

2008夏のターボセミナー講義資料

計算数理科学と数理モデリング

2008年8月18日－20日

宮城県蔵王

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 ラプラス方程式を手計算
5. 流れの数理モデル
 - 5-1 流れの支配方程式
 - 5-2 ポテンシャル流れ
 - 5-3 渦度方程式
 - 5-4 流れ関数
6. 数理モデルの数値計算法
 - 6-1 ラプラス方程式の反復解法
 - 6-2 三次元ポテンシャル流れのSOR解法
 - 6-3 熱伝導方程式の差分解法
 - 6-4 線形安定理論
 - 6-5 直接法と反復法
 - 6-6 熱伝導方程式の反復解法
 - 6-7 二次元熱伝導方程式のExcel計算
 - 6-8 三次元熱伝導方程式の差分解法
 - 6-9 特性方程式
 - 6-10 特性曲線法
7. 数理モデリングと計算
 - 7-1 非平衡凝縮流れの数理モデル
 - 7-2 反応拡散方程式の数理モデル
 - 7-3 反応拡散方程式系の数理モデル
 - 7-4 バクテリアと白血球のバトル

1. はじめに

計算数理学という名前は、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はできます。

さて、計算数理学は、「計算」と「数理学」からなっています。まず、数理学ですが、自然科学や社会科学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。特に、ここで説明する計算数理学は、数学的・理論的というよりは、実用的で役に立つような「計算数理学」です。たとえば、環境・エネルギー問題解決に役立つ計算数理学、新しい機器開発に役立つ計算数理学、生活に役立つ計算数理学、などです。

2. 計算数理学の典型としての数値流体力学

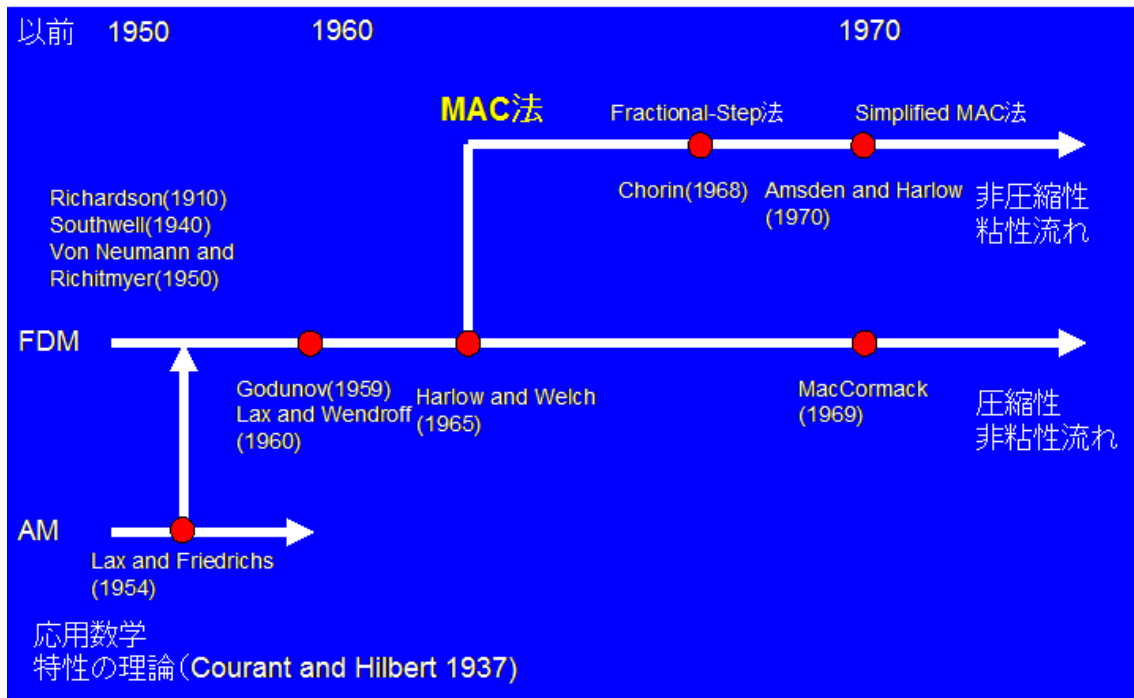
数値流体力学という学問があります。これは、Computational Fluid Dynamics の日本語訳で、英語の頭文字を取って、通称、CFD と呼ばれます。本来ですと、計算流体力学と訳されるのが自然ですが、最初に訳した日本人の研究者がこのように訳したことから、その後、CFD のことを数値流体力学と呼ぶのが一般的になりました。数値流体力学は、計算数理学の典型的な成功例の一つです。

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。

流体力学の完成形としての偏微分方程式は、ナビエ・ストークス方程式 (Navier-Stokes Equations) と呼ばれます。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、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 の論文がその始まりという説もあります。創世記の代表的な研究を系統図にまとめてみました。



FDMは差分法のことです。AMはApplied Mathematics、応用数学のことです。実は、FDM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。ここではFDMの話に限定します。

FDM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、もしくはタイガー計算機 (手動の手回し計算機) 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はご存知でしょうか？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。FDMの基礎になる線形安定性理論 (Linear Stability Analysis) というのも提案しています。

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ていきます。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では、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、このうち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きます。

3. 数値計算とは

数値計算はやった経験がない人の方が多いでしょうが、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、パソコン、自動車、携帯、地デジ、コンビニなど、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想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人材養成機関がほとんど見当たりません。数学は数学屋、工学は工学屋がやるのが当たり前だからです。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数値流体力学を念頭においた研究をしています。

日本ではその応用数学の研究者が不足しているため、工学屋が肩代わりしているのが現状です。日本からたとえば、数値流体力学のための有用な計算手法がほとんど提案されていないのも応用数学分野の研究が貧弱であるためでしょう。

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。この式のことを微分方程式 (Differential Equation) と呼びます。正確には常微分方程式 (Ordinary Differential Equation) です。 $du/dt = -u$ と書けます。

一方、差分法 (Finite Difference Method) とは、その英語の頭文字をとって、FDM と呼ばれ、数値流体力学や計算数理学の基礎として広く用いられている計算手法です。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。

$$(u^{n+1} - u^n) / dt = -u^n$$

ここで、 n は時間ステップと呼ばれます。べき乗ではありません。この式をさらに変形して、

$$u^{n+1} = u^n - dt u^n$$

これは、コンピュータプログラムの代入文に近い式で、 n の値を 1, 2, 3, ... と増やしながら繰り返し計算する式です。いま、 $n = 0$ で、 $t = 0$, $u = 1$ として、 t をたとえば 0.9 ずつ、すなわち $dt = 0.9$ で繰り返し計算すれば、

n	t	u
1	0.9	0.1
2	1.8	0.01
3	2.7	0.001
4	3.6	1E-04
5	4.5	1E-05
6	5.4	1E-06
7	6.3	1E-07
8	7.2	1E-08
9	8.1	1E-09
10	9	1E-10

となり、 u がだんだんゼロに漸近していくのがわかります。もともと、 $u = e^{-t}$ でしたから、近似的にこの指数関数の解を差分法により求めることができます。ただし、注意しなければならないのは、 t の値を十分大きくして、 u の値が変わらなくなったところが収束した解ですので、そこに至るまでの値は正確ではなく、あまり意味がありません。

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 が何か物質だとすれば、その物質が十分時間が経過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厳密解は e^t になりますので、 t を十分大きくすると今度は u は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) と呼びます。

・ 4-2 反応方程式系

反応方程式を 2 つ同時に解いて、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。

$$du/dt = -v u$$

$$dv/dt = u v$$

これらから、差分法により、

$$u^{n+1} = u^n - dt v^n u^n$$

$$v^{n+1} = v^n + dt u^n v^n$$

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、 $dt=0.9$ で実際に計算すると、

n	t	u	v
1	0.9	0.98109	0.01891
2	1.8	0.964393	0.035607
3	2.7	0.933487	0.066513
4	3.6	0.877608	0.122392
5	4.5	0.780937	0.219063
6	5.4	0.626969	0.373031
7	6.3	0.416478	0.583522
8	7.2	0.197757	0.802243
9	8.1	0.054973	0.945027
10	9	0.008217	0.991783

のようになります。 $n=10$ 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。

・ 4-3 偏微分方程式

反応方程式 $du/dt = -u$ の t は時間でした。数学では独立変数と呼ばれます。一方、偏微分方程式とは、英語では、Partial Differential Equation (略して PDE) で、独立変数が 2 つ以上の微分方程式のことをそう呼びます。たとえば、

$$\partial u / \partial t = \partial u / \partial x$$

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。数値流体力学や計算数理科学の分野では、移流方程式 (Convective Equation) に分類されます。

この方程式の解としては、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、結局上記の移流方程式が成り立ちます。実は、独立変数 x 、 t からなる任意の関数(正確には偏微分を含みますので、偏導関数と呼ばれます) を f とすれば、

$$u = f(x + t)$$

であれば、すべて上記の移流方程式が成り立ちます。移流方程式は、数値流体力学、特に圧縮性流れの計算手法にとって極めて重要な偏微分方程式です。

$\partial u / \partial t = \partial u / \partial x$ 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

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

と差分近似できます。 $u(x)$ の x は、ある x における値を意味します。ここでの x は、空間の1次元座標です。 $x + \Delta x$ は x から Δx だけ離れたところを意味します。 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。さらに変形しますと、

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

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 $\Delta t = 0.01$ で計算してみましょう。なお、 $x = 0$ と $x = 1$ の u には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。厳密解と同じ値になりました。それでは、 Δt を一桁増やして $\Delta t = 0.1$ と
して同じ計算をしてみましょう。厳密解は、

n	t	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1	x
1	0.1	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1	1.1	
2	0.2	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1	1.1	1.2	
3	0.3	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1	1.1	1.2	1.3	
4	0.4	0.4	0.5	0.6	0.7	0.8	0.9	1	1.1	1.2	1.3	1.4	
5	0.5	0.5	0.6	0.7	0.8	0.9	1	1.1	1.2	1.3	1.4	1.5	
6	0.6	0.6	0.7	0.8	0.9	1	1.1	1.2	1.3	1.4	1.5	1.6	
7	0.7	0.7	0.8	0.9	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	
8	0.8	0.8	0.9	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	
9	0.9	0.9	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	
10	1	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2	

となります。一方、差分法では、

n	t	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1	x
1	0.1	0.1	0.11	0.21	0.31	0.41	0.51	0.61	0.71	0.81	0.91	1.1	
2	0.2	0.2	0.12	0.22	0.32	0.42	0.52	0.62	0.72	0.82	0.92	1.2	
3	0.3	0.3	0.12	0.23	0.33	0.43	0.53	0.63	0.73	0.83	0.94	1.3	
4	0.4	0.4	0.11	0.24	0.34	0.44	0.54	0.64	0.74	0.84	0.97	1.4	
5	0.5	0.5	0.11	0.25	0.35	0.45	0.55	0.65	0.75	0.85	0.99	1.5	
6	0.6	0.6	0.09	0.26	0.36	0.46	0.56	0.66	0.76	0.86	1.03	1.6	

7	0.7	0.7	0.08	0.28	0.37	0.47	0.57	0.67	0.77	0.88	1.06	1.7
8	0.8	0.8	0.05	0.29	0.38	0.48	0.58	0.68	0.78	0.89	1.11	1.8
9	0.9	0.9	0.03	0.31	0.39	0.49	0.59	0.69	0.79	0.91	1.15	1.9
10	1	1	-0	0.33	0.4	0.5	0.6	0.7	0.8	0.93	1.2	2

になりました。今度は、厳密解と違い少し変な値が計算されています。この違いは、フォンノイマンが提案した線形安定性理論により説明することができます。ただし、ここでは省略します（後ほど紹介）。

・ 4-4 拡散方程式

拡散とは、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 (Diffusive Equation) と呼ばれます。拡散現象は、数学的には2階 (要するに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、

$$\partial u / \partial t = \partial^2 u / \partial x^2$$

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。具体的な例としては、お風呂です。お風呂の温度は刻々と変化します。暖めている熱が温度の低いところへ伝わっていってお風呂はだんだん熱くなっていきます(ただし、対流は無視)。このような熱の伝わりを専門用語で熱伝導 (Heat Conduction) といいます。そのようなことから、上記の式は熱伝導方程式 (Equation of Heat Conduction) とも呼ばれます。正確には右辺に係数がついて、

$$\partial u / \partial t = \kappa \partial^2 u / \partial x^2$$

のようになります。この係数のことを、熱伝導係数 (Coefficient of Heat Conduction) といいます。熱の伝わり方は物質ごとに異なります。この係数の値は物質ごとに違った値になっています。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、

