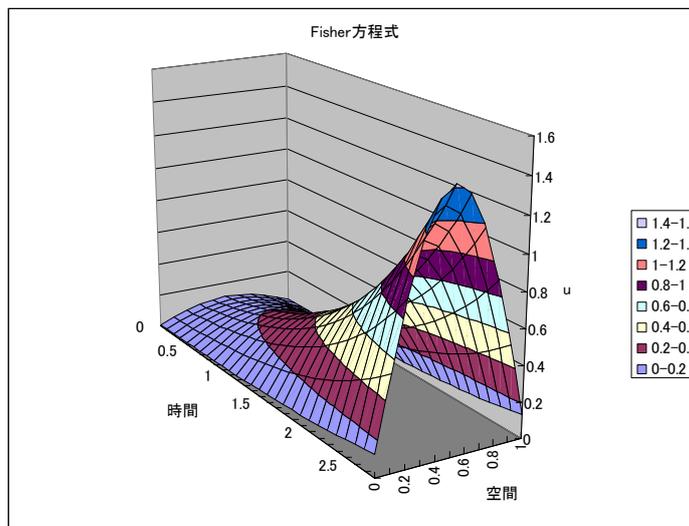


Fig. 6.1 Fisher 方程式の計算結果その 1

時間の経過とともに、 u の値が増加して、特に $x=0.5$ に近いほど増加する傾向が強いと
いう結果になっています。

それでは次に、 $a=1$ の場合にはどうなるでしょう。 $a=1$ にして同様に計算した結果、
次の図のような結果が得られました。時間の経過とともに、 u の値が減少していること
がわかります。このように、反応項にある係数 a を変化させることにより、 u の増加や
減少を模擬することができ、たとえば u がある生態系の個体数であるとすれば、個体の
増殖や消滅が模擬できます。

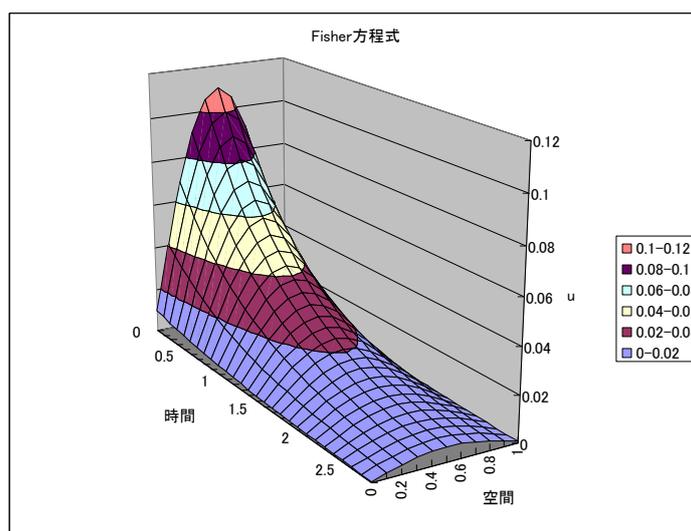


Fig. 6.2 Fisher 方程式の計算結果その 2

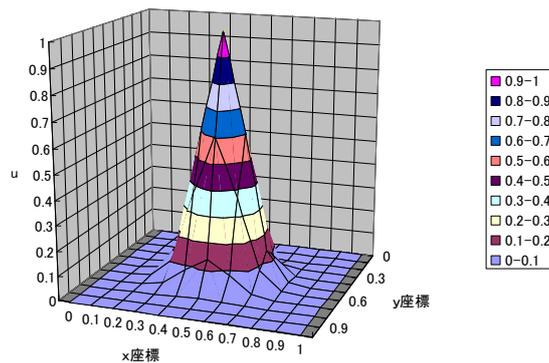
6-2 反応拡散方程式系

先に反応方程式系について説明しましたが、これに拡散項を付加したものが、反応拡散方程式系 (System of Reaction-diffusion Equations) になり、偏微分方程式系になります。反応拡散方程式を連立して解くことにより、さらに複雑な物理現象や社会事象を計算することができます。

