

5K-1

一般曲線座標系における流線作図手法

石原誠治* 秋葉幸範** 土肥 俊** 速水 謙**

* 日本電気技術情報システム開発(株) **日本電気(株) C & C 情報研究所

1. はじめに

3次元流体解析のような大規模科学技術計算において、膨大な計算結果を表示するグラフィック技術は重要な位置を占めている。流体解析では流線の表示は流れのベクトル場を把握する上で重要である。しかし、この表示アルゴリズムを詳細に記した論文¹⁾は少ない。定常流の場合、粒子(質量、抗力ともに零)の軌跡は流線と一致する。一般曲線座標系に差分法を適用する境界適合法は、物理空間の曲線格子から写像空間の立方格子へ座標変換して解析を行う手法である。粒子の軌跡を計算する場合、物理空間の曲線格子では粒子近傍の格子点の検索に時間がかかると言う問題がある。本稿では、この問題を克服するために粒子の軌跡の計算を写像空間の立方格子で行い、流線を表示する方法を提案する。

2. 軌跡計算アルゴリズム

2.1 座標系の写像

Thompsonらの方法¹⁾によると、境界適合法における物理空間の曲線格子から写像空間の立方格子への写像はPoisson方程式を差分法で解いて行われる。本手法ではこの写像を容易にするために図1に示すように物理空間の曲線格子において8つの格子点(i, j, k), (i+1, j, k), (i+1, j+1, k), (i, j+1, k), (i, j, k+1), (i+1, j, k+1), (i+1, j+1, k+1), (i, j+1, k+1)に囲まれる領域(以下、6面体セルと呼ぶ。)が、写像空間の立方格子においてそれぞれ対応する格子点に囲まれる領域(以下、立方体セルと呼ぶ。)へ写像されるとして次の式を用いる。

$$\begin{aligned} X &= \alpha_0 + \alpha_1 \xi + \alpha_2 \eta + \alpha_3 \zeta + \alpha_4 \xi \eta + \alpha_5 \eta \zeta + \alpha_6 \zeta \xi + \alpha_7 \xi \eta \zeta \\ Y &= \beta_0 + \beta_1 \xi + \beta_2 \eta + \beta_3 \zeta + \beta_4 \xi \eta + \beta_5 \eta \zeta + \beta_6 \zeta \xi + \beta_7 \xi \eta \zeta \quad \dots\dots\dots (1) \\ Z &= \gamma_0 + \gamma_1 \xi + \gamma_2 \eta + \gamma_3 \zeta + \gamma_4 \xi \eta + \gamma_5 \eta \zeta + \gamma_6 \zeta \xi + \gamma_7 \xi \eta \zeta \end{aligned}$$

但し、X, Y, Zは物理空間での座標、ξ, η, ζは写像空間での座標である。また、α_i, β_i, γ_i(i=0-7)の係数は(1)式に6面体セルを形成する格子点の座標と立方体セルを形成する格子点の座標を代入して得られる8つの式からそれぞれ求める。

2.2 粒子の軌跡計算

粒子の軌跡の計算は写像空間において次の式をオイラー法で数値積分することにより行われる。

$$d\mathbf{p} / dt = \mathbf{v}(\mathbf{p}) \quad \dots\dots\dots (2)$$

但し、pは写像空間における粒子の座標(ξ, η, ζ)、vは粒子の写像空間における速度(u, v, w)である。粒子の写像空間における速度を求めるには先ず、格子点での物理空間における速度を写像空間における速度に変換する。これは(3)式に示すように、物理空間における速度にJacobian行列を乗じて変換する。

$$(\mathbf{u}, \mathbf{v}, \mathbf{w})^t = [\partial(\xi, \eta, \zeta) / \partial(X, Y, Z)] (\mathbf{U}, \mathbf{V}, \mathbf{W})^t \quad \dots\dots\dots (3)$$

但し、(u, v, w)は写像空間における速度、(U, V, W)は物理空間における速度である。次に粒子の速度を粒子近傍の格子点の速度から補間して求める。この格子点は粒子を包含する立方体セルを形成する格子点である。粒子の速度をこの格子点の速度から次式を用いて補間する。

$$\begin{aligned} \mathbf{u} &= \varepsilon_0 + \varepsilon_1 \xi + \varepsilon_2 \eta + \varepsilon_3 \zeta + \varepsilon_4 \xi \eta + \varepsilon_5 \eta \zeta + \varepsilon_6 \zeta \xi + \varepsilon_7 \xi \eta \zeta \\ \mathbf{v} &= \delta_0 + \delta_1 \xi + \delta_2 \eta + \delta_3 \zeta + \delta_4 \xi \eta + \delta_5 \eta \zeta + \delta_6 \zeta \xi + \delta_7 \xi \eta \zeta \quad \dots\dots\dots (4) \\ \mathbf{w} &= \sigma_0 + \sigma_1 \xi + \sigma_2 \eta + \sigma_3 \zeta + \sigma_4 \xi \eta + \sigma_5 \eta \zeta + \sigma_6 \zeta \xi + \sigma_7 \xi \eta \zeta \end{aligned}$$

但し、ε_i, δ_i, σ_i(i=0-7)の係数は立方体セルを形成する格子点での速度と座標を(4)式に代入してそれぞれ求める。

また、(2)式を数値積分する時の時間刻み幅は、積分の精度を維持するために写像空間における立方体セル毎に、次のような手順で設定する。先ず、粒子の立方体セルの始点座標における速度で粒子が立方体セルを抜ける時間(以下、脱出時間と呼ぶ。)を求める。この時間を基準時間刻み幅として脱出時間分だけ、(2)式を積分して粒子の到達座標を求め、次にこの刻み幅の半分の刻み幅で脱出時間分だけ同様に積分して粒子の到達座標を求める。これらを比較して、この差が予め指定されている精度以下であるならば、この基準時間刻み幅で粒子が立方体セルを抜けるまで積分を行うことにする。この差が精度以下でないならば、この差が精度以下になるまで基準時間刻み幅を半減して、同様な処理を繰り返す。

A Method of Drawing Stream Lines for General Curvilinear Coordinate Systems.

