

水底構造物を越える安定成層流

衛藤, 鑑

九州大学大学院生物資源環境科学府生産環境科学専攻生産環境情報学研究室

<https://hdl.handle.net/2324/1498281>

出版情報 : 九州大学, 2001, 修士, 修士
バージョン :
権利関係 :

修士論文

水底構造物を越える安定成層流

九州大学大学院生物資源環境科学府生産環境科学専攻

生産環境情報学講座

衛藤 鑑

2002年3月

目次

第1章	序論	1
第2章	数値解析法	2
	2. 1. 1 基礎方程式	2
	2. 1. 2 コントロールボリューム法による対流項の離散化	3
	2. 1. 3 対流項の風上差分	10
	2. 1. 4 指数法およびハイブリット法	14
	2. 1. 5 スタガード格子	19
	2. 1. 6 運動方程式および密度方程式の離散化	22
	2. 1. 7 SIMPLE 法	35
	2. 1. 8 フローチャート	40
第3章	計算条件	42
第4章	無次元成層度 K の流れへの影響	44
第5章	無次元成層度 K と渦の長さとの関係	49
第6章	結論	51
	謝辞	52
	引用文献	53
	APPENDIX	54

第1章 序論

エスチャリー等の水域水底に堆積した豊富な栄養塩の有効利用を目的として、水底に設置した構造物等によって地形性湧昇を促進する試みがある。しかし、密度成層流場においては、密度界面の存在が底層水塊の上層への輸送（湧昇効果）を抑制するものと考えられる。つまり、安定な成層流が水底構造物を過ぎる場合、負の浮力が作用し、様々な流動現象や振動現象が出現するものと考えられる。したがって、密度成層流に及ぼす水底構造物の影響について検討することは重要である。そこで、本研究は、密度場として連続成層場を対象とし、水底構造物である鉛直壁を過ぎる安定成層流について数値実験を行い検討した。

第2章 コントロールボリューム法

2.1 数値解析法

2.1.1 基礎方程式

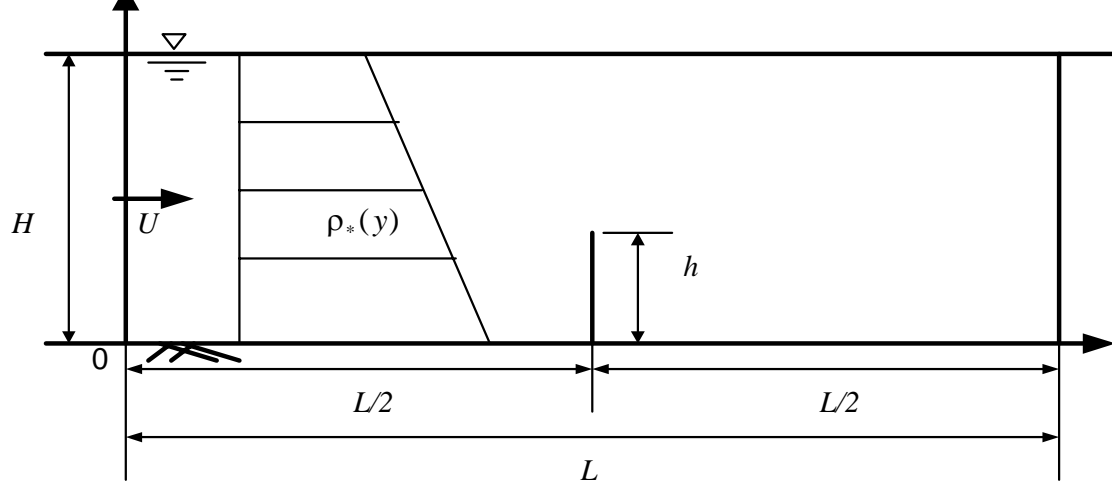


Fig.2.1 流域モデル

Fig-1 に示すように水底に設置した鉛直壁を過ぎる安定成層流について考える．計算領域は Fig-1 に示すように $x-y$ 軸(鉛直二次元)である．鉛直壁の十分上流において，流速は一様流速 U ，密度分布は鉛直方向に線形な勾配 $d\rho_*/dy = -\alpha$ ($\alpha > 0$) を有する基本密度場 $\rho_*(y)$ ，さらにこれに静水圧平衡で対応する基本圧力場 $p_*(y)$ とする．密度と圧力は $\rho_*(y)$ ， $p_*(y)$ からのずれとして $\rho' = \rho - \rho_*$ ， $p' = p - p_*$ と定義する．流れは非圧縮で，ブシネスク近似を仮定する．ここで未知数 x, y 方向の速度分布 u, v ，密度 ρ' ，圧力 p' に対する基礎方程式は以下のようなになる．なお，以下の式で ρ' ， p' のダッシュは省略している．

[連続の式]

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (2.1)$$

[N-S 方程式]

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{1}{\rho_0} \frac{\partial p}{\partial x} + \frac{\partial}{\partial x} (v_H \frac{\partial u}{\partial x}) + \frac{\partial}{\partial y} (v_V \frac{\partial u}{\partial y}) \quad (2.2)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{1}{\rho_0} \frac{\partial p}{\partial y} + \frac{\partial}{\partial x} (v_H \frac{\partial v}{\partial x}) + \frac{\partial}{\partial y} (v_V \frac{\partial v}{\partial y}) - \frac{\rho}{\rho_0} g \quad (2.3)$$

[密度方程式]

$$\frac{\partial \rho}{\partial t} + u \frac{\partial \rho}{\partial x} + v \frac{\partial \rho}{\partial y} = -v \frac{d\rho}{dy} + \frac{\partial}{\partial x} (v d_H \frac{\partial \rho}{\partial x}) + \frac{\partial}{\partial y} (v d_V \frac{\partial \rho}{\partial y}) \quad (2.4)$$

ここに ρ_0 は代表密度， g は重力加速度， v_H および v_V は水平方向および鉛直方向の渦動粘性係数， $v d_H$ および $v d_V$ は水平方向および鉛直方向の拡散係数である。

2.1.2 コントロールボリューム法による対流項と拡散項の離散化

以上の方程式をコントロールボリューム法で離散化する。しかしここで、輸送方程式を離散化する際に、対流項と拡散項の取り扱いに注意を払う必要がある。つまり、対流項が大きくなるにつれて中心差分的な離散化を採用すると、数値計算の不安定が生じることになり、それを防ぐ何らかの対策が必要となる。ここでは、輸送方程式(2.5)の最も簡単な場合として、定常1次元の対流・拡散方程式(2.6)について考察する。

$$\frac{\partial}{\partial t}(\rho\phi) + \frac{\partial}{\partial x}(\rho u\phi) + \frac{\partial}{\partial y}(\rho v\phi) = \frac{\partial}{\partial x}\left(\Gamma \frac{\partial\phi}{\partial x}\right) + \frac{\partial}{\partial y}\left(\Gamma \frac{\partial\phi}{\partial y}\right) + S \quad (2.5)$$

$$\frac{\partial}{\partial x}(\rho u\phi) = \frac{\partial}{\partial x}\left(\Gamma \frac{\partial\phi}{\partial x}\right) \quad (2.6)$$

$$\frac{d}{dx}(\rho u) = 0 \quad \text{すなわち} \quad \rho u = \text{const} \quad (2.7)$$

なお，式(2.7)は1次元の連続の方程式である．Fig.2.2に示すように，計算の対象となる点を P ，その東と西側の点をおのおの E, W ，また，コントロールボリューム法で積分する領域の境界点を東，西側のおのおので e, w とする．式(2.6)を点 P を含む区間 $w e$ で積分することにより，次の離散式を得る．

(左辺)

$$\int_w^e \frac{d}{dx}(\rho u\phi) dx = (\rho u\phi)_e - (\rho u\phi)_w \quad (2.8)$$

(右辺)

$$\int_w^e \frac{d}{dx}\left(\Gamma \frac{d\phi}{dx}\right) dx = \left(\Gamma \frac{d\phi}{dx}\right)_e - \left(\Gamma \frac{d\phi}{dx}\right)_w \quad (2.9)$$

まとめて，

$$(\rho u\phi)_e - (\rho u\phi)_w = \left(\Gamma \frac{d\phi}{dx}\right)_e - \left(\Gamma \frac{d\phi}{dx}\right)_w \quad (2.10)$$

点 e, w は区間 PE, PW のおのおのの中間点であると仮定し，Fig.2.2に示すように物理量 ϕ の値は

$$\phi_e = \frac{1}{2}(\phi_P + \phi_E), \phi_w = \frac{1}{2}(\phi_W + \phi_P) \quad (2.11)$$

と与えられ，式(2.10)の次のように変形される．

$$\frac{1}{2}(\rho u)_e(\phi_P + \phi_E) - \frac{1}{2}(\rho u)_w(\phi_W + \phi_P) = \frac{\Gamma_e(\phi_E - \phi_P)}{(\delta x)_e} - \frac{\Gamma_w(\phi_P - \phi_W)}{(\delta x)_w} \quad (2.12)$$

式の理解を容易にするために，対流項と拡散項の係数を

$$F = \rho u, D = \frac{\Gamma}{\delta x} \quad (2.13)$$

と書きなおすことにより，物理量 ϕ の関係式(2.16)を得ることができる．

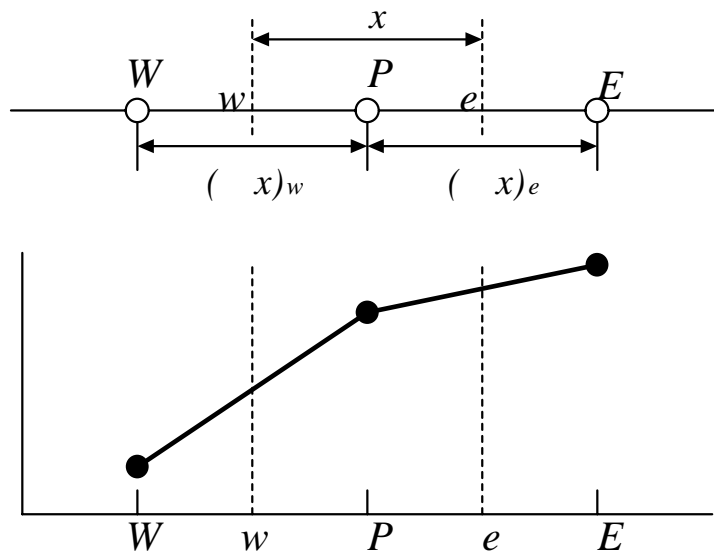


Fig.2.2 コントロールボリューム法における

物理量 ϕ の分布の仮定

$$a_p \phi_p = a_E \phi_E + a_w \phi_w \quad (2.14)$$

$$a_E = D_e - \frac{1}{2} F_e, \quad a_w = D_w - \frac{1}{2} F_w \quad (2.15)$$

$$a_p = D_e + D_w + \frac{1}{2} F_e - \frac{1}{2} F_w = a_E + a_w + (F_e - F_w) \quad (2.16)$$

対流項が大きいときに，計算が不合理な解を与える例をしてみる．

拡散項の係数 $D_e = D_w = 1$ に対し，対流項の係数を $F_e = F_w = 4$ と大きくする． ϕ に対する係数の値は $a_E = -1, a_w = 3, a_p = 2$ となり

$$\phi_E = 20, \phi_w = 10 \text{ のとき } \phi_p = 5$$

$$\phi_E = 10, \phi_w = 20 \text{ のとき } \phi_p = 25$$

となり， $10 < \phi_p < 20$ の範囲を越えて，非現実的な解を与えることになる．繰り返し計算を行う場合には解の発散の原因となる．生成項が存在しないから求めている方程式の解は単調増加か減少でなくてはならない．このような矛盾は未知数の中心の係数 a_p の絶対値が他の係数 a_w よりも小さくなっていること，および各係数の符号が一致していないことに起因している．

ここで，境界点での値の決め方における次数（オーダー）について調べる．式（2.14）を得るときに，西側境界点における物理量とその微分値を

$$\phi_w = \frac{1}{2} (\phi_w + \phi_p), \quad \left(\frac{d\phi}{dx} \right)_w = \frac{\phi_p - \phi_w}{(\delta x)_w} \quad (2.17)$$

と近似している．これは ϕ_w, ϕ_p を点 w の値によるテイラー展開

$$\phi_w = \phi_w - \left(\frac{d\phi}{dx} \right)_w \frac{(\delta x)_w}{2} + \left(\frac{d^2\phi}{dx^2} \right)_w \frac{(\delta x)_w^2}{8} + O((\delta x)_w^3) \quad (2.18)$$

$$\phi_p = \phi_w + \left(\frac{d\phi}{dx} \right)_w \frac{(\delta x)_w}{2} + \left(\frac{d^2\phi}{dx^2} \right)_w \frac{(\delta x)_w^2}{8} + O((\delta x)_w^3) \quad (2.19)$$

で与えて算出したのである。つまり, (2.18) + (2.19) より

$$\phi_w = \frac{1}{2} (\phi_w + \phi_p) + O((\delta x)_w^2) \quad (2.20)$$

また, (2.18) - (2.19) から

$$\left(\frac{d\phi}{dx} \right)_w = \frac{\phi_p - \phi_w}{(\delta x)_w} + O((\delta x)_w^2) \quad (2.21)$$

したがって, いずれの値も 2 次精度で与えられていることになる。さらに, 両隣の値を利用して中心差分と同等である。

1 次元の対流・拡散問題の方程式 (2.6) において, 速度 u , 拡散係数 Γ が一定であると仮定すると, 次式が得られる。

$$\rho u \frac{\partial \phi}{\partial x} = \Gamma \frac{\partial^2 \phi}{\partial x^2} \quad (2.22)$$

これを境界条件

$$x=0 \text{ で } \phi = \phi_0, \quad x=L \text{ で } \phi = \phi_L \quad (2.23)$$

のもとで解析解を求める． $\phi = Ae^{mx}$ とおき， $d\phi/dx = mAe^{mx}$ ， $d^2\phi/dx^2 = m^2Ae^{mx}$ を式 (2.22) に代入して解析解を求めると

$$\frac{\phi - \phi_0}{\phi_L - \phi_0} = \frac{\exp\left(P \frac{x}{L}\right) - 1}{\exp(P) - 1} \quad (2.24)$$

で与えられる．ここに P はペクレ数であり

$$P = \frac{\rho u L}{\Gamma} \quad (2.25)$$

と定義される．つまり， P は対流項の係数と領域の長さに比例し，拡散項の係数に反比例する．数値解との比較を容易にするために， $x=0$ を x_W ， $x=L$ を x_E に対応させる．Fig2.2 に示されている数値解の格子の大きさは， $\Delta x = (\delta x)_W = (\delta x)_E$ の等分割とし，新たにペクレ数

$$P_e = \frac{\rho u L}{\Gamma} \quad (2.26)$$

を定義すると，数値解と比較できる解析解は

$$\frac{\phi - \phi_W}{\phi_E - \phi_W} = \frac{\exp(P_e (x - x_W) / \Delta x) - 1}{\exp(P_e) - 1} \quad (2.27)$$

と与えられる．Fig2.3 は， $\phi_W = 0$ ， $\phi_E = 1$ として解析解をプロットしたものであり，ペクレ数が大きくなるにつれて，物理量 ϕ の分布が東側に強く偏っていく，つまり上流値 ϕ_W によって多く影響を受けることがわかる．ペクレ数が 0 になる限界では，拡散項

のみが働くから ϕ の分布が線形勾配を持つ。また、ペクレ数が負の値になると、図の逆になる。すなわち、物理量が負の x 方向に働くとき、 ϕ_E が上流値となり、その値は物理量 ϕ の値を支配することになる。数値解を解析解と同様の条件で求めてみる。式 (2.15) の係数を算出すると

$$a_E = \frac{\Gamma}{\Delta x} \left(1 - \frac{1}{2} P_e \right), \quad a_W = \frac{\Gamma}{\Delta x} \left(1 + \frac{1}{2} P_e \right), \quad a_P = \frac{2\Gamma}{\Delta x} \quad (2.28)$$