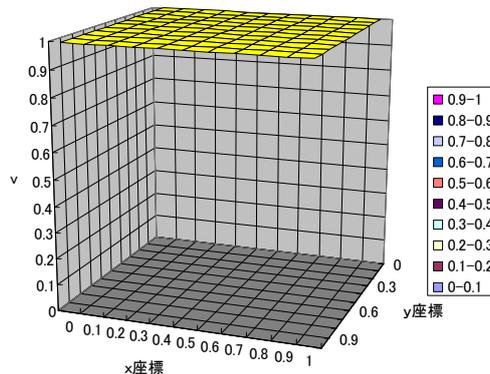
ここでは、次のような二次元の反応拡散方程式系を考えます。

$$\begin{aligned} \frac{\partial u}{\partial t} &= \varepsilon^2 \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + avu \\ \frac{\partial v}{\partial t} &= \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} - uv \end{aligned} \quad (6.5)$$

$0 \leq x \leq 1, 0 \leq y \leq 1, \varepsilon = 0.2, a = 5$ として、境界条件は、 $\partial u / \partial n = 0, \partial v / \partial n = 0$ とします。ただし、 n は境界の法線方向座標です。また、初期値を、 $u(x, y, 0) = \exp\{[(x - 0.5)^2 + (y - 0.5)^2] * 50\}, v = 1$ と与えると、次のように図示されます。



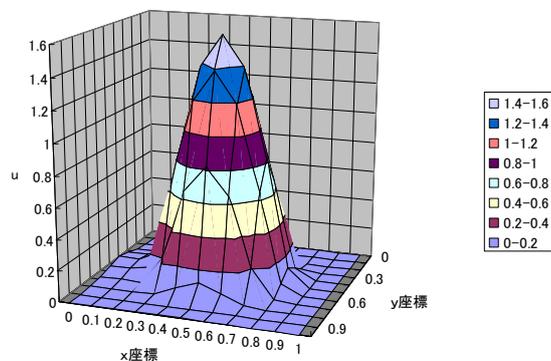
(a) u の初期値



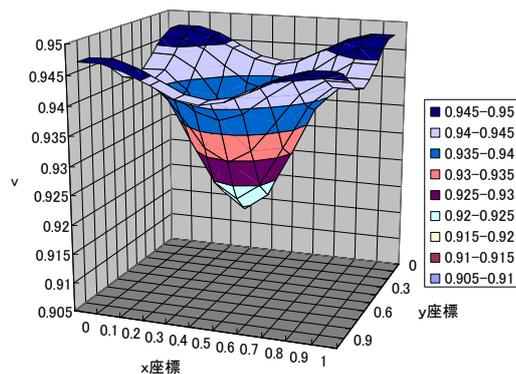
(b) v の初期値

Fig. 6.3 拡散方程式系の初期値

この初期値から、 $\Delta t = 0.001$ で時間を進行させて、 $t = 3.0$ まで計算した結果が次のようになります。



(a) u の初期値



(b) v の初期値

Fig. 6.4 拡散方程式系の計算結果

u の分布は、中心部が約 1.5 倍の値になり、かつ分布の裾野が広がっています。一方、 v の分布は中心部分に最小値を持った分布になっています。これらの分布は、係数 ε や a の値を変えることにより、まったく違った傾向の分布にすることもできます。ここで得られた分布は、たとえば、 u がバクテリアの個体数、 v がその栄養源の濃度であるとすれば、栄養源を吸収したバクテリアが増殖し、一方で栄養源はバクテリアの個体数が多いところほど減少が著しいという一連のメカニズムを近似していると解釈することができます。

たとえば、物質 A と物質 B が化学反応して、物質 C になる反応式



を数理モデル化すると、次のような反応拡散方程式系になります。

$$\begin{aligned}
\frac{\partial u}{\partial t} &= D_u \nabla^2 u - avu \\
\frac{\partial v}{\partial t} &= D_v \nabla^2 v - auv \\
\frac{\partial w}{\partial t} &= D_w \nabla^2 w + auv
\end{aligned} \tag{6.7}$$

ここで、 u 、 v 、 w は物質A、物質B、物質Cの密度もしくは濃度で、 D_u 、 D_v 、 D_w はそれぞれの拡散係数、 a は反応係数です。なお、 $\nabla^2 u$ は拡散項で、 u 、 v 、 w の2階微分の偏導関数です。

