自己組織化マップを用いた 3 次元 PTV の開発*

森 原 崇 文^{*1}, 村 田 滋^{*2}

Development of Three-dimensional PTV with a Self-organizing Map

Takafumi MORIHARA, Shigeru MURATA^{*3}^{*3} Department of Mechanical and System Engineering, Kyoto Institute of Technology,
Matsugasaki, Sakyo-ku, Kyoto-shi, Kyoto, 606-8585 Japan

This paper reports the particle tracking algorithm using a self-organizing map (SOM) developed for 3D-PTV measurement and its performance evaluation in numerical simulations. The algorithm using SOM has high applicability to many engineering problems including the particle identification in a time-series of visualization images with tracer particles. This algorithm has been applied to the two models of the uniform flow in 3D space and the cubic cavity flow obtained by numerical simulation. In the performance tests for the two flow models, the normalized displacement defined for 3D flow is introduced to evaluate the high reliability of the present algorithm in 3D particle tracking. The test results show that the present algorithm can be successfully applied to 3D flow measurements even for higher normalized displacement up to about 3 and the correct ratio of particle tracking in the preset method is influenced mainly by the number of isolated particle images.

Key Words: Flow Measurements, Particle Tracking Velocimetry, Self-organizing Maps, Three-dimensional Velocimetry, Neural Network, Flow Visualization, Velocity Distribution, Three-dimensional Flow, Digital Image Processing

1. 結 言

場計測法である粒子画像流速測定法 (PTV) は従来の点計測手法と相補的な特徴をもつため注目を集め、照明装置と画像装置からなる観測機器の高速化および大容量化を背景に、現在では流体を取り扱う各種分野における主要な実験解析手法の一つに成長を遂げた。PTV では可視化手法から観測機器の性能・画像解析法さらに後処理法に至るまで様々な要素が相互に関連し、その測定性能に複雑に影響するが、観測機器等ハードウェアの改善には経済的負担を伴うことが多く、測定結果の信頼性を低コストで向上させる最も重要なポイントは、ソフトウェアである画像解析法の工夫にあると思われる。

従来から欧米に比べ我が国では PTV の画像解析手法の開発が盛んであり、多くの解析アルゴリズムが世界に先駆け提案され、性能検討されてきた。とりわけ我が国が開発の中心的役割を果たした PTV と略称される粒子追跡法 (Particle Tracking Velocimetry) は、計測速度分布の空間分解能が高いためせん断の大きな流れの解析等に適するなど利点が多い。一方、個々の粒子像の移動を時々刻々追跡するため、一般に PTV のアルゴリズムは複雑な場合が多く、より簡便な PTV アルゴリズムが開発されればその利用が広まるものと期待される。

その中で、ニューラルネットを PIV に適用して測定性能改善を試みたいいくつかの報告⁽¹⁾⁽²⁾がなされている。特に Kohonen⁽³⁾により一般化された自己組織化マップ (Selforganizing Map, SOM) を粒子画像に適用した手法はアルゴリズムがシンプルで簡便な粒子追跡法として成功している。この SOM を PIV に初めて適用した LaBonté⁽⁴⁾は、2次元 PTV に対して SOM モデルを用いたアルゴリズムが Grant ら⁽⁵⁾によるニューラルネットモデルを用いたものより優れていることを示している。さらに、近江⁽⁶⁾は LaBonté のアルゴリズムをベースに同一画面内において粒子の変位量が大きい場合や、時間経過による粒子像の消失や出現がある場合にも追跡可能なアルゴリズムを開発している。

本研究では、近江による自己組織化マップを用いた粒子追跡アルゴリズムを 3次元流れに適用できるよう拡張し、そのアルゴリズムの性能分析を数値シミュレーションにより行なっている。また、時系列粒子画像の特徴を表す相対移動距離を 3次元計測に利用できるよう拡張し、これを用いて本手法の性能評価結果を整理している。

2. 粒子追跡アルゴリズム

自己組織化マップは Kohonen により簡潔に定式化されたため、情報検索や地盤崩壊予測システムなど工学分野を始め、経営戦略・経済分析や健康診断など

* 原稿受付 2003 年 9 月 22 日。

^{*1} 京都工芸繊維大学大学院工芸科学研究科(〒606-8585 京都市左京区松ヶ崎御所海道町)。^{*2} 正員、京都工芸繊維大学工芸学部。
E-mail: murata@kit.jp

