

動的モード分解を用いた動力学的特性に基づくクラスタリング Clustering based on dynamical properties using Dynamic Mode Decomposition

竹原 颯汰¹⁾ 河原 吉伸²⁾
Sota Takehara Yoshinobu Kawahara

1 はじめに

一般的なものづくりにおける設計分野では、美しく、高性能でエネルギー効率の高いデザインが求められる。デザインの完成には、設計、シミュレーション、評価といったプロセスを何度も繰り返すことが求められる。例えば自動車の設計者は、設計パラメータを調整することによって複数のデザインを用意し、空力抵抗係数である Cd 値を小さくするために繰り返し設計を行う。しかし、Cd 値の推定には、物理的な風洞実験や計算量の多いシミュレーションが必要で、高コストで時間のかかる作業であるという問題が存在し、これがサイクルにおけるボトルネックとなり得る。

この問題を解消するためには、機械学習を用いてプロセスの自動化、効率化することが挙げられるが、シミュレーションの動力学的特性の比較という技術的な課題が内在している。そのため、本研究では、シミュレーション・データの動力学的特性を比較することを目的としたアルゴリズムの構築を行った。

近年、動的モード分解 (Dynamic Mode Decomposition, DMD) は多次元時系列データにおける主要な動力学的特性に関する情報を取り出す手法として注目されている。動的モード分解は、流体解析の分野において、初めて提案された手法である。同様の目的がある主成分分析 (Principal Component Analysis, PCA) とは異なり、空間的、時間的構造を同時に取得可能ということが動的モード分解の特徴である。つまり、ダイナミクス中の各モードの減衰や周期的な振動を知ることができ、全体的なコヒーレント構造を知ることができる。さらに、ダイナミクス自体の比較よりも計算量が大幅に小さくなるといった利点がある。本研究における提案手法では、動的モード分解を用いることでダイナミクスを解析する。DMD 出力を介して類似度を計算し、クラスタリングを行うことで、その精度について示す。

以上を踏まえ、本研究ではオープンソース CFD ソフトウェアである OpenFOAM を用いて、円柱、角柱、三角柱周りのシミュレーションを行い、提案手法による解析手法を適用した。

2 関連研究

2.1 動的モード分解

動的モード分解 [1] は時間的に成長、減衰、振動するダイナミクスや空間的なコヒーレント構造という観点からダイナミクスを近似することができる手法である。以下、説明する動的モード分解は、特に厳密動的モード分解と呼ばれる。このコヒーレント構造を DMD モードと呼び、各 DMD モードは一つの固有値で定義される対応する時間ダイナミクスを持つ。初めに $\mathbf{x}(t_0), \mathbf{x}(t_1), \dots, \mathbf{x}(t_m)$ の $m+1$ 時点から成る時系列デー

タを考える。ここで、 $\mathbf{x}(t_i) \in \mathbb{R}^n$ である。以下のように行列を定義する。

$$\mathbf{X} = \begin{bmatrix} | & | & \dots & | \\ \mathbf{x}(t_0) & \mathbf{x}(t_1) & \dots & \mathbf{x}(t_{m-1}) \\ | & | & \dots & | \end{bmatrix} \quad (1)$$

$$\mathbf{X}' = \begin{bmatrix} | & | & \dots & | \\ \mathbf{x}(t_1) & \mathbf{x}(t_2) & \dots & \mathbf{x}(t_m) \\ | & | & \dots & | \end{bmatrix} \quad (2)$$

DMD アルゴリズムは二つの時間発展する行列に関連する行列 $\mathbf{A} \in \mathbb{C}^{n \times n}$ を固有値分解する。 \mathbf{A} は次のように仮定することができる。

$$\mathbf{X}' \approx \mathbf{A}\mathbf{X} \quad (3)$$

また、これを満たす \mathbf{A} は数学的に以下のようにも表すことができる。

$$\mathbf{A} = \arg \min_{\mathbf{M} \in \mathbb{C}^{n \times n}} \|\mathbf{X}' - \mathbf{M}\mathbf{X}\|_F = \mathbf{X}'\mathbf{X}^\dagger \quad (4)$$

$\|\cdot\|_F$ はフロベニウスノルム、 \mathbf{X}^\dagger は \mathbf{X} の擬似逆行列である。この \mathbf{A} について固有値分解を適用することが目標だが、 \mathbf{A} のサイズは $n \times n$ であるため、擬似逆行列を求めるための特異値分解の計算量は、 n が増加するにつれて、 $O(n^2)$ で大きくなってしまふ。そこで、DMD アルゴリズムでは \mathbf{A} を直接分解するのではなく、 \mathbf{X} を特異値分解して得られた特異ベクトルへ射影した $\tilde{\mathbf{A}}$ を考える。 $\tilde{\mathbf{A}}$ のサイズは最大でも $m \times m$ となる。