6-3 バクテリアと白血球のバトル

偏微分方程式系からなる数理モデルの構築では、特に各方程式にある反応項のバランスを理解することが重要です。これを理解すれば、各自いろいろな物理現象や社会現象をモデリングすることができます。

ここでは、バクテリア (Bacteria) とそれを捕食・自爆する白血球(White cell)からなる生態系を数理モデル化します。これの元になるものは、Alt and Lauffenberger [9]のバクテリアとそれを捕食する白血球からなる一次元生態系モデルで、次のように提案されています。

$$\begin{aligned}
\frac{\partial u}{\partial t} &= D_u \nabla^2 u + (a - bw)u \\
\frac{\partial v}{\partial t} &= D_v \nabla^2 v + cu \\
\frac{\partial w}{\partial t} &= D_w \nabla^2 w - (d + eu)w - \nabla \cdot (\alpha w \nabla v)
\end{aligned} \tag{6.8}$$

u はバクテリアの濃度、 v はバクテリアが排出する化学誘因物質(Attractant)の濃度、 w は白血球の濃度。 a, b, c, d, e, α は反応項の経験定数で、 D_u, D_v, D_w はそれぞれの拡散係数です。これらの偏微分方程式が模擬している物理メカニズムを簡単に説明すれば、まずバクテリアは増殖(減少)しながら、化学誘因物質を排出します。一方、白血球はバクテリアの排出物であるこの化学誘因物質の濃度が高い方に移動する性質があります。そしてバクテリアを捕食して、かつ自分自身も自爆します。この一連のメカニズムを模擬しています。ただし、式の中で第3式の右辺第3項は、次式のように展開されます。化学誘因物質の濃度が高い方に移動するメカニズムをモデル化している部分です。

$$\begin{aligned} & \nabla \cdot (\alpha w \nabla v) \\ &= \left(\frac{\partial}{\partial x} \quad \frac{\partial}{\partial y} \quad \frac{\partial}{\partial z} \right) \cdot \left(\alpha w \frac{\partial v}{\partial x} \quad \alpha w \frac{\partial v}{\partial y} \quad \alpha w \frac{\partial v}{\partial z} \right) \\ &= \frac{\partial}{\partial x} \left(\alpha w \frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial y} \left(\alpha w \frac{\partial v}{\partial y} \right) + \frac{\partial}{\partial z} \left(\alpha w \frac{\partial v}{\partial z} \right) \end{aligned}$$

Alt and Lauffenberger が提案したモデルに、バクテリアの栄養補給を考慮した独自の数理モデルを作成してみました。数理モデルは次の4つの連立した偏微分方程式系になります。栄養源 (Nutritive) の濃度を n とすれば、

$$\begin{aligned} \frac{\partial u}{\partial t} &= D_u \nabla^2 u + (a - bw)u + \beta nu \\ \frac{\partial v}{\partial t} &= D_v \nabla^2 v + cu \\ \frac{\partial w}{\partial t} &= D_w \nabla^2 w - (d + eu)w - \nabla \cdot (\alpha w \nabla v) \\ \frac{\partial n}{\partial t} &= D_n \nabla^2 n - \beta un \end{aligned} \tag{6.9}$$

β は反応項の経験定数です。

前出の2次元熱伝導方程式の Excel シートを改良して、二次元問題を解いてみます。いま、経験定数を次の表のように設定します。

Table 1 経験定数

a	b	c	d	e	α	β
1.0	100.0	1.0	1.0	100.0	1.0	10.0

また、拡散係数は $D_u = D_v = D_w = D_n = 1.0$ とします。バクテリアの初期濃度を $u(x, y, 0) = \exp[\{(x - 0.5)^2 + (y - 0.5)^2\} * 50]$ と、かつ化学誘因物質、白血球、栄養源の濃度をそれぞれ、 $v(x, y, 0) = 0.0$ 、 $w(x, y, 0) = 1.0$ 、 $n(x, y, 0) = 1.0$ と与えます。ただし $0 \leq x \leq 1$ 、 $0 \leq y \leq 1$ 。これらの初期値は図 6.5 のように可視化されます。

