

VIII Российская конференция
ВЫЧИСЛИТЕЛЬНЫЙ ЭКСПЕРИМЕНТ
В АЭРОАКУСТИКЕ И АЭРОДИНАМИКЕ



20—25 сентября 2021 г., ГЕЛЕНДЖИК



Идентификация вихрей и анализ мод неустойчивости, полученных из данных нестационарного расчёта LES

И.С. Босняков, Н.А. Ключев

Содержание

- Описание рабочего процесса (workflow)
- Обзор критериев вихревого течения
- Декомпозиция по динамическим модам DMD

Введение

- Объём расчётных сеток растёт, сложность течений увеличивается
- Актуален вопрос идентификации турбулентных структур внутри потока газа
 - Например, в отрыве за механизацией крыла
 - Предмет рабочего пакета в проекте HiFiTurb (Горизонт 2020)
 - Изучение физического процесса с целью создания упрощенных или дополненных моделей
- Ручная обработка результатов затруднена
- В качестве модельного исследуемого течения используется вихревой след за самолётом
- Автоматизация достигается за счёт написания скриптов
 - Скрипты подготавливаются на **Python** для среды обработки и визуализации **ParaView**

Работа выполнена при финансовой поддержке РФФИ
(проект №19-01-00163)

Описание рабочего процесса

Workflow

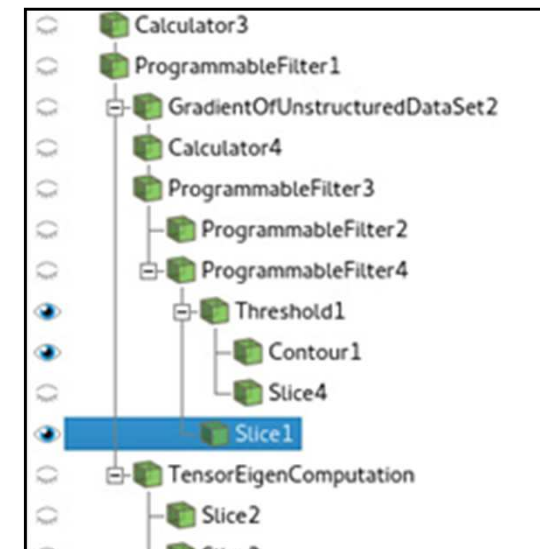
Численное исследование

- Подготовка нестационарного расчёта задачи вихревого следа
- Расчётная область + Уравнения + Начальные-граничные условия
- Расчётная сетка (7.7 млн узлов) + Солвер LES DG (K2, 10 DOF/на ячейку)
- Результаты нестационарного расчёта записываются как серия полей в заданные моменты времени
 - 4 Gb – один файл результатов в формате солвера
- Преобразование в универсальный формат (cgns)
 - Сетка мельчится для визуализации решения высокого порядка обычной программой
 - Функции газодинамических величин записываются в центрах новых ячеек
 - cgns с результатами занимает 17 Gb на каждый момент времени