次に具体的な DMD アルゴリズムでの計算手順について述べる。

1. \mathbf{X} の特異値分解を実行し、低位の切り捨てを行う。

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^* \quad (5)$$

ただし、 $\mathbf{U} \in \mathbb{C}^{n \times r}$ 、 $\mathbf{\Sigma} \in \mathbb{C}^{r \times r}$ 、 $\mathbf{V} \in \mathbb{C}^{m \times r}$ であり、近似するランク r の選び方が重要となる。また、 \mathbf{U} には、 $\mathbf{U}^*\mathbf{U} = \mathbf{I}$ という特徴がある。

2. (4)、(5) と \mathbf{X} の擬似逆行列を合わせて \mathbf{A} を表す。

$$\mathbf{A} = \mathbf{X}'\mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}^* \quad (6)$$

\mathbf{A} を \mathbf{U} に射影すると、

$$\tilde{\mathbf{A}} = \mathbf{U}^*\mathbf{A}\mathbf{U} = \mathbf{U}^*\mathbf{X}'\mathbf{V}\mathbf{\Sigma}^{-1} \quad (7)$$

ここで最も大切なことは次元削減した $\tilde{\mathbf{A}}$ と \mathbf{A} は同じ固有値を持つということである。

3. $\tilde{\mathbf{A}}$ を固有値分解する。

$$\tilde{\mathbf{A}}\mathbf{W} = \mathbf{W}\mathbf{\Lambda} \quad (8)$$

1) 大阪大学大学院情報科学研究科

2) 大阪大学大学院情報科学研究科、理化学研究所革新知能統合研究センター

Λ は固有値 Λ_i を対角成分に持つ行列であり、DMD 固有値と呼ばれる。また、 Λ は行列 A の固有値に相当する。 W は A の固有ベクトルである w_i を列成分に持つ。

4. W を用いて A の固有ベクトルである DMD モード Φ を再構成する。

$$\Phi = X'V\Sigma^{-1}W \quad (9)$$

確認のため、 $A\Phi$ を計算すると、

$$\begin{aligned} A\Phi &= (X'V\Sigma^{-1}U^*)(X'V\Sigma^{-1}W) \\ &= X'V\Sigma^{-1}\tilde{A}W \\ &= X'V\Sigma^{-1}W\Lambda \\ &= \Phi\Lambda \end{aligned}$$

これにより全ての時刻での状態を記述することができる。時間間隔を $\Delta t = t_1 - t_0$ とおき、 $\omega_i = \log(\Lambda_i)/\Delta t$ とおくと、任意の時間 t での状態 $x(t)$ は、

$$x(t) = \sum_{j=1}^r \Phi_j e^{\omega_j t} b_j = \Phi \exp(\Omega t) b \quad (10)$$

と書くことができる。ここで、 Ω は ω_j を対角成分に持つ対角行列である。また、 b は初期状態を表し、 $b = \Phi^\dagger x(t_0)$ で計算できる。

2.2 グラスマンカーネル

グラスマンカーネル [2][3] は機械学習における課題としてしばしば見られる、対象の特微量の数が異なる場合に用いられる手法である。 \mathbb{R}^D 上の m 次元線形部分空間の集合は $G = (m, D)$ と記述され、グラスマン多様体と呼ばれる。

特微量の集合全体は、 \mathbb{R}^D の対応する部分空間 $\text{span}(Y_i) = \text{span}\{u_1, \dots, u_p\}$ と表せる。ここで、 $\text{span}(Y_i)$ は $Y_i = [u_1 \dots u_p]$ の列ベクトルで表される部分空間であり、各列ベクトルは正規直交する。部分空間 $\text{span}(Y_i)$, $\text{span}(Y_j)$ はグラスマン多様体上 $G = (m, D)$ の 2 点である。グラスマン多様体上の 2 点の距離はグラスマン距離と呼ばれ、正準角に基づいた類似度である。

二つの部分空間 $\text{span}(Y_1)$, $\text{span}(Y_2)$ の正準角 $0 \leq \theta_1 \leq \dots \leq \theta_m \leq \pi/2$ は以下のように再帰的に定義される。

$$\cos \theta_k = \max_{u_k \in \text{span}(Y_1)} \max_{v_k \in \text{span}(Y_2)} u_k' v_k \quad (11)$$

ただし、次のような制約条件を持つ。

$$\begin{aligned} u_k' u_k &= 1, v_k' v_k = 1 \\ u_k' u_i &= 0, v_k' v_i = 0, (i = 1, \dots, k-1) \end{aligned} \quad (12)$$

正準角は $Y_1' Y_2$ の特異値分解によって得られる。

$$Y_1' Y_2 = U(\cos \Theta) V' \quad (13)$$

ここで、 $U = [u_1 \dots u_m]$, $V = [v_1 \dots v_m]$, $\cos \Theta$ は対角行列であり、 $\cos \Theta = \text{diag}\{\cos \theta_1, \dots, \cos \theta_m\}$ である。以上の定義によりグラスマン距離は複数定義されるが、代表的なものに projection metric が挙げられる。本研究では projection metric に基づいたカーネルである projection kernel を用いる。