様々な分野の問題に応用されてきた。本来、自己組織化とは系外からの影響がない状態で、系そのものがあるメカニズムに従って系内の秩序が自然に形成されることを示すものであり、予備的な知識を要することなく多次元の入力情報を分類できるという特徴を有する。自己組織化マップを用いた本追跡アルゴリズムも PIV 計測の経験が少なくても適切な粒子追跡を実現してくれるという側面を持つ。

今回行った粒子追跡法では、二時刻の画面を用いて粒子像の追跡を行う方法を採用している。粒子追跡に用いる画面として、ある時刻 t での流体内の粒子の状態を第一画面、時刻 t から Δt だけ進んだ時刻 $t + \Delta t$ での粒子の状態を第二画面として用いる。ここでそれぞれの画面より二つのサブネットワークを図 1 のように構成する。これらのサブネットワークは対応する画面内の粒子と同数のニューロンにより構成され、それらは互いに競争させるため相互に連結されている。

今、三次元内の領域について考えているので、それぞれの画面内の粒子は三次元の位置ベクトルを持っており、第一画面内の M 個の粒子の位置ベクトルを $\vec{x}_i (i=1, \dots, M)$ 、第二画面内の N 個の粒子の位置ベクトルを $\vec{y}_j (j=1, \dots, N)$ とする。また、各ニューロンはそれぞれ三次元の荷重ベクトルの情報を持っている。この荷重ベクトルは、初期には対応する粒子の 3 次元座標が与えられ、自己組織化マップ法による反復計算により値が更新される。そして、第一及び第二サブネットワークのニューロンの荷重ベクトルが一致した時、それらのニューロンが表す粒子の対応付けがなされる。すなわち、荷重ベクトルは、各粒子が対応付けられるべき他時刻の粒子に徐々に近づいていく仮想的な状態の粒子位置ベクトルを表している。近江の方法との相違点はこれらベクトルが 3 成分になっている点である。

荷重ベクトルの更新の手順としては、第一サブネットワークの荷重ベクトルを $\vec{w}_i (i=1, \dots, M)$ 、第二サブネットワークの荷重ベクトルを $\vec{w}_j (j=1, \dots, N)$ とすると、

Fig.1 Relation between the particle image and the sub-network

以下になる。最初のステップとして第二サブネットワークの j 番目のニューロンの荷重ベクトル $\vec{w}_j$ を第一サブネットワークに入力し、ユークリッド距離 $\|\vec{w}_j - \vec{v}_i\|$ を最も小さくする荷重ベクトル $\vec{v}_i$ をもつニューロンをそのときの勝者ニューロンとして定める。この勝者ニューロンを c としその荷重ベクトルを $\vec{v}_c$ とすると、第一サブネットワークの i 番目のニューロンは荷重ベクトルの変化率 $\alpha_i (0 \leq \alpha_i < 1)$ を用いて次のような変化量が与えられる。

$$\Delta \vec{v}_i(c(\vec{w}_j)) = \alpha_i (\vec{w}_j - \vec{v}_c)$$

$$\left(\alpha_i = \begin{cases} \alpha & (\|\vec{x}_i - \vec{x}_c\| < r) \\ 0 & (\|\vec{x}_i - \vec{x}_c\| \geq r) \end{cases} \right) \dots\dots\dots (1)$$

式 (1) の $\|\vec{x}_i - \vec{x}_c\|$ は勝者ニューロンと各ニューロンとのユークリッド距離であり、これが閾値 r より小さい場合に荷重変化率 α が値を持ち、式 (1) より与えられる変化量だけ荷重ベクトルが変化する。これは、勝者ニューロンに対応する粒子の位置ベクトル $\vec{x}_c$ を中心とする半径 r の球内に、 i 番目のニューロンが存在するかどうかに対応する。そして、第二サブネットワークの N 個すべての荷重ベクトルが第一サブネットワークに対して入力され、第一サブネットワークの各ニューロンがそのすべての荷重ベクトルの変化を式 (1) に基づいて受けていく。次に第一サブネットワークのすべての荷重ベクトルを第二サブネットワークに入力し、それらの荷重ベクトルを変化させる。各サブネットワークのニューロンの荷重ベクトルは、最終的に次のように変化する。

$$\left. \begin{aligned} \vec{v}_i &\leftarrow \vec{v}_i + \sum_{j=1}^N \Delta \vec{v}_i(c(\vec{w}_j)) : \text{The 1st sub-network} \\ \vec{w}_j &\leftarrow \vec{w}_j + \sum_{i=1}^M \Delta \vec{w}_j(c(\vec{v}_i)) : \text{The 2nd one} \end{aligned} \right\} \dots\dots\dots (2)$$