であるから、数値解は式 (2.28) によって次式で与えられる。

$$\phi_P = \frac{1}{2} \left(1 - \frac{1}{2} P_e \right) \phi_E + \frac{1}{2} \left(1 + \frac{1}{2} P_e \right) \phi_W \quad (2.29)$$

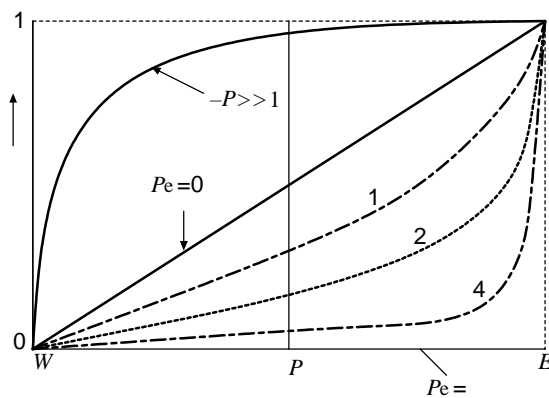


Fig.2.3 解析解

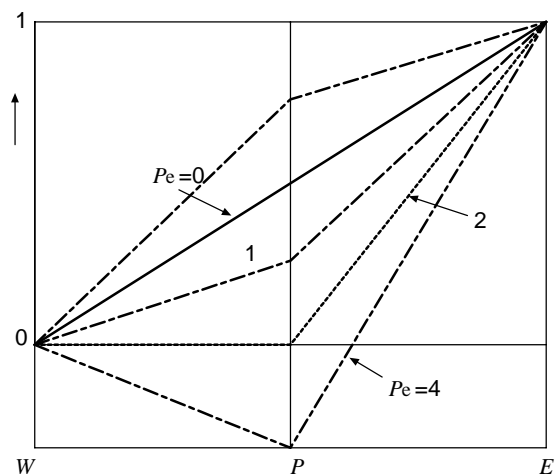


Fig.2.4 数値解(中心差分)

Fig.2.4 は境界条件 $\phi_w = 0$ $\phi_E = 1$ のもとに数値解をプロットしたものであり，ここでは中心の点 P の値 ϕ_p だけを求めることができる．ペクレ数が 2 を越えると ϕ_p が負になり，解析解と大きく異なっていることがわかる．

2.1.3 対流項の風上差分

前述のように，対流・拡散の輸送方程式において，2 次精度の中心差分を使った数値計算は，対流項が大きくなると不合理な解を与える．これを防ぐために，Fig.2.5 に示すように，境界点の値 ϕ_e と ϕ_w は上流側の値をそのまま利用する．つまり

$$u_w > 0 \text{ のとき } \phi_w = \phi_W \quad (2.30)$$

$$u_w < 0 \text{ のとき } \phi_w = \phi_P \quad (2.31)$$

であり，この近似法を対流項に対して適用する．上流側の値を使うことから，これを風上差分 (S.V.Ptankar, 1985) と呼び，式 (2.18), (2.19) から，この近似法は 1 次精度であることがわかる．西側の境界点からの対流による流束を評価してみる． $u_w > 0$ のとき式 (2.18) を利用すると

$$\begin{aligned} F_w \phi_w &= \rho u_w \phi_w \\ &= \rho u_w \phi_W + \rho \left(\frac{d\phi}{dx} \right)_w \frac{u_w \Delta x}{2} + \dots \end{aligned} \quad (2.32)$$

であるから，打ち切り誤差は

$$\Delta F_w = \rho \frac{u_w \Delta x}{2} \left(\frac{d\phi}{dx} \right)_w + \dots \quad (2.33)$$

と表現される．右辺の第 2 項は拡散項と同じであり，第 1 項はその係数とみなすことができる．したがって風上差分を利用することは，係数

$$\Gamma_n = \rho \frac{u_w \Delta x}{2} = \frac{1}{2} P_e \cdot \Gamma \quad (2.34)$$

の数値拡散を導入したこと，つまり人工的な拡散を新たに付加したことと同等である．

風上差分を利用したときの対流・拡散の輸送方程式を作る．

記号

$$\max(A, B) = (A \text{ と } B \text{ の大きい方の値}) \quad (2.35)$$

を導入すると，東側・西側の境界点からの対流流束は，おのこの

$$F_e \phi_e = \phi_p \max(F_e, 0) - \phi_E \max(-F_e, 0) \quad (2.36)$$

$$F_w \phi_w = \phi_W \max(F_w, 0) - \phi_p \max(-F_w, 0) \quad (2.37)$$

と表現される．したがって，風上差分による物理量 ϕ の代数方程式とその係数の値は，

中心差分による式 (2.14) ~ (2.16) と同じ形式で

$$a_p \phi_p = a_E \phi_E + a_W \phi_W \quad (2.38)$$

$$a_E = D_e + \max(-F_e, 0), \quad a_W = D_w + \max(F_w, 0) \quad (2.39)$$

$$\begin{aligned} a_p &= D_e + \max(F_e, 0) + D_w + \max(-F_w, 0) \\ &= a_E + a_W + (F_e - F_w) \end{aligned} \quad (2.40)$$

と書くことができる．係数は常に正に保たれており，さらに連続のしきから $F_e = F_w$ が成立するなら，中心の係数 a_p が他の係数よりも大きくなっていることがわかる．したがって数値計算上，安定な解を得ることができる．前述と同様の条件（速度 u ，拡散係数 Γ が一定）で離散化方程式を作ると，速度 $u_w > 0$ のとき

$$\phi_p = \frac{1}{2 + P_e} \phi_E + \frac{1 + P_e}{2 + P_e} \phi_W \quad (2.41)$$

と与えられる．境界条件を $\phi_w = 0$ ， $\phi_E = 1$ としたときの計算結果を，Fig.2.6 に示す．解析解の Fig.2.3 と比べると，ペクレ数が大きいときは解析解とほぼ一致するが，ペクレ数が小さいときは ϕ_p が解析解よりも大きくなり，数値拡散項が働いていることを示している．Fig.2.4 との中心差分による数値解との比較では， ϕ_p が負となるような不合理さを解消していることがわかる．

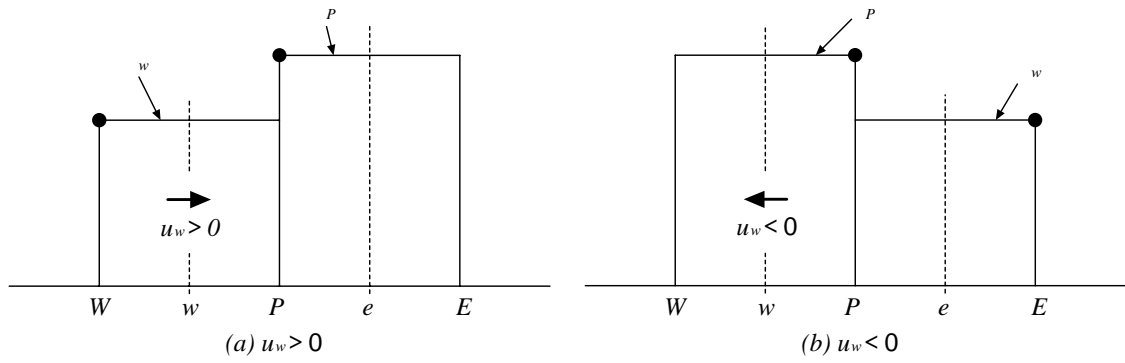


Fig.2.5 風上差分における ϕ の分布の仮定

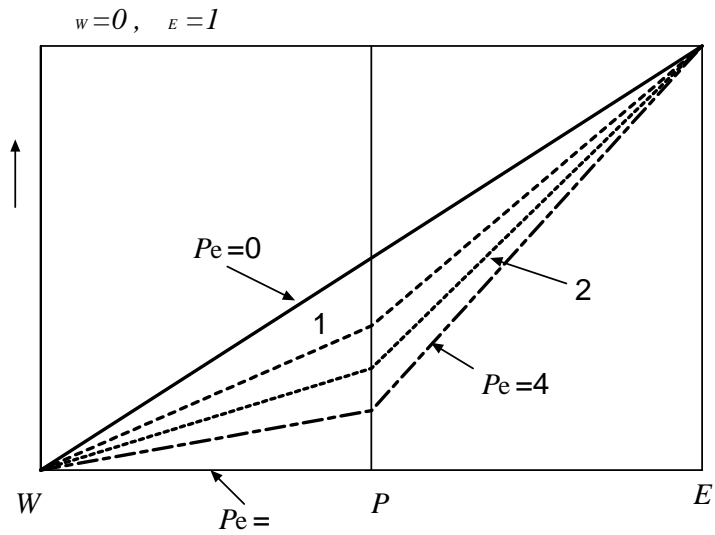


Fig.2.6 風上差分

2.1.4 指数法およびハイブリッド法

ここまで離散化方程式を得るときにテイラー展開を利用して北が、解析解を組み合わせることにより、合理的な離散化方程式を得る方法を述べる。

対流・拡散による 1 次元の輸送方程式において、全流束は

$$J = \rho u \phi - \Gamma \frac{d\phi}{dx} \quad (2.42)$$

と定義される。このとき、輸送方程式とその離散化方程式は

$$\frac{dJ}{dx} = 0 \quad (2.43)$$

$$J_e - J_w = 0 \quad (2.44)$$

と表される。解析解 (2.27) を利用し、 $x=0$ を点 P 、 $x=L=(\delta x)_e$ を点 E に置き換えたものを式 (2.42) に代入する。

$$J_e = F_e \left(\phi_P + \frac{\phi_P - \phi_E}{\exp(P_e) - 1} \right) \quad (2.45)$$

$$\text{ここに} \quad P_e = \frac{(\rho u)_e (\delta x)_e}{\Gamma_e} = \frac{F_e}{D_e} \quad (2.46)$$

同様に、 $x=0$ を点 W 、 $x=L=(\delta x)_w$ を点 P と置換すると

$$J_w = F_w \left(\phi_w + \frac{\phi_w - \phi_p}{\exp(P_w) - 1} \right) \quad (2.47)$$

$$\text{ここに } P_w = \frac{(\rho u)_w (\delta x)_w}{\Gamma_w} = \frac{F_w}{D_w} \quad (2.48)$$

が得られる．式(2.45),(2.47)を離散式(2.44)に代入すると, ϕ に対する離散化方程式は

$$a_p \phi_p = a_E \phi_E + a_w \phi_w \quad (2.49)$$

$$a_E = \frac{F_e}{\exp(F_e/D_e) - 1}, \quad a_w = \frac{F_w \exp(F_w/D_w)}{\exp(F_w/D_w) - 1} \quad (2.50)$$

で与えられる．この指数法(S.V.Ptankar,1985)で得られる点 P における数値解は,当然のことながら解析解と一致する．係数 a_E のペクレ数に対する変化を調べるため,拡散項の係数 D_e に対する a_E の比

$$\frac{a_E}{D_e} = \frac{P_e}{\exp(P_e) - 1} \quad (2.51)$$

を Fig.2.7 に実践でプロットする． P_e の値が正なら格子点 E は下流側隣接点であり,その影響は P_e の増加とともに減少する． P_e が負なら点 E は上流側隣接点であり,大きな影響をもつ．この指数法は解析解に一致する便利さがあるものの,その厳密性が保たれるのは1次元問題に限られることや,指数関数が計算に時間を要するなどの欠点も持ち合わせている．そこで,指数法のもつ定量的な正確さをもった計算しやすい方法,ハイブリッド法(S.V.Ptankar,1985)を説明する． a_E/D_e の正確な変化の特徴は次のよ

うになる .

1. $P_e \rightarrow \infty$ のとき

$$\frac{a_E}{D_e} \rightarrow 0$$

2. $P_e \rightarrow -\infty$ のとき

$$\frac{a_E}{D_e} \rightarrow -P_e$$

3. $P_e = 0$ での接線は

$$\frac{a_E}{D_e} = 1 - \frac{P_e}{2}$$

これらの極限の場合を表した 3 本の直線を , Fig.2.7 に示す . これらの直線は解析解の包括線をなしていて , 妥当な近似を表していると考えられる . ハイブリッド法はこれら 3 本の直線から成り立っている .

$P_e < -2$ のとき

$$\frac{a_E}{D_e} = -P_e \tag{2.52}$$

$-1 < P_e \leq 2$ のとき

$$\frac{a_E}{D_e} = 1 - \frac{P_e}{2} \tag{2.53}$$

$P_e > 2$ のとき

$$\frac{a_E}{D_e} = 0 \tag{2.54}$$

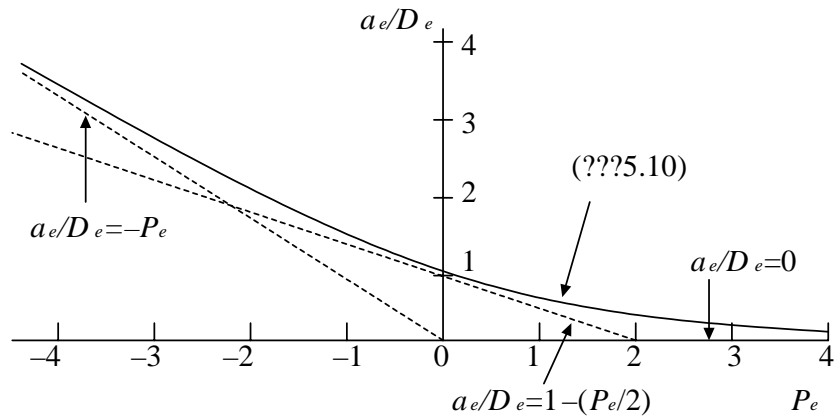


Fig.2.7 指数法・ハイブリッド法におけるペクレ数 P_e に対する影響係数 a_E の変化

指数法で得られた係数 a_E を，式 (2.52) ~ (2.54) を考慮して式で表すと

$$a_E = D_e \max\left(-P_e, 1 - \frac{P_e}{2}\right) = \max\left(-F_e, D_e - \frac{F_e}{2}\right) \quad (2.55)$$

となる．

同様に，点 W の係数は

$$a_W = \max\left(F_w, D_w + \frac{F_w}{2}\right) \quad (2.56)$$

で与えられ，さらに式 (2.49) を参照して中心点 P の係数が決められる．

$$a_p = a_E + a_W + (F_e - F_w) \quad (2.57)$$

つまり、係数としてペクレ数が $-2 < P_e < 2$ と、対流が拡散に比べて小さいときは中心差分による値をとり、ペクレ数が $P_e < -2$ 、 $2 < P_e$ のように対流成分が大きくなったときは、拡散を 0 とした 1 次精度の風上差分の値をとることを意味している。このハイブリッド法はペクレ数が小さいとき対流項を中心差分で離散化していることや、指数関数などを含んでいないため計算時間を短縮できる。本実験では、ハイブリッド法を採用する。

2.1.5 スタガード格子

以下において数値計算の対象とする 2 次元定常の運動方程式と連続の式および密度方程式は 2.1.1 で示したようになる。

方程式の数値計算には，スタガード格子 (S.V.Ptankar, 1985) と呼ばれる，未知数によってくい違った異なる格子を用いる。これは，1 次元ではジグザグの振動解を，2 次元ではチェスボード状の千鳥格子の振動解を防ぐために工夫されたものである。x 方向の運動方程式 (2.2) の圧力勾配項を例に取り，振動解の発生原因とその防止法を検討する。圧力勾配 $-\partial p / \partial x$ をコントロールボリューム法で離散化すると

$$-\iint \frac{\partial p}{\partial x} dx dy = \Delta y [p_w - p_e] = \Delta y \left[\frac{p_w + p_p}{2} - \frac{p_p + p_e}{2} \right] = \Delta y \left[\frac{p_w - p_e}{2} \right] \quad (2.58)$$