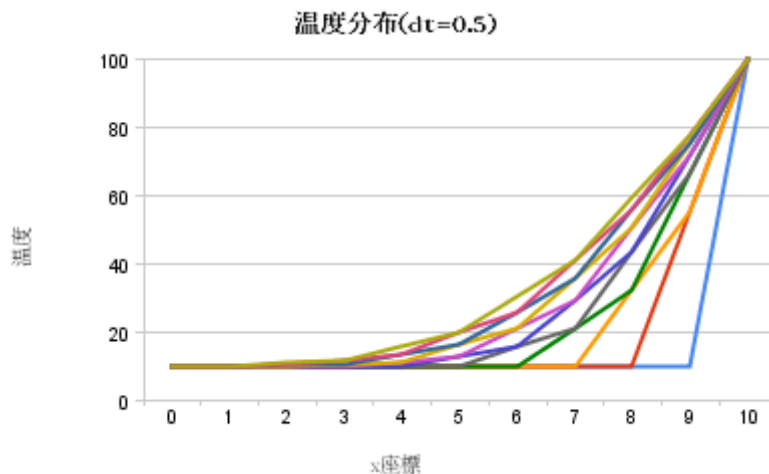
$$[u^{n+1}(x) - u^n(x)] / \Delta t = [u^n(x + \Delta x) - 2u^n(x) + u^n(x - \Delta x)] / (\Delta x)^2$$

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 での $u^n(x)$ 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、

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

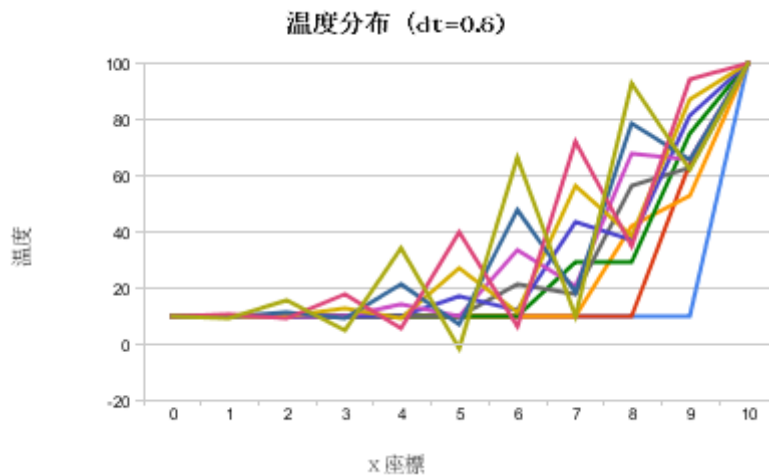
が得られます。

いま長さ 10 で温度 10 度の任意の物質を考えます。 $x = 0$ を 10 度で冷やし、 $x = 10$ 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。 $\Delta x = 1$ 、 $\Delta t = 0.5$ として、 $n=10$ まで計算した結果をグラフにしますと、



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 度を直線を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 $\Delta t = 0.6$ で再度計算してみますと、



になりました。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でおかしい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。

$$A \partial^2 u / \partial x^2 + 2B \partial^2 u / \partial x \partial y + C \partial^2 u / \partial y^2 = f(x, y, u, \partial u / \partial x, \partial u / \partial y)$$

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、

$$\partial^2 u / \partial x^2 - \partial^2 u / \partial y^2 = 0$$

放物型では、

$$\partial^2 u / \partial x^2 = \partial u / \partial y$$

楕円型では、

$$\partial^2 u / \partial x^2 + \partial^2 u / \partial y^2 = 0$$

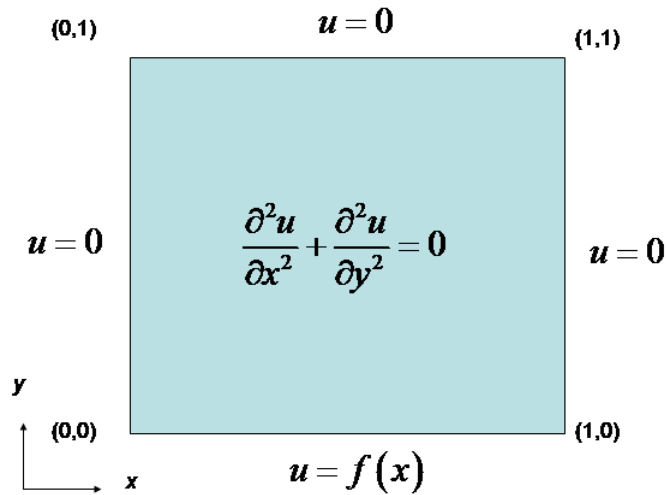
などです。これらの式は標準形と呼ばれます。

双曲型の式は、計算数理学では波動方程式 (Wave Equation) と呼ばれます。放物型の式はよく見ますと、 y が t になれば、「拡散方程式」で紹介した熱伝導方程式と同じであることがわかります。したがって、熱伝導方程式と呼ばれます。そして、楕円型の式は、ラプラス方程式 (Laplace Equation) と呼ばれます。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) と呼ばれます。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を与えます。



この問題は、典型的な偏微分方程式の計算問題ですが、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。詳細が知りたいときは、既存の偏微分方程式に関する教科書にはどれにでも同じことが書かれていますので、参照してください。結局のところ、導出された解析解は次のようになります。

$$u(x, y) = 2 \sum \sinh(n\pi(y-1)) / \sinh(-n\pi) * \sin(n\pi x) \int f(z) \sin(n\pi z) dz$$

ただし、 \sum は $n = 1$ から ∞ までの和、 \int は $z = 0$ から 1 までの積分です。また、

$$\sinh(y) = (e^y - e^{-y}) / 2$$

です。かなり複雑な式になっていますが、上記の図に示した領域内の任意の点の解が求まりますので、たいへんエレガントな式です。

このように、ラプラス方程式の解析解は紙と鉛筆があれば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の内、 $u=0$ のひとつでも 0 以外の値にすると、途端に解析解を求めるのが困難になります。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、数値計算により求めるしかなくなります。

・ 4-7 ラプラス方程式の差分解法

ラプラス方程式は二次元で次式のようになります。

$$\partial^2 u / \partial x^2 + \partial^2 u / \partial y^2 = 0$$

これを中心差分で近似すると、

$$\begin{aligned} & [u(x+\Delta x) - 2u(x) + u(x-\Delta x)] / (\Delta x)^2 \\ & + [u(y+\Delta y) - 2u(y) + u(y-\Delta y)] / (\Delta y)^2 = 0 \end{aligned}$$

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

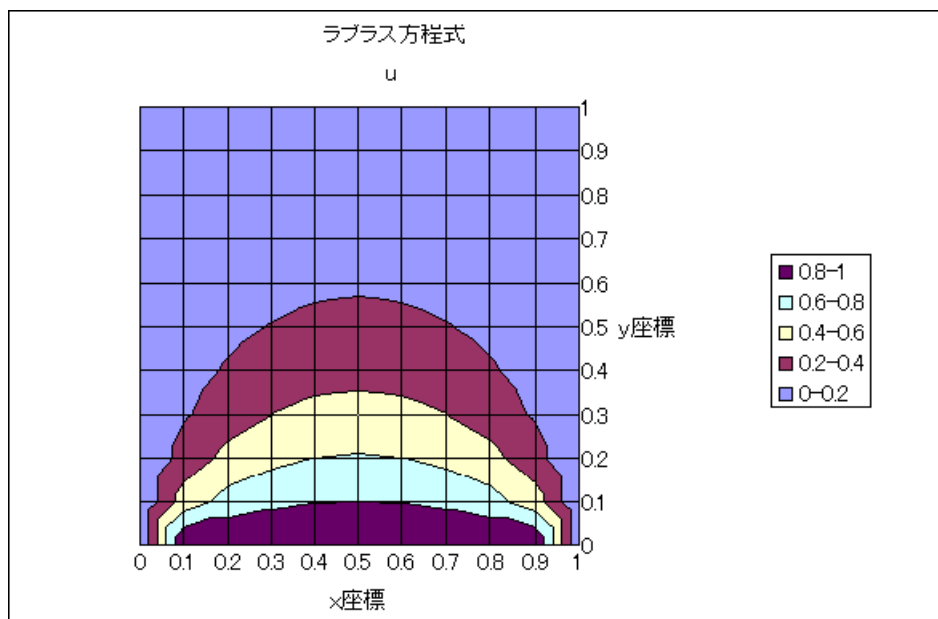
$$u(x) = [u(x+\Delta x) + u(x-\Delta x) + u(y+\Delta y) + u(y-\Delta y)] / 4$$

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。さらに反復計算を施せば、

$$u^{n+1}(x) = [u^n(x + \Delta x) + u^n(x - \Delta x) + u^n(y + \Delta y) + u^n(y - \Delta y)] / 4$$

になります。

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 $f(x) = 1$ として実際に反復計算をしますと次のような計算結果を得ることができます。



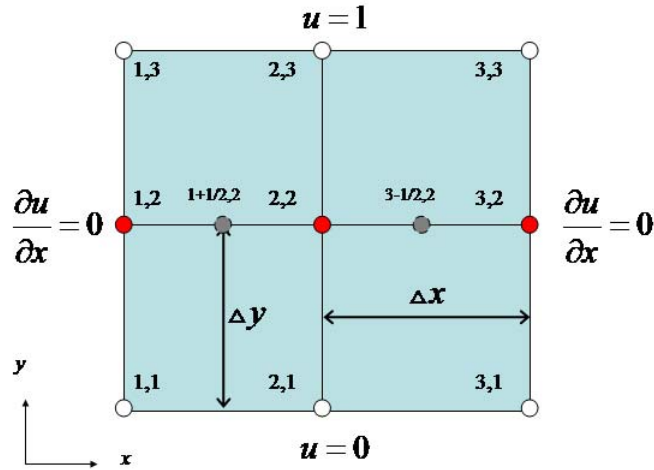
境界条件が底辺角の点で不連続のため解が多少波打っていますが、あまり気にしないことにしましょう。若干のマクロプログラムとともにシート上の計算だけで、Excelでもラプラス方程式を差分法で解くことができます。

・ 4-8 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。次の図にあるような計算格子点がたった9点の問題を考えます。



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。

ラプラス方程式のような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。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。このように未知変数である u そのものを規定する境界条件のことを第1種境界条件、もしくは考えた研究者の名前で、ディリクレ境界条件 (Dirichlet's Boundary Condition) 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、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。このように未知変数である u の1階偏微分を規定する境界条件のことを第2種境界条件、もしくは考えた研究者の名前でノイマン境界条件 (Neumann's Boundary Condition) と呼びます。ノイマンとは、先に紹介したフォンノイマンです。これら2点では、 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。

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。

ラプラス方程式の差分近似式は

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

になります。簡単にするため、 $\Delta x = \Delta y = 1$ と仮定します。

まず、(2,2)における差分近似式を求めてみます。そのために、上記差分近似式に具体的な計算格子点番号を当てはめると、

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

が求まります。同じ項で整理すれば、

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

次に、(1,2)はどうでしょうか。同様に格子点番号を当てはめると、

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

と求まりました。としたいところですが、よく見ると、 $u_{0,2}$ は求めることができません。元の式の $u_{i-1,j}$ が左辺の境界からはみ出してしまいました。2次精度差分近似式では、いま計算しようとする点と両隣の点の値が必要になるため、いま計算しようとする点が境界上にあると、両隣のいずれかの点が境界からはみ出してしまい、差分近似ができなくなります。境界から点をはみ出さないようにするためには、片側差分近似式というもの新たに導出する必要があります。

図には、黒塗りの点を2つ示しており、それぞれの格子点番号は、 $(3-1/2, 2)$ 、 $(1+1/2, 2)$ と定義されています。実際にはこれらの格子点はありません。あくまでバーチャルな格子点です。たとえば、格子点(1,2)では、 x 方向の差分近似で片側差分する必要があります。そのために、(1,2)における $\partial^2 u / \partial x^2$ の片側差分近似式は、まず1階の偏導関数を用いて、

$$(\partial u / \partial x)_{1,2} = [(\partial u / \partial x)_{1+1/2,2} - (\partial u / \partial x)_{1,2}] / (\Delta x / 2)$$

のように定義できます。ここで、 $\Delta x = 1$ でしたので、

$$(\partial u / \partial x)_{1,2} = 2[(\partial u / \partial x)_{1+1/2,2} - (\partial u / \partial x)_{1,2}]$$

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、

$$(\partial u / \partial x)_{1,2} = 2(\partial u / \partial x)_{1+1/2,2}$$

になります。 $(\partial u / \partial x)_{1+1/2,2}$ は u の2次精度中心差分近似で

$$(\partial u / \partial x)_{1+1/2,2} = (u_{2,2} - u_{1,2}) / \Delta x = u_{2,2} - u_{1,2}$$

と計算されますので、結局

$$(\partial^2 u / \partial x^2)_{1,2} = 2(u_{2,2} - u_{1,2})$$

といった、ちょっと特殊な形の片側差分近似式が導出されます。

同様に、格子点(3,2)では、

$$(\partial^2 u / \partial x^2)_{3,2} = 2[(\partial u / \partial x)_{3,2} - (\partial u / \partial x)_{3-1/2,2}]$$

となります。境界条件として $(\partial u / \partial x)_{3,2} = 0$ が与えられていますので、

$$(\partial^2 u / \partial x^2)_{3,2} = -2(\partial u / \partial x)_{3-1/2,2}$$

となり、結局

$$(\partial^2 u / \partial x^2)_{3,2} = -2(u_{3,2} - u_{2,2})$$

が得られます。

これより、(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、(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、