- projection metric (正準角の正弦の 2 ノルム)

$$d_p(Y_1, Y_2) = \left(\sum_{i=1}^m \sin^2 \theta_i \right)^{\frac{1}{2}} = \left(m - \sum_{i=1}^m \cos^2 \theta_i \right)^{\frac{1}{2}} \quad (14)$$

(14) に基づき、projection kernel は次のように計算される。

$$k(Y_1, Y_2) = |Y_1' Y_2|_F^2 = \text{tr}[(Y_1 Y_1')(Y_2 Y_2')] \quad (15)$$

projection kernel を DMD 特微量 $\hat{\Phi}_i$, $\hat{\Phi}_j$ に適用する場合、 $k(\hat{\Phi}_i, \hat{\Phi}_j)$ として、グラム行列を生成して用いる。

3 提案手法

本節では、提案手法のアルゴリズムについて説明する。シミュレーションにより得られた速度データに動的モード分解を実行する。そして、得られた DMD モードにグラスマンカーネルを適用することで、DMD モード間の類似度を測り、グラム行列を構成する。最後にグラム行列を用いてカーネル k-means 法 [4][5] により、クラスタリングを行う。以下、この手法について詳しく説明する。

本手法では時系列データごとに動的モード分解を実行する。得られたセル数 n , タイムステップ数 m の p 個のデータについて、 $d_i(t_0), d_i(t_1), \dots, d_i(t_{m-1})$ のように定義する。ただし、添え字 i はデータ番号を表しており、 $i \in \{1, \dots, p\}$, ベクトル $d_i(t_j)$ は時間 $j \in \{0, \dots, m-1\}$ におけるデータであり、 $d_i(t_j) \in \mathbb{R}^n$ である。これらのベクトルを 1 つの行列 $D_i = [d_i(t_0) \dots d_i(t_{m-1})]$ とすることで、 D_i に対して動的モード分解を行い、 D_i を近似できるような DMD モードを取得する。ただし、行列 $D_i \in \mathbb{C}^{n \times m}$ である。次に、行列 D_i を二つに分割し次のように定義する。

$$X_i = [d_i(t_0) \dots d_i(t_{m-2})] \quad (16)$$

$$Y_i = [d_i(t_1) \dots d_i(t_{m-1})] \quad (17)$$

(16), (17) の定義により次のような式を満たす行列 A_i を求める。

$$A_i = \arg \min_{M_i \in \mathbb{C}^{n \times n}} \|Y_i - M_i X_i\|_F = Y_i X_i^\dagger \quad (18)$$

ここで、 X_i を特異値分解することによりダイナミクスへの影響が少ない波を切り捨てる。特異値が大きいほど、ダイナミクスへの影響が強く、任意の値 r によって選択するモードの数を決定する。 X_i の特異値分解は次のように計算される。

$$X_i = U_i \Sigma_i V_i' \quad (19)$$

ただし、 $U_i \in \mathbb{C}^{n \times r}$, $\Sigma_i \in \mathbb{C}^{r \times r}$, $V_i \in \mathbb{C}^{(m-1) \times r}$ である。

次に、行列 A_i を U_i に射影する。射影した行列を \tilde{A}_i と定義する。

$$\tilde{A}_i = U_i^* A_i U_i = U_i^* Y_i V_i \Sigma_i^{-1} \quad (20)$$

ここで、切り捨てを行ったことにより、 $\tilde{A}_i \in \mathbb{C}^{r \times r}$ となっている。そして、 \tilde{A}_i を固有値分解していくことで、

DMD モードを求める。\$\tilde{A}_i\$ の固有値分解は次の式で与えられる。

$$\tilde{A}_i W_i = W_i \Lambda_i \quad (21)$$

\$\tilde{A}_i\$ の固有ベクトルを列成分に持つ行列 \$W_i\$ を用いることで、DMD モード \$\Phi_i \in \mathbb{C}^{n \times r}\$ を計算することができる。

$$\Phi_i = Y_i V_i \Sigma_i^{-1} W_i \quad (22)$$

これらの操作により求められた DMD モード \$\Phi_i\$ が、\$i\$ 番目のデータに対するダイナミクスの主要な情報を保持している。

次に、DMD モード間の類似度を計算するため、2.2 節で述べた projection kernel をカーネルとし採用する。DMD モードはデータに一对一対応しているため、与えられたデータ数が \$p\$ 個のとき、DMD モードは \$p\$ 個存在している。データ \$i\$ に対して計算した DMD モードと、データ \$j\$ に対して計算した DMD モードの類似度は projection kernel を用いて次のような式で計算される。

$$k(\Phi_i, \Phi_j) = |\Phi_i' \Phi_j|_F^2 = \text{tr}[(\Phi_i \Phi_i')(\Phi_j \Phi_j')]$$