となる。Fig.2.8 に示すように，大文字の点 W, P, E で圧力の離散値を取ることにすると，ジグザグの圧力振動解を許してしまう。これを防ぐためには，圧力 p は他の未知数 (x 方向の運動方程式では速度 u) とは異なった点で値を与えればよい。連続の式および密度方程式についても，同様のことが発生する。後述するように，連続の式は，圧力 p を得るために積分され，離散化される。1 次元に関しては次式のように積分し，離散化方程式を得る。

$$\int \frac{du}{dx} dx = u_e - u_w = \frac{u_p + u_e}{2} - \frac{u_w + u_p}{2} = u_e - u_w = 0 \quad (2.59)$$

したがって、速度 u と圧力 p が同一の点で定義されると、振動解が発生する恐れがある。圧力を大文字の点 W, P, E などで与えるなら、速度は境界点 w, e で値を与える、とすることにより振動解を防ぐことができる。これらの 1 次元におけるジグザグ状の振動解は、2 次元ではチェスボード状の振動解となって現れる。1 次元における考察から、それを防ぐためには、Fig.2.9 に示すように、大文字の点で圧力 p を定義するなら、速度 u および v は p に対するコントロールボリュームの境界点で与えられればよい。したがって、3 個の未知数 u, v, p に対する格子は、互いにくい違った位置に配置する必要があり、スタガード(くい違い)格子と呼ばれる所以である。なお一般的に圧力 p と密度 ρ については同じ点で与えられるので、本研究でも圧力 p と密度 ρ は同じ点で与えている。

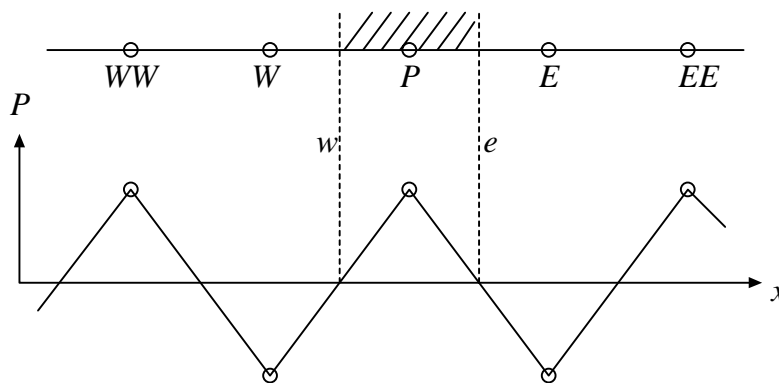


Fig.2.8 振動解

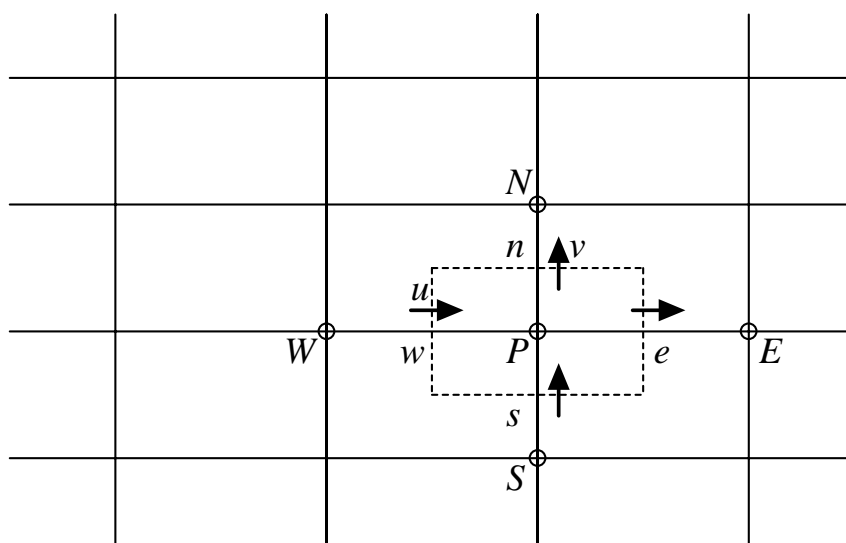
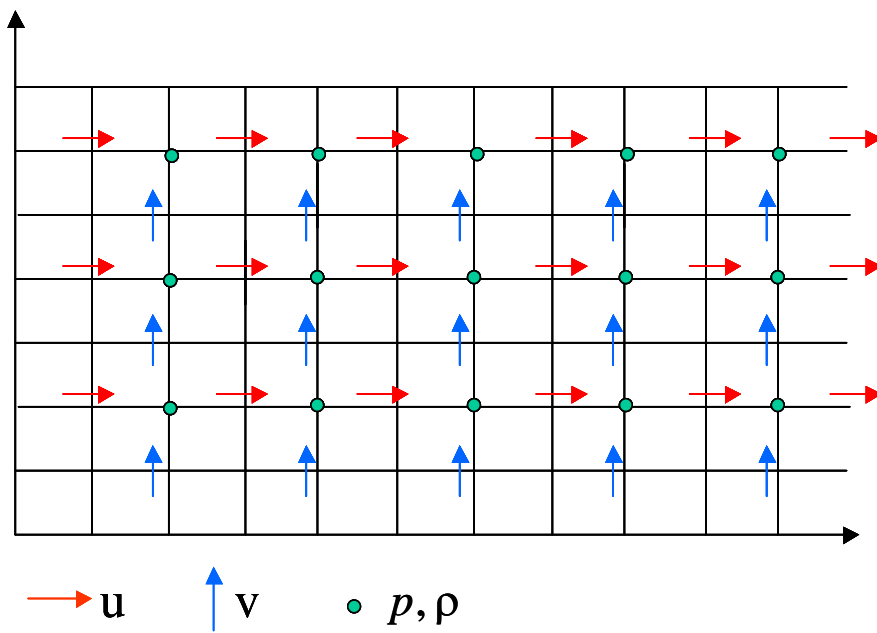


Fig.2.9 スタガード格子

2.1.6 運動方程式および密度方程式の離散化

まず式(2.60)の運動方程式を、速度 u を未知数としてコントロールボリューム法で離散化する。

$$\frac{\partial uu}{\partial x} + \frac{\partial vv}{\partial y} = -\frac{1}{\rho_0} \frac{\partial p}{\partial x} + \frac{\partial}{\partial x} (v_H \frac{\partial u}{\partial x}) + \frac{\partial}{\partial y} (v_V \frac{\partial u}{\partial y}) \quad (2.60)$$

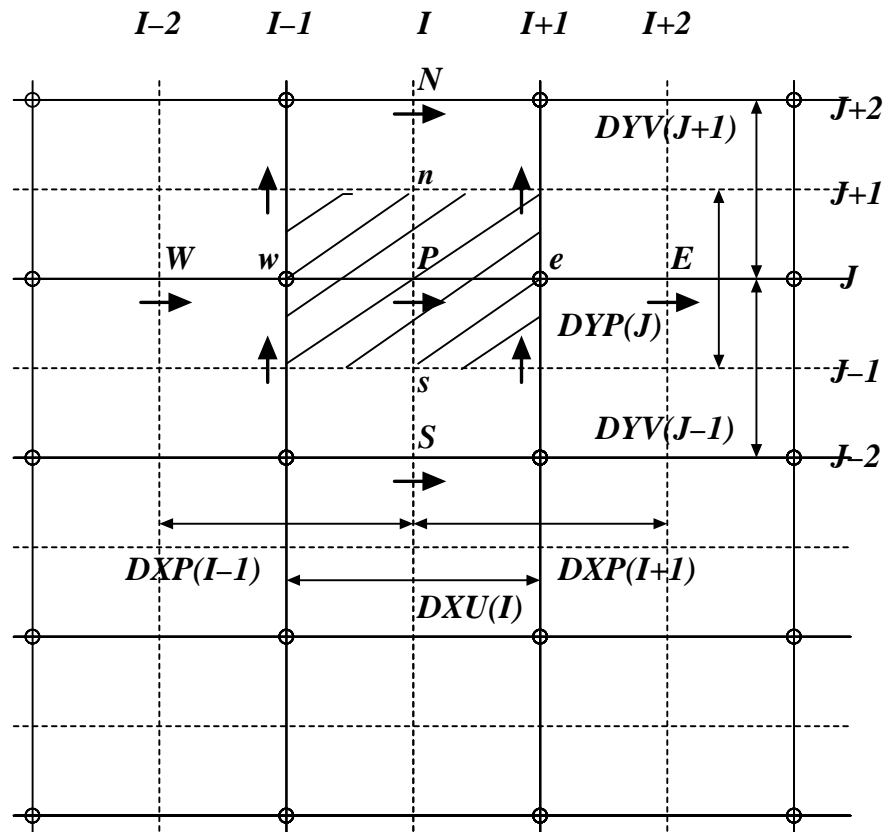


Fig.2.10 x方向運動方程式の離散化の説明

Fig.2.10 は，離散化したときの位置や長さの記号を表したものであり，斜線は積分する領域，すなわちコントロールボリュームを示す．添え字 (I, J) に対する中心点 P の位置で，速度 $U(I, J)$ を定義する．スタガード格子をとるために，圧力 p はコントロールボリュームの境界点 w, e で与えられ，速度 v はそのコーナーで与えられる．添え字に関してはすべて同じものに対応している．なお，大文字で表した変数は，プログラムで利用されている変数名にほぼ一致する．

対流項の第 1 項の離散化を，次のように進める．

$$\begin{aligned}
 \iint \frac{\partial}{\partial x}(uu) dx dy &= \Delta y \int \frac{\partial}{\partial x}(uu) dx \\
 &= DYP(J) * [(uu)_e - (uu)_w] \\
 &= DYP(J) * \{ 0.5[U(I+2, J) + U(I, J)] * u_e \\
 &\quad - 0.5 * [U(I, J) + U(I-2, J)] * u_w \} \\
 &= CE * u_e - CW * u_w \\
 &= CE * 0.5 * [U(I+2, J) + U(I, J)] \\
 &\quad - CW * 0.5 * [U(I, J) + U(I-2, J)] \tag{2.61}
 \end{aligned}$$

つまり， y 方向の積分については平均値に y 方向の区間幅 $DYP(J)$ を乗じることによって表現し， x 方向の積分については原始関数 uu が与えられているため，その積分区間の差として表現されている．境界面の値は，2 次精度の中心差分と等価である，2 点の平均値で与えている．未知数である速度 u の線形代数方程式を得るために， uu の一方の u は既知の値を利用して係数に含ませておき，最終的な式 (2.61) を得る．対流項

の第 2 項も，同様の道筋でコントロールボリューム法による離散化方程式 (2.62) を得る．速度 v はこの方程式では未知数扱いしていないため，係数に組み込まれていく．

$$\begin{aligned}
 \iint \frac{\partial}{\partial x}(vu) dx dy &= \Delta x \int \frac{\partial}{\partial x}(vu) dy \\
 &= DXU(I) * [(vu)_n - (vu)_s] \\
 &= DXU(I) * \{0.5[V(I+1, J+1) + V(I-1, J+1)] * u_n \\
 &\quad - 0.5 * [V(I+1, J-1) + V(I-1, J-1)] * u_s\} \\
 &= CN * u_n - CS * u_s \\
 &= CN * 0.5 * [U(I, J+2) + U(I, J)] \\
 &\quad - CS * 0.5 * [U(I, J) + U(I, J-2)] \quad (2.62)
 \end{aligned}$$

粘性項第 1 項において， y 方向の積分は平均値に区間幅を乗じて表現し， x 方向の積分については原始関数 $v_H \partial u / \partial x$ の差で表す．この微分項の東側境界 e における値は，速度 u の点 E, P の差を両点間の距離 $DXP(I+1)$ で除して与え，2 次精度である．式 (2.63) は粘性項第 1 項の離散化された式を，式 (2.64) は同様にして得られる粘性項第 2 項を示す．

$$\begin{aligned}
 \iint \frac{\partial}{\partial x} \left(v_H \frac{\partial u}{\partial x} \right) dx dy &= \Delta y \int \frac{\partial}{\partial x} \left(v_H \frac{\partial u}{\partial x} \right) dx \\
 &= DYP(J) * \left[\left(v_H \frac{\partial u}{\partial x} \right)_e - \left(v_H \frac{\partial u}{\partial x} \right)_w \right] \\
 &= DYP(J) * \left[VISH * \frac{1}{DXP(I+1)} (\partial u)_e - VISH * \frac{1}{DXP(I-1)} (\partial u)_w \right]
 \end{aligned}$$

$$\begin{aligned}
&= DE * (\partial u)_e - DW * (\partial u)_w \\
&= DE * [U(I+2, J) - U(I, J)] - DW * [U(I, J) + U(I-2, J)] \quad (2.63)
\end{aligned}$$

$$\begin{aligned}
\iint \frac{\partial}{\partial y} \left(v_v \frac{\partial u}{\partial y} \right) dx dy &= \Delta x \int \frac{\partial}{\partial y} \left(v_v \frac{\partial u}{\partial y} \right) dy \\
&= DXU(I) * \left[\left(v_v \frac{\partial u}{\partial y} \right)_n - \left(v_v \frac{\partial u}{\partial y} \right)_s \right] \\
&= DXU(I) * \left[VISV * \frac{1}{DYV(J+1)} (\partial u)_n - VISV * \frac{1}{DYV(J-1)} (\partial u)_s \right] \\
&= DN * (\partial u)_n - DS * (\partial u)_s \\
&= DN * [U(I, J+2) - U(I, J)] - DS * [U(I, J) + U(I, J-2)] \quad (2.64)
\end{aligned}$$

圧力勾配は、y方向については平均値に区間幅を乗じ、x方向は境界点w,eに圧力の格子点を与えられているため、それらの差として直接与えられる。

$$\begin{aligned}
\iint \left(-\frac{\partial p}{\partial x} \right) dx dy &= -\Delta y \int \frac{\partial p}{\partial x} dx = -\Delta y [p_e - p_w] \\
&= DYP(J) * [P(I-1, J) - P(I+1, J)] \quad (2.65)
\end{aligned}$$

以上の各項をまとめると、x方向運動方程式の離散化式は次式のようになる。

$$\begin{aligned}
AP(I, J) * U(I, J) &= AN(I, J) * U(I, J+2) + AS(I, J) * U(I, J-2) \\
&\quad + AE(I, J) * U(I+2, J) + AW(I, J) * U(I-2, J) + SU(I, J) \quad (2.66)
\end{aligned}$$

$$AN(I, J) = -0.5 * CN + DN \rightarrow DN + \max(-CN, 0)$$

(風上差分)

$$\rightarrow \max(-CN, DN - 0.5CN, 0) \quad (2.67)$$

(ハイブリッド法)

$$AS(I, J) = 0.5 * CS + DS \rightarrow DS + \max(CS, 0)$$

$$\rightarrow \max(CS, DS + 0.5CS, 0) \quad (2.67)$$

$$AE(I, J) = -0.5 * CE + DE \rightarrow DE + \max(-CE, 0)$$

$$\rightarrow \max(-CE, DE - 0.5CE, 0) \quad (2.67)$$

$$AW(I, J) = 0.5 * CW + DW \rightarrow DW + \max(CW, 0)$$

$$\rightarrow \max(CW, DW + 0.5CW, 0) \quad (2.67)$$

$$AP(I, J) = AN(I, J) + AS(I, J) + AE(I, J) + AW(I, J)$$

$$+ (CN - CS + CE - CW) \quad (2.68)$$

$$SU(I, J) = DYP(J) * [P(I-1, J) + P(I+1, J)] \quad (2.69)$$

y方向運動方程式の離散化式は、式(2.70)のy方向の運動方程式をもとに、速度vを求める離散化式をX方向運動方程式と同様に作る。また、密度方程式の離散化については式(2.71)の密度方程式をもとに、密度ρを求める離散化式をX方向運動方程式と同様に作る。

$$\frac{\partial uv}{\partial x} + \frac{\partial vw}{\partial y} = -\frac{1}{\rho_0} \frac{\partial p}{\partial y} + \frac{\partial}{\partial x} (v_H \frac{\partial v}{\partial x}) + \frac{\partial}{\partial y} (v_V \frac{\partial v}{\partial y}) - \frac{\rho}{\rho_0} g \quad (2.70)$$

$$\frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} = -v\alpha + \frac{\partial}{\partial x} (v d_H \frac{\partial \rho}{\partial x}) + \frac{\partial}{\partial y} (v d_V \frac{\partial \rho}{\partial y}) \quad (2.71)$$