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

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

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

のように求められました。

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、 $u = 0$ でしたから、

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

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

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

になります。

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。

このことからわかりますが、ラプラス方程式の差分法による数値計算は、それをひも解いていけば、未知の u の数だけの連立1次方程式の計算に最終的には帰着されます。言い換えれば、連立1次方程式から構成される行列の逆行列を求めることと同意義です。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になります。

実は、物理的なセンスがある人は、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度、下面を温度0度としたならば、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。

計算数理科学が目指す数理モデルの構築は、もともと物理現象を数学的に記述することでしたから、もとの物理現象に立ち返って数理モデルを考察することが重要です。

・ 5 流れの数理モデル

・ 5-1 流れの支配方程式

偏微分方程式として、反応方程式、移流方程式、拡散方程式、そしてラプラス方程式を紹介してきました。それでは次の方程式はどうでしょう。

$$\partial \rho / \partial t + \partial (\rho u) / \partial x + \partial (\rho v) / \partial y + \partial (\rho w) / \partial z = 0$$

これは、三次元流れの連続の式 (Continuity Equation) です。 t は時間、 (x, y, z) は三次元直交座標、 (u, v, w) は (x, y, z) 方向の流速、 ρ は流体の密度です。すべて 1 階の偏導関数から成っており移流方程式に似ています。

次はどうでしょう。

$$\partial (\rho u) / \partial t + u \partial (\rho u) / \partial x + v \partial (\rho u) / \partial y + w \partial (\rho u) / \partial z$$

$$= -\partial p / \partial x + \mu (\partial^2 u / \partial x^2 + \partial^2 u / \partial y^2 + \partial^2 u / \partial z^2)$$

$$\partial (\rho v) / \partial t + u \partial (\rho v) / \partial x + v \partial (\rho v) / \partial y + w \partial (\rho v) / \partial z$$

$$= -\partial p / \partial y + \mu (\partial^2 v / \partial x^2 + \partial^2 v / \partial y^2 + \partial^2 v / \partial z^2)$$

$$\partial (\rho w) / \partial t + u \partial (\rho w) / \partial x + v \partial (\rho w) / \partial y + w \partial (\rho w) / \partial z$$

$$= -\partial p / \partial z + \mu (\partial^2 w / \partial x^2 + \partial^2 w / \partial y^2 + \partial^2 w / \partial z^2)$$

これらは比較的遅い流れの運動方程式 (Momentum Equations) です。三次元の場合には 3 つの方程式からなります。 p は圧力、 μ は粘性係数です。1 階と 2 階の偏導関数からできているのがわかります。移流方程式と拡散方程式が混ざったような形をしています。ただし、厳密には 2 階微分項の部分はもう少し複雑になります。

最後に、

$$\partial (\rho h) / \partial t + u \partial (\rho h) / \partial x + v \partial (\rho h) / \partial y + w \partial (\rho h) / \partial z$$

$$= \kappa (\partial^2 T / \partial x^2 + \partial^2 T / \partial y^2 + \partial^2 T / \partial z^2)$$

は比較的遅い流れのエネルギー方程式 (Energy Equation) です。 h は比エンタルピー、 T は温度、 κ は熱伝導係数です。これも移流方程式と拡散方程式が混ざったような形をしているのがわかります。特に、右辺は熱伝導方程式の拡散項と同じです。

ここで紹介した 5 つの方程式は、流体力学の完成形である三次元ナビエ・ストークス方程式 (Navier-Stokes Equations, 略して N-S 式) のひとつの表記形です。いろいろな仮定の導入で、N-S 式の記述方法も違ってきます。ところで、一般的に非圧縮性の N-S 式とは運動方程式そのものを指すことが多いようです。

数値流体力学では、流体力学の基本知識は重要ですが、とりあえず N-S 式を構成する各項、すなわち、時間微分項 (Time-derivative Term)、対流項 (Convection Term)、圧

力項(Pressure Term)、そして粘性項 (Viscosity Term)がどのような偏導関数から成り立っているかがわかれば、数値計算することはできます。

流れの連続の式は、仮に流体が非圧縮性ですと密度 ρ は変化しませんので、連続の式は次のように簡略化されます。

$$\partial u / \partial x + \partial v / \partial y + \partial w / \partial z = 0$$

1階の偏導関数のみからなる見た目にも簡単な偏微分方程式になります。しかしながら、この方程式を直接解くことができる数値計算手法は見当たりません。事実上、この方程式は直接解けないと考えた方がよさそうです。

実は、非圧縮性流れの連続の式がこのような形をしていることが、非圧縮性流れの数値流体力学の展開にとって極めて重要な意味を持ちます。これまで紹介した、反応方程式、移流方程式、そして拡散方程式にはいずれも、時間微分項 $\partial u / \partial t$ が含まれており、差分法では

$$u^{n+1} = u^n - \Delta t f^n$$

のように反復計算します。このような計算手法を時間進行法 (Time-marching Method) と呼びます。時間微分項がある方程式は時間進行法を用いて比較的簡単に解くことができます。しかしながら、この項がないと別の手法で解かなければなりません。連続の式を解く適当な手法が見当たらないというのが現状です。

運動方程式のうち、 x 方向の流速 u の運動方程式に対して非圧縮性流れを仮定すれば、

$$\begin{aligned} \partial u / \partial t + u \partial u / \partial x + v \partial u / \partial y + w \partial u / \partial z \\ = -1 / \rho \partial p / \partial x + \mu / \rho (\partial^2 u / \partial x^2 + \partial^2 u / \partial y^2 + \partial^2 u / \partial z^2) \end{aligned}$$

となります。運動方程式の左辺は、時間微分項 (Time-derivative Term) と対流項 (Convection Term)、右辺は圧力項 (Pressure Term) と粘性項 (Viscosity Term) からなっています。非圧縮性流体は、圧力の高いところから低いところへ移流します。また粘性があり物体壁などに引きずられ移流に影響を与えます。上の式は、これらの項のバランスから成り立っています。 y 方向、 z 方向についても同様です。

無次元化 (Nondimensionalization) という操作をしますと、

$$\begin{aligned} \partial u / \partial t + u \partial u / \partial x + v \partial u / \partial y + w \partial u / \partial z \\ = -\partial p / \partial x + 1/Re (\partial^2 u / \partial x^2 + \partial^2 u / \partial y^2 + \partial^2 u / \partial z^2) \end{aligned}$$

となります。 $1 / \rho$ がなくなり、 μ / ρ が $1/Re$ に変わりました。このときの Re がレイノルズ数 (Reynolds number) と呼ばれるもので、流体力学では一番有名な無次元パラメータです。レイノルズ数は慣性力と粘性力の比で定義されます。レイノルズ数が極めて大きい場合には、粘性力は慣性力に比べて取るに足りないとして仮定されて、粘性項を省略した式を計算する場合があります。ところで無次元化の操作ですが、説明が少し面倒ですのでここでは省略します。

エネルギー式に非圧縮性流れを仮定して、さらに $h = CpT$ という熱力学の関係式を導入しますと、

$$\begin{aligned} \partial T / \partial t + u \partial T / \partial x + v \partial T / \partial y + w \partial T / \partial z \\ = \kappa / \rho C_p (\partial^2 T / \partial x^2 + \partial^2 T / \partial y^2 + \partial^2 T / \partial z^2) \end{aligned}$$

となります。ただし、 C_p は定圧比熱と呼ばれる物質固有の物性値です。温度を変数とした式に変形することができます。この式を見ると、流れがない場合には対流項はすべてなくなり、結局、熱伝導方程式と同じ形の式になることがわかります。

無次元化された非圧縮性粘性流れの基礎方程式は、これまで紹介した式をまとめれば、結局次のようになります。

$$\begin{aligned} \partial u / \partial x + \partial v / \partial y + \partial w / \partial z &= 0 \\ \partial u / \partial t + u \partial u / \partial x + v \partial u / \partial y + w \partial u / \partial z \\ &= -\partial p / \partial x + 1/Re (\partial^2 u / \partial x^2 + \partial^2 u / \partial y^2 + \partial^2 u / \partial z^2) \\ \partial v / \partial t + u \partial v / \partial x + v \partial v / \partial y + w \partial v / \partial z \\ &= -\partial p / \partial y + 1/Re (\partial^2 v / \partial x^2 + \partial^2 v / \partial y^2 + \partial^2 v / \partial z^2) \\ \partial w / \partial t + u \partial w / \partial x + v \partial w / \partial y + w \partial w / \partial z \\ &= -\partial p / \partial z + 1/Re (\partial^2 w / \partial x^2 + \partial^2 w / \partial y^2 + \partial^2 w / \partial z^2) \\ \partial T / \partial t + u \partial T / \partial x + v \partial T / \partial y + w \partial T / \partial z \\ &= \kappa / \rho C_p (\partial^2 T / \partial x^2 + \partial^2 T / \partial y^2 + \partial^2 T / \partial z^2) \end{aligned}$$

これらの式を連立して解くことにより、非圧縮性粘性流れを計算します。

実は、変数の数が方程式の数より1つ多いため、これらの式を解いても解は求まりません。圧力 p が不定です。 p を求める方法には大きくわけて二つあります。一つは理想気体を仮定して、圧力の状態方程式 $p = \rho RT$ を合わせて計算する方法です。空気の遅い流れは、この仮定により非圧縮性粘性流れと仮定して解くことができます。それでは水の場合にはどうでしょう。理想気体の状態方程式は使えません。連続の式が直接解けないのに加えて、圧力場を如何に求めるかが、非圧縮性粘性流れの数値流体力学にとっては大きな問題です。

・5-2 ポテンシャル流れ

ラプラス方程式が差分法で簡単に計算できることを紹介しましたが、流れ問題をラプラス方程式で近似することができます。

いま、流速 u, v をポテンシャルと呼ぶ記号 ϕ で次のように定義します。

$$u = \partial \phi / \partial x, \quad v = \partial \phi / \partial y$$

ところで、流れの渦度 ω は二次元で、

$$\omega = \partial v / \partial x - \partial u / \partial y$$

となりますが、これにポテンシャルで定義された流速を代入すると、

$$\omega = \partial (\partial \phi / \partial y) / \partial x - \partial (\partial \phi / \partial x) / \partial y = 0$$

です。すなわち、ポテンシャルで表わされた流れ場は渦なし流れになります。

さて、連続の式は、

$$\partial u / \partial x + \partial v / \partial y = 0$$

ですので、これに渦なし流れの流速を代入すると、

$$\partial (\partial \phi / \partial x) / \partial x + \partial (\partial \phi / \partial y) / \partial y = \partial^2 \phi / \partial x^2 + \partial^2 \phi / \partial y^2 = 0$$

となります。これより、渦なし流れはポテンシャル ϕ のラプラス方程式を解いて求めることができます。渦なし流れはポテンシャル流れ (Potential Flow) とも呼ばれます。

・ 5 - 3 渦度方程式

二次元非圧縮性粘性流れの運動方程式は、

$$\partial u / \partial t + u \partial u / \partial x + v \partial u / \partial y = -\partial p / \partial x + 1/Re(\partial^2 u / \partial x^2 + \partial^2 u / \partial y^2)$$

$$\partial v / \partial t + u \partial v / \partial x + v \partial v / \partial y = -\partial p / \partial y + 1/Re(\partial^2 v / \partial x^2 + \partial^2 v / \partial y^2)$$

と定義されます。

渦なし流れはラプラス方程式で表されることを示しましたが、いま、流速 v の運動方程式を x でさらに偏微分した式から、流速 u の運動方程式を y でさらに偏微分した式を引いて整理すると、

$$\partial \omega / \partial t + u \partial \omega / \partial x + v \partial \omega / \partial y = 1/Re(\partial^2 \omega / \partial x^2 + \partial^2 \omega / \partial y^2)$$

が得られます。ここで、 ω はもともと、 $\omega = \partial v / \partial x - \partial u / \partial y$ で、二次元の渦度の式です。ポテンシャル流れでは、 $\omega = 0$ を仮定していましたが、二次元の2つの運動方程式から、渦度の方程式、すなわち、渦度方程式 (Vorticity Equation) を導出することができます。これにより流れの渦度を直接計算することができます。よく見ると、運動方程式にあった圧力項が渦度方程式にはありません。式の導出の過程で消えました。したがって、圧力場の情報がなくても渦度方程式は計算できます。しかしながら、流速 u , v の情報は必要です。

・ 5 - 4 流れ関数

渦度方程式では、あいかわらず流速 u, v の値が必要です。いま、流れ関数 (Stream Function) と呼ばれる新たなパラメータを導入します。流れ関数には ϕ を用います。流速 u, v は、 ϕ の関数として次のように定義されます。

$$u = \partial \phi / \partial y, \quad v = -\partial \phi / \partial x$$

これを、渦度方程式に代入すれば、

$$\begin{aligned} \partial \omega / \partial t + \partial \phi / \partial y \cdot \partial \omega / \partial x - \partial \phi / \partial x \cdot \partial \omega / \partial y \\ = 1/Re(\partial^2 \omega / \partial x^2 + \partial^2 \omega / \partial y^2) \end{aligned}$$