全 \$p\$ 個の時系列データに対して類似度を求め、グラム行列 \$\mathbf{G} \in \mathbb{C}^{p \times p}\$ にすると次のようになる。

$$\mathbf{G} = \begin{pmatrix} k(\Phi_1, \Phi_1) & \cdots & k(\Phi_1, \Phi_j) & \cdots & k(\Phi_1, \Phi_p) \\ \vdots & & \vdots & & \vdots \\ k(\Phi_i, \Phi_1) & \cdots & k(\Phi_i, \Phi_j) & \cdots & k(\Phi_i, \Phi_p) \\ \vdots & & \vdots & & \vdots \\ k(\Phi_p, \Phi_1) & \cdots & k(\Phi_p, \Phi_j) & \cdots & k(\Phi_p, \Phi_p) \end{pmatrix} \quad (23)$$

作成されたグラム行列を用い、カーネル k-means 法を行うことで DMD モードをクラスタリングする。

カーネル k-means 法ではクラスタ数 \$k\$ の時、\$c\$ 番目のクラスタを \$\pi_c\$ とすると、次の目的関数を最小化するようなクラスタを求める。

$$\mathcal{D}(\{\pi_c\}_{c=1}^k) = \sum_{c=1}^k \sum_{\Phi_i \in \pi_c} k(\Phi_i, \Phi_i) - \frac{2 \sum_{\Phi_j \in \pi_c} k(\Phi_i, \Phi_j)}{|\pi_c|} + \frac{\sum_{\Phi_j, \Phi_l \in \pi_c} k(\Phi_j, \Phi_l)}{|\pi_c|^2} \quad (24)$$

つまり、(23) で作成したグラム行列のみを用いて計算することができる。

これにより、時系列データに対して、解析を行い、その結果を可視化することが可能である。また、クラスタリング結果のマッピングの際、高次元のため、プロットできない場合がある。そのような場合は、高次元データを可視化するための次元削減手法である t-SNE を用いることで解消するのだが、詳細は 4.2 節で述べる。

4 研究方法

4.1 実験方法

実験はオープンソース CFD ソフトウェアである OpenFOAM を用いて行った。使用した幾何形状は \$0.06\text{m} \times 0.13\text{m}\$ であり、円柱の直径 \$D\$ は \$2.0 \times 10^{-3}\text{m}\$、角柱の一辺は \$2.0 \times 10^{-3}\text{m}\$、三角柱の一辺は \$\sqrt{3} \times 10^{-3}\text{m}\$ と

した。総セル数は円柱、角柱、三角柱において、それぞれ 41060, 43336, 43986 となっている。

本研究では、3 種類の図形に対して \$x\$ 軸の正の方向に一樣流速 \$u_\infty\$ を 20 パターンずつ与えた。\$y\$ 軸方向の流速 \$u_y\$ と圧力 \$p\$ については、初期条件として 0 を与えている。また、動粘性係数は \$1.0 \times 10^{-5}\text{m}^2/\text{s}\$ である。自由流れ境界条件であることから流体が外部に対して制約を受けず、自由に流れ、スリップなし条件であることから流体は固体表面で滑らない。

本実験におけるシミュレーションでは、時間幅を 0s から 0.80s まで \$1.0 \times 10^{-3}\text{s}\$ ごとのステップで設定しており、全部で 800 ステップ存在している。そして、これら 20 パターンの流速ごとに全てのセル、全ての時間ステップでの速度データを取得する。

4.2 解析方法

3 種類の図形それぞれに対して 20 パターンごとに得られた速度データを \$\mathbf{d}_i(t_0), \mathbf{d}_i(t_1), \dots, \mathbf{d}_i(t_{m-1})\$ とする。これらのベクトルを用いて、データを 1 つの行列 \$\mathbf{D}_i = [\mathbf{d}_i(t_0) \dots \mathbf{d}_i(t_{m-1})]\$ とした。ここで、\$\mathbf{D}_i \in \mathbb{C}^{n \times m}\$ であり、\$\mathbf{d}_i(t_j)\$ はパターン \$i\$ における時刻 \$t_j\$ での速度ベクトル、\$n\$ は総セル数、\$m\$ はタイムステップ数を表す。ただし、\$i \in \{1, \dots, p\}\$、\$j \in \{0, \dots, m-1\}\$ である。ここで、1 つのセルあたり、3 次元方向に速度をもつため、データの列ベクトルはセル数の 3 倍となる。本研究の場合、\$\mathbf{D}_i\$ の行列サイズは円柱、角柱、三角柱とそれぞれ \$123180 \times 800\$、\$130008 \times 800\$、\$131958 \times 800\$ となる。加えて 2 つの行列を次のように定義し、2.1 節での動的モード分解を適用する。