この処理は、ニューロン同士の距離の平均が設定された微小値よりも小さくなった時点で繰り返しを終了する。そして最近法により第一サブネットワークと第二サブネットワークのニューロンの対応関係を確定し、そのニューロンに対応する粒子が第一画面と第二画面における同一粒子と定める。これにより、粒子の追跡結果が得られる。

ここで、より迅速に対応化を行なうために、荷重変更するニューロンを定める閾値 r を、一連のステップが終了するたびに減衰率 $\beta (0 < \beta < 1)$ を用いて減少させる。それと同時に荷重変化率 α も β を用いて増加させる。

$$\left. \begin{aligned} r &\leftarrow \beta \cdot r \\ \alpha &\leftarrow \alpha/\beta \end{aligned} \right\} \dots\dots\dots (3)$$

さて、2時刻の粒子群には本来対応付くべきでない出沒粒子が含まれる。これらは最終的な対応結果に悪影響を及ぼす原因となるため、荷重ベクトルの更新過程で出沒粒子に対応するニューロンの影響を排除する。その排除方法は次の手順に従う。ここでは、一方のサブネットワーク内のニューロンAが他方のサブネットワークのニューロンXに対応付くべき状況とする。(i) 荷重ベクトルの更新繰り返し数が50回を超えている時、以下の処理を行う。(ii) 1回の更新繰り返しの際に、他方のサブネットワーク内のあるニューロンXを勝者ニューロンに指定したニューロンA, B, C, …を取り上げる。(iii) 取り上げたニューロンの内、勝者ニューロンとの距離が最小でないものを敗者ニューロンB, C, …とする。(iv) この敗者ニューロンB, C, …が先程と同様に3回敗者になった場合、出沒粒子としてサブネットワークから消去する。以上のような排除方法に従い、出沒粒子の影響を消去する。

粒子対応組数を N_0 とすると、一連の対応付けの処理を終了させるための条件として、対応ニューロンとの荷重ベクトルの平均ユークリッド距離 ε を用いる。

$$\varepsilon = \frac{1}{N_0} \sum_{i=1}^{N_0} \|\vec{v}_i - \vec{w}_{j(i)}\| \dots\dots\dots (4)$$

この ε が、閾値 ε_0 以下になったとき繰り返し処理を終了させる。なお、本実験で使用したアルゴリズムの主要なパラメータとして以下のものがあり、その設定値は次のように定めた。

荷重変化率 $\alpha = 0.001, 0.003, 0.005$

減衰率 $\beta = 0.90 \sim 0.99$

初期半径 r_0 = 測定領域により決定

処理終了の閾値 $\varepsilon_0 = 1.0 \times 10^{-6}$

α と β の値の違いは、 α が小さいほど、また β が1に近いほど変化量が小さくなり対応付けの精度は高くなるが、繰り返し数が多くなり処理時間を要する。逆の場合、変化量が大きくなり対応付けの精度が落ちるが、繰り返し数が少なくなる。初期半径 r_0 はさきほど述べた閾値 r の初期値であり、測定領域の全域を含めるぐらいの大きさが望ましいので、測定する領域により変化させた。以上のパラメータ設定を用いて、以下の数値シミュレーションを行なった。

3. シミュレーション結果及び考察

Fig.2 Uniform flow in 3-D space

以上のような手順により作成した粒子追跡を行なうアルゴリズムが、三次元流れにおいてどの程度の性能を持つのか把握するために数値シミュレーションによる性能分析を試みた。以下にその数値シミュレーションの設定とアルゴリズムを適用した結果について述べる。以下のシミュレーションの全てにおいて各粒子の位置座標は、第一画面では一様乱数により決定し、第二画面ではそれに基づき流れの速度分布により決定した。