一方、渦度は、

$$\omega = \partial v / \partial x - \partial u / \partial y$$

でしたから、これに流れ関数で定義された流速を代入すると、

$$\omega = -\partial (\partial \phi / \partial x) / \partial x - \partial (\partial \phi / \partial y) / \partial y$$

もう少し変形して、

$$\partial^2 \phi / \partial x^2 + \partial^2 \phi / \partial y^2 = -\omega$$

となり、流れ関数のポアソン方程式が導出されました。結局、渦度方程式とこの式の2式を連立して解くことにより、二次元非圧縮性粘性流れを数値計算することができます。渦度方程式は時間進行法で解き、流れ関数のポアソン方程式は、先に説明したラプラス方程式の差分解法と同じ方法で解きます。

通常、二次元非圧縮性粘性流れの基礎方程式は、連続の式と2つの運動方程式の計3つの式からなりますが、渦度と流れ関数を用いることで、計算する式の数を一つ減らすことができるため、計算量を削減することができます。ただし、たとえば圧力の情報が要らないことが、逆に境界条件の設定に不都合であったり、三次元の渦度の定義が複雑なことから三次元流れに拡張するのが難しいなどの問題点もあります。

・ 6 数値モデルの数値計算法

・ 6-1 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、

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

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、

$$u_{i,j} = (u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1}) / 4$$

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、時間ステップ n を当てはめた式は、

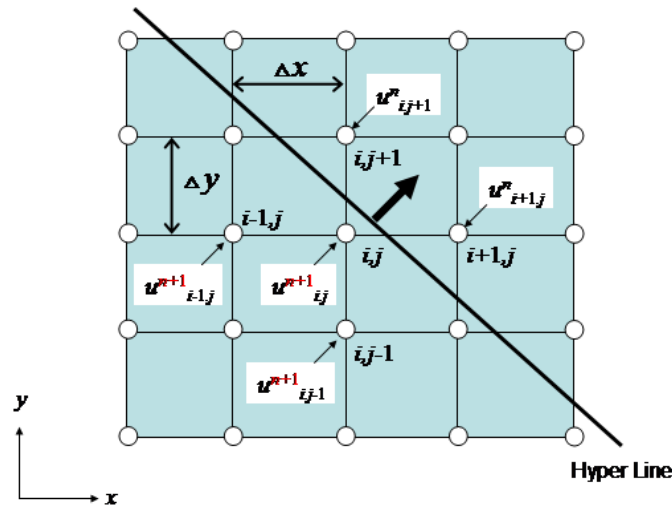
$$u_{i,j}^{n+1} = (u_{i+1,j}^n + u_{i-1,j}^n + u_{i,j+1}^n + u_{i,j-1}^n) / 4$$

のように定義されます。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。また、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。 $n+1$ 時間ステップを計算するとき、 n 時間ステップ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時間ステップの計算は1回の計算で済んでしまいます。

ただし、手元のパソコンでは並列計算は無理かもしれません。ヤコビ法は、上記の式をひたすら繰り返して、最終的に $n+1$ 時間ステップと n 時間ステップ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、かなりの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、



に示すような、ハイパーライン (Hyper Line) と呼ばれる、いわゆ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 update されますから、 $n+1$ 時間ステップの値になります。

これを式で記述すれば、

$$u^{n+1}_{i,j} = (u^{n+1}_{i+1,j} + u^{n+1}_{i-1,j} + u^{n+1}_{i,j+1} + u^{n+1}_{i,j-1}) / 4$$

となります。すでに update 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を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 $u^{n+1}_{i,j}$ を、 $(u^{n+1}_{i,j})^{GS}$ とおいてその計算式を導出すれば、

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

と定義されます。この式は、 n 時間ステップにおける $u^{n+1}_{i,j}$ とガウス・ザイデル法で求めた値の ω による線形結合により $u^{n+1}_{i,j}$ を求めるという形の式になっています。通常、線形結合においては、緩和係数 ω は、 $0 < \omega < 1$ なのですが、ここでの ω は、 $1 < \omega$ になります。そのため、この ω 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 近傍の値が上限値になります。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) と呼びます。これらのことから、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。

・ 6-2 三次元ポテンシャル流れのSOR解法

ポテンシャル流れが、ラプラス方程式を解いて求められることを紹介しました。ここでは具体的に、三次元のポテンシャル流れにSOR法を適用した際の式を導出します。

まず、ポテンシャル ϕ の三次元ラプラス方程式は、

$$\partial^2 \phi / \partial x^2 + \partial^2 \phi / \partial y^2 + \partial^2 \phi / \partial z^2 = 0$$

になります。ただし、流速 u , v , w はポテンシャル ϕ で

$$u = \partial \phi / \partial x, \quad v = \partial \phi / \partial y, \quad w = \partial \phi / \partial z$$

と定義されます。もともとこれを連続の式に代入すればラプラス方程式が得られます。

三次元ラプラス方程式を差分近似すると

$$\begin{aligned} & (\phi_{i+1,j,k} - 2\phi_{i,j,k} + \phi_{i-1,j,k}) / (\Delta x)^2 \\ & + (\phi_{i,j+1,k} - 2\phi_{i,j,k} + \phi_{i,j-1,k}) / (\Delta y)^2 \\ & + (\phi_{i,j,k+1} - 2\phi_{i,j,k} + \phi_{i,j,k-1}) / (\Delta z)^2 = 0 \end{aligned}$$

になります。 z 方向に新たに計算格子点 k を用います。

これまで、 Δx , Δy を簡単にして式を導出してきましたが、ここではこれらの値に忠実に式を導出します。

上記の式を $\phi_{i,j,k}$ の式に変形すれば、

$$\begin{aligned} \phi_{i,j,k} = & \{ (\phi_{i+1,j,k} + \phi_{i-1,j,k}) / (\Delta x)^2 \\ & + (\phi_{i,j+1,k} + \phi_{i,j-1,k}) / (\Delta y)^2 \\ & + (\phi_{i,j,k+1} + \phi_{i,j,k-1}) / (\Delta z)^2 \} / \{ 2/(\Delta x)^2 + 2/(\Delta y)^2 + 2/(\Delta z)^2 \} \end{aligned}$$

になります。さらにSOR法を適用すれば、

$$\begin{aligned} \phi_{i,j,k}^{n+1} = & (1 - \omega) \phi_{i,j,k}^n \\ & + \omega \{ (\phi_{i+1,j,k}^n + \phi_{i-1,j,k}^{n+1}) / (\Delta x)^2 \\ & + (\phi_{i,j+1,k}^n + \phi_{i,j-1,k}^{n+1}) / (\Delta y)^2 \\ & + (\phi_{i,j,k+1}^n + \phi_{i,j,k-1}^{n+1}) / (\Delta z)^2 \} / \{ 2/(\Delta x)^2 + 2/(\Delta y)^2 + 2/(\Delta z)^2 \} \end{aligned}$$

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数値計算することができます。

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 $i = IF$ 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、ポテンシャル ϕ が求められます。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、流速ベクトルを求めることができます。Fortranによる本計算プログラムは、研究室ホームページの下記場所に掲載してあります。

・ 6 - 3 熱伝導方程式の差分解法

熱伝導方程式をもう一度示せば、

$$\partial u / \partial t = \kappa \partial^2 u / \partial x^2$$

でした。これを差分近似すれば、

$$(u_i^{n+1} - u_i^n) / \Delta t = \kappa (u_{i+1}^n - 2u_i^n + u_{i-1}^n) / (\Delta x)^2$$

となり、さらに、

$$u_i^{n+1} = u_i^n + \kappa \Delta t (u_{i+1}^n - 2u_i^n + u_{i-1}^n) / (\Delta x)^2$$

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。 $n+1$ 時間ステップの値を、 n 時間での値のみから計算する方法のことを、陽解法 (Explicit Method) と呼びますが、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、上記式の場合には、 $\kappa \Delta t / (\Delta x)^2$ 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。

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) 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。

$$(u_i^{n+1} - u_i^n) / \Delta t = \kappa \{ (u_{i+1}^n - 2u_i^n + u_{i-1}^n) / (\Delta x)^2 + (u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}) / (\Delta x)^2 \} / 2$$

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) と呼びます。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ます。 Δt をいくらでも大きく取れるということの意味します。ただし、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。

・ 6 - 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。

いま、熱伝導方程式を

$$\partial u / \partial t = \kappa \partial^2 u / \partial x^2$$

とします。陽解法で差分近似した式は、

$$u^{n+1}_j = u^n_j + \kappa \Delta t (u^n_{j+1} - 2u^n_j + u^n_{j-1}) / (\Delta x)^2$$

でした。フォン・ノイマンの線形安定性理論では、時間ステップ n における格子点 j の解 u^n_j 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、

$$u^n_j = G^n \exp(ji\theta)$$

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 についている 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)。これを差分近似式に代入すると、

$$G = 1 - 2\kappa \Delta t (1 - \cos \theta) / (\Delta x)^2$$

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、

$$\kappa \Delta t / (\Delta x)^2 \leq 1/2$$

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、 $1/2$ 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

$$\partial u / \partial t = \kappa \partial^2 u / \partial x^2$$

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、

$$\begin{aligned} (u^{n+1}_j - u^n_j) / \Delta t = \kappa \{ & (u^n_{j+1} - 2u^n_j + u^n_{j-1}) / (\Delta x)^2 \\ & + (u^{n+1}_{j+1} - 2u^{n+1}_j + u^{n+1}_{j-1}) / (\Delta x)^2 \} / 2 \end{aligned}$$

でした。陽解法のとおり同様に、

$$u^n_j = G^n \exp(ji\theta)$$

を代入して整理すれば、

$$G = \{1 - \kappa \Delta t (1 - \cos \theta) / (\Delta x)^2\} / \{1 + \kappa \Delta t (1 - \cos \theta) / (\Delta x)^2\}$$

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ない、すなわち無条件安定な方法であることがわかります。 κ や Δx は定数値ですから、 Δt の値が無限大にまで大きくできるということになりますが、実際には、初期値や境界条件の制約により無限大にとれるわけではありません。

6-5 直接法と反復法

熱伝導方程式の差分式において、 $\theta = \kappa \Delta t / (\Delta x)^2$ として変形すると、

$$-\theta u^{n+1}_{j-1} / 2 + (1 + \theta) u^{n+1}_j + \theta u^{n+1}_{j+1} / 2 = \theta u^n_{j-1} / 2 + (1 - \theta) u^n_j + \theta u^n_{j+1} / 2$$

となります。この式の右辺は、すでに既知の値です。既知量を簡単な記号に置き換えて、この式をよりわかりやすくすると、

$$a u^{n+1}_{j-1} + b u^{n+1}_j + c u^{n+1}_{j+1} = d_j$$

ただし、右辺は格子点 j ごとに値が違いますので、右辺のみ d_j とします。

この式は、

$$\begin{array}{c}
 \cdot \\
 \cdot \\
 \cdot \\
 a u_{j-2}^{n+1} + b u_{j-1}^{n+1} + c u_j^{n+1} = d_{j-1} \\
 a u_{j-1}^{n+1} + b u_j^{n+1} + c u_{j+1}^{n+1} = d_j \\
 a u_j^{n+1} + b u_{j+1}^{n+1} + c u_{j+2}^{n+1} = d_{j+1} \\
 \cdot \\
 \cdot \\
 \cdot
 \end{array}$$

のように、格子点の数だけある連立1次方程式であることがわかります。したがって、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差分近似式は、この連立1次方程式を解けば計算することができます。

連立1次方程式を解くとは、係数 a, b, c からなる行列 A と、既知量 d_j からなるベクトルから、 u_j^{n+1} からなるベクトルを求めることに等しいですから、結局、行列 A の逆行列を計算することに帰着されます。

このように各時間ステップ n ごとに逆行列を1回だけ計算する方法のことを、直接法 (Direct Method) と呼びます。直接法にはいくつかの方法がありますが、連立1次方程式の逆行列計算を体系化した方法を、ガウス消去法 (Gauss Elimination Method) と呼びます。また、逆行列を上三角形、下三角形領域に分割して計算する方法は、LU分解法 (Lower-upper Decomposition Method) と呼びます。

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、行列計算式自体が作れませんので、そのままでは直接法は用いることができません。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。

・ 6-6 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、

$$\begin{aligned}
 (u_i^{n+1} - u_i^n) / \Delta t = \kappa \{ & (u_{i+1}^n - 2u_i^n + u_{i-1}^n) / (\Delta x)^2 \\
 & + (u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}) / (\Delta x)^2 \} / 2
 \end{aligned}$$

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

$$u_i^{n+1} = \theta (u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}) / 2 + d_i$$