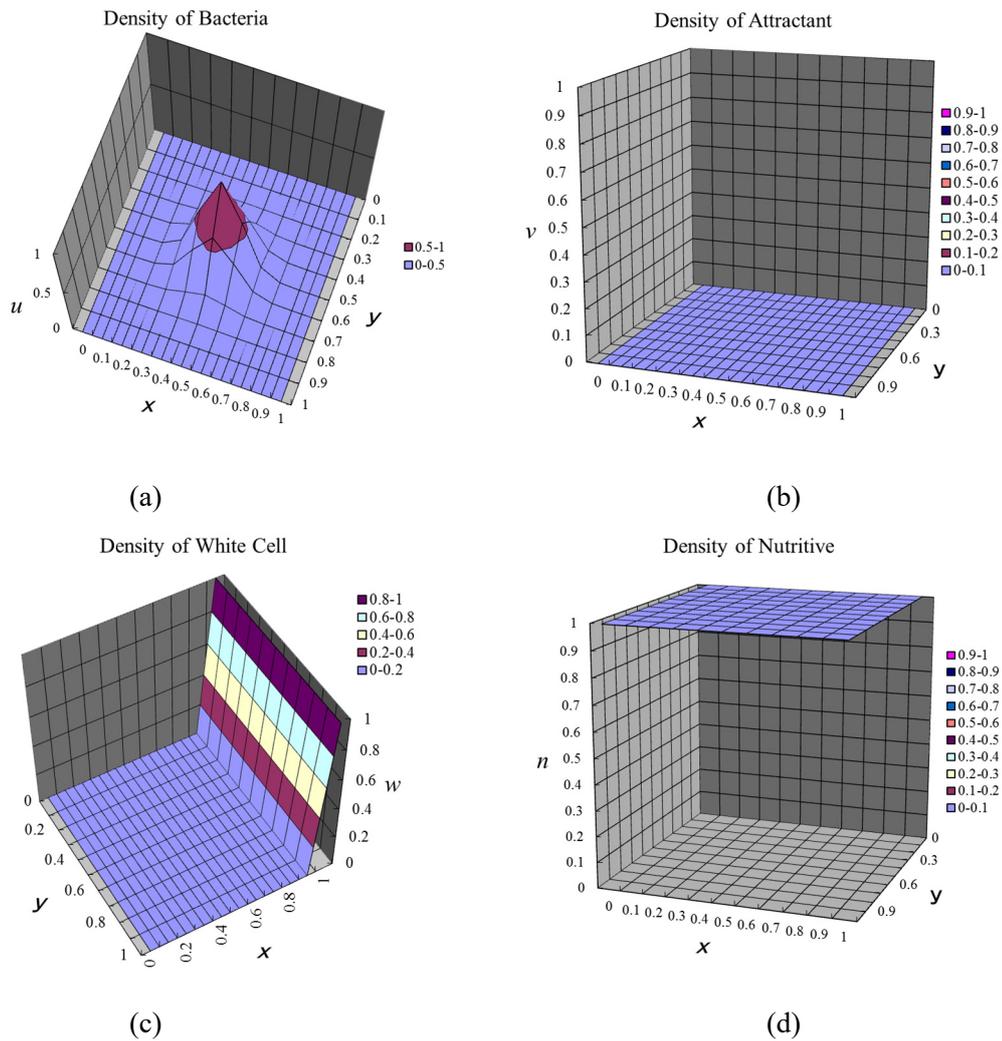


Fig. 6.5 濃度の初期値分布[(a) バクテリア, (b) 化学誘因物質, (c) 白血球, (d) 栄養源].

$\Delta t = 0.001$ で400時間ステップ計算した後に得られた結果を図6.6に示します。バクテリアの濃度は徐々に減少しながらその最大値が $-x$ 方向に移動しています。化学誘因物質は全領域で有限値を示しており、バクテリアの濃度が最大値の辺りで相対的に高い値になっています。 $x=1$ 面に初期値1と与えていた白血球の濃度は、 $-x$ の方向に徐々に減少している結果となっていますが、これは白血球がバクテリアの方に移動していることを意味します。栄養源は、バクテリア濃度が高いところでは相対的に減少している結果になっています。これらの結果から、定性的ながら当初モデル化したバクテリアと白血球間のメカニズムを計算結果はよく再現しました。