3.1 二画面内で粒子数が一定の場合

まず、三次元一様流モデルを用いて、二画面内で粒子数が一定の場合における粒子追跡アルゴリズムの性能分析を行なった。この一様流モデルの測定領域の大きさは $256 \times 256 \times 256$ (voxel) とし、粒子の変位量は $(x, y, z) = (18, 18, 19)$ (pixel) とした。また、粒子の数は100, 200, 300, 400, 500, 1000個とし、第一画面に存在した粒子はすべて第二画面に存在するとした。この条件で自己組織化マップ法を用いた三次元粒子追跡アルゴリズムを適用した。その結果、すべての粒子像が一致できており粒子数が一定ならば1000個というたくさんの粒子が領域内に存在していても誤対応を出さずに対応できた。図2は、粒子数が200個のときの対応結果を変位ベクトルで表したものである。

3.2 粒子数密度パラメータを用いた性能分析

次に自己組織化マップ法を利用した粒子追跡アルゴリズムを、粒子数密度パラメータを用いて性能分析を行なった。対象とした流れは3.1節と同じ一様流モデルである。

可視化画像の粒子像の状態を表すものとして相対移動距離 θ があり、ここではそれを参考にして画面内

Fig.3 Convergence history and the change of the number of particles

の粒子の密度や移動距離を表すパラメータを用いることにする。粒子の最大移動距離を $r_{\max}$ 、領域内の粒子数を N_0 、測定領域の大きさを V_0 と表すと、あるひとつの粒子が次時刻までに移動する可能性を持つ領域の大きさは $4\pi r_{\max}^3/3$ 、一様に分布した場合にひとつの粒子が割り当てられる領域の大きさは V_0/N_0 となる。両者の比を取り立方根をとると、次式が得られる。

$$\rho' = \sqrt[3]{\frac{4\pi r_{\max}^3}{3} \frac{N_0}{V_0}} \quad \text{.....(5)}$$

この ρ' は、粒子数密度を基準とした相対的な無次元移動量であるため単位を持たない。また、二次元計測で定義された相対移動距離と同様に、粒子追跡アルゴリズムの適用範囲の目安を $\rho' < 1$ として与えるパラメータである。これを用いて粒子の密度や粒子の変位を変化させた場合の対応結果について検討した。また前節とは違い、より実際の流れ場に近い状態について

Fig.4 Rate of correct pairs for 100~150 particle images and the normalized displacements 1.0, 1.5, 2.0 and 3.0

Fig.5 Cubic cavity flow

調べるため、第一画面と第二画面に出没粒子を存在させて粒子の対応結果を調べた。

出没粒子の処理がどのように行なわれるのかを図3に示す。図3は、二画面に出没粒子が10個ずつ存在する場合において、荷重ベクトルの更新回数と出没粒子の消去、粒子間距離が小さくなっていく様子を表している。横軸に繰り返し数、左縦軸に粒子数、右縦軸には各粒子間の距離をとっており、出没粒子の消去は繰り返し数が50回を超えた時点で始まり、86回で全ての出没粒子が消去された。また、出没粒子が消去されることにより急に粒子間の距離も小さくなっている。

前述した相対移動距離 ρ' を1.0, 1.5, 2.0, 3.0の四種類に変化させ、出没粒子が存在する場合における全粒子数に対して正しく対応できた粒子数の割合(正解率)を調べる。基本粒子の数を100個として、出没粒子の数を10~50個に設定し、二画面の粒子位置は図2の一樣流モデルと同様に決定した。この条件で粒子の変位を変化させる。その結果を図4に示す。図4の横軸は基本粒子+出没粒子の数、縦軸は正解率を表している。図4より出没粒子が30%以上になると誤対応が増加し、出没粒子が50%になると $\rho' > 1$ では粒子の対応付けはほとんどできなかった。しかし、出没粒子が全体の10%までなら $\rho' = 3.0$ の場合でもほぼ100%の正解率を示した。

3.3 Cubic cavity flow を用いた性能分析

今までは、一樣流モデルを用いた様々な粒子数や粒子の変位における粒子の対応結果について述べてきたが、一樣流の代わりに Cubic cavity flow⁽⁷⁾ (以下 OCF と呼ぶ) を用いて粒子追跡アルゴリズムによる対応結果について述べる。図5のような流体を満たした立方体容器があり、その上面のみ可動でその他の面は固定

Fig. 6 Rate of correct pairs for 100~1000 particle images at the time intervals 0.1, 0.5 and 1.0

Fig. 7 Correlation between correct pairs and displacement for 1000 particle images at the time intervals 0.5 and 1.0