Fig.2.11 と Fig.2.12 のように，斜線で示したコントロールボリュームを設定し，その中心点 P に速度 v および密度 ρ に対応する未知数 $V(I, J)$ および $\rho(I, J)$ を定める．スタガード格子を採用するため，前の記号と対応していない部分もあるが，離散化式を得る手順は x 方向の運動方程式と同様である．以下に示す．

・ y 方向運動方程式の離散化

対流項の離散化

$$\begin{aligned} \iint \frac{\partial}{\partial x} (uv) dx dy &= \Delta y \int \frac{\partial}{\partial x} (uv) dx \\ &= DYV(J) [(uv)_e - (uv)_w] \\ &= DYV(J) \{ 0.5 [U(I+1, J+1) + U(I+1, J-1)] v_e \\ &\quad - 0.5 [U(I-1, J+1) + U(I-1, J-1)] v_w \} \\ &= CEv_e - CWv_w \\ &= 0.5CE [V(I+2, J) + V(I, J)] - 0.5CW [V(I, J) + V(I-2, J)] \end{aligned} \quad (2.72)$$

$$\begin{aligned} \iint \frac{\partial}{\partial y} (vw) dx dy &= \Delta x \int \frac{\partial}{\partial y} (vw) dy \\ &= DXP(I) [(vw)_n - (vw)_s] \end{aligned}$$

$$\begin{aligned}
&= DXP(I)\{0.5[V(I, J+2)+V(I, J)]v_n \\
&\quad -0.5[V(I, J)+V(I, J-2)]v_s\} \\
&= CNv_n - CSv_s \\
&= 0.5CN[V(I, J+2)+V(I, J)] - 0.5CS[V(I, J)+V(I, J-2)] \quad (2.73)
\end{aligned}$$

粘性項の離散化

$$\begin{aligned}
\iint \frac{\partial}{\partial x} \left(v_H \frac{\partial v}{\partial x} \right) dx dy &= \Delta y \int \frac{\partial}{\partial x} \left(v_H \frac{\partial v}{\partial x} \right) dx \\
&= DYV(J) \left[\left(v_H \frac{\partial v}{\partial x} \right)_e - \left(v_H \frac{\partial v}{\partial x} \right)_w \right] \\
&= DYV(J) \left[VISH \frac{1}{DXU(I+1)} (\partial v)_e - VISH \frac{1}{DXU(I-1)} (\partial v)_w \right] \\
&= DE(\partial v)_e - DW(\partial v)_w \\
&= DE[V(I+2, J) - V(I, J)] - DW[V(I, J) - V(I-2, J)] \quad (2.74)
\end{aligned}$$

$$\begin{aligned}
\iint \frac{\partial}{\partial y} \left(v_V \frac{\partial v}{\partial y} \right) dx dy &= \Delta x \int \frac{\partial}{\partial y} \left(v_V \frac{\partial v}{\partial y} \right) dy \\
&= DXP(I) \left[\left(v_V \frac{\partial v}{\partial y} \right)_n - \left(v_V \frac{\partial v}{\partial y} \right)_s \right] \\
&= DXP(I) \left[VISV \frac{1}{DYP(J+1)} (\partial v)_n - VISV \frac{1}{DYP(J-1)} (\partial v)_s \right] \\
&= DN(\partial v)_n - DS(\partial v)_s \\
&= DN[V(I, J+2) - V(I, J)] - DS[V(I, J) - V(I, J-2)] \quad (2.75)
\end{aligned}$$

圧力項の離散化

$$\begin{aligned} \iint \frac{1}{\rho_0} \left(-\frac{\partial p}{\partial y} \right) dx dy &= -\frac{\Delta x}{\rho_0} \int \frac{\partial p}{\partial y} dy \\ &= -\frac{\Delta x}{\rho_0} [p_n - p_s] \\ &= DXP(I) \frac{1}{\rho_0} [P(I, J-1) - P(I, J+1)] \end{aligned} \quad (2.76)$$

質量力項の離散化

$$\int -\frac{\rho}{\rho_0} g dx dy = -0.5 \frac{g}{\rho_0} \Delta x \Delta y [\rho(I, J+1) + \rho(I, J-1)] \quad (2.77)$$

以上の各項をまとめると, y 方向運動方程式の離散化式は次式のようになる.

$$\begin{aligned} AP(I, J) * V(I, J) &= AN(I, J) * V(I, J+2) + AS(I, J) * V(I, J-2) \\ &\quad + AE(I, J) * V(I+2, J) + AW(I, J) * V(I-2, J) + SU(I, J) \end{aligned} \quad (2.78)$$

$$AN(I, J) = -0.5 * CN + DN \rightarrow DN + \max(-CN, 0)$$

(風上差分)

$$\rightarrow \max(-CN, DN - 0.5CN, 0) \quad (2.79)$$

(ハイブリッド法)

$$AS(I, J) = 0.5 * CS + DS \rightarrow DS + \max(CS, 0)$$

$$\rightarrow \max(CS, DS + 0.5CS, 0) \quad (2.80)$$

$$AE(I, J) = -0.5 * CE + DE \rightarrow DE + \max(-CE, 0)$$

$$\rightarrow \max(-CE, DE - 0.5CE, 0) \quad (2.81)$$

$$\begin{aligned}
 AW(I, J) &= 0.5 * CW + DW \rightarrow DW + \max(CW - \rho) \\
 &\rightarrow \max(CW, DW + 0.5CW, 0)
 \end{aligned}
 \tag{2.82}$$

$$\begin{aligned}
 AP(I, J) &= AN(I, J) + AS(I, J) + AE(I, J) + AW(I, J) \\
 &\quad + (CN - CS + CE - CW)
 \end{aligned}
 \tag{2.83}$$

$$\begin{aligned}
 SU(I, J) &= DXP(I) \frac{1}{\rho_0} [P(I, J-1) + P(I, J+1)] \\
 &\quad - \frac{g}{2\rho_0} DXP(I) DYV(J) [\rho(I, J+1) + \rho(I, J-1)]
 \end{aligned}
 \tag{2.84}$$

・ 密度方程式の離散化

対流項の離散化

$$\begin{aligned}
 \iint \frac{\partial}{\partial x}(\rho u) dx dy &= \Delta y \int \frac{\partial}{\partial x}(\rho u) dx \\
 &= DYP(J) [(\rho u)_e - (\rho u)_w] \\
 &= DYP(J) \{U(I+1, J) \rho_e - U(I-1, J) \rho_w\} \\
 &= CE \rho_e - CW \rho_w \\
 &= 0.5CE \{ \rho(I+2, J) + \rho(I, J) \} - 0.5CW \{ \rho(I, J) + \rho(I-2, J) \} \quad (2.85)
 \end{aligned}$$

$$\begin{aligned}
 \iint \frac{\partial}{\partial y}(\rho v) dx dy &= \Delta x \int \frac{\partial}{\partial y}(\rho v) dy \\
 &= DYP(J) [(\rho v)_n - (\rho v)_s] \\
 &= DXP(J) \{V(I, J+1) \rho_n - V(I, J-1) \rho_s\} \\
 &= CN \rho_n - CS \rho_s \\
 &= 0.5CN \{ \rho(I, J+2) + \rho(I, J) \} - 0.5CS \{ \rho(I, J) + \rho(I, J-2) \} \quad (2.86)
 \end{aligned}$$

粘性項の離散化

$$\begin{aligned}
 \iint \frac{\partial}{\partial x} \left(v d_H \frac{\partial \rho}{\partial x} \right) dx dy &= \Delta y \int \frac{\partial}{\partial x} \left(v d_H \frac{\partial \rho}{\partial x} \right) dx \\
 &= DYP(J) \left[\left(v d_H \frac{\partial \rho}{\partial x} \right)_e - \left(v d_H \frac{\partial \rho}{\partial x} \right)_w \right] \\
 &= DYP(J) \left[VISUD \frac{1}{DXU(I+1)} (\partial \rho)_e - VISUD \frac{1}{DXU(I-1)} (\partial \rho)_w \right] \\
 &= DE (\partial \rho)_e - DW (\partial \rho)_w
 \end{aligned}$$

$$= DE[\rho(I+2, J) - \rho(I, J)] - DW[\rho(I, J) - \rho(I-2, J)] \quad (2.87)$$

$$\iint \frac{\partial}{\partial y} \left(v d_v \frac{\partial \rho}{\partial y} \right) dx dy = \Delta x \int \frac{\partial}{\partial y} \left(v d_v \frac{\partial \rho}{\partial y} \right) dy$$

$$= DXP(I) \left[\left(v d_v \frac{\partial \rho}{\partial y} \right)_n - \left(v d_v \frac{\partial \rho}{\partial y} \right)_s \right]$$

$$= DXP(I) \left[VISVD \frac{1}{DYV(J+1)} (\partial \rho)_n - VISVD \frac{1}{DYV(J-1)} (\partial \rho)_s \right]$$

$$= DN(\partial \rho)_n - DS(\partial \rho)_s$$

$$= DN[\rho(I, J+2) - \rho(I, J)] - DS[\rho(I, J) - \rho(I, J-2)] \quad (2.88)$$

密度勾配項の離散化

$$\iint -v \alpha dx dy = -0.5 DXP(I) DYP(J) \alpha [V(I, J+1) + V(I, J-1)] \quad (2.89)$$

以上の各項をまとめると、密度方程式の離散化式は次式のようになる。

$$\begin{aligned} AP(I, J) * \rho(I, J) = & AN(I, J) * \rho(I, J+2) + AS(I, J) * \rho(I, J-2) \\ & + AE(I, J) * \rho(I+2, J) + AW(I, J) * \rho(I-2, J) + SU(I, J) \end{aligned} \quad (2.90)$$

$$AN(I, J) = -0.5 * CN + DN \rightarrow DN + \max(-CN, 0)$$

(風上差分)

$$\rightarrow \max(-CN, DN - 0.5CN, 0) \quad (2.91)$$

(ハイブリッド法)

$$AS(I, J) = 0.5 * CS + DS \rightarrow DS + \max(CS, 0)$$

$$\rightarrow \max(CS, DS + 0.5CS, 0) \quad (2.92)$$

$$\begin{aligned}
 AE(I, J) &= -0.5 * CE + DE \rightarrow DE + \max(-CE, 0) \\
 &\rightarrow \max(-CE, DE - 0.5CE, 0) \quad (2.93)
 \end{aligned}$$

$$\begin{aligned}
 AW(I, J) &= 0.5 * CW + DW \rightarrow DW + \max(CW, 0) \\
 &\rightarrow \max(CW, DW + 0.5CW, 0) \quad (2.94)
 \end{aligned}$$

$$\begin{aligned}
 AP(I, J) &= AN(I, J) + AS(I, J) + AE(I, J) + AW(I, J) \\
 &\quad + (CN - CS + CE - CW) \quad (2.95)
 \end{aligned}$$

$$SU(I, J) = -0.5\alpha DXP(I) DYP(J) [V(I, J+1) + V(I, J-1)] \quad (2.96)$$

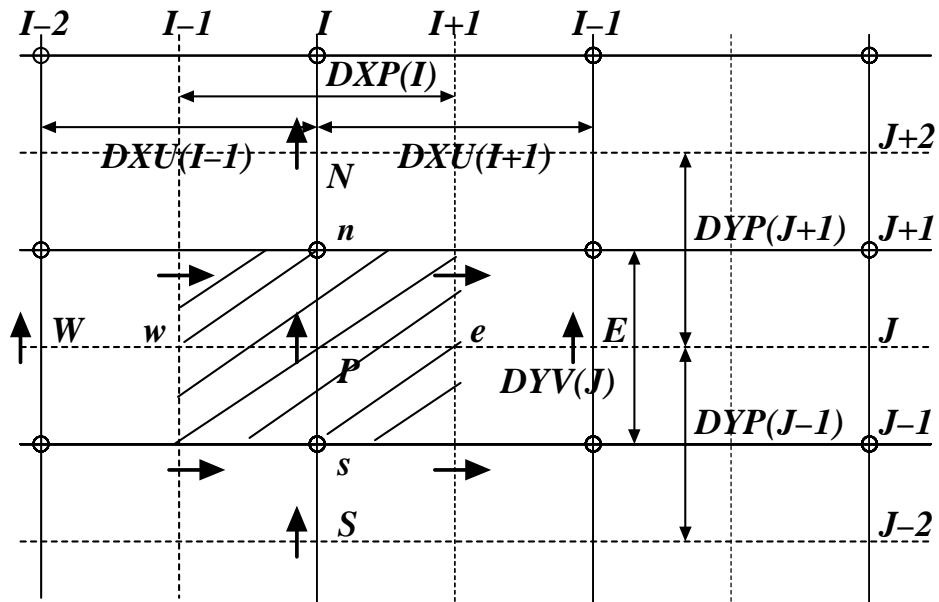


Fig.2.11 y方向運動方程式の離散化の説明

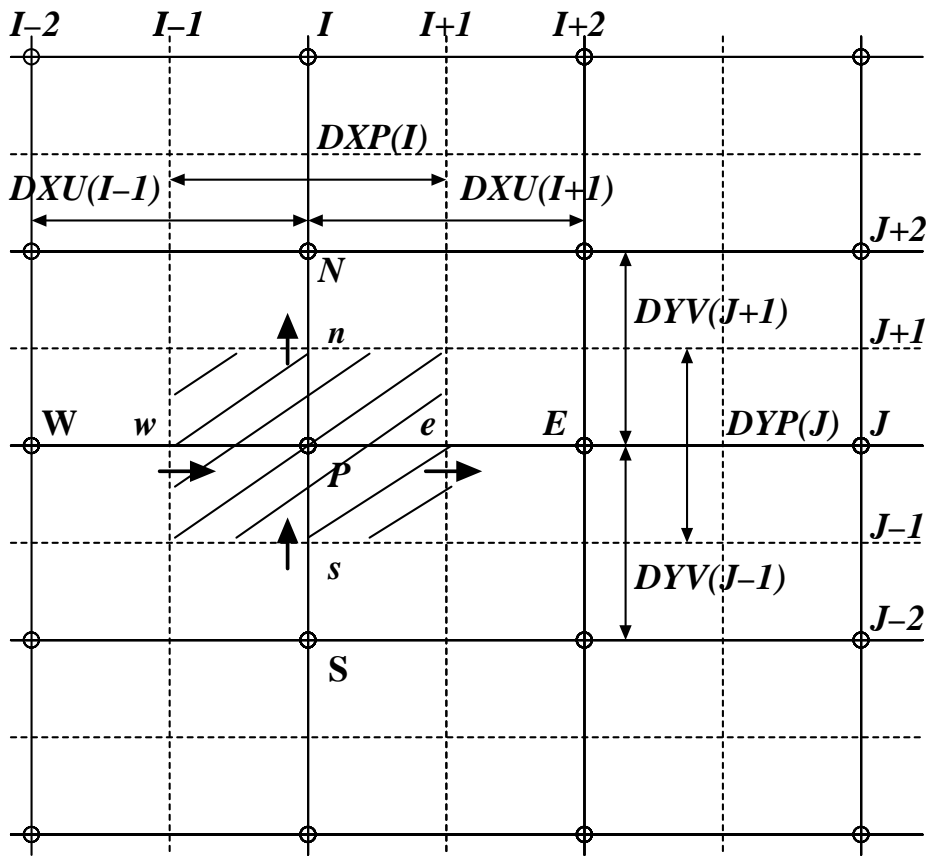


Fig.2.12 密度方程式の離散化の説明

2.1.7 SIMPLE 法

現在，非静水圧モデルによる圧力の計算には，圧力方程式 (S.V.Ptankar,1985) による解法と，SIMPLE 解法 (尹ら，1995) の二つの解法がある．前者はポアソン方程式を直接解くことになり，境界条件の取り扱いが厳しいことと，計算の不安定が生じ易いことが欠点として指摘されている．これに対して，後者は境界条件の取り扱いや計算の安定性が優れていると言われている．そこで，本研究では SIMPLE 解法を採用する．