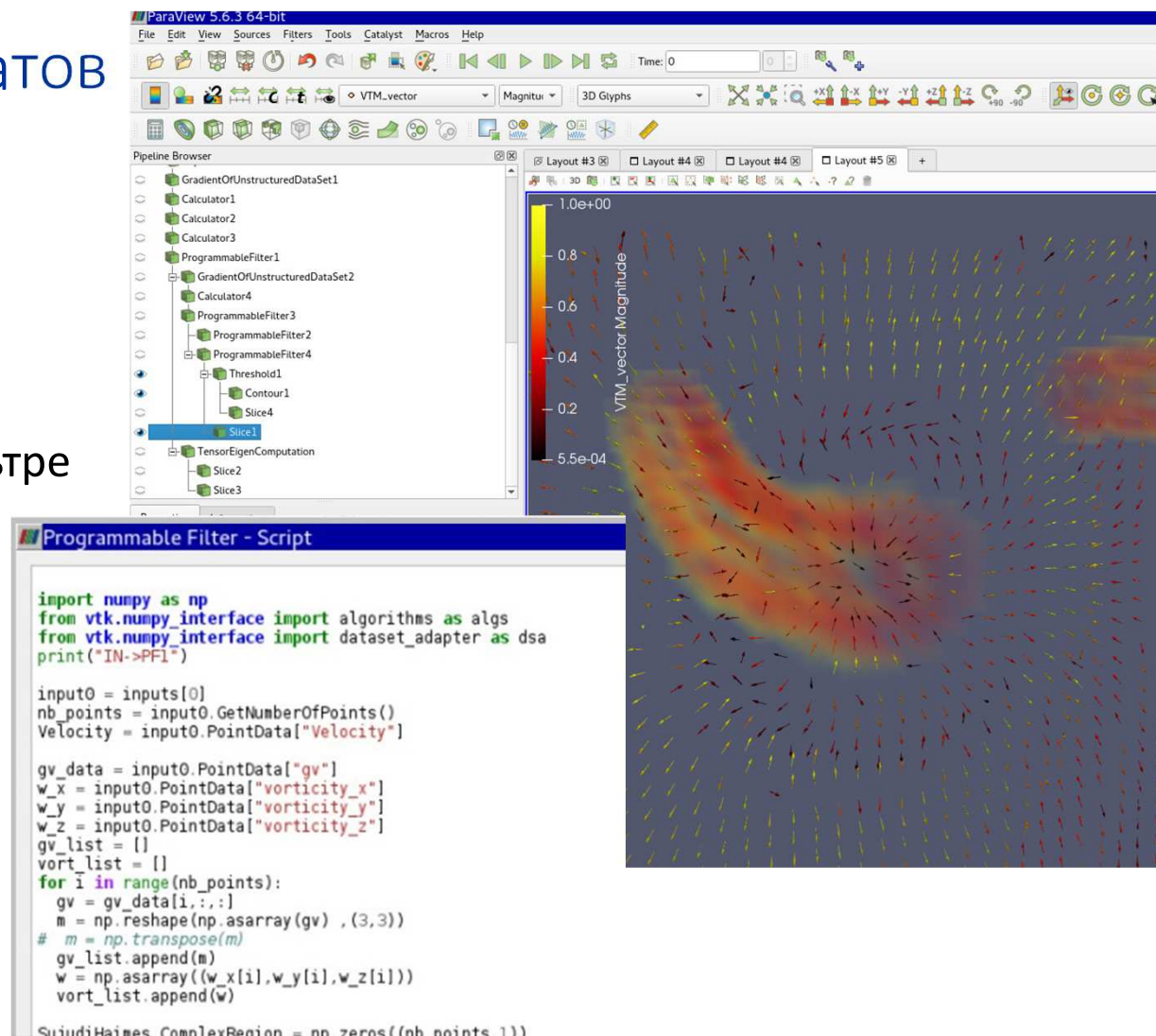
Программа обработки результатов

- Используется программа ParaView v. > 5.6
- Два режима работы
 - Графический в окне
 - Через Скрипты на *Python*
- Обработка данных организуется в виде дерева с соответствующим разграничением доступа
- Набор стандартных опций постпроцессинга:
 - Построение изоповерхностей, контуров, линий тока, графиков функций по линиям и пр.
- Можно вытащить данные в массивы и обрабатывать их вручную
 - Например, с помощью *numpy*
- Имеется возможность обработки на параллельных процессах



Процесс обработки результатов

- Базовый алгоритм разрабатывается в графической моде на ограниченном наборе данных
- Прототип сложной обработки полей пишется в т.н. Программируемом фильтре
 - Например поиск собственных векторов тензоров с помощью *numpy*
- Затем сохраняется два файла:
 - Скрипт *python* модифицируется для работы в консоли
 - Скрипт *pvsm* создается для финальной отрисовки CFD