ただし、

$$d_i = u_i^n + \varrho(u_{i+1}^n - 2u_i^n + u_{i-1}^n)/2$$

になります。

ラプラス方程式の反復計算に用いた時間ステップ n における解は、収束解が得られるまで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。したがって、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 とは別に、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、上記式にヤコビ法を適用すると、

$$(1 + \varrho)(u_i^{m+1}) = \varrho[(u_{i+1}^m + u_{i-1}^m)]/2 + d_i$$

と定義される式が導出されます。 d_i は既知量ですから、反復法の計算からは、はずされます。 u_i^{m+1} の値を求めるために、 m を $1, 2, 3, \dots$ と増やして計算し、 m と $m+1$ の値が同じになったとき、 u_i^{m+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、ヤコビ法をガウス・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、

$$(1 + \varrho)(u_i^{m+1}) = \varrho\{(u_{i+1}^m + u_{i-1}^{m+1})\}/2 + d_i$$

と導出することができます。

同じく、ガウス・ザイデル法をSOR法に改良することができます。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、

$$(1 + \varrho)(u_i^{m+1}) = (1 - \omega)(1 + \varrho)(u_i^m) + \omega[\varrho\{(u_{i+1}^m + u_{i-1}^{m+1})\}/2 + d_i]$$

となります。

一般的に、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ます。

・ 6 - 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簡単に計算することができます。しかしながら、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。

$$\partial u / \partial t = \kappa (\partial^2 u / \partial x^2 + \partial^2 u / \partial y^2)$$

これを陽解法で差分近似すると、

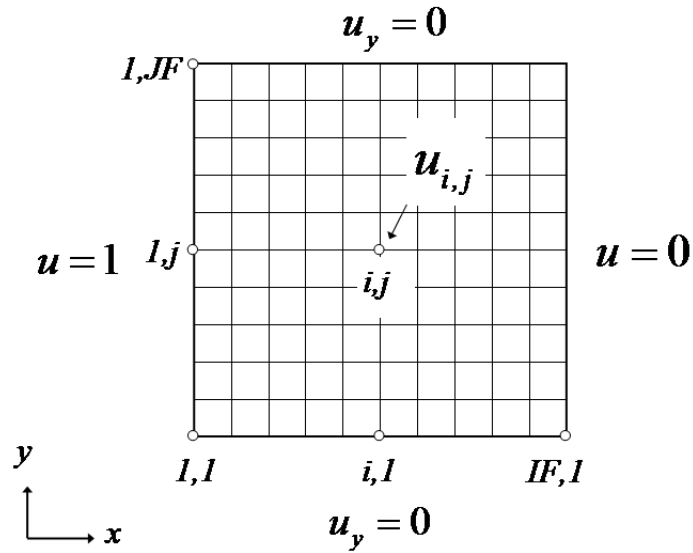
$$(u_{i,j}^{m+1} - u_{i,j}^m) / \Delta t = \kappa [(u_{i+1,j}^m - 2u_{i,j}^m + u_{i-1,j}^m) / (\Delta x)^2 + (u_{i,j+1}^m - 2u_{i,j}^m + u_{i,j-1}^m) / (\Delta y)^2]$$

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

$$u^{n+1}_{i,j} = u^n_{i,j} + \varrho (u^n_{i+1,j} + u^n_{i-1,j} + u^n_{i,j+1} + u^n_{i,j-1} - 4u^n_{i,j})$$

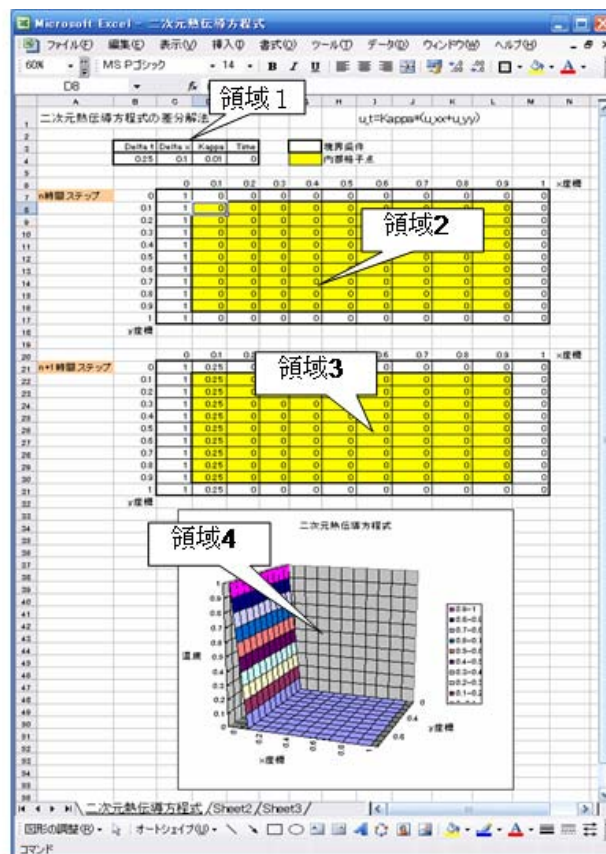
と比較的簡単な式を導出することができます。

具体的な熱伝導問題として、次の図に示す境界条件を設定します。



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、次に示します。



シートは大きく分けて、初期値を与える「領域1」、 n 時間ステップの計算結果を保管する「領域2」、 $n+1$ 時間ステップ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 n 時間ステップ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 C17 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 $n+1$ 時間ステップ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、 $n+1$ 時間ステップで計算された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、 $fx= D22$ と置きます。

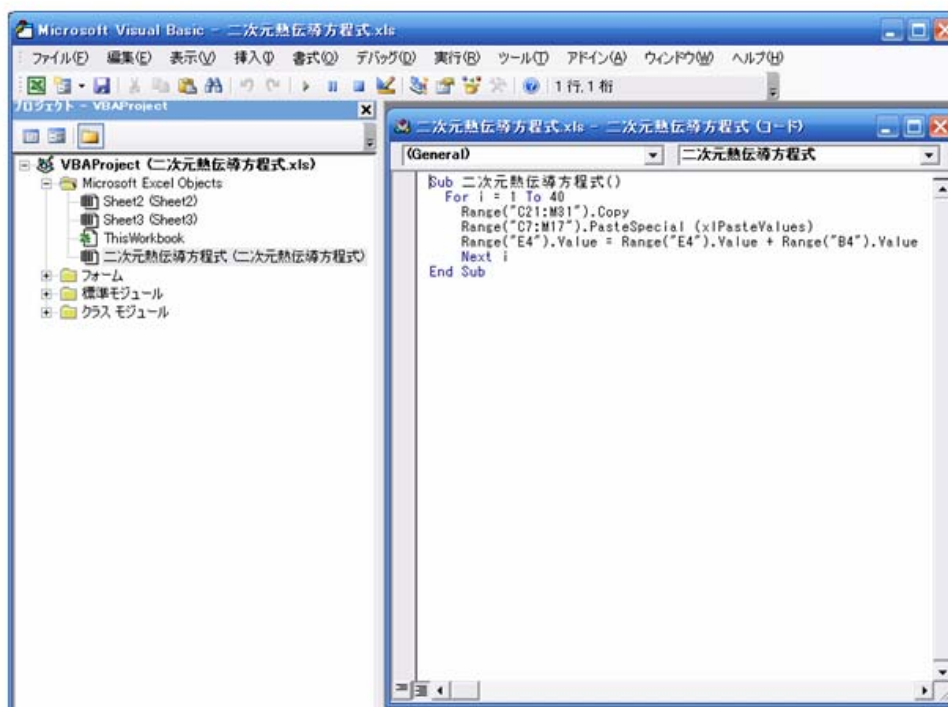
境界と内点の計算式を用いた計算がすべて終了した段階で、領域3には $n+1$ 時間ステップの計算結果が保管されます。

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。

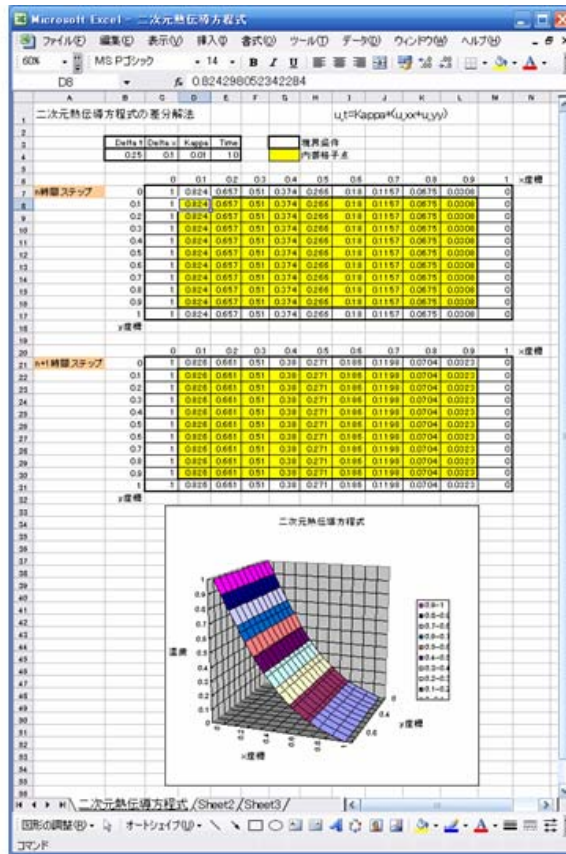
このプログラムを、Excel のツールバーにある、ツール/マクロ/Visual Basic Editor を立ち上げて、左ボックスに表示された、Sheet1 をダブルクリックして開き、その中にコピーします。



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。

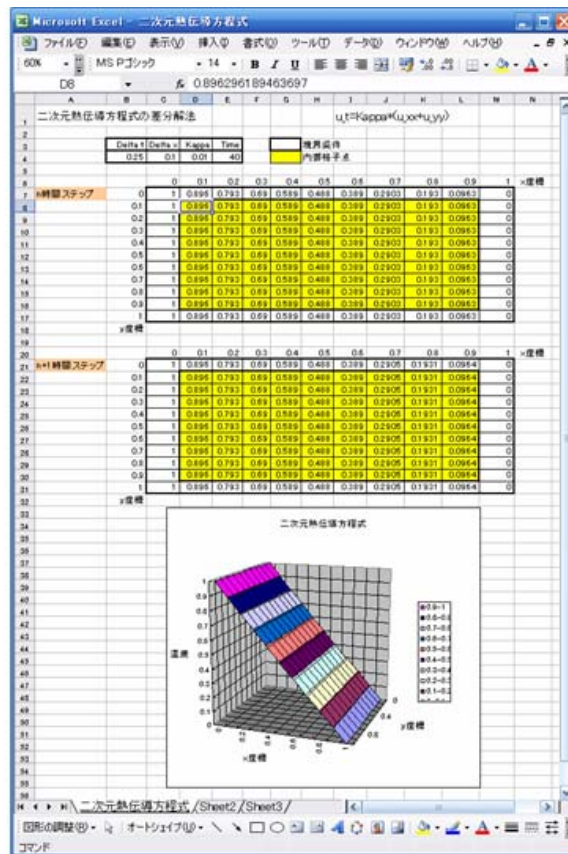
マクロプログラムを実行した結果を示します。このマクロプログラムでは、40 時間ステップ計算するように設定されており、その間、領域 3 の二次元セル領域 C21:M31 の値を、領域 2 の二次元セル領域 C7:M17 へのコピーを繰り返し、かつセル番号 E4 にある Time にセル番号 B4 にある Delta t の値を加算します。Time の値は最終的には、 $Time=0.25 \times 40=10$ となります。時間ステップの数を増やしたい場合には、繰り返し回数の数値を変更するか、繰り返しマクロプログラムを実行します。

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。



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。



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ので、ご自由にお使いください

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

・ 6 - 8 三次元熱伝導方程式の差分解法

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 で計算するのは現実的ではありません。

ここでは、三次元熱伝導方程式をクラunkに・ニコルソン法で差分近似して、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。

$$\partial T / \partial t = \kappa (\partial^2 T / \partial x^2 + \partial^2 T / \partial y^2 + \partial^2 T / \partial z^2)$$

T は温度です。これをクラunk・ニコルソン法で差分近似すると、

$$\begin{aligned} (T^{n+1}_{i,j,k} - T^n_{i,j,k}) / \Delta t = & \kappa [(T^{n+1}_{i+1,j,k} - 2T^{n+1}_{i,j,k} + T^{n+1}_{i-1,j,k}) / (\Delta x)^2 \\ & + (T^{n+1}_{i,j+1,k} - 2T^{n+1}_{i,j,k} + T^{n+1}_{i,j-1,k}) / (\Delta y)^2 \\ & + (T^{n+1}_{i,j,k+1} - 2T^{n+1}_{i,j,k} + T^{n+1}_{i,j,k-1}) / (\Delta z)^2 \\ & + (T^n_{i+1,j,k} - 2T^n_{i,j,k} + T^n_{i-1,j,k}) / (\Delta x)^2 \\ & + (T^n_{i,j+1,k} - 2T^n_{i,j,k} + T^n_{i,j-1,k}) / (\Delta y)^2 \end{aligned}$$

$$+ (T^n_{i,j,k+1} - 2T^n_{i,j,k} + T^n_{i,j,k-1}) / (\Delta z)^2] / 2$$

とたいへん長い式になります。

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、

$$\begin{aligned} L(T^{n+1}_{i,j,k})^{m+1} = & (1 - \omega) L(T^n_{i,j,k})^m \\ & + \omega \kappa [\varrho_x \{ (T^{n+1}_{i+1,j,k})^m + (T^{n+1}_{i-1,j,k})^{m+1} \} / 2 \\ & + \varrho_y \{ (T^{n+1}_{i,j+1,k})^m + (T^{n+1}_{i,j-1,k})^{m+1} \} / 2 \\ & + \varrho_z \{ (T^{n+1}_{i,j,k+1})^m + (T^{n+1}_{i,j,k-1})^{m+1} \} / 2 + d_{i,j,k}] \end{aligned}$$