ここで紹介したExcelのプログラムは、下記ホームページに置いてあります。

<http://www.caero.mech.tohoku.ac.jp/publicData/Modeling/index.html>

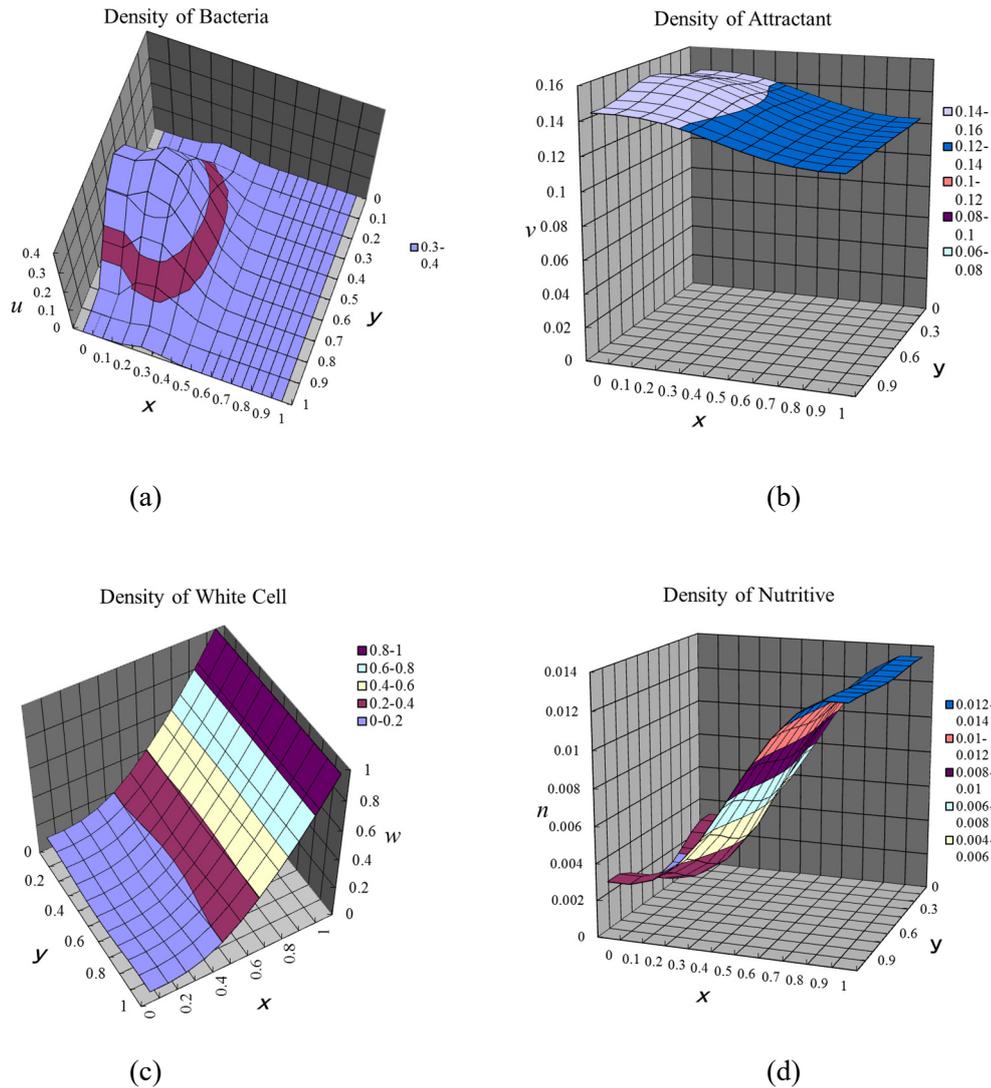


Fig. 6.6 計算結果[(a) バクテリア, (b) 化学誘因物質, (c) 白血球, (d) 栄養源].