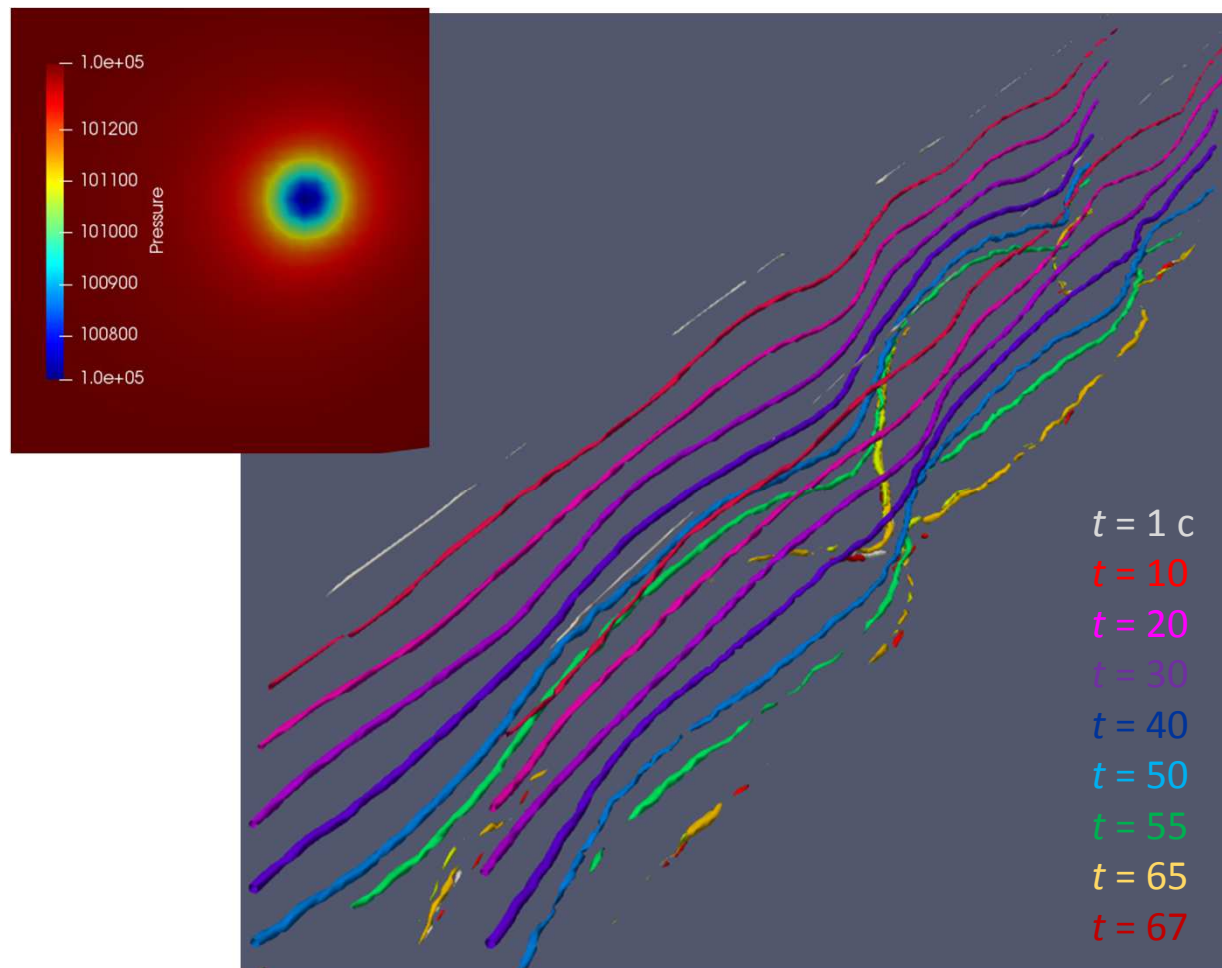


Обзор критериев вихревого течения

- По давлению
- Q-критерий
- λ_2 -критерий
- Sujudi-Haimes
- Levy

Изоповерхности, линии давления

- В центре вихря давление минимально
- Критерий изоповерхности
 - $P_{min} + 1/16 \rho v^2$
- + Легко сделать
- + Идентифицируются самые мощные вихри
- - Ручная обработка поля
- - Легко спутать с другими структурами
 - Например струями