式が複雑になりわかりにくいですが、

$$\begin{aligned} d_{i,j,k} = & T^n_{i,j,k} + \varrho_x (T^n_{i+1,j,k} - 2T^n_{i,j,k} + T^n_{i-1,j,k}) / 2 \\ & + \varrho_y (T^n_{i,j+1,k} - 2T^n_{i,j,k} + T^n_{i,j-1,k}) / 2 \\ & + \varrho_z (T^n_{i,j,k+1} - 2T^n_{i,j,k} + T^n_{i,j,k-1}) / 2 \end{aligned}$$

ただし、

$$\begin{aligned} \varrho_x = \Delta t / (\Delta x)^2, \quad \varrho_y = \Delta t / (\Delta y)^2, \quad \varrho_z = \Delta t / (\Delta z)^2 \\ L = 1 + \varrho_x + \varrho_y + \varrho_z \end{aligned}$$

式をさらに変形すると、

$$\begin{aligned} (T^{n+1}_{i,j,k})^{m+1} = & (T^n_{i,j,k})^m + \omega \kappa [\varrho_x \{ (T^{n+1}_{i+1,j,k})^m + (T^{n+1}_{i-1,j,k})^{m+1} \} / 2L \\ & + \varrho_y \{ (T^{n+1}_{i,j+1,k})^m + (T^{n+1}_{i,j-1,k})^{m+1} \} / 2L \\ & + \varrho_z \{ (T^{n+1}_{i,j,k+1})^m + (T^{n+1}_{i,j,k-1})^{m+1} \} / 2L \\ & + d_{i,j,k} / L - (T^n_{i,j,k})^m] \end{aligned}$$

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

計算例として、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。

立方体内外の熱伝導率はそれぞれ10、1と設定します。境界条件は、入口境界 $i = 1$ 面で $T = 100$ 度、出口境界 $i = IF$ 面で $T = 50$ 度とします。それ以外の外部境界は、法線方向に $\partial T / \partial n = 0$ となるようノイマン境界条件を与える。初期値として、入口境界以外の点で $T = 50$ 度を与えます。格子間隔は $\Delta x = \Delta y = \Delta z = 0.1$ で、過緩和係数は $\omega = 1.5$ とします。

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/datafiles/>

・6-9 特性方程式

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定義すると、

$$A \partial^2 u / \partial x^2 + 2B \partial^2 u / \partial x \partial y + C \partial^2 u / \partial y^2 = f(x, y, u, \partial u / \partial x, \partial u / \partial y)$$

になります。

いま、偏導関数を、

$\partial u / \partial x = p, \partial u / \partial y = q, \partial^2 u / \partial x^2 = r, \partial^2 u / \partial x \partial y = s, \partial^2 u / \partial y^2 = t$
とおいて、上式に代入します。すると、

$$Ar + 2Bs + Ct = f$$

になります。

ところで、 p, q の全微分は、

$$dp = \partial p / \partial x \cdot dx + \partial p / \partial y \cdot dy = r dx + s dy$$

$$dq = \partial q / \partial x \cdot dx + \partial q / \partial y \cdot dy = s dx + t dy$$

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、

$$A(dp - sdy) / dx + 2Bs + C(dq - sdx) / dy = f$$

が得られます。これをさらに s で整理すると、

$s\{A(dy/dx)^2 - 2B(dy/dx) + C\} - \{A(dp/dx)(dy/dx) + Cdq/dx - f dy/dx\} = 0$
となります。この式がすべての s に対して常に成り立つためには、

$$A(dy/dx)^2 - 2B(dy/dx) + C = 0$$

でかつ、

$$A(dp/dx)(dy/dx) + Cdq/dx - f dy/dx = 0$$

でなければなりません。前者の式のことを、特性方程式 (Characteristics Equation) と呼びます。

特性方程式は、 dy/dx の 2 次方程式になっています。したがって、根は、

$$dy/dx = \{B + (B^2 - AC)^{1/2}\} / A$$

ならびに、

$$dy/dx = \{B - (B^2 - AC)^{1/2}\} / A$$

になります。

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、

$$D > 0 \text{ なら、双曲型 (Hyperbolic)}$$

$$D = 0 \text{ なら、放物型 (Parabolic)}$$

$$D < 0 \text{ なら、楕円型 (Elliptic)}$$

となることを紹介しました。

実は、この D の値は上記根の平方根の中にある値です。すなわち、 $D > 0$ なら、二つの実根、 $D = 0$ なら重根、 $D < 0$ なら二つの複素根になります。

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

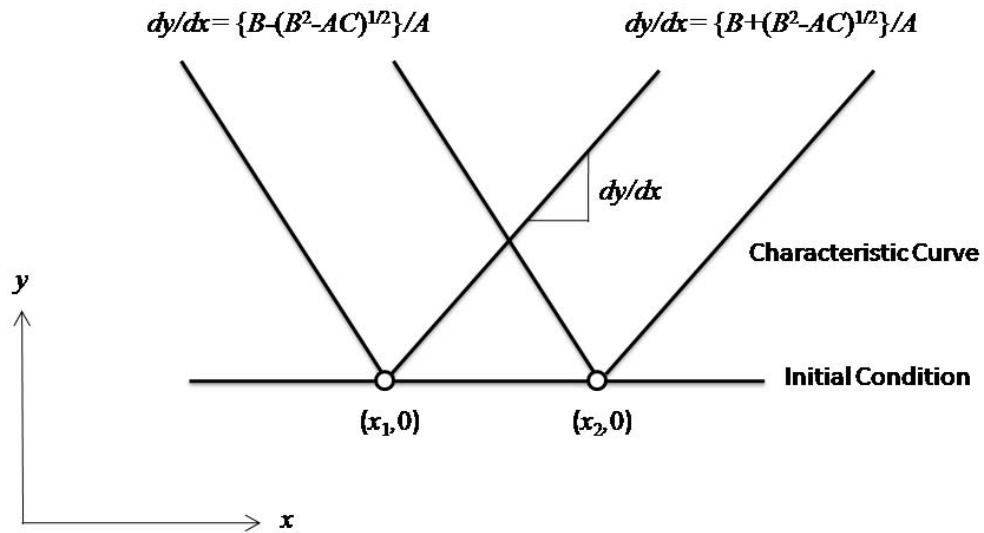
に分類されます。

特性方程式の根は、係数 A, B, C が定数なら定数になります。 $\lambda = dy/dx$ とおけば、

$$A \lambda (dp/dx) + Cdq/dx - f \lambda = 0$$

となり、独立変数が x だけですので、この式は常微分方程式になります。結局、偏微分方程式が常微分方程式に簡単化されました。

特性方程式の根をわかりやすく説明するために、下のように図示してみます。

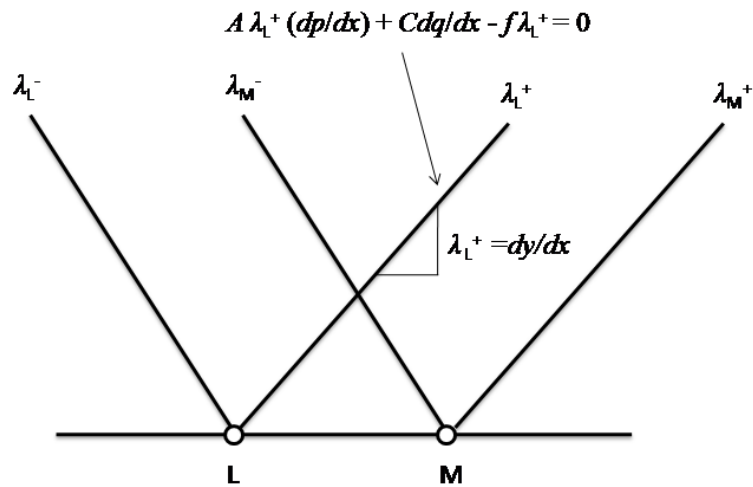


独立変数 (x, y) からなる 2 階偏微分方程式は、 xy 空間上で図示することができます。

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るれば、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) と呼びます。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。

特性曲線の図を簡略化して、改めて示します。



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図示します。たとえば、L点から伝播した解を示す2つの直線の内、 $dy/dx = \{B + (B^2 - AC)^{1/2}\} / A$ の勾配を、 λ_L^+ とします。これを言い換えれば、「特性曲線の勾配」です。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。特性方程式の根とは、すなわち、特性曲線の勾配です。 xy 空間は、特性方程式の根を勾配に持つ特性曲線で埋め尽くされています。

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、

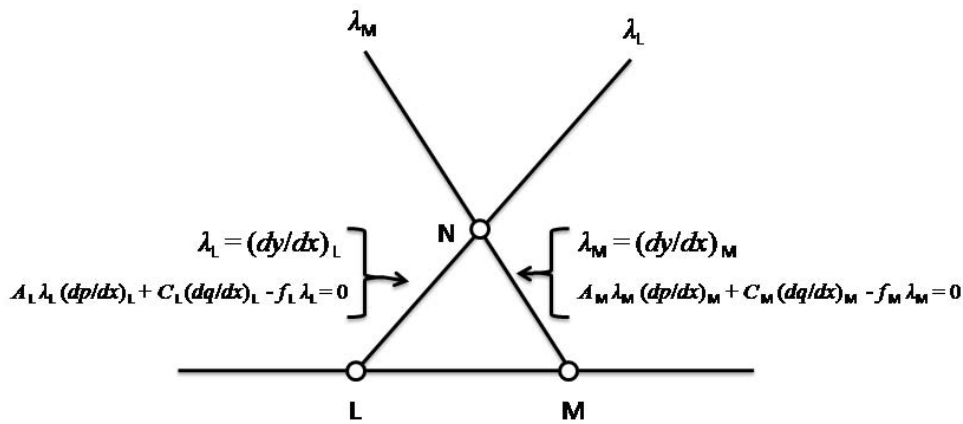
$$A \lambda_L^+ (dp/dx) + C dq/dx - f \lambda_L^+ = 0$$

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、解析的に解くこともできます。

・ 6 - 10 特性曲線法

特性曲線法 (Method of Characteristic Curves)について紹介します。特性曲線法とは、偏微分方程式を特性曲線上で解く方法です。コンピュータが進歩した今日では、あまり使われない方法ですが、むしろコンピュータがまだなかった時代に、手計算やタイガー計算機を用いて、広く用いられていた方法です。特に双曲型偏微分方程式の特徴を理解する上で、大変役に立つ方法です。

準備として、次の図を用意します。



図中には、2本の特性曲線が示されています。まず、点Lから延びた $dy/dx = \{B + (B^2 - AC)^{1/2}\}/A$ を勾配を持った特性曲線を λ_L 、点Mから延びた $dy/dx = \{B - (B^2 - AC)^{1/2}\}/A$ を勾配を持った特性曲線を λ_M とします。さらに、これらの特性曲線が交差した点を、点Nとします。それぞれの特性曲線上では、図中に示したそれぞれ2つの式が成り立ちます。すなわち、

λ_L では、

$$\lambda_L = (dy/dx)_L$$

ならびに

$$A_L \lambda_L (dp/dx)_L + C_L (dq/dx)_L - f_L \lambda_L = 0$$

λ_M では、

$$\lambda_M = (dy/dx)_M$$

ならびに

$$A_M \lambda_M (dp/dx)_M + C_M (dq/dx)_M - f_M \lambda_M = 0$$

が成り立ちます。これら4つの式を連立して解くことにより、解を求めることができます。

λ_L では、

$$\lambda_L = (dy/dx)_L$$

ですから、これが点Nを通るとすれば、次の差分近似式が成り立ちます。

$$\lambda_L = (y_N - y_L) / (x_N - x_L)$$

少し書き換えれば、

$$y_N - y_L = \lambda_L (x_N - x_L)$$

同様に、 λ_M では、

$$\lambda_M = (dy/dx)_M$$

ですから、これが点Nを通るとすれば、

$$\lambda_M = (y_N - y_M)/(x_N - x_M)$$

で、

$$y_N - y_M = \lambda_M(x_N - x_M)$$

です。ところで、点L,Nにおける (x_L, y_L) , (x_M, y_M) は既知の値です。未知の値は (x_N, y_N) です。したがって、

$$y_N - y_L = \lambda_L(x_N - x_L)$$

$$y_N - y_M = \lambda_M(x_N - x_M)$$

の2式を連立して計算することにより、 (x_N, y_N) を求めることができます。

次に、 λ_L では、

$$A_L \lambda_L (dp/dx)_L + C_L (dq/dx)_L - f_L \lambda_L = 0$$

λ_M では、

$$A_M \lambda_M (dp/dx)_M + C_M (dq/dx)_M - f_M \lambda_M = 0$$

の各式が成り立ちますので、それぞれを、直線LN、LM上で差分近似します。

すると、

$$A_L \lambda_L (p_N - p_L)/(x_N - x_L) + C_L (q_N - q_L)/(x_N - x_L) - f_L \lambda_L = 0$$

$$A_M \lambda_M (p_N - p_M)/(x_N - x_M) + C_M (q_N - q_M)/(x_N - x_M) - f_M \lambda_M = 0$$