連続の式は，前述のように 2 次元定常流れのとき

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (2.102)$$

である．既に運動方程式から速度を求めているから，連続の式からは圧力を得たい．しかし，連続の式には圧力が含まれていないために，運動方程式と組み合わせる必要がある．速度 u を求める x 方向の運動方程式の離散化式 (2.66) を

$$a_e u_e = \sum a_{nb} u_{nb} + b + A_e (p_p - p_E) \quad (2.103)$$

と書き直す．添え字は，連続の式に対するコントロールボリュームを説明する Fig.2.13 に対応している．速度 v を求める y 方向の運動方程式についても，次式のように表す．

$$a_n v_n = \sum a_{nb} v_{nb} + b + A_n (p_p - p_N) \quad (2.104)$$

圧力に対して適切な推定値 p^* を与えると，速度に対しての近似値 u^*, v^* を，上の 2 つ式から以下のように求めることができる．

$$a_e u_e^* = \sum a_{nb} u_{nb}^* + b_e + A_e (p_P^* - p_E^*) \quad (2.105)$$

$$a_n v_n^* = \sum a_{nb} v_{nb}^* + b_n + A_n (p_P^* - p_N^*) \quad (2.106)$$

正しい圧力および速度を p, u, v とし，推定値 p^* に対する補正量を p' ，それに対する速度補正を u', v' とする．すなわち

$$p = p^* + p' \quad (2.107)$$

$$u = u^* + u' \quad (2.108)$$

$$v = v^* + v' \quad (2.109)$$

の関係式を定義する．この 3 式を式(2.103)(2.104)に代入し，おのこの式(2.105)，(2.106)を差し引くと

$$a_e u_e' = \sum a_{nb} u_{nb}' + A_e (p_P' - p_E') \quad (2.110)$$

$$a_n v_n' = \sum a_{nb} v_{nb}' + A_n (p_P' - p_N') \quad (2.111)$$

となる．ここで，隣の格子点の速度補正量の影響が小さいと考えて，右辺第 1 項を省略する．仮に収束解を得ることができれば，補正量はすべて 0 になるため，この省略は収束解を得ることを前提として認めることのできる合理的な仮定である．したがって速度補正量は

$$u'_e = d'_e(p'_P - p'_E), \quad d'_e \equiv A_e/a_e \quad (2.82)$$

$$u'_n = d'_n(p'_P - p'_N), \quad d'_n \equiv A_n/a_n \quad (2.83)$$

で与えられるから，連続のしきに必要な速度は次式で与えられる．

$$u_e = u_e^* + d'_e(p'_P - p'_E) \quad (2.84)$$

$$v_n = v_n^* + d'_n(p'_P - p'_N) \quad (2.85)$$

連続の式(2.72)をコントロールボリュームで積分すると

$$(u_e - u_w)\Delta y + (v_n - v_s)\Delta x = 0 \quad (2.86)$$

となり，この式に式(2.84),(2.85)および u_w, v_s に関する同様な式を代入し，圧力補正量 p' に関する次式のような圧力補正式を得る．

$$a_p p'_p = a_E p'_E + a_W p'_W + a_N p'_N + a_S p'_S + b \quad (2.87)$$

$$a_E = \rho d'_e \Delta y, \quad a_W = \rho d'_w \Delta y, \quad a_N = \rho d'_n \Delta x, \quad a_S = \rho d'_s \Delta x,$$

$$a_p = a_E + a_W + a_N + a_S,$$

$$b = (u_w^* - u_e^*)\Delta y + (v_n^* - v_s^*)\Delta x \quad (2.88)$$

速度を求めるときと同様に，適当な緩和法で p' を算出する． p' を得られれば，式(2.77),(2.78),(2.79)などを利用して新しい p, u, v を求めることができる．

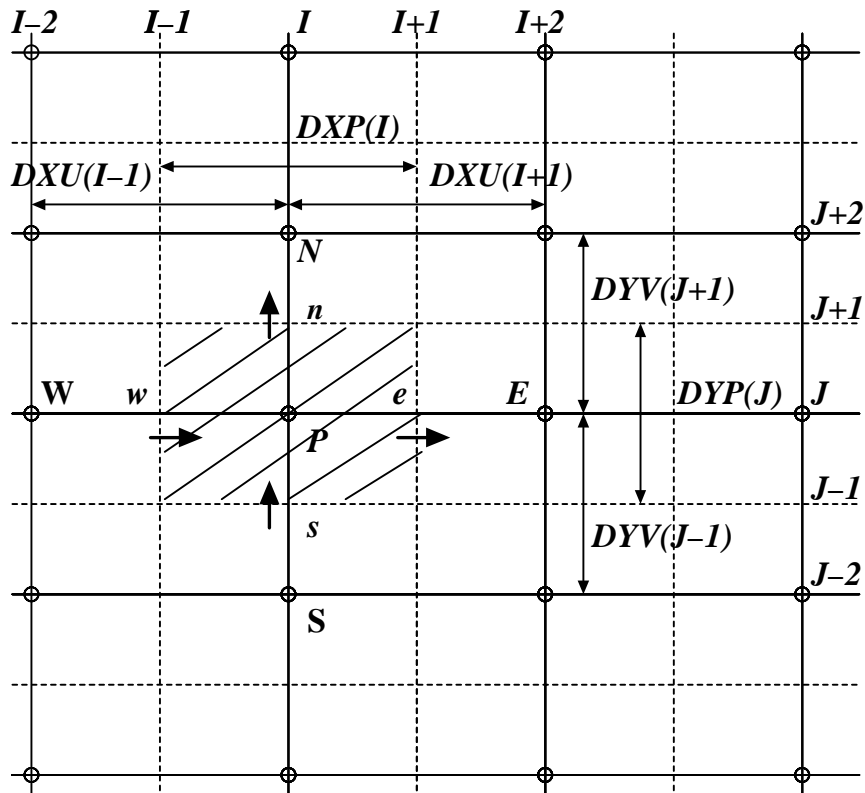


Fig.2.13 連続の式に対する離散化の説明

離散化された運動方程式，圧力補正式の代数方程式を効率的に算出するため，列内の格子点数がどんなに多くても，境界条件の情報が1回の計算で列の端から内部まで伝わるため収束が早い3重対角行列解法（TDMA法）と緩和法を組み合わせた線順法（S.V.Ptankar, 1985）を利用する．非線形性のため，おのこの代数方程式の係数に未知数が含まれていること，および2個の運動方程式と圧力補正式が連立であるため，計算の途中において必ずしもおのこの代数方程式を厳密に解ききる必要はない．したがって，TDMAを組み合わせた緩和法をおのこの代数方程式に独立に適用しながら，最終的な収束解を得る．緩和係数としては，非線形方程式の解法があることや，TDMA

の線順法の緩和であることから，1 以下の値，つまり不足緩和を利用する．圧力補正式については，次式に示すような単純な形で緩和係数 α_p を導入する．

$$p = p' + \alpha_p p' \quad (2.89)$$

運動方程式，数値計算において未知数の配列が 1 個で済むように，解くべき代数方程式に緩和係数を導入しておく．つまり，代数方程式 (2.73) に対して，緩和係数 α_u を導入した式は

$$\frac{a_e}{a_u} u_e = \sum a_{nb} u_{nb} + b + A_e (p_p - p_E) + \frac{1 - \alpha_u}{a_u} a_e u_e^{old} \quad (2.90)$$

で与えられる．ここで求める u は不足緩和された値となる．SIMPLE 解法で利用される緩和係数の値は，流れの安定性や物体形状の複雑さなどによって変化させる必要がある．すなわち，計算が発散する傾向があるときには緩和係数を小さく，また収束が遅いときにはより大きな値をとることが必要になる．

2.1.8 フローチャート

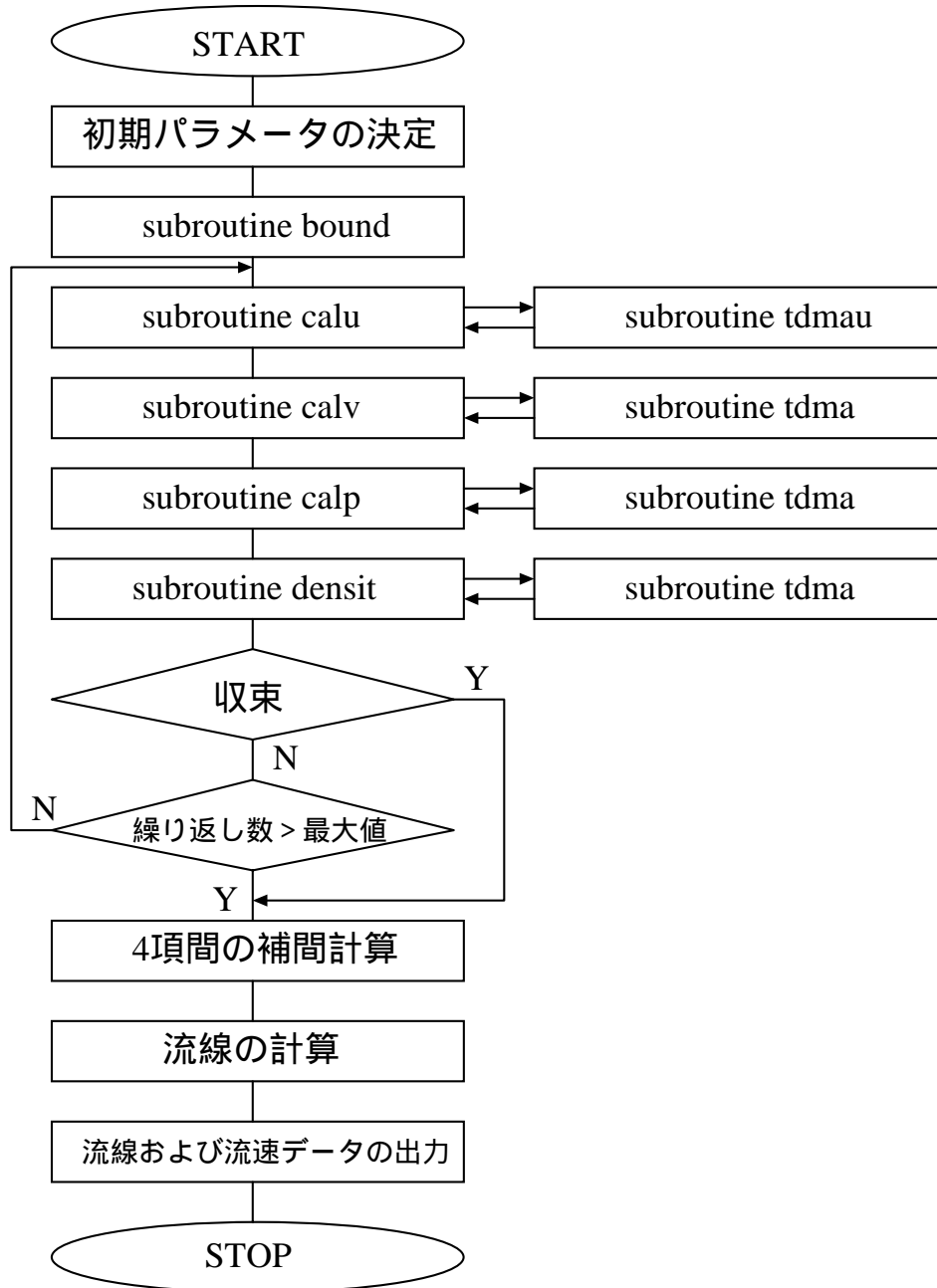


Fig.2.14 フローチャート

- (1) 初期パラメータや格子を決定する .
- (2) 適切な近似値 u^* , v^* , p^* , ρ^* を初期値として与える . (subroutine bound)
- (3) x 方向の運動方程式を , 与えられた u^* を出発点として緩和し , 最新の近似値 u^* を得る . ここで方程式を完全に解ききる必要はなく , 同一係数を用いた線順法で 2 , 3 回程度掃引きするだけで十分である . (subroutine calu)
- (4) y 方向の運動方程式を , (3) と同様に与えられた v^* を出発点として緩和し , 最新の v^* を得る . (subroutine calv)
- (5) 圧力補正式を緩和することによって圧力補正量 p' を得る . 補正量を求めているから , (3) ,(4)の場合とは異なり , p' に対する出発点は常に 0 である . 得られた補正量から最新の p^* と , 速度補正の式を用いた新しい近似値 u^* と v^* を得る . (subroutine calp)
- (6) 密度方程式を , (3) と同様に与えられた ρ^* を出発点として緩和し , 最新の ρ^* を得る . なお , (3) ~ (6)の緩和には TDMA による線順法を用いる . ただし(3)についてはダミーの格子が存在するので(4) ~ (6)とは多少異なるサブルーチンを用いる . (subroutine tdmau , subroutine tdma)

- (7) 得られた u^* , v^* , p^* , ρ^* が収束の条件を満たしていなければ(3)に戻り, より良い近似値を求めて繰り返し計算を実行する。もし, 収束条件を満たしていれば, それらの値が解 u , v , p , ρ となる。収束は運動方程式, 連続の式, 密度方程式の無次元化された残差, つまり方程式の矛盾量がある近似値になったとき, あるいは繰り返し回数をオーバーしたときに, 繰り返し計算を打ち切る。収束結果は, 一定の形式で出力される。

第3章 計算条件

計算方法はコントロールボリューム法を, 格子には計算値の振動を抑えるためにスタガードメッシュを用い陰解法により計算を進めた。計算スキームは中心差分と風上差分およびハイブリッド法を, 連続の式に関しては SIMPLE 法を用いて圧力補正式を導いた。また, 各方程式を解いて得られた近似値の緩和には不足緩和係数を利用し, TDMA による線順法を用いて緩和を行った。なお解析は倍精度計算である。計算の条件を Table.3.1 に示す。

Table.3.1 計算条件

パラメータ	使用した値
密度 ρ_0 (Kg/m ³)	1000.0
渦動粘性係数(水平) ν_H	1.0×10^{-4}
渦動粘性係数(鉛直) ν_V	1.0×10^{-6}
拡散係数(水平) νd_H	1.0×10^{-3}
拡散係数(鉛直) νd_V	1.0×10^{-5}
速度 u の補正係数 αu	0.5
速度 v の補正係数 αv	0.5
圧力 p の補正係数 αp	0.8
密度 ρ の補正係数 $\alpha \rho$	0.5
レイノルズ数	500, 1000, 2000 4000, 6000
構造物の高さ h (m)	0.03, 0.05
水路長さ L (m)	3.0
水路深さ H (m)	0.2
格子幅 $\Delta X, \Delta Y$ (m)	0.01
無次元成層パラメータ K	0.0, 0.1, 0.5, 1.0 1.5, 2.0, 2.5, 3.0