壁であるとする。図 5 の立方体容器の上面を黒い矢印の方向に無次元速度 1 で動かすと、容器内の流体は粘性により上面に引きずられて白い矢印のように流動する。この CCF の数値解析結果を用いて、粒子追跡アルゴリズムの性能分析を行なった。ただし、この数値解析結果は流れの Re 数が 1000 のときのものである。また、以下に示す空間座標、変位量は立方体容器の 1 辺の長さを代表長さとした無次元量 ($0 \leq x, y, z \leq 1$) で整理している。

各粒子位置は一様乱数により決定し、各粒子の変位は CCF の速度分布より与えらる。ただし、変位、時間ともに無次元量である。測定領域は一辺の大きさが 1 の立方体容器全体とし、領域内の粒子数は 100, 200, 300, 400, 500, 1000 個とした。粒子の変位量の

変化によるアルゴリズムの性能を調べるため、第一画面と第二画面の時間的な間隔(ここでは Δt)を変化させて粒子の変位量を変化させた。また、測定領域を立方体容器全体としたため出沒粒子は存在しないとする。

これらの条件で CCF に粒子追跡アルゴリズムを適用させた結果を図 6 に示す。図 6 は、横軸は領域内の全粒子数、縦軸はそれに対する正解率を表している。また、 Δt としては 0.1, 0.5, 1.0 の三つの場合について調べた。図 6 では、粒子数の変化による対応結果の変動はあまり見られないが、 Δt が大きくなるに従い対応結果が悪くなっているのが分かる。これは、この CCF は出沒粒子が存在しないので粒子数の変化による誤対応は生じないが、 Δt が変化していくことにより、粒子の変位が大きくなり粒子の対応付けが困難に

(a) X - Y plane (1000 vectors)

(b) Z - X plane (1000 vectors)

Fig. 8 Displacement field determined by true particle pairs

Fig.9 Displacement field determined by the present algorithm

なっているためである。粒子数 100 個の場合、 $\Delta t = 0.1, 0.5, 1.0$ のときの ρ' は、それぞれ 0.6, 2.6, 5.0 となった。次に、粒子の変位量と対応付け精度がどのような関係にあるのかを調べた。この CCF では、 $\Delta t = 1.0$ のとき粒子の変位量は 0.0~1.0 の値をとっている。この変位量を 0.0~0.1 のように幅 0.1 でクラス分けし、その変位区分毎での正解率を求めてみた。図 7 は粒子数が 1000 個、 $\Delta t = 0.5, 1.0$ のときの結果で、横軸が変位の大きさを表しており、縦軸はその変位の大きさでの正解率を表している。図 7 より変位量 0.0~0.2 では 70% 以上が正しく対応付けができていたが、変位量が 0.2 以上になるとほとんど対応できていないことがわかる。つまり変位が測定領域の 20% 以下でないと対応付けができないことになる。

さらに、図 7 の $\Delta t = 1.0$ のときの粒子像のペア（二画面での同一粒子像）をベクトルとして表し、それを (a) X - Y 平面、(b) Z - X 平面にそれぞれ投影したものを図 8、図 9 に示す。図 8 は本来対応すべき粒子像のペアを、図 9 はアルゴリズムにより得られた粒子像のペアをそれぞれ表している。図 8 のベクトル数は 1000、図 9 のベクトル数は 941 であり、そのうち誤対応をしたベクトルの数は 111 である。両者を比較すると、図 8 の (a) の $Y = 1.0$ と $X = 0.5 \sim 1.0$ の位置に存在する大きなベクトルは図 9 では見られない。また (b) でも、図 8 で見られる大きなベクトルは、図 9 では見られない。これは、粒子の変位量が大きいため、出沒粒子を処理するアルゴリズムにより対応すべき粒子像が消去され、より近傍にある粒子像と対応してしまっ

Fig.10 RMS error for particle velocity at the time intervals 1.0 and 0.5

たことが考えられる。このことは図 7 の結果と同様で変位の大きいもののほど対応付けができないことを示している。

変位量と対応精度の関係を定量的に評価するために幅 0.1 の速度区分毎の速度 (単位時間間隔あたりの変位) の RMS 誤差を求める。

$$RMS\ Error = \sqrt{\frac{\sum_{i=1}^N (u'_i - u_i)^2 + (v'_i - v_i)^2 + (w'_i - w_i)^2}{N}} \quad \dots(6)$$