*ISHIHARA Seiji, **AKIBA Yukinori, **DOI Shun, **HAYAMI Ken

*NEC Scientific Information System Development Ltd.

**Information Technology Res. Labs., NEC Corporation

2.3 アルゴリズム

粒子軌跡計算の手順を次に示す(図2参照)。① 軌跡の出発点(最初の粒子の座標)は物理空間における座標で与えられているので、物理空間における軌跡の出発点を包含する6面体セルを検索する。この6面体セルが写像空間において対応する立方体セルを包含立方体セル(粒子を包含する立方体セル)とする。(1)式より軌跡の出発点の座標を写像空間に変換する。以後の軌跡の計算は全て写像空間で行われる。最初の包含立方体セルを検索すれば、粒子が包含立方体セルを抜け出た面を調べるにより次の包含立方体セルが求められる。② 包含立方体セルにおける粒子座標の写像関数(1)式、速度の補間関数(4)式の係数を計算する。③ 包含立方体セルにおける粒子の始点座標を求める。④ (2)式を数値積分する時の時間刻み幅を設定する。⑤ (2)式を数値積分して、粒子の新しい座標を求める。⑥ 粒子の座標を(1)式より物理空間の座標に変換する。粒子の座標を線で結び、軌跡として表示する。⑦ 粒子が現在の包含立方体セルを抜けるまで⑤、⑥を繰り返す。⑧ 粒子が解析領域を抜けるまで②~⑦を繰り返す。

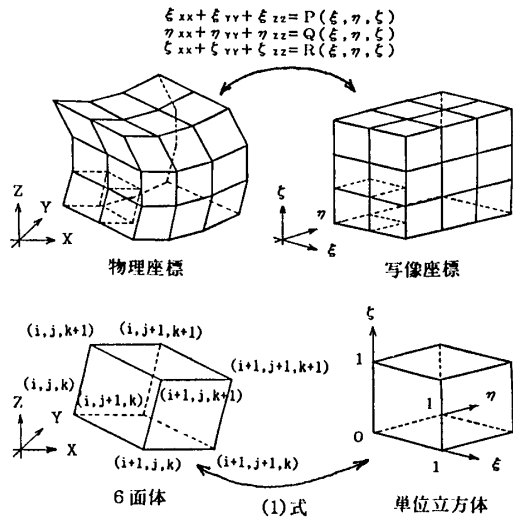


図1 座標系の写像

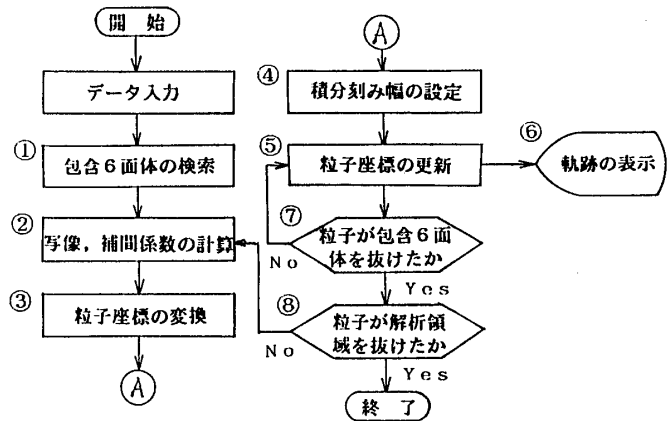


図2 フローチャート

3. 適用例

図3に球体まわりの流れに本手法を適用して流線を表示した例を示す。格子の数は31×31×36であり、本図作成のための軌跡計算の計算時間はスーパーコンピュータSX-2のCP(Control Processor)を用いて約2.0秒であった。

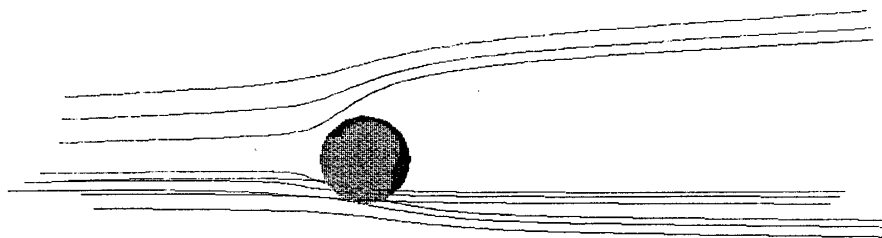


図3 球体まわりの流線表示

4. むすび

粒子の追跡計算を物理座標系の曲線格子で行うのではなく、立方格子に写像して行うことにより、軌跡を簡単に求めることができる。なお、ここで提示した流線作図手法は、非定常流において粒子の軌跡を描くのにも適用できる。

参考文献

1) 例えば、金田、ほか3名：三次元空間に分布するベクトル値データ視覚化のための一手法，情報処理学会論文誌 Vol. 27, No. 6 pp. 567-574 (1986)
 2) Thompson, J.F. et al.: Numerical Grid Generation ; Foundations and applications, North-Holland (1985)