$$\mathbf{X}_i = [\mathbf{d}_i(t_0) \dots \mathbf{d}_i(t_{m-2})] \quad (25)$$

$$\mathbf{Y}_i = [\mathbf{d}_i(t_1) \dots \mathbf{d}_i(t_{m-1})] \quad (26)$$

これによって、3 種類の図形ごとに DMD モード \$\Phi_1, \Phi_2, \dots, \Phi_{20}\$ が得られる。これら 60 個の行列のうち、\$\Phi_1, \dots, \Phi_{20}\$ を円柱、\$\Phi_{21}, \dots, \Phi_{40}\$ を角柱、\$\Phi_{41}, \dots, \Phi_{60}\$ を三角柱の DMD モードとし、2.2 節の projection kernel を用いて、パターンごとの類似度の情報であるグラム行列 \$\mathbf{G} \in \mathbb{C}^{60 \times 60}\$ を作成する。

ここで、グラスマンカーネルでは、次元数の異なる部分空間同士の距離を測ることはできない。そのため、得られた DMD モードの次元数を揃える必要がある。次元の揃え方には、部分空間を相手の部分空間のグラスマン多様体に射影して得られる集合との距離を計算する手法 [6] が挙げられるが、本研究では簡単のため、低次元の DMD モードを 0 でパディングすることにより、高次元の DMD モードと次元を揃える。

$$\mathbf{G} = \begin{pmatrix} k(\Phi_1, \Phi_1) & \cdots & k(\Phi_1, \Phi_j) & \cdots & k(\Phi_1, \Phi_{60}) \\ \vdots & & \vdots & & \vdots \\ k(\Phi_i, \Phi_1) & \cdots & k(\Phi_i, \Phi_j) & \cdots & k(\Phi_i, \Phi_{60}) \\ \vdots & & \vdots & & \vdots \\ k(\Phi_{60}, \Phi_1) & \cdots & k(\Phi_{60}, \Phi_j) & \cdots & k(\Phi_{60}, \Phi_{60}) \end{pmatrix} \quad (27)$$

グラム行列 \$\mathbf{G}\$ の \$(i, j)\$ 成分はパターン \$i\$ とパターン \$j\$ の DMD モードのカーネル値である。すなわち、\$\mathbf{G}\$ は対称行列である。

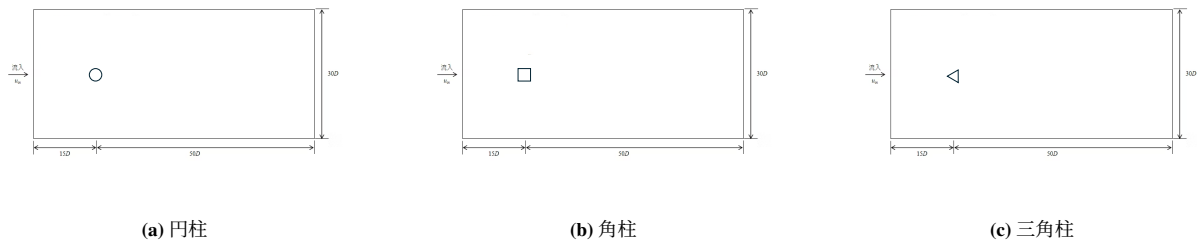


図 1: 幾何モデル

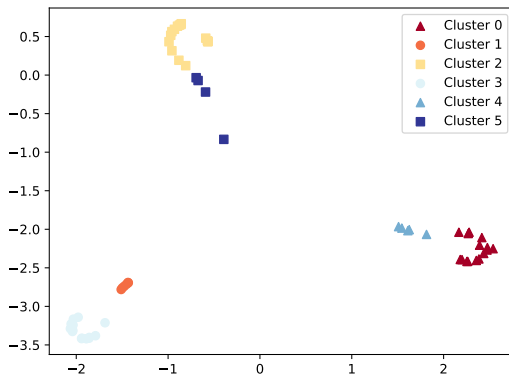


図 2: クラスタリング結果

最後にグラム行列を用いてカーネル k-means 法を行う。カーネル k-means 法では、クラスタ数を 6 とし、図形ごとに層流とカルマン渦とのクラスタリングを行った。

4.2.1 データの可視化

クラスタリング結果の図示にはグラム行列を用いるが、 G は 60×60 とそのままでは次元が大きいため、t-SNE [7] を用い、データを 2 次元にすることで可視化を行った。

t-SNE とは、高次元データを 2 次元又は 3 次元に変換して可視化するための次元削減アルゴリズムである。高次元での距離分布が低次元での距離分布にもできるだけ合致するように変換するという特徴がある。高次元空間上の類似度と低次元空間上の類似度をモデル化し、分布間の距離を最小化するようにパラメータを最適化する。これにより、非線形な構造を保持しやすく、高次元データの非線形なクラスタリングや構造を視覚的に理解しやすいことや局所的なクラスタが保持されやすいといった利点がある。