ここで u, v, w は、正解の粒子像ペアの x, y, z 方向の速度成分で、 u', v', w' はアルゴリズムにより決定された粒子像ペアの x, y, z 方向の速度成分である。また、 N は粒子の数である。各速度区分毎の RMS 誤差を式 (6) より求め、それをグラフにしたものが図 10 である。図 10 の (a), (b) はそれぞれ $\Delta t = 1.0, 0.5$ のときの速度の RMS 誤差を表しており、横軸は速度、縦軸は RMS 誤差を表している。また、対応する粒子像が消失してしまった場合は粒子の速度が得られないので RMS 誤差の結果には含めていない。ゆえに図 10 (a) にみられるように速度の大きい範囲では結果が示されていない。

(a) と (b) を比較すると時間間隔 Δt の大きさに関わらず各速度区分での RMS 誤差は、多少の変動があるものの同じ形のグラフを描いている。また、粒子数を 100, 200, 300, 400, 500, 1000 に変化させたが、粒子数の変化による RMS 誤差に違いは見られない。しかし、速度 0.0~0.4 のときの RMS 誤差を比較すると、 $\Delta t = 1.0$ に比べて $\Delta t = 0.5$ の場合は RMS 誤差が最大で 50% 程度まで小さくなっている。これは図 7 の変位が 0.2 以下 (速度が $\Delta t = 1.0$ のとき 0.0~0.2, $\Delta t = 0.5$ のとき 0.0~0.4) の粒子像しか対応付けられなかった結果に一致する。また、RMS 誤差の値が速度に比例しているので、誤対応した場合に選択される粒子像は真の粒子像よりかなり近い位置のものが選択されていることになる。

以上の結果よりこの粒子追跡アルゴリズムは、一様流では出沒粒子が存在しない場合の正解率は 100% であり、出沒粒子が全体の 10% までなら $\rho' = 3.0$ の場合でもほぼ 100% の正解率を示した。また CCF では二つの画面の時間間隔 Δt が小さい場合はほとんど対応でき、 Δt が大きい場合でも 80% 近くの対応付けができていた。更に、粒子の変位量が測定領域の大きさの 20% 以内なら対応付けが可能であることが示された。従って、このアルゴリズムは出沒粒子の数や粒子の変位量などを考慮した上で、三次元流れの場合でも有効に活用できると考えられる。

Table.1 Comparison of processing time among various numbers of particles

Number of Particles	Time (sec)	Iteration number
100	2	110
200	4	110
300	7	134
400	10	144
500	21	172
1000	96	288

また、今回のアルゴリズムの処理時間を表 1 に示す。計算機は Dell Dimension 4100 (CPU: Pentium III 800 MHz, memory: 128 MB) を用いた。表 1 には、 $\Delta t = 0.5$ のときの CCF モデルの粒子数が 100 個~1000 個の場合の処理時間を示している。

4. 結 言

PTV に有用な自己組織化マップ法を用いた粒子追跡アルゴリズムに対して三次元流れの数値シミュレーションによる性能分析を行なった。結果として三次元の場合でも二次元の場合と同様に出沒粒子や粒子の変位が対応結果に大きく影響するということが定量的に示された。今後は出沒粒子を処理するアルゴリズムを改善し、この SOM アルゴリズムを出沒粒子が多い場合でも粒子の追跡を可能とするようなより有益なものにすることが望まれる。

なお、自己組織化マップ法について大産大、近江和生先生に情報提供を頂いた。ここに謝辞を述べる。また、本研究の一部は平成 14 年度科学研究費補助金 (基盤(c)(2), 12650166) により実施された。記して謝意を表する。

文 献

- (1) 村田 滋・山本貴幸, 日本機械学会論文集, 63-609, B (1997), 1628-1634.
- (2) Grant, L., Pan, X., Experiments in Fluids, Vol.19 (1995), 159-166
- (3) Kohonen, T., Selforganization and Associative Memory, Second Edition, (1988), Springer-Verlag, Berlin.
- (4) LaBonté, G., Experiments in Fluids, Vol.26, (1999), 340-346
- (5) 近江和生・吉田直弘, 可視化情報, Vol.21, Suppl.1, (2001), 125-128
- (6) 可視化情報学会編, PIV ハンドブック, (2002), 67-68, 森北出版
- (7) Murata, S. et al., Proc. of 2nd Japan-Soviet union joint symposium on computational fluid dynamics, (1990), 71-78