Table.3.1 に示した無次元成層パラメータ K は、 $K = NH / \pi U$ で定義される。ここに、 $N (= \sqrt{(-g/\rho_0)(\partial\rho/\partial y)})$ はバイサラ振動数、 U は一様水平流速である。均一流体の場合には $K=0$ 、成層度が増すにつれて K の値は大きくなる。本研究では水底構造物の高さ h を 0.03m と 0.05m の場合について、それぞれレイノルズ数 Re の値を 500~6000 と変化させた。またそれぞれのレイノルズ数に対して無次元成層パラメータ K の値 0.0~3.0 の範囲で変化させ検討した。

第4章 無次元成層度 K の流れへの影響

コントロールボリューム法により得られた計算結果はスタガードメッシュを用いているため、そのままでは流れ場の密度分布図や細かな流線図を描くことに用いることができない。そのため全ての格子点において二項間、四項間の補間式を用いて流速 u と流速 v そして密度 ρ を補間したものをデータとして出力させた。この Fig.4.1, Fig.4.2 は $Re=2000$, 構造物高さ $h=0.03\text{m}$ とし、無次元成層場 $K=0.0\sim 3.0$ の流れの計算結果から得られた流線図と密度分布図を示している。Fig.4.3, Fig.4.4 は $Re=2000$, 構造物高さ $h=0.05\text{m}$ とし、無次元成層場 $K=0.0\sim 3.0$ の流れの計算結果から得られた流線図と密度分布図を示している。流れは左から右である。Fig.4.1 と Fig.4.3 の流線図より鉛直壁背後に循環流域(定在渦)が見られ、成層度 K の増加とともに循環領域が縮小しているのがわかる。また Fig.4.2 と Fig.4.4 の密度分布図からも成層度 K の増加とともに構造物直前と直上の湧昇のスケールが縮小しているのがわかる。

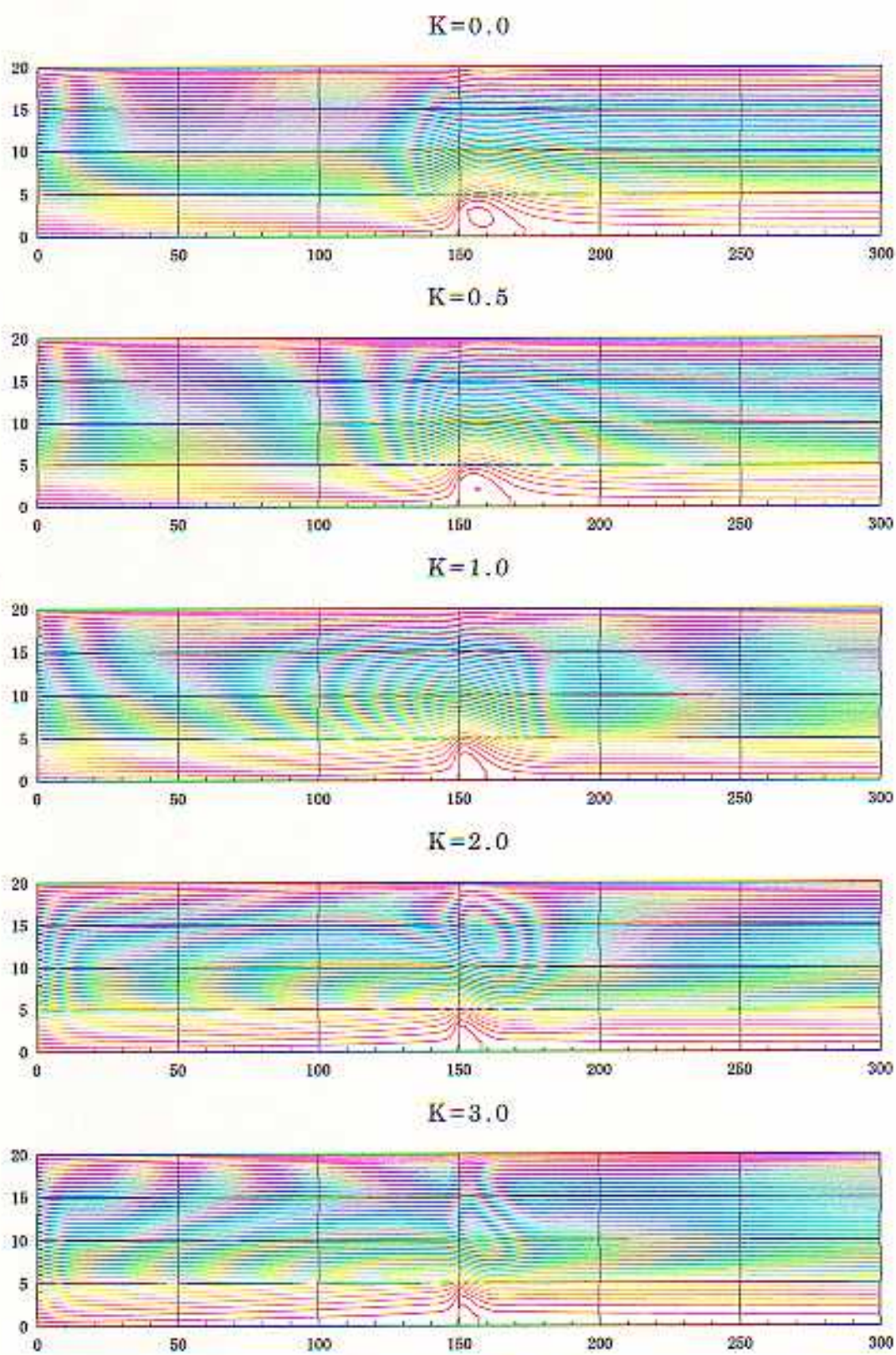


Fig.4.1 流線圖

$Re=2000, h=0.03m$

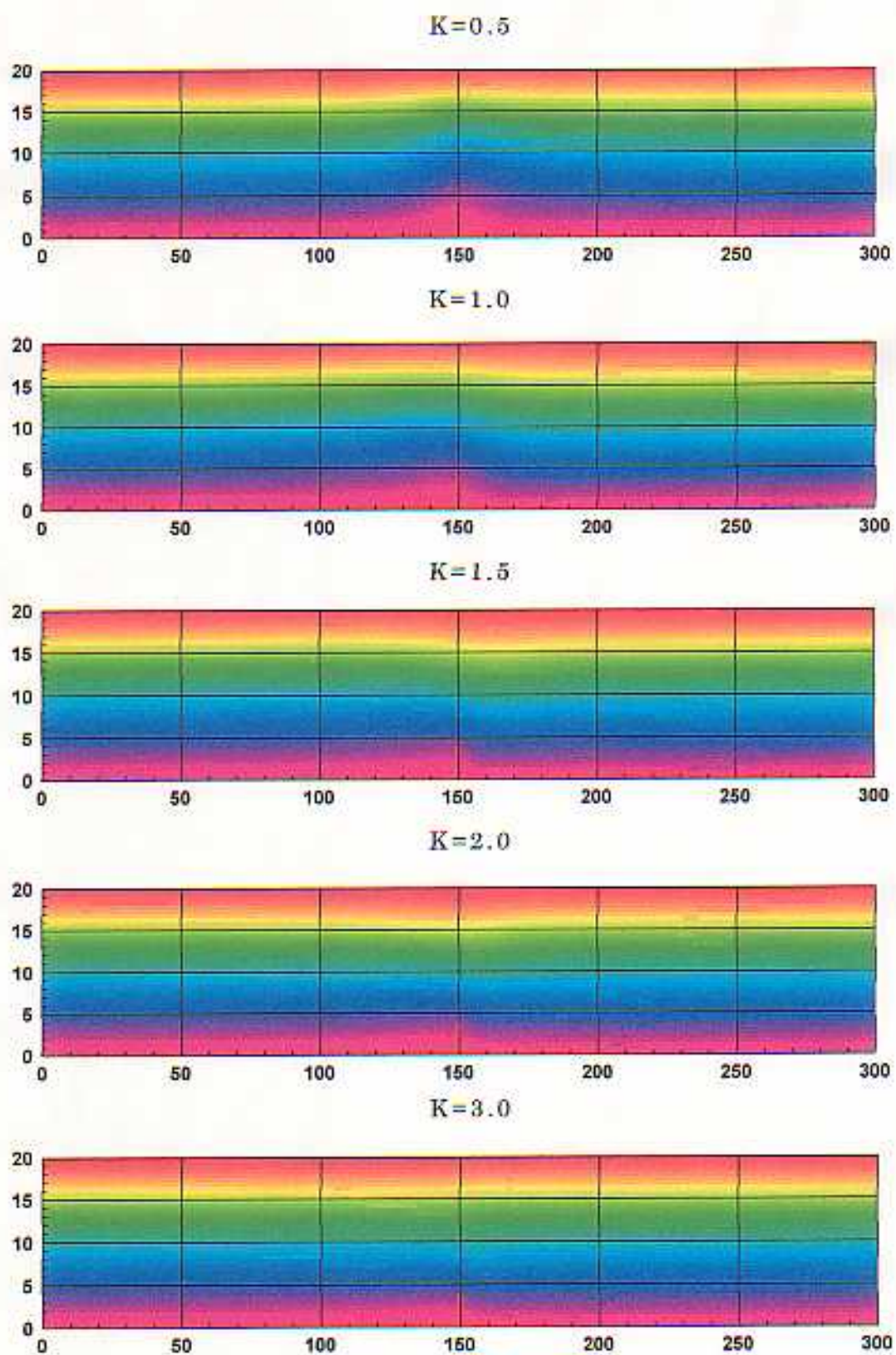


Fig.4.2 密度分布图

$Re=2000, h=0.03m$

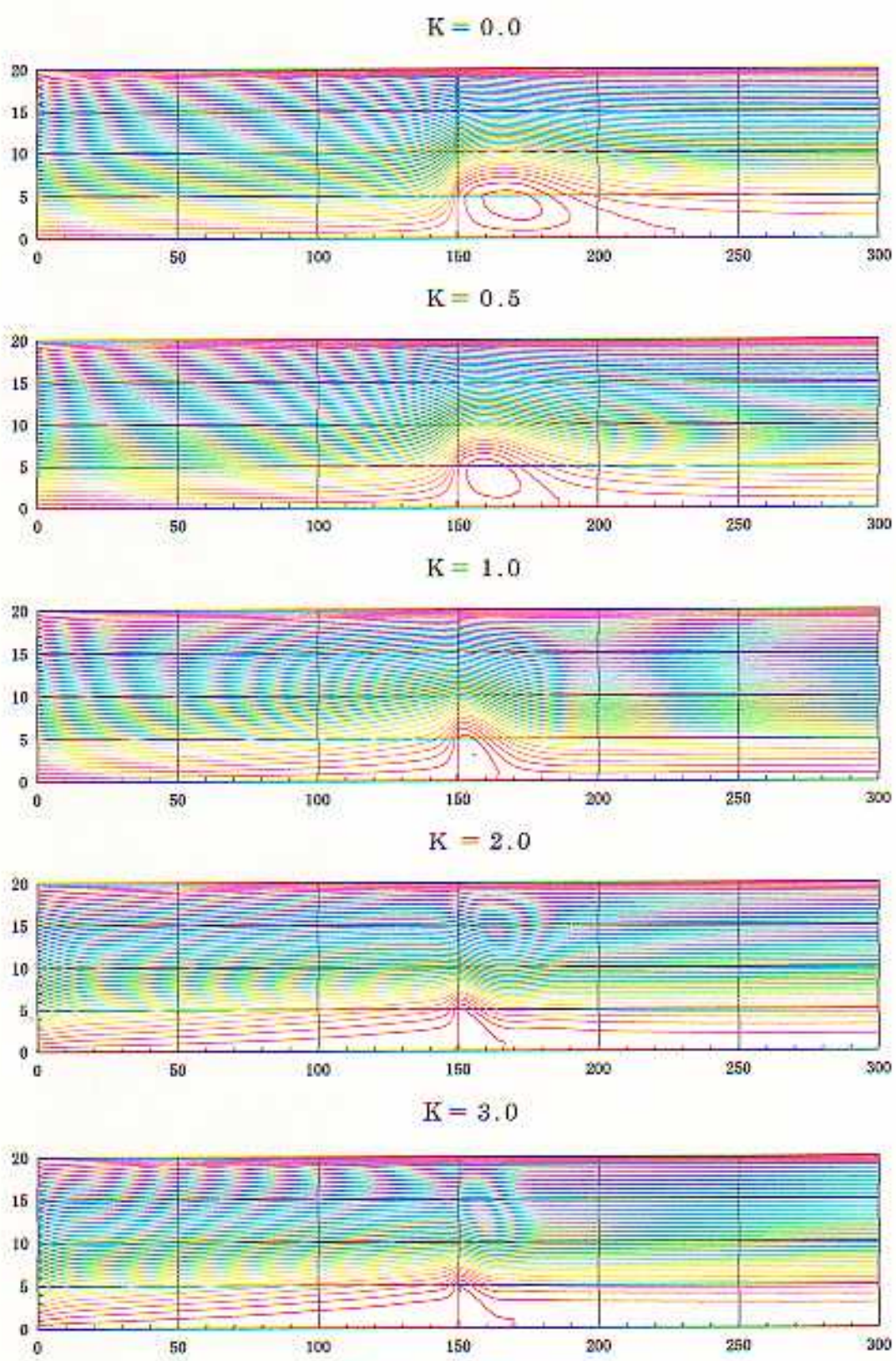


Fig.4.3 流線圖

$Re = 2000, h = 0.05m$

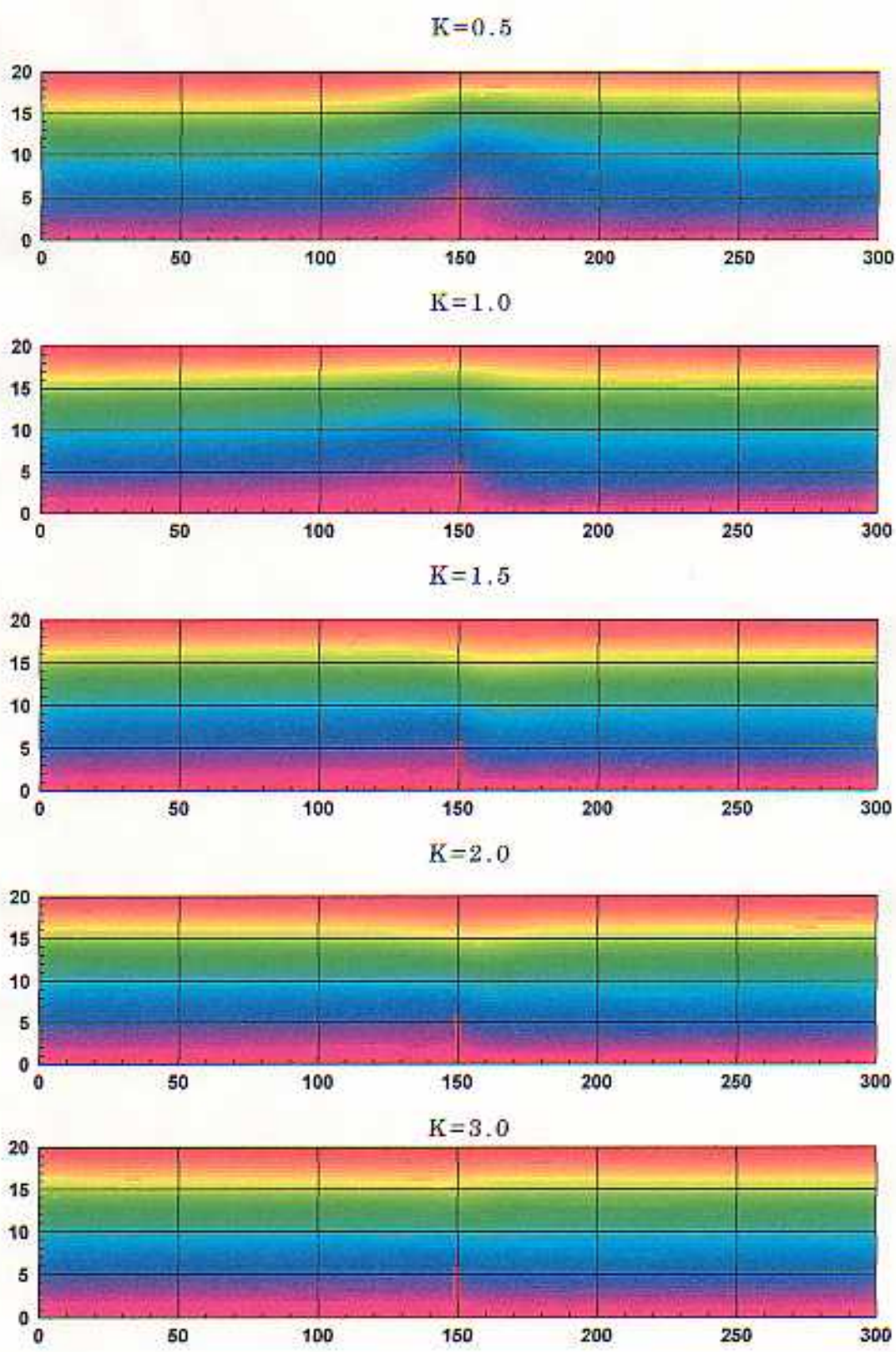


Fig.4.4 密度分布图
 $Re=2000, h=0.05m$

第5章 無次元成層度 K と渦の長さとの関係

以上の計算結果から密度成層度 K と構造物背後の渦の大きさとの関係を調べた。そこで、Fig.5.1 のように構造物から構造物背後の渦の再付着点までの水平長さ x を構造物高さ h で割ったものを無次元パラメータ渦域 Z と定義し、成層度 K と渦域 Z との関係を検討した。