の2式が得られます。

これらの式における未知変数は (p_N, q_N) のみです。

したがって、上記2式を連立することで (p_N, q_N) を求めることができます。

もともと、 p, q は、 $p = \partial u / \partial x$, $q = \partial u / \partial y$ でした。点Nにおける u_N の値を求めるために、 u の全微分式、

$$du = \partial u / \partial x \cdot dx + \partial u / \partial y \cdot dy$$

を差分近似します。すなわち、

$$u_N - u_L = \{(p_N + p_L)(x_N - x_L) + (q_N + q_L)(y_N - y_L)\} / 2$$

ただし、 p, q の値は、点Lと点Nにおける値を平均します。

また、直線MNで成り立つ差分近似式

$$u_N - u_M = \{(p_N + p_M)(x_N - x_M) + (q_N + q_M)(y_N - y_M)\} / 2$$

を解いてもかまいません。解は同じになります。これより、点Nにおける解 u_N を求めることができます。特性曲線法では、点Nの解を求める計算を他の任意の点でも行い、求められた複数の点における解を用いて、さらに次の計算を同様に繰り返すことにより、解の伝播を手計算で求めることができます。

双曲型方程式

$$\partial^2 u / \partial x^2 - u^2 \partial^2 u / \partial y^2 = 0$$

を特性曲線法で具体的に解いてみます。まず、 $D = B^2 - AC = u^2 > 0$ ですので、この方程式は双曲型です。

いま、 $x=0.2$ にある点L、 $x=0.3$ にある点Mの間にあり、 $y > 0$ にある特性曲線が交叉した点Nの座標 (x_N, y_N) における u_N を求めてみます。ただし、 u は初期値 $y = 0$ 上で、 $0 \leq x \leq 1$ のとき、

$$u = 0.2 + 5x^2$$

で、かつ

$$\partial u / \partial y = 3x$$

を満足するとします。

まず、偏導関数 p, q は、

$$p = du/dx = 10x, \quad q = du/dy = 3x$$

と求まります。次に特性方程式は、

$$(dy/dx)^2 - u^2 = 0$$

になります。したがって、この根は、

$$\lambda = dy/dx = u,$$

ならびに

$$\lambda = dy/dx = -u$$

です。

つぎに、 x_N, y_N を求めます。

$$\lambda_L = u_L = 0.2 + 5 \cdot 0.2^2 = 0.4$$

$$\lambda_M = -u_M = -0.2 - 5 \cdot 0.3^2 = -0.65$$

ですので、

$$y_N - 0 = 0.4(x_N - 0.2)$$

$$y_N - 0 = -0.65(x_N - 0.3)$$

となり、これらを連立して解けば、

$$x_N = 0.26190, \quad y_N = 0.024562$$

と点Nの座標が求まります。

次に、点Nにおける p_N, q_N の値を求めます。

その際に、

$$A_L \lambda_L (p_N - p_L) / (x_N - x_L) + C_L (q_N - q_L) / (x_N - x_L) = 0$$

$$A_M \lambda_M (p_N - p_M) / (x_N - x_M) + C_M (q_N - q_M) / (x_N - x_M) = 0$$

を連立して解く必要があります。ただし、これらの式は、

$$A_L \lambda_L (p_N - p_L) + C_L (q_N - q_L) = 0$$

$$A_M \lambda_M (p_N - p_M) + C_M (q_N - q_M) = 0$$

のように簡略化できます。

$$A_L = A_M = 1$$

$$C_L = -u_L^2 = -0.4^2 = -0.16, \quad C_M = -u_M^2 = -(-0.65)^2 = 0.4225$$

$$p_L = 10 \cdot 0.2 = 2, \quad p_M = 10 \cdot 0.3 = 3, \quad q_L = 3 \cdot 0.2 = 0.6, \quad q_M = 3 \cdot 0.3 = 0.9$$

ですので、結局、

$$0.4(p_N - 2.0) - 0.16(q_N - 0.6) = 0$$

$$-0.65(p_N - 3.0) - 0.4225(q_N - 0.9) = 0$$

から、 p_N , q_N を求めれば、

$$p_N = 2.45524, \quad q_N = 1.73810$$

となります。

求められた、 x_N , y_N , p_N , q_N を、

$$u_N - u_L = (p_N + p_L)(x_N - x_L) + (q_N + q_L)(y_N - y_L)$$

に代入すれば、

$$u_N = 0.4 + (2.0 + 2.45524)(0.26190 - 0.2) / 2 \\ + (0.6 + 1.73810)(0.024762 - 0) / 2$$

で、これを計算すれば、

$$u_N = 0.56684$$

となります。これが、特性曲線法により求められた解です。

・ 7 数理モデリングと計算

・ 7-1 非平衡凝縮流れの数理モデル

現在、我々が研究しているテーマの一つに非平衡凝縮流れの数値シミュレーションがあります。ここでは、その数値解法で用いられている数理モデルについて簡単に紹介します。非平衡凝縮流れの研究は、1887年にR. Von Helmholtzが湿り蒸気の過飽和現象を発見した辺りに始まります。その後、1897年にWilsonが非平衡凝縮現象を発見しました。1950年から1980年にかけて、数多くの実験データや凝縮モデルが発表され、非平衡凝縮流れの数値計算も数多く発表されています。ただし、詳細な非平衡凝縮モデルの説明はここでは省略します。むしろ、支配方程式がこれまで紹介した偏微分方程式系の形をしていることを、ここでは簡単に紹介します。

非平衡凝縮は、湿り空気ならびに湿り蒸気中で起こります。その流れの支配方程式としては、混合気体の質量、運動量、エネルギー保存式に、水蒸気の質量、液滴の質量、ならびに液滴の数密度の保存式を解きます。具体的には、まず混合気体の質量、運動量、エネルギー保存式として、

$$\partial \rho / \partial t + \partial (\rho u) / \partial x + \partial (\rho v) / \partial y + \partial (\rho w) / \partial z = 0$$

$$\partial (\rho u) / \partial t + \partial (\rho uu) / \partial x + \partial (\rho vu) / \partial y + \partial (\rho wu) / \partial z \\ = -\partial p / \partial x + \partial \tau_{xx} / \partial x + \partial \tau_{xy} / \partial y + \partial \tau_{xz} / \partial z$$

$$\partial (\rho v) / \partial t + \partial (\rho uv) / \partial x + \partial (\rho vv) / \partial y + \partial (\rho wv) / \partial z \\ = -\partial p / \partial y + \partial \tau_{yx} / \partial x + \partial \tau_{yy} / \partial y + \partial \tau_{yz} / \partial z$$

$$\partial (\rho w) / \partial t + \partial (\rho uw) / \partial x + \partial (\rho vw) / \partial y + \partial (\rho ww) / \partial z \\ = -\partial p / \partial z + \partial \tau_{zx} / \partial x + \partial \tau_{zy} / \partial y + \partial \tau_{zz} / \partial z$$

$$\begin{aligned}
& \partial e / \partial t + \partial (ue) / \partial x + \partial (ve) / \partial y + \partial (we) / \partial z \\
& = - \partial (up) / \partial x - \partial (vp) / \partial y - \partial (wp) / \partial z - \partial (up) / \partial x \\
& \quad + \partial (u\tau_{xx}) / \partial x + \partial (v\tau_{xy}) / \partial y + \partial (w\tau_{xz}) / \partial z \\
& \quad + \partial (u\tau_{yx}) / \partial x + \partial (v\tau_{yy}) / \partial y + \partial (w\tau_{yz}) / \partial z \\
& \quad + \partial (u\tau_{zx}) / \partial x + \partial (v\tau_{zy}) / \partial y + \partial (w\tau_{zz}) / \partial z \\
& \quad - \partial (uq_x) / \partial x - \partial (vq_y) / \partial y - \partial (wq_z) / \partial z
\end{aligned}$$

が解かれます。ただし、 τ_{xx} 、 τ_{xy} 、 τ_{xz} 、 \dots は粘性応力成分、 e は単位体積当たりの全内部エネルギー、 q_x 、 q_y 、 q_z は熱流束成分です。この偏微分方程式系は、通常の圧縮性ナビエ・ストークス方程式と表面的には同じです。非平衡凝縮流れの支配方程式では、これに加えて、水蒸気の質量、液滴の質量、ならびに液滴の数密度の保存式として

$$\begin{aligned}
& \partial \rho_v / \partial t + \partial (\rho_v u) / \partial x + \partial (\rho_v v) / \partial y + \partial (\rho_v w) / \partial z = - S_w \\
& \partial \rho_w / \partial t + \partial (\rho_w u) / \partial x + \partial (\rho_w v) / \partial y + \partial (\rho_w w) / \partial z = S_w \\
& \partial (\rho n) / \partial t + \partial (\rho n u) / \partial x + \partial (\rho n v) / \partial y + \partial (\rho n w) / \partial z = S_n
\end{aligned}$$

が連立して解かれます。このうち、水蒸気の質量保存式と液滴の質量保存式には符号の違う同じ生成項 S_w が付いています。 S_w は液滴の生成速度を近似する数理モデルであり、 S_w の増加により、液滴の質量 ρ_w が増加して、その分水蒸気の質量 ρ_v が減少するような偏微分方程式系になっています。 S_w の中には、非平衡凝縮における均一核生成率ならびに凝縮核の成長がモデル化されています。

・ 7-2 反応拡散方程式

反応方程式と拡散方程式については、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です。すなわち、

$$\partial u / \partial t = \kappa \partial^2 u / \partial x^2 + a u$$

の形をしている偏微分方程式です。ここで、 κ は拡散係数、 a は反応係数です。

典型的な反応拡散方程式としては、たとえば、

$$\partial u / \partial t = \varepsilon^2 \partial^2 u / \partial x^2 + u(u - a)(1 - u)$$

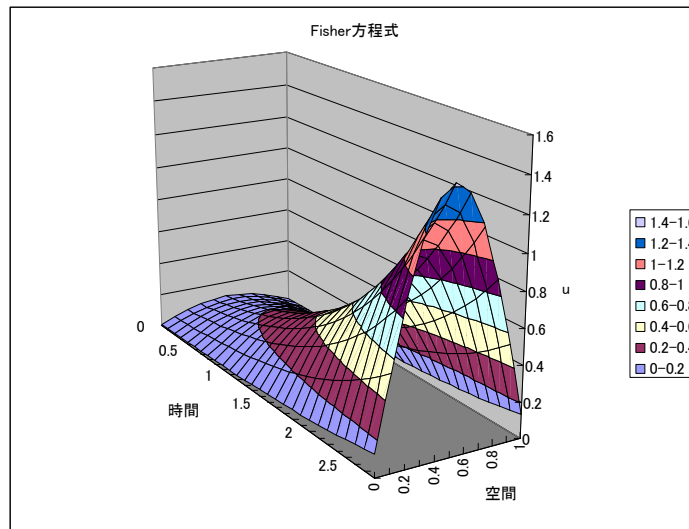
のように反応項が三次関数になっているものがあります。これは、Nagumo 方程式と呼ばれ、神経系の信号伝播を模擬する反応拡散方程式型の数理モデルです。また、

$$\partial u / \partial t = \varepsilon^2 \partial^2 u / \partial x^2 + a u(1 - u)$$

は、Fisher 方程式と呼ばれ、生態系の増殖・拡散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となり、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。

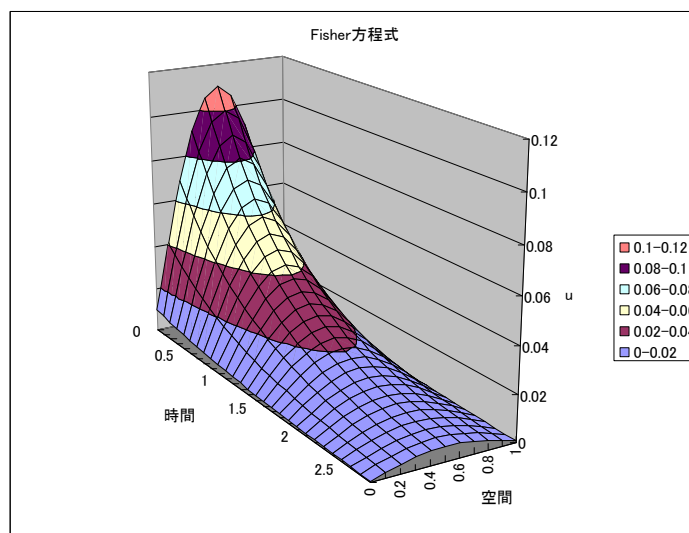
いま、Fisher 方程式を解いてみましょう。

$0 \leq x \leq 1$ 、 $\varepsilon = 0.001$ 、 $a = -1$ 、 $u(x, 0) = 0.01 + 0.1 \sin(\pi x)$ として、 $\Delta t = 0.1$ で時間を進行させます。その結果、 $t = 3$ のときに次のような図の結果が得られます。