本研究では、t-SNE の処理を経て、グラム行列を 60×2 の大きさに圧縮する。そして、圧縮された行列の i 行目をパターン i のプロットとして採用した。

5 結果と考察

5.1 クラスタリング結果

流速 0.05m/s - 1.00m/s の範囲における 3 種類の図形ごとの 20 個のシミュレーション結果のクラスタリングを示す。

図 2 ではプロットの形状が対応する幾何モデルを表している。また、Cluster0, Cluster2, Cluster3 はそれぞれカルマン渦が生じているクラスタである。カルマン渦が生

じていない層流のクラスタはどの形状においても図の中央に配置されていることから、形状に関係なく層流のダイナミクスが似ていることを反映している。

また、クラスタ境界部分のシミュレーションの様子について図 3a-3f に示す。速度は画像中のカラーマップに従う。

図 3a, 図 3f, 図 3e では若干の波の揺れを観測することができるが、振幅が小さく、層流の影響が大部分を占めているため、これらのシミュレーションは層流とみなしている。対して、図 3b, 図 3d, 図 3f では明らかにカルマン渦の特徴が見て取れる。以上より、適切にクラスタリングができていると言える。

このような渦による揺れを区別するためには適切なモードの選択が必要である。そこで、5.2 節では、モードの選択方法についての考察を述べる。

5.2 DMD モードの選択

層流とカルマン渦のクラスタリングにおいて、DMD による次元削減の際に選択するモードの数が重要であった。以下、円柱での例を挙げることにより、選択する DMD モードの数について考察をする。選択するモードが少なすぎる場合、流れ場の複雑なダイナミクスや構造についての情報が不足する可能性が生じる。一方、選択するモードが多すぎる場合、一部のモードが冗長であり、流れ場の本質的でない情報が含まれる可能性が生じる。得られたシミュレーションデータに対して DMD を行った結果を以下に示す。層流データとして流速 0.20m/s 、カルマン渦データとして流速 0.60m/s を例に挙げる。

また、図 4 は特異値の累積寄与率を表している。この図より、主要なモードの数は 10 個程度であったことから例では DMD モードを 10 個取り出した場合の図を示す。

DMD では、DMD 固有値の値によって対応する DMD モードの振る舞いを知ることができる。固有値が単位円内であればモードは減衰し、単位円外であればモードは成長する。0 以外の虚部を持つ場合、そのモードには振動がある。図 5 における mode3 に対応する固有値を見てみると単位円の外側にあり、0 以外の虚部を持つ。つまり、mode3 は振動しながら成長する波であるということが分かる。図 6b と比較すると、波が振動しながら成長していることが見て取れる。DMD の固有値では、偏角が定数倍になっている固有値が存在することがある。それらは同一のモードを示しているため、複数考慮する必要はない。また、mode1 と mode2, mode3 と mode4, mode7 と mode8, mode9 と mode10 のように対になっているものが見つかっている。そのため、選択する