さらに三次元問題を解いてみました。三次元熱伝導方程式を解く Fortran プログラムを式 6.9 が解けるように拡張しました。初期値として、 $0 \leq x \leq 1$, $0 \leq y \leq 1$, $0 \leq z \leq 1$ の無次元空間で、バクテリア分布、 $u(x, y, z, 0) = \exp\{[(x - 0.5)^2 + (y - 0.5)^2 + (z - 0.5)^2] * 50\}$ を与えます。栄養源は、全領域に $n(x, y, z, 0) = 1.0$ 。化学誘因物質は、 $v(x, y, z, 0) = 0.0$ 。そして、白血球は、 $w(1, y, z, 0) = 1.0$ を与えます。計算格子点数を $20 \times 20 \times 20$ として計算しました。バクテリアと白血球の初期値分布を図 6.7 に示します。計算領域の中心部分にバクテリアが分布し、右面に白血球が分布しています。白血球の集団がいまバクテリアの近くにいる状態を模擬しました。

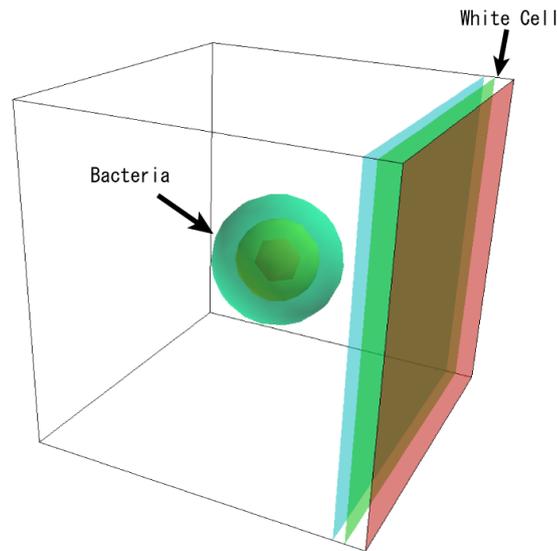


Fig. 6.7 三次元問題の初期値

この初期値から、無次元時間 $t=0.1$ まで計算を進行させて得られた計算結果を図 6.8 に示します。白血球がバクテリアに迫ってきている様子が見られるのと同時に、バクテリアはその分布をお椀形に変形しながら後退している様子が見られます。4つの反応拡散方程式系を数値計算するだけで、このような計算結果が見られるところはたいへん興味深いものがあります。ただし、多くの拡散係数や経験定数を設定しており、これらの値を変えることにより、計算結果も変わってきますから、得られた解の信頼性については保証の限りではありません。言い換えれば、数理モデリングの本質はその係数を物理現象や社会事象に合わせて如何に設定するかということになります。

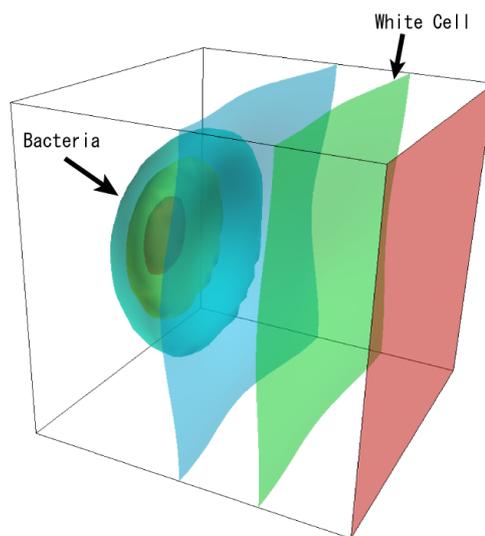


Fig. 6.8 三次元問題の時間依存解

6-4 自己複製パターンの数理モデル

最後に少し変わった数理モデルを紹介します。60年以上前に暗号解読の機械学習を提案したことで有名な Turing [10] は、この研究とは別に、「反応拡散方程式で空間非一様な解を得ることができる」ということを提唱して、細胞分裂 (Cell differentiation) や形態形成 (Morphogenesis) も拡散現象により支配されているということを論文で発表しました。同じ頃、Hodgkin and Huxley [11] は反応拡散方程式系により、神経膜 (Nerve membrane) を出入りするイオンのパルス状信号を模擬できると発表しました。その後40年経って、Pearson [12] は簡単な反応拡散方程式系で複雑なパターンを得ることができることを数値計算により示し、Turing や Hodgkin and Huxley の理論を証明しました。

Pearson が解いた反応拡散方程式系は Gray-Scott モデルと呼ばれ、単純な2つの反応拡散方程式からなります。まず、Gray and Scott [13] はゲル媒体 (Gel medium) 中で観察される次のような自己触媒反応を発見しました。