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。



・ 7 - 3 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。

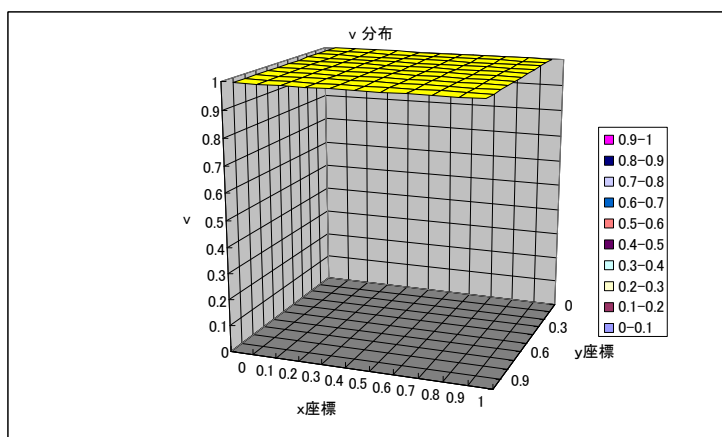
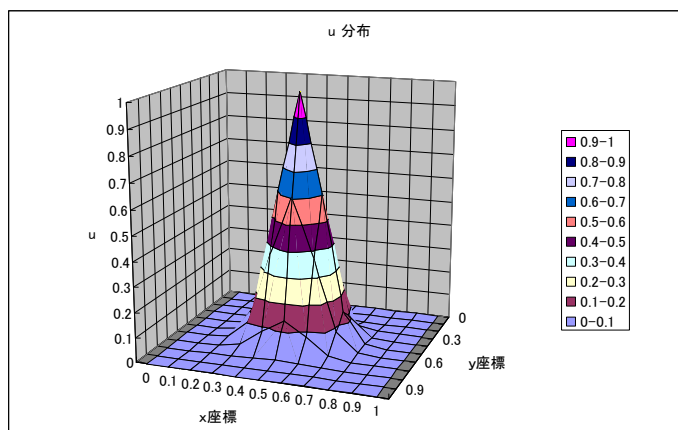
ここでは、次のような二次元の反応拡散方程式系を考えます。

$$\partial u / \partial t = \varepsilon^2 (\partial^2 u / \partial x^2 + \partial^2 u / \partial y^2) + avu$$

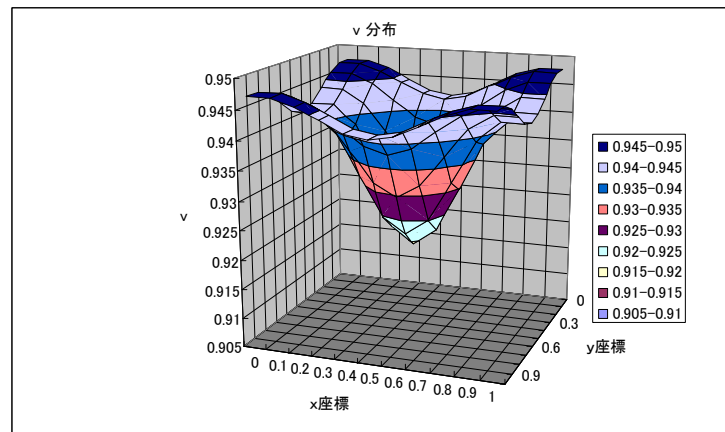
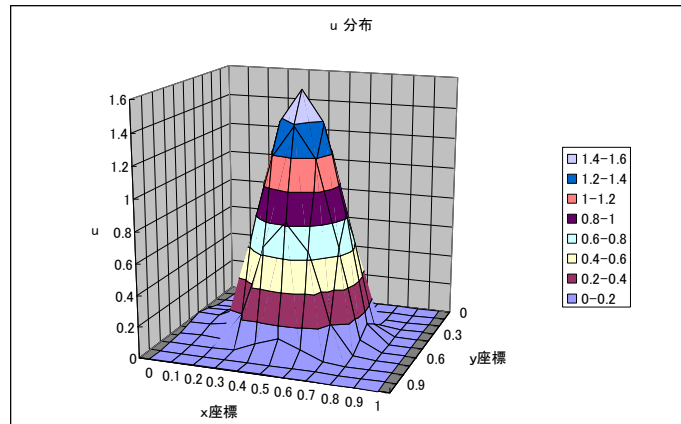
$$\partial v / \partial t = \partial^2 v / \partial x^2 + \partial^2 v / \partial y^2 - uv$$

$0 \leq x \leq 1$ 、 $0 \leq y \leq 1$ で、 $\varepsilon = 0.2$ 、 $a = 5$ として、境界条件は u 、 v それぞれ、 $\partial u / \partial n = 0$ 、 $\partial v / \partial n = 0$ とします。ただし、 n は境界の法線方向座標です。また、初期値は、 $u(x, y, 0) = \exp\{[(x - 0.5)^2 + (y - 0.5)^2] * 50\}$ ならびに $v = 0$ とします。

初期値を次のように図示されます。



この初期値から、 $\Delta t = 0.001$ で時間を進行させて、 $t = 0.3$ まで計算した結果は次のように図示されます。



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。

・ 7-4 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の基本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事象をモデリングすることができます。

ここでは、最終的な応用例として、栄養補給しているバクテリアとそれを捕食・自爆する白血球からなる生態系を三次元で数理モデル化します。これの元になるものは、すでに論文に発表されている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す。

W. Alt and D. A. Lauffenberger, J. Math. Biology, 24(1987), 691-722.

それに前節の栄養源を追加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。

$$\begin{aligned}\frac{\partial u}{\partial t} &= \alpha n u (\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}) + (c_{u1} - c_{u2} w) u + c_{n1} n u \\ \frac{\partial n}{\partial t} &= \varepsilon (\frac{\partial^2 n}{\partial x^2} + \frac{\partial^2 n}{\partial y^2} + \frac{\partial^2 n}{\partial z^2}) - c_{n1} u n \\ \frac{\partial v}{\partial t} &= \beta (\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}) + c_{v1} u \\ \frac{\partial w}{\partial t} &= \gamma (\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2}) + (c_{w1} + c_{w2} u) w \\ &\quad - \partial \{ \theta w (\frac{\partial v}{\partial x}) \} / \partial x - \partial \{ \theta w (\frac{\partial v}{\partial y}) \} / \partial y \\ &\quad - \partial \{ \theta w (\frac{\partial v}{\partial z}) \} / \partial z\end{aligned}$$

u はバクテリアの個体数、 n は栄養源の濃度、 v はバクテリアが排出する化学誘因物質の濃度、 w は白血球の個体数。 α 、 ε 、 β 、 γ はそれぞれの拡散係数。 c_{u1} 、 c_{u2} 、 c_{n1} 、 c_{v1} 、 c_{w1} 、 c_{w2} は反応項の経験定数です。さらに、白血球の個体数の偏微分方程式には化学誘因物質 v に起因する拡散項が付加されており、その θ はその拡散係数です。

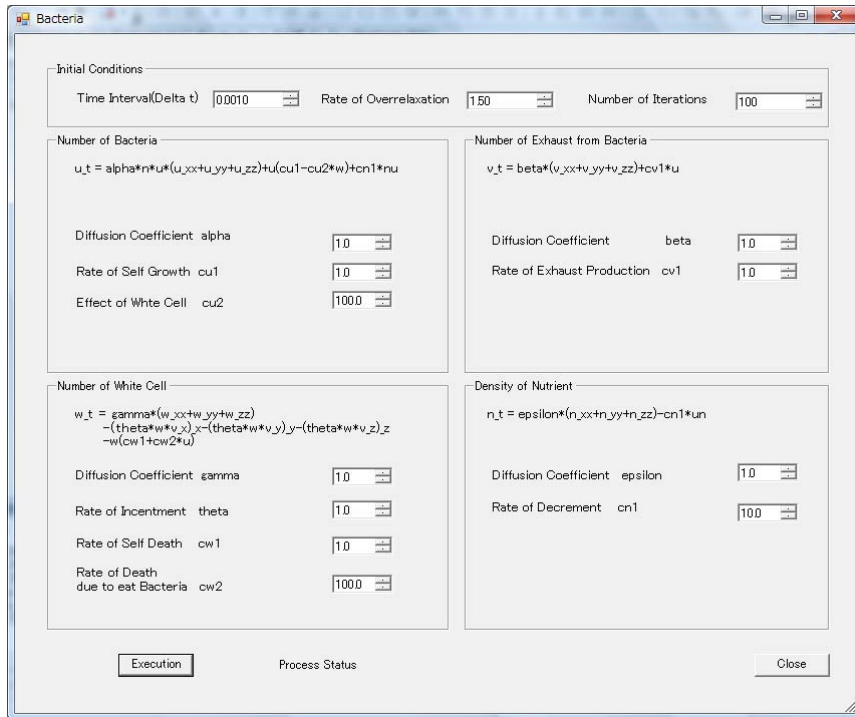
これらの偏微分方程式が模擬している物理メカニズムを簡単に説明すれば、まずバクテリアは栄養源を吸収しながら成長します。それに伴い、バクテリアは老廃物(化学誘因物質)を排出します。一方、白血球はバクテリアが排出する老廃物である化学物質の濃度を検出して濃度が高い方に移動する性質があります。そしてバクテリアを見つけると、それを捕食して、かつ自分自身も自爆します。この一連のメカニズムを模擬しています。ただし、多くの拡散係数や経験定数からなっているため、それら係数の設定がたいへんです。その係数を物理現象や社会事象に合わせて如何に設定するかというのが、実は数理モデリングの本質といえるでしょう。

計算プログラムは、Visual Studio2005 と Intel Fortran による環境で開発しました。先に紹介した下記の三次元熱伝導方程式の差分解法がベースになっています。

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

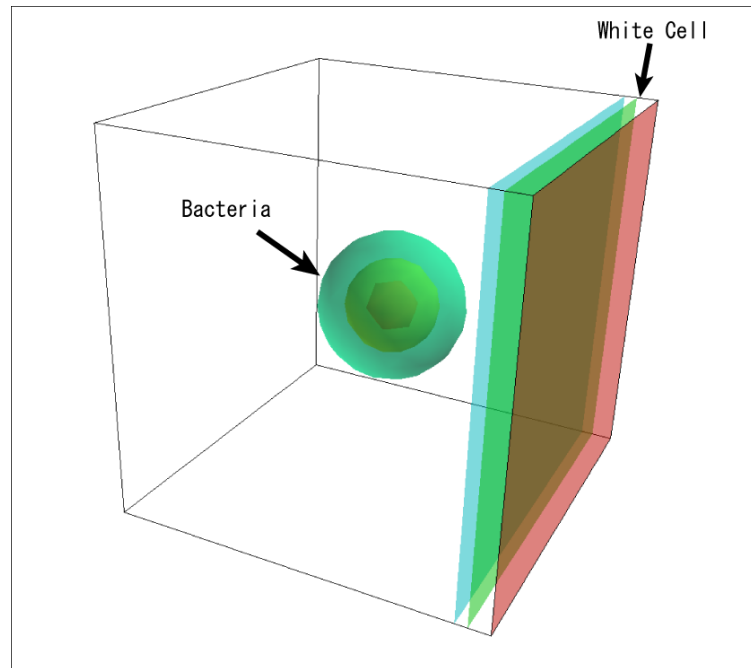
これを Windows に移植して、次に示す GUI 環境から計算できるソフトウェアを開発しました。この画面から、各反応拡散方程式の係数、反復回数、過緩和係数、時間間隔を設定して Execution ボタンを押すと計算が始まります。

いま、下記画面に設定された係数を用いて、具体的に計算してみます。



$0 \leq x \leq 2, 0 \leq y \leq 2, 0 \leq z \leq 2$ の無次元空間で、バクテリア分布の初期値として、 $u(x, y, z, 0) = \exp[\{(x - 1.0)^2 + (y - 1.0)^2 + (z - 1.0)^2\} * 10]$ 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、 $x = 0$ 面に、 $w(0, y, z, 0) = 1.0$ を与え、その他はゼロとしました。計算格子点数は $20 \times 20 \times 20$ とします。

バクテリアと白血球の初期値分布を下の図に示します。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。



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下の図に示します。白血球が細菌に迫ってきている様子が見られるのと同時に、細菌はその分布をお丸形に変形しながら後退している様子が見られています。この解の信憑性についてはなんとも言えない状況ですが、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得られるところはたいへん興味深いものがあります。