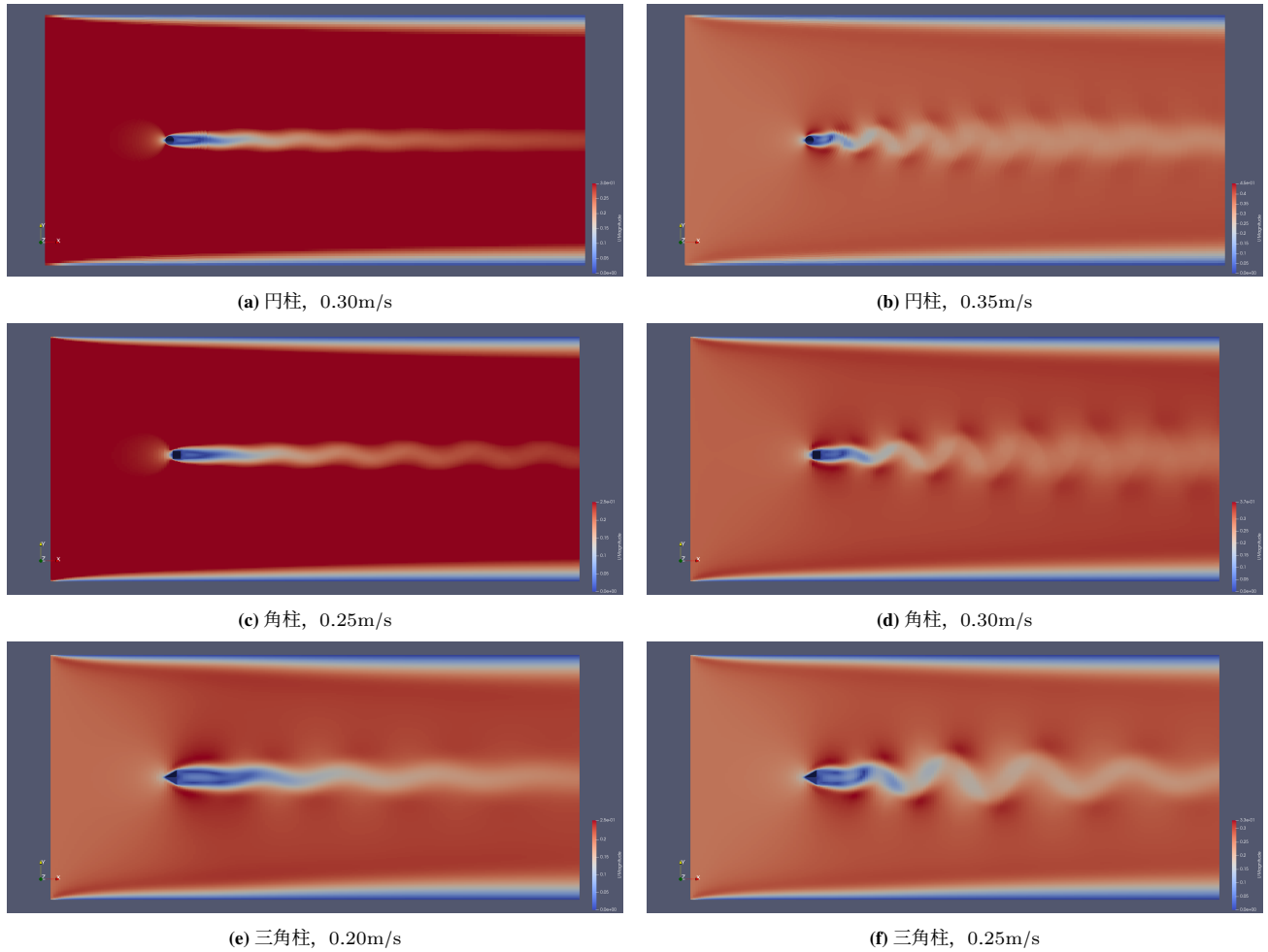


図 3: 0.80s 地点におけるシミュレーション

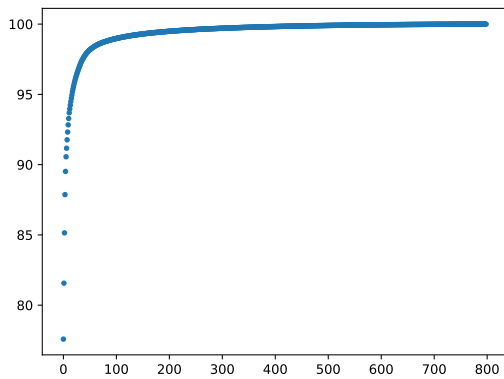


図 4: 0.60m/s の特異値の累積寄与率

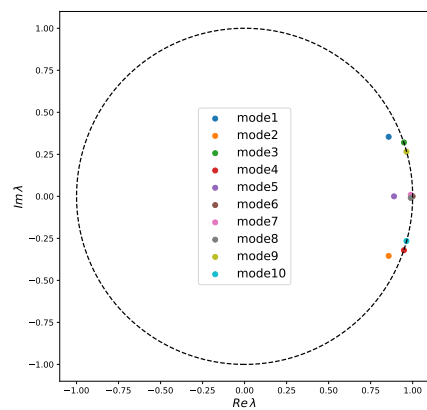


図 5: 0.60m/s の DMD 固有値分布

モードの数も偶数個であることが望ましい。以上の考察と図 6a, 6b より mode3-mode4 がカルマン渦を示していることから, mode3-4 を含め, かつ元のダイナミクスの情報を失わないようモードを選択し, クラスタリングを行う必要がある。これらを踏まえ, 本研究では選択する DMD モードの数を 6 とした。

6 結論

本稿では, 動的モード分解を用いたシミュレーション・データの動力学的特性の比較とクラスタリングについての手法を提案した。提案手法の有用性を検証するために, 円柱, 角柱, 三角柱の 3 つの図形後方での流れ場について解析を行った。結果として, 3 つの図形とそれぞれについて層流とカルマン渦とをクラスタリングでき