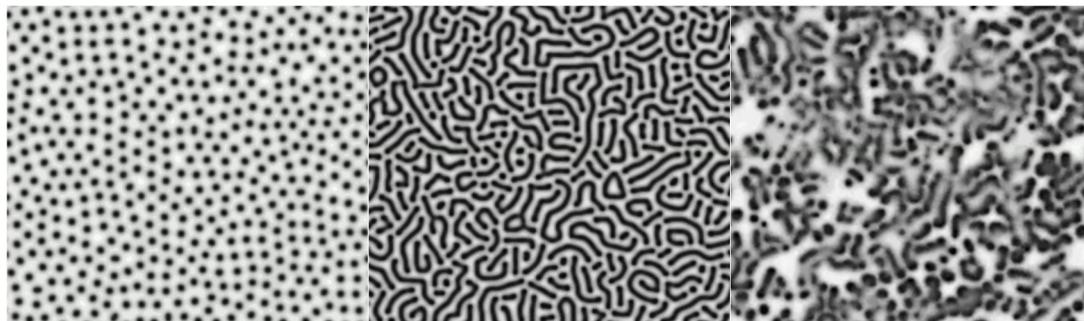


A, B, C は化学反応物質で、そのうち A は触媒として機能します。Pearson は式(6.10)で示される化学反応を反応拡散方程式で次のように数理モデル化しました。

$$\begin{aligned} \frac{\partial u}{\partial t} &= D_u \nabla^2 u - uv^2 + f(1-u) \\ \frac{\partial v}{\partial t} &= D_v \nabla^2 v + uv^2 - (f+k)v \\ \left(\frac{\partial w}{\partial t} &= D_w \nabla^2 w + kv \right) \end{aligned} \tag{6.11}$$

ここで、 u , v , w は化学物質 A, B, C で、 f , k は経験定数。Pearson はこれら2つの経験定数の値をわずかに変えることで、12種類の異なる非一様パターンが得られることを数値計算により示しました。

ここでは、拡散係数を $D_u = 2 \times 10^{-5}$, $D_v = 10^{-5}$ として、2次元 Gray-Scott モデルを計算してみます。計算に用いた格子点数は 256×256 格子点です。図 6.9 に f , k の値をわずかに変化させて計算して結果を示します。まず、図 6.9(a) には、 $f = 0.024$, $k = 0.063$ として計算して得られた結果を示します。この場合、初期値から点状スポットが自己増殖を繰り返して増加し続け、一面を覆う結果が得られました。Gray-Scott モデルが、自己複製パターンを模擬できるモデルであることがわかります。次に、図 6.9(b) に $f = 0.044$, $k = 0.063$ とした場合に得られた結果を示します。今度は迷路のようなパターンが定常解として得られました。さらに図 6.9(c) に示した $f = 0.02$, $k = 0.052$ の場合には、カオス的な非定常解が得られました。たった2つの反応拡散方程式で定義された Gray-Scott モデルから、このような複雑なパターンが得られることは驚きです。



(a) $f = 0.024$; $k = 0.063$ (b) $f = 0.044$; $k = 0.063$ (c) $f = 0.02$; $k = 0.052$

Fig. 6.9 Gray-Scott model により得られた複雑パターン

さらに、3-D Gray-Scott モデルも計算してみました。2種類の異なる経験定数の値を設定して得られた時間依存解を次に示します。動画で示した方がわかりやすいのですが、左の解は周期的に同じ解が繰り返される時間依存解であったのに対して、右の解は、カオス的な時間依存解になりました。これらは三次元熱伝導方程式の Fortran プログラムに基づき計算しました。