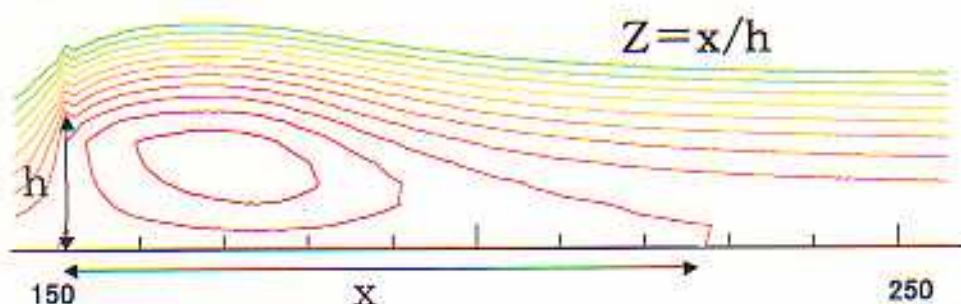


Fig.5.1 無次元パラメータ渦域 Z

構造物高さを $h=0.03\text{m}$, $h=0.05\text{m}$ の2ケースにわけ、それぞれケースについてレイノルズ数を $Re=500, 1000, 2000, 4000, 6000$ と変化させた。また各レイノルズ数に対して、成層度 K を $K=0.0\sim 3.0$ と変化させ、それぞれに対する渦域 Z の値を求めた。その結果を Fig.5.2, Fig.5.3 に示す。Fig.5.2, Fig.5.3 が示すように、 $h=0.03\text{m}, 0.05\text{m}$ とともに $0.0\leq K\leq 0.5$ の間では渦域 Z は十分に大きな値を示し、 K の増大とともにその大きさは縮小していくのが分かる。また $h=0.05\text{m}$ の方が渦がよく発達しているのが分かる。また $1.5\leq K$ になると $h=0.03\text{m}, 0.05\text{m}$ とともに渦域 Z の値は2~3程度に収束してく

ることが分かる。

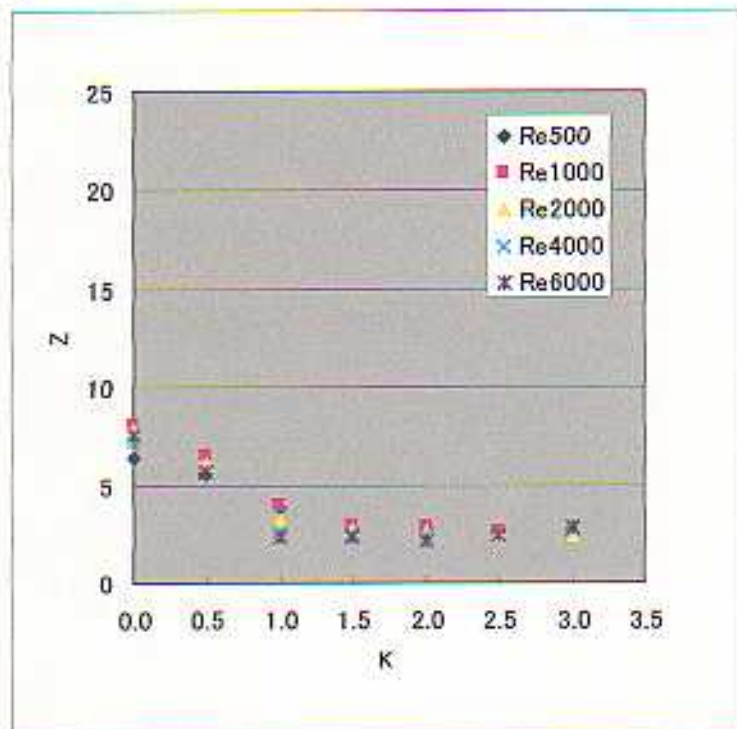


Fig.5.2 Z-K の関係 (h=0.03m)

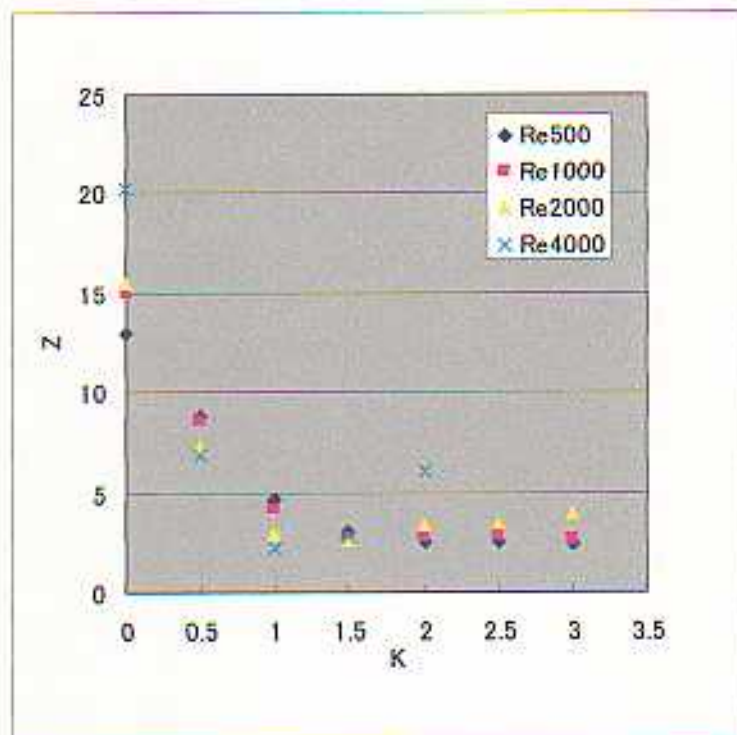


Fig.5.3 Z-K の関係 (h=0.05m)

第 6 章 結論

本研究では、連続成層定常流場を対象とし、水底構造物として鉛直壁を設置した場合の流れ場への影響について数値実験を行い検討を行った。その結果、成層流の構造物の背後流れの動態は、成層度 K に強く影響を受けることがわかった。成層度 K が大きくなるほど水底構造物背後での渦領域が縮小していくことがわかった。特に $K \geq 1.5$ の場合では、構造物背後の渦領域は一様流にくらべ極端に小さくなることが分かった。また成層度 K の増加とともに構造物直前から直上にかけての湧昇の大きさが縮小していくことから成層流の構造物の背後流れの動態は、成層度 K に強く影響を受けることがわかった。

今後の課題としては、水底構造物の形状を鉛直壁だけでなく種々の形状について数値実験し、構造物背後における動態について検討したい。また二層流についても解析していきたい。

謝辞

本修士論文を作成するにあたり、生産環境情報学講座の森健教授、井上英二助教授、原口智和先生そして先輩方々に多大なご指導、ご協力を賜りましたことを心から感謝するとともに、深甚なる謝意を表します。

引用文献

- 1) 内田孝紀, 大屋祐二: 有限流路内の二次元丘陵地形を過ぎる安定成層流の非定常性, ながれ 17(1998)45-56.
- 2) スハン V. パタンカー原著(1985): コンピュータによる熱移動と流れの数値解析, 森北出版株式会社, p11-142.
- 3) 池田昌平: 水底構造物を過ぎる成層流, 九州大学大学院修士論文(2001).

APPENDIX

```

program main
  implicit none
  integer,parameter :: nx=303,ny=23
  integer :: i,j,m,n,niter,maxit,nsydd
  real(8) :: sormax,lx,ly,flowin,xmonin,ymonin,arden,sorce,cn,cs,ce,cw,ds,dn,de,dw
&
      ,resor,ardenx,durf,urfd,grav
  integer :: iend,jend                ! all
  real(8) :: dx,dy                    ! grid
  real(8),dimension(nx) :: dxu        ! grid
  real(8),dimension(nx+1) :: dxp      ! grid
  real(8),dimension(ny) :: dyv        ! grid
  real(8),dimension(ny+1) :: dyp      ! grid
  real(8) :: resoru,nswpv,urfv        ! uvel
  real(8) :: resorv,nswpv,urfv        ! vvel
  integer :: nswpp                     ! pcor
  real(8) :: urfp,resorm,resord        ! pcor
  real(8),dimension(nx,ny) :: du,dv    ! pcor
  real(8) :: ustart                    ! var
  real(8),dimension(nx,ny) :: v,p,pp,psi,d,dd ! var
  real(8),dimension(nx,0:ny) :: u      ! var
  real(8) :: visu,visv,visud,visvd,den ! fluid
  integer :: obsi,obsj                  ! obs
  real(8),dimension(0:nx,ny-1) :: ap,an,as,ae,aw,su ! coef
  real(8),dimension(nx,ny) :: dk
  integer,dimension(nx,ny) :: s
  integer,dimension(nx,ny) :: t

  niter=0
  nswpu=2
  nswpv=2
  nswpp=3
  nsydd=3
! fluid properties
  den=1000.0

```

```

visu=1.0D-4
visv=1.0D-6
visud=1.0D-3
visvd=1.0D-5
grav=9.8
! 密度勾配
  durf=-203.7315
!  durf=0.0D0
! program control and monitor
  maxit=10000000
  urfu=0.5
  urfv=0.5
  urfp=0.8
  urfd=0.5
  sormax=1.0E-1

! 格子と初期値の設定
! 格子
  obsi=151
  obsj=6
  lx=3.0

  ly=0.2
  m=300
  n=20
  iend=m+1
  jend=n+1
  dx=lx/real(m)
  dy=ly/real(n)
! 初期値
  call init

! 無次元化するための代表値
  flowin=0.0
  xmonin=0.0
  ymonin=0.0

```

```

do j=2,jend,2
  arden=dyp(j)
  ardenx=dyp(j)
  flowin=flowin+arden*ustart
  xmonin=xmonin+ardenx*ustart*ustart
  ymonin=ymonin+ardenx*ustart*den
end do

```

```

open (1,file='c:\¥my documents¥修士論文衛藤結果¥output1_1.txt',status='new')

```

! 反復計算

```

100 niter=niter+1
  call bound
  call calu
  call calv
  call calp
  call densit
  resorm=resorm/flowin
  resoru=resoru/xmonin
  resorv=resorv/xmonin
  resord=resord/ymonin
write(*,*) niter

```

! 反復計算の出力

```

do i=2,iend-1,2
  do j=2,jend-1,2
    u(i,j)=0.5D0*(u(i-1,j)+u(i+1,j))
    v(i,j)=0.5D0*(v(i,j-1)+v(i,j+1))
  end do
end do

```

```

do i=3,iend-2,2
  do j=2,jend-1,2
    v(i,j)=0.25D0*(v(i-1,j+1)+v(i+1,j+1)+v(i-1,j-1)+v(i+1,j-1))
    p(i,j)=0.5D0*(p(i-1,j)+p(i+1,j))
    d(i,j)=0.5D0*(d(i-1,j)+d(i+1,j))
  end do
end do

```

```

    end do
end do

do i=2,iend-1,2
  do j=3,jend-2,2
    u(i,j)=0.25D0*(u(i-1,j+1)+u(i+1,j+1)+u(i-1,j-1)+u(i+1,j-1))
    p(i,j)=0.5D0*(p(i,j-1)+p(i,j+1))
    d(i,j)=0.5D0*(d(i,j-1)+d(i,j+1))
  end do
end do

```

```

do i=3,iend-2,2
  do j=3,jend-2,2
    u(i,j)=0.25D0*(u(i-1,j+1)+u(i+1,j+1)+u(i-1,j-1)+u(i+1,j-1))
    v(i,j)=0.25D0*(v(i-1,j+1)+v(i+1,j+1)+v(i-1,j-1)+v(i+1,j-1))
    p(i,j)=0.25D0*(p(i-1,j+1)+p(i+1,j+1)+p(i-1,j-1)+p(i+1,j-1))
    d(i,j)=0.25D0*(d(i-1,j+1)+d(i+1,j+1)+d(i-1,j-1)+d(i+1,j-1))
  end do
end do

```

```

do i=1,iend
  u(i,jend)=u(i,jend-1)
  p(i,jend)=p(i,jend-1)
  d(i,jend)=d(i,jend-1)
end do

```

```

do j=1,obsj
  u(obsi,j)=0.0D0
  v(obsi,j)=0.0D0
  p(obsi,j)=0.0D0
  d(obsi,j)=0.0D0
end do

```

! 流線の計算

```

do i=1,iend
  psi(i,2)=dy*u(i,2)
end do

```

```
do i=1,iend
  do j=2,jend-2
    psi(i,j+1)=psi(i,j)+dy*u(i,j+1)
  end do
end do
```

```
do i=1,iend
  psi(i,jend)=ustart*ly
end do
```

```
do j=1,obsj
  psi(obsi,j)=0.0D0
end do
```

!計算結果の出力

```
if (MOD(niter,15000)==0) then
```

!実密度計算

```
do i=1,iend
  do j=1,jend
    dk(i,j)=(durf*(j-1)*(1.0D-2))+(1000.0D0+(-durf*0.2D0))
  end do
end do
```

```
do j=1,obsj
  dk(obsi,j)=0.0D0
end do
```

```
do i=1,iend
  do j=1,jend
    d(i,j)=d(i,j)+dk(i,j)
  end do
end do
```

!実圧力計算

```
! do i=1,iend
!   do j=1,jend
!     ps(i,j)=grav((
```

! 流線の出力

```
  do j=0,jend
    psi(0,j)=j-1
  end do
  do i=0,iend
    s(i,j)=i-1
    write(1,'(1x,i5,22f15.11)') s(i,j),psi(i,1:jend)
  end do
```

! 流速の出力

```
  do i=1,iend
    do j=1,jend
      s(i,j)=i-1
      t(i,j)=j-1
      write(1,'(1x,2i5,2f15.12)') s(i,j),t(i,j),u(i,j),v(i,j)
    end do
  end do
```

!密度の出力

```
  do j=0,jend
    d(0,j)=j-1
  end do
  do i=0,iend
    s(i,j)=i-1
    write(1,'(1x,i5,22f15.8)') s(i,j),d(i,1:jend)
  end do
```