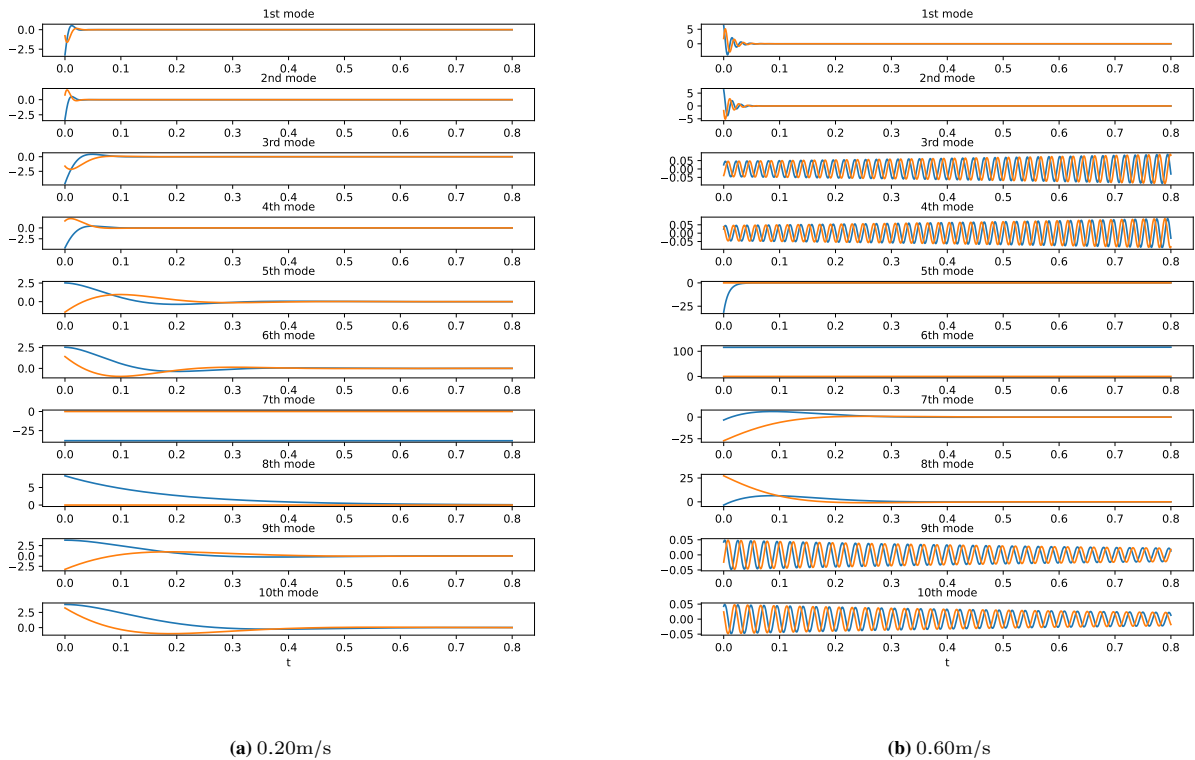


図 6: DMD モードに対応する波形

ていることが確認できた。また、動的モード分解はダイナミクスの主要なモードを取り出す手法ではあるが、モードの取り出し方について結果とともに考察を行った。取り出すモードが少なすぎる場合、流れ場の複雑なダイナミクスや構造についての情報が不足する可能性が生じる。一方、選択するモードが多すぎる場合、一部のモードが冗長であり、流れ場の本質的でない情報が含まれる可能性が生じることについて議論を行った。動的モード分解により得られた特性は、自動車や航空機、建造物などさまざまな分野でも重要なため、提案手法による動力学的特性の比較が有効である。

しかし、本研究では DMD モードの次元を揃えるために 0 でパディングしている部分のように、簡略化している部分がある。今後の展望としては、簡略化している部分に対して手法 [6] のような次元数を揃える手法を用いること、厳密動的モード分解以外の動的モード分解手法を適用するといった改善を予定している。

参考文献

- [1] J. Nathan Kutz, Steven L. Brunton, Bingni W. Brunton, and Joshua L. Proctor, *Dynamic mode decomposition: data-driven modeling of complex systems*, SIAM, 2016.
- [2] Yoshiyuki Shiraishi, Yoshinobu Kawahara, Okito Yamashita, Ryohei Fukuma, Shota Yamamoto, Youichi Saitoh, Haruhiko Kishima, and Takufumi Yanagisawa, “Neural decoding of electrocorticographic signals using dynamic mode decomposition,” *Journal of Neural Engineering*, vol. 17, no. 3, 2020.
- [3] J. Hamm and D. D. Lee, “Grassmann discriminant analysis: a unifying view on subspace-based learning,” *ICML '08: Proceedings of the 25th international conference on Machine learning*, pp. 376–383, 2008.
- [4] Inderjit S. Dhillon, Yuqiang Guan, and Brian Kulis, “Kernel k-means: spectral clustering and normalized cuts,” in *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004.
- [5] Inderjit S. Dhillon, Yuqiang Guan, and Brian Kulis, “Weighted Graph Cuts without Eigenvectors A Multilevel Approach,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 11, pp. 1944–1957, 2007.
- [6] Ke Ye and Lek-Heng Lim, “Schubert varieties and distances between subspaces of different dimensions,” arXiv:1407.0900 [math.NA], July 2014.
- [7] L. v. d. Maaten and G. Hinton, “Visualizing Data using t-SNE,” *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.