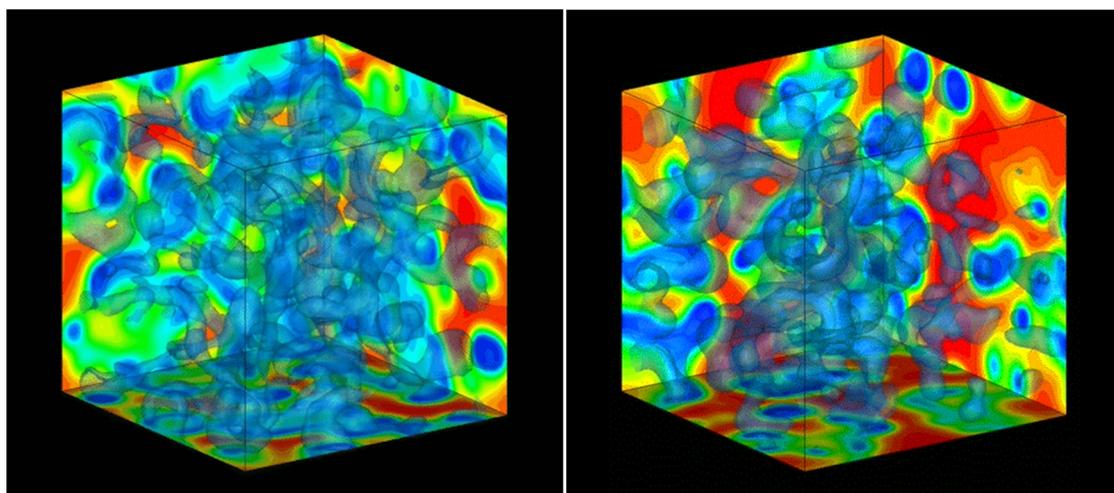


Fig. 6.10 3D Gray-Scott により得られた時間依存パターン

7 まとめ

偏微分方程式系に基づく数理モデルの構築方法ならびに、差分法を用いた数値計算手法について説明しました。また、反応拡散方程式系の具体的な数理モデルについて、計算結果をいくつか紹介しました。皆さんがこの講義ノートを読み終えた段階で、自ら数理モデルを構築して数値計算できるようになっていれば幸いです。

参考文献

1. L.F. Richardson, The Approximate Arithmetical Solution by Finite Differences of Physical Problems Involving Differential Equations with an Application to the Stresses in a Masonry Dam, *Philos. Trans. R. Soc. London, Ser. A*, 210 (1910), 307-357.
2. W.O. Kermack and A.G. McKendrick, Contributions to the Mathematical Theory of Epidemics I, *Proc. of the Royal Society*, 115A (1927), 700-721. (reprinted in *Bulletin of Mathematical Biology*, 53(1/2) (1991), 33-55.
3. W.O. Kermack and A.G. McKendrick, Contributions to the Mathematical Theory of Epidemics II, *Proc. of the Royal Society*, 138A (1932), 55-83. (reprinted in *Bulletin of Mathematical Biology*, 53(1/2) (1991), 57-87.
4. W.O. Kermack and A.G. McKendrick, Contributions to the Mathematical Theory of Epidemics III, *Proc. of the Royal Society*, 141A (1933), 94-122. (reprinted in *Bulletin of Mathematical Biology*, 53(1/2) (1991), 89-118.
5. S.P. Frankel, Convergence Rates of Iterative Treatments of Partial Differential Equations, *Mathematical Tables and Other Aids to Computation*, 4 (1950), 65-75.
6. J. Crank and P. Nicholson, A Practical Method for Numerical Evaluation of Solution of Partial Differential Equations of the Heat-Conduction type, *Proc. Cambridge Philos. Soc.*, 43 (1947), 50-67.
7. H.A. Abdusalam, Analytic and Approximate Solutions for Nagumo Telegraph Reaction Diffusion Equation, *Applied Mathematics and Computation*, 157 (2004), 515-522.
8. R.A. Fisher, *The Genetical Theory of Natural Selection*. Oxford University Press, (1930), [New Ed Edition, (2000)].
9. W. Alt and D.A. Lauffenberger, Transient Behavior of a Chemotaxis System Modelling Certain Types of Tissue Inflammation, *Journal of Mathematical Biology*, 24 (1987), 691-722.
10. A.M. Turing, The Chemical Basis of Morphogenesis, *Phil. Trans. Royal Society of London*, B327 (1952), 37-72.
11. A.L. Hodgkin and A.F. Huxley, A Quantitative Description of Membrane Current and Its Application to Conduction and Excitation in Nerve, *Journal of Physiology*, 117 (1952), 500-544.

12. J.E. Pearson, Complex Patterns in A Simple System, *Science*, 61 (1993), 189-192.
13. P. Gray and S.K. Scott, Autocatalytic Reaction in the Isothermal, Continuous, Stirred Tank Reactor: Oscillation and Instabilities in the System $A + 2B \rightarrow 3B$, B to C, *Chemical Engineering Science*, 39 (1984), 1087-1097.