!圧力の出力

```
! do i=1,iend
!   write(1,'(1x,22f15.8)') p(i,1:jend)
! end do
```

```
end if
```

```
!計算回数 1 0 0 回ごとの残差の出力 (収束確認)
```

```
! if(MOD(niter,10)==0) then
```

```
! write(1,'(1x,4e15.8)' resord,resorm,resoru,resorv
```

```
! end if
```

```
! 収束の確認
```

```
sorce=amax1(resord,resorm,resoru,resorv)
```

```
if ((niter==140).and.(sorce>1.0E4*sormax).or.(niter==maxit)) then
```

```
write(*,*)'計算結果が発散しているか、さらなる反復計算が必要です'
```

```
stop
```

```
end if
```

```
! if ((niter>1 .and. sorce>sormax) .or. (niter==1)) go to 100
```

```
if (niter<15000) go to 100
```

```
close(1)
```

```
stop
```

```
contains
```

```
subroutine init
```

```
implicit none
```

```
do i=1,iend,2
```

```
dxu(i)=2D0*dx
```

```
end do
```

```
do i=2,iend-1,2
```

```
dxp(i)=2D0*dx
```

```
end do
```

```
do j=1,jend,2
```

```
dyv(j)=2D0*dy
```

```
end do
```

```
do j=2,jend-1,2
```



```
    dyp(j)=2D0*dy
end do
```

```
do i=1,iend
  do j=1,jend
    u(i,j)=0.0
    v(i,j)=0.0
    p(i,j)=0.0
    pp(i,j)=0.0
    du(i,j)=0.0
    dv(i,j)=0.0
    psi(i,j)=0.0
    d(i,j)=0.0
    dd(i,j)=0.0
  end do
end do
```

```
do j=1,jend
  ustart=0.03
  u(1,j)=ustart
end do
```

```
return
end subroutine init
```

```
subroutine calu
implicit none
```

```
do j=2,jend-1,2
  do i=3,iend-2,2
    if (i==obsi .and. j<=obsj) go to 400
```

! 対流項の係数計算

```
    cn=0.5D0*(v(i-1,j+1)+v(i+1,j+1))*dxu(i)
    cs=0.5D0*(v(i-1,j-1)+v(i+1,j-1))*dxu(i)
    ce=0.5D0*(u(i+2,j)+u(i,j))*dyp(j)
    cw=0.5D0*(u(i,j)+u(i-2,j))*dyp(j)
```

```

! 粘性項の係数計算
    ds=visv*dxu(i)/dyv(j-1)
! if (i==obsi .and. j==obsj+2) ds=0.0
    dn=visv*dxu(i)/dyv(j+1)
    de=visu*dyp(j)/dyp(i+1)
! if (i==obsi-2 .and. j<=obsj) de=0.0
    dw=visu*dyp(j)/dyp(i-1)
! if (i==obsi+2 .and. j<=obsj) dw=0.0
! メインの係数の計算
    an(i,j)=Dmax1(Dabs(0.5D0*cn),dn)-0.5D0*cn
    as(i,j)=Dmax1(Dabs(0.5D0*cs),ds)+0.5D0*cs
    ae(i,j)=Dmax1(Dabs(0.5D0*ce),de)-0.5D0*ce
    aw(i,j)=Dmax1(Dabs(0.5D0*cw),dw)+0.5D0*cw
    du(i,j)=dyp(j)                ! 面積 = y × 1
    su(i,j)=du(i,j)*(p(i-1,j)-p(i+1,j))/den
400 end do
    end do
! top water
! do i=3,iend-2,2
!   an(i,jend-1)=0.0
! end do
! bottom wall
! do i=3,iend-2,2
!   as(i,2)=0.0
! end do
! 構造物頂上
! as(obsi,obsj+2)=0.0
resoru=0.0
do j=2,jend-1,2
    do i=3,iend-2,2
        if (i==obsi .and. j<=obsj) go to 200
        ap(i,j)=an(i,j)+as(i,j)+ae(i,j)+aw(i,j)+(ce+cn-cw-cs)
        du(i,j)=du(i,j)/ap(i,j)                ! de=Ae/ae
        resor=an(i,j)*u(i,j+2)+as(i,j)*u(i,j-2)+ae(i,j)*u(i+2,j)    &
        +aw(i,j)*u(i-2,j)-ap(i,j)*u(i,j)+su(i,j)
        resoru=resoru+Dabs(resor)
    end do
end do

```

! simple 法に対する不足緩和係数の適用

```
ap(i,j)=ap(i,j)/urfu
su(i,j)=su(i,j)+(1.0D0-urfu)*ap(i,j)*u(i,j)
du(i,j)=du(i,j)*urfu
```

200 end do

end do

! 方程式の計算

```
do n=1,nswpu
  call tdmaw(3,2,iend-2,jend-1,u)
end do
return
end subroutine calu
```

subroutine calv

implicit none

```
do j=3,jend-2,2
```

```
  do i=4,iend-3,2
```

! 対流項の係数計算

```
cn=0.5D0*(v(i,j+2)+v(i,j))*dyp(i)
cs=0.5D0*(v(i,j)+v(i,j-2))*dyp(i)
ce=0.5D0*(u(i+1,j+1)+u(i+1,j-1))*dyv(j)
cw=0.5D0*(u(i-1,j+1)+u(i-1,j-1))*dyv(j)
```

! 粘性項の係数計算

```
ds=visv*dyp(i)/dyp(j-1)
```

```
dn=visv*dyp(i)/dyp(j+1)
```

```
de=visu*dyv(j)/dxu(i+1)
```

```
! if (i==obsi-1 .and. j<obsj) de=0.0
```

```
dw=visu*dyv(j)/dxu(i-1)
```

```
! if (i==obsi+1 .and. j<obsj) dw=0.0
```

! メインの係数の計算

```
an(i,j)=Dmax1(Dabs(0.5D0*cn),dn)-0.5D0*cn
```

```
as(i,j)=Dmax1(Dabs(0.5D0*cs),ds)+0.5D0*cs
```

```
ae(i,j)=Dmax1(Dabs(0.5D0*ce),de)-0.5D0*ce
```

```
aw(i,j)=Dmax1(Dabs(0.5D0*cw),dw)+0.5D0*cw
```

dv(i,j)=dxp(i)

! 面積 = $x \times 1$

su(i,j)=(dv(i,j)*(p(i,j-1)-p(i,j+1))-0.5D0*grav*(d(i,j+1)+d(i,j-1))*dxp(i)*dyv(j))/den

end do

end do

! 構造物前方

! do j=3,obsj-1,2

! ae(obsi-1,j)=0.0

! end do

! 構造物後方

! do j=3,obsj-1,2

! aw(obsi+1,j)=0.0

! end do

resorv=0.0

do j=3,jend-2,2

do i=4,iend-3,2

ap(i,j)=an(i,j)+as(i,j)+ae(i,j)+aw(i,j)+(ce+cn-cw-cs)

dv(i,j)=dv(i,j)/ap(i,j) ! dn=An/an

resor=an(i,j)*v(i,j+2)+as(i,j)*v(i,j-2)+ae(i,j)*v(i+2,j) &

+aw(i,j)*v(i-2,j)-ap(i,j)*v(i,j)+su(i,j)

resorv=resorv+Dabs(resor)

! simple 法に対する不足緩和係数の適用

ap(i,j)=ap(i,j)/urfv

su(i,j)=su(i,j)+(1.0D0-urfv)*ap(i,j)*v(i,j)

dv(i,j)=dv(i,j)*urfv

end do

end do

! 方程式の計算

do n=1,nswpv

call tdma(4,3,iend-3,jend-2,v)

end do

return

end subroutine calv

```

subroutine calp
implicit none
real(8) :: smp
resorm=0.0
do i=2,iend-1,2
  do j=2,jend-1,2
! メインの係数の計算
    an(i,j)=dxp(i)*dv(i,j+1)
    as(i,j)=dxp(i)*dv(i,j-1)
    ae(i,j)=dyp(j)*du(i+1,j)
    aw(i,j)=dyp(j)*du(i-1,j)
! 対流項の係数計算
    cn=dxp(i)*v(i,j+1)
    cs=dxp(i)*v(i,j-1)
    ce=dyp(j)*u(i+1,j)
    cw=dyp(j)*u(i-1,j)
    smp=cn-cs+ce-cw
    su(i,j)=-smp
    resorm=resorm+Dabs(smp)
  end do
end do

! 構造物前方
! do j=2,obsj,2
!   ae(obsi-1,j)=0.0
! end do
! 構造物後方
! do j=2,obsj,2
!   aw(obsi+1,j)=0.0
! end do
do i=2,iend-1,2
  do j=2,jend-1,2
    ap(i,j)=an(i,j)+as(i,j)+ae(i,j)+aw(i,j)
  end do
end do

```

```

do n=1,nswpp
  call tdma(2,2,iend-1,jend-1,pp)

end do
! 流速の計算
do i=3,iend-2,2
  do j=2,jend-1,2
    if (i==obsi .and. j<obsj) then
      u(i,j)=0.0
    else
      u(i,j)=u(i,j)+du(i,j)*(pp(i-1,j)-pp(i+1,j))
    end if
  end do
end do

do i=4,iend-3,2
  do j=3,jend-2,2
    v(i,j)=v(i,j)+dv(i,j)*(pp(i,j-1)-pp(i,j+1))
  end do
end do

! 圧力の計算
do i=2,iend-1,2
  do j=2,jend-1,2
    p(i,j)=p(i,j)+urfp*pp(i,j)
    pp(i,j)=0.0
  end do
end do

return
end subroutine calp

subroutine densit
implicit none

```

```

! real(8) :: dpref
do i=2,iend-1,2
  do j=2,jend-1,2
! 対流項の係数計算
    cn=0.5D0*v(i,j+1)*dyp(i)
    cs=0.5D0*v(i,j-1)*dyp(i)
    ce=0.5D0*u(i+1,j)*dyp(j)
    cw=0.5D0*u(i-1,j)*dyp(j)
! 拡散項の計算
    ds=visvd*dyp(i)/dyv(j-1)
    dn=visvd*dyp(i)/dyv(j+1)
    de=visud*dyp(j)/dxu(i+1)
    dw=visud*dyp(j)/dxu(i-1)
! メインの係数の計算
    an(i,j)=Dmax1(Dabs(cn),dn)-cn
    as(i,j)=Dmax1(Dabs(cs),ds)+cs
    ae(i,j)=Dmax1(Dabs(ce),de)-ce
    aw(i,j)=Dmax1(Dabs(cw),dw)+cw
    dd(i,j)=dyp(i)*dyp(j)
    su(i,j)=-0.5D0*durf*(v(i,j+1)+v(i,j-1))*dyp(i)*dyp(j)
  end do
end do
! 構造物前方
! do j=2,obsj,2
!   ae(obsi-1,j)=0.0
! end do
! 構造物後方
! do j=2,obsj,2
!   aw(obsi+1,j)=0.0
! end do

resord=0.0
do i=2,iend-1,2
  do j=2,jend-1,2
    ap(i,j)=an(i,j)+as(i,j)+ae(i,j)+aw(i,j)+2*(ce+cn-cw-cs)
    resor=an(i,j)*d(i,j+2)+as(i,j)*d(i,j-2)+ae(i,j)*d(i+2,j) &

```

```

      +aw(i,j)*d(i-2,j)-ap(i,j)*d(i,j)+su(i,j)
      resord=resord+Dabs(resor)
      ap(i,j)=ap(i,j)/urfd
      su(i,j)=su(i,j)+(1.0D0-urfu)*ap(i,j)*d(i,j)
      dd(i,j)=dd(i,j)*urfd
    end do
  end do

  do n=1,nswdd
    call tdma(2,2,iend-1,jend-1,d)
  end do

! 密度計算
!  do i=2,iend-1,2
!    do j=2,jend-1,2
!      d(i,j)=d(i,j)+urfd*dd(i,j)
!      dd(i,j)=0.0
!    end do
!  end do

return
  end subroutine densit

subroutine tdmau(istart,jstart,iend,jend,phi)
  implicit none
  integer :: jstm1,istart,jstart,iend,jend,jj
  real(8) :: term
  real(8),dimension(nx,0:ny) :: phi
  real(8),dimension(0:ny-1) :: a,b,c,d

  jstm1=jstart
  a(jstm1)=0.0
  a(0)=a(2)
  c(0)=c(2)

```



```

do i=istart,iend,2
  c(jstm1)=phi(i,jstm1)
  do j=jstart,jend,2
    if (i==obsi .and. j<=obsj) go to 300
! TDMA 法の係数
    a(j)=an(i,j)
    b(j)=as(i,j)
    c(j)=ae(i,j)*phi(i+2,j)+aw(i,j)*phi(i-2,j)+su(i,j)
    d(j)=ap(i,j)
! 係数 Pi と Qi の計算
    term=1.0D0/(d(j)-b(j)*a(j-2))
    a(j)=a(j)*term
    c(j)=(c(j)+b(j)*c(j-2))*term
  end do
! PHI の後退代入計算
  do jj=jstart,jend,2
    j=jend+jstm1-jj
    phi(i,j)=a(j)*phi(i,j+2)+c(j)
  end do
300 end do
return
end subroutine tdmau

```

```

subroutine tdma(istart,jstart,iend,jend,phi)
implicit none
integer :: jstm1,istart,jstart,iend,jend,jj
real(8) :: term
real(8),dimension(nx,ny) :: phi
real(8),dimension(0:ny) :: a,b,c,d

```

```

jstm1=jstart
a(jstm1)=0.0
a(0)=a(2)
c(0)=c(2)
a(1)=a(3)

```

```

c(1)=c(3)
do i=istart,iend,2
  c(jstm1)=phi(i,jstm1)
  do j=jstart,jend,2
! TDMA 法の係数
    a(j)=an(i,j)
    b(j)=as(i,j)
    c(j)=ae(i,j)*phi(i+2,j)+aw(i,j)*phi(i-2,j)+su(i,j)
    d(j)=ap(i,j)
! 係数 Pi と Qi の計算
    term=1.0D0/(d(j)-b(j)*a(j-2))
    a(j)=a(j)*term
    c(j)=(c(j)+b(j)*c(j-2))*term
  end do
! PHI の後退代入計算
  do jj=jstart,jend,2
    j=jend+jstm1-jj
    phi(i,j)=a(j)*phi(i,j+2)+c(j)
  end do
end do
return
end subroutine tdma

```

```

subroutine bound
implicit none

```

```

do i=1,iend
  u(i,1)=0.0
  v(i,1)=0.0
  v(i,jend)=0.0
end do

```

```

do i=1,iend
  u(i,jend+1)=u(i,jend-1)
  d(i,jend+1)=d(i,jend-1)
  u(i,0)=-u(i,2)

```

```

    d(i,0)=d(i,2)
end do

do i=1,iend
    u(i,jend)=u(i,jend-1)
    d(i,jend)=d(i,jend-1)
end do

do j=1,jend
    u(iend,j)=u(iend-2,j)
    du(iend,j)=du(iend-2,j)
    v(iend-1,j)=v(iend-3,j)
    dv(iend-1,j)=dv(iend-3,j)
    v(iend,j)=v(iend-1,j)
    dv(iend,j)=dv(iend-1,j)
    v(2,j)=v(4,j)
    dv(2,j)=dv(4,j)
    du(1,j)=du(3,j)
    d(iend+1,j)=d(iend-1,j)
    dd(iend+1,j)=dd(iend-1,j)
    d(iend,j)=d(iend-1,j)
    p(iend,j)=p(iend-1,j)
    pp(iend,j)=pp(iend-1,j)
end do

do j=1,obsj
    u(obsi,j)=0.0
    v(obsi,j)=0.0
    p(obsi,j)=0.0
    d(obsi,j)=0.0
end do

return
end subroutine bound
end program main

```