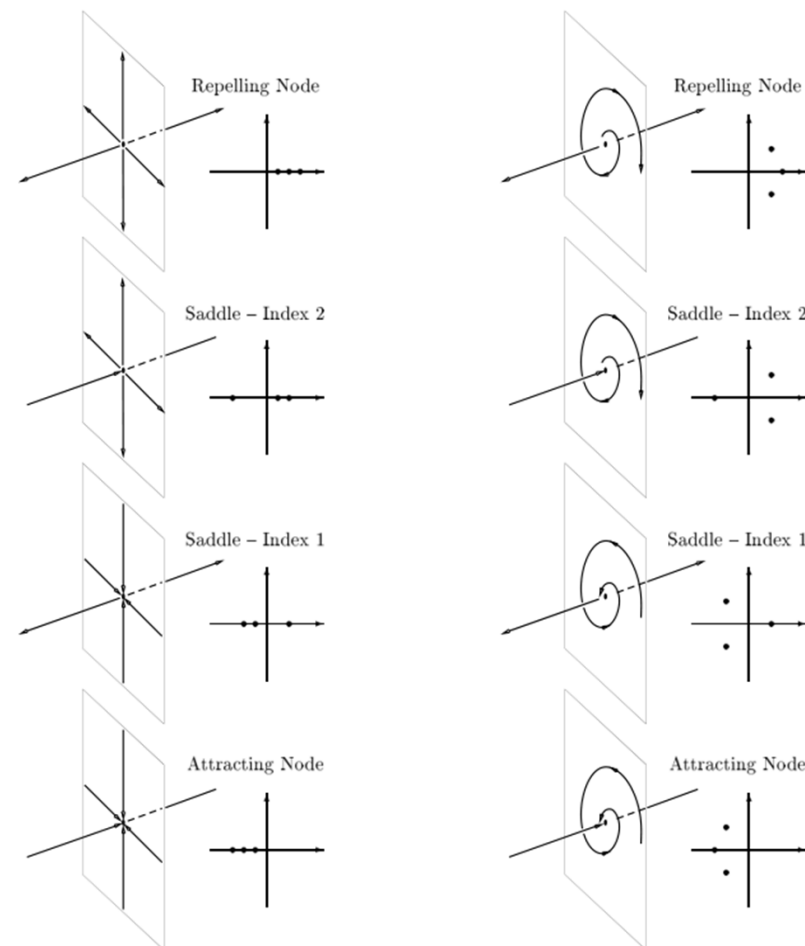


Критерии на основе анализа тензора градиента скорости

- Рассматриваются тензоры на базе градиента вектора скорости

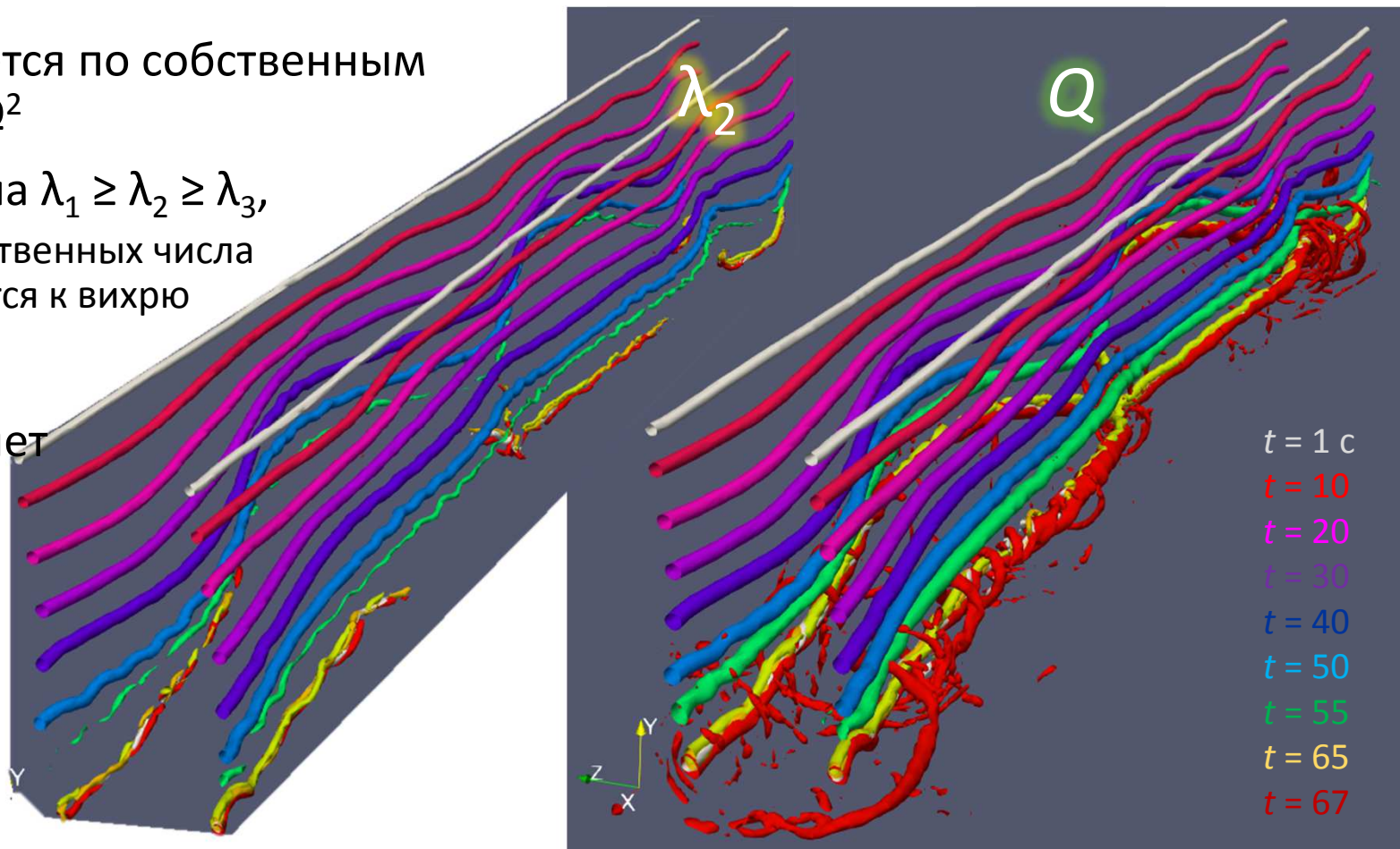
$$\mathfrak{D} = \left\{ \frac{\partial v_i}{\partial x_k} \right\} = \begin{pmatrix} \frac{\partial v_x}{\partial x} & \frac{\partial v_y}{\partial x} & \frac{\partial v_z}{\partial x} \\ \frac{\partial v_x}{\partial y} & \frac{\partial v_y}{\partial y} & \frac{\partial v_z}{\partial y} \\ \frac{\partial v_x}{\partial z} & \frac{\partial v_y}{\partial z} & \frac{\partial v_z}{\partial z} \end{pmatrix} \quad \begin{aligned} \mathbf{S} &= \frac{1}{2}(\mathfrak{D} + \mathfrak{D}^T) \\ \mathbf{\Omega} &= \frac{1}{2}(\mathfrak{D} - \mathfrak{D}^T) \end{aligned}$$

- Тензор скоростей деформации в главных осях показывает направления сжатия расширения газа
- Тензор ротации дополняется вектором завихренности $\mathbf{\Omega} = \boldsymbol{\omega} \times \mathbf{E} = \mathbf{E} \times \boldsymbol{\omega}$.
- По характеру собственных чисел \mathbf{D} можно судить о характере течения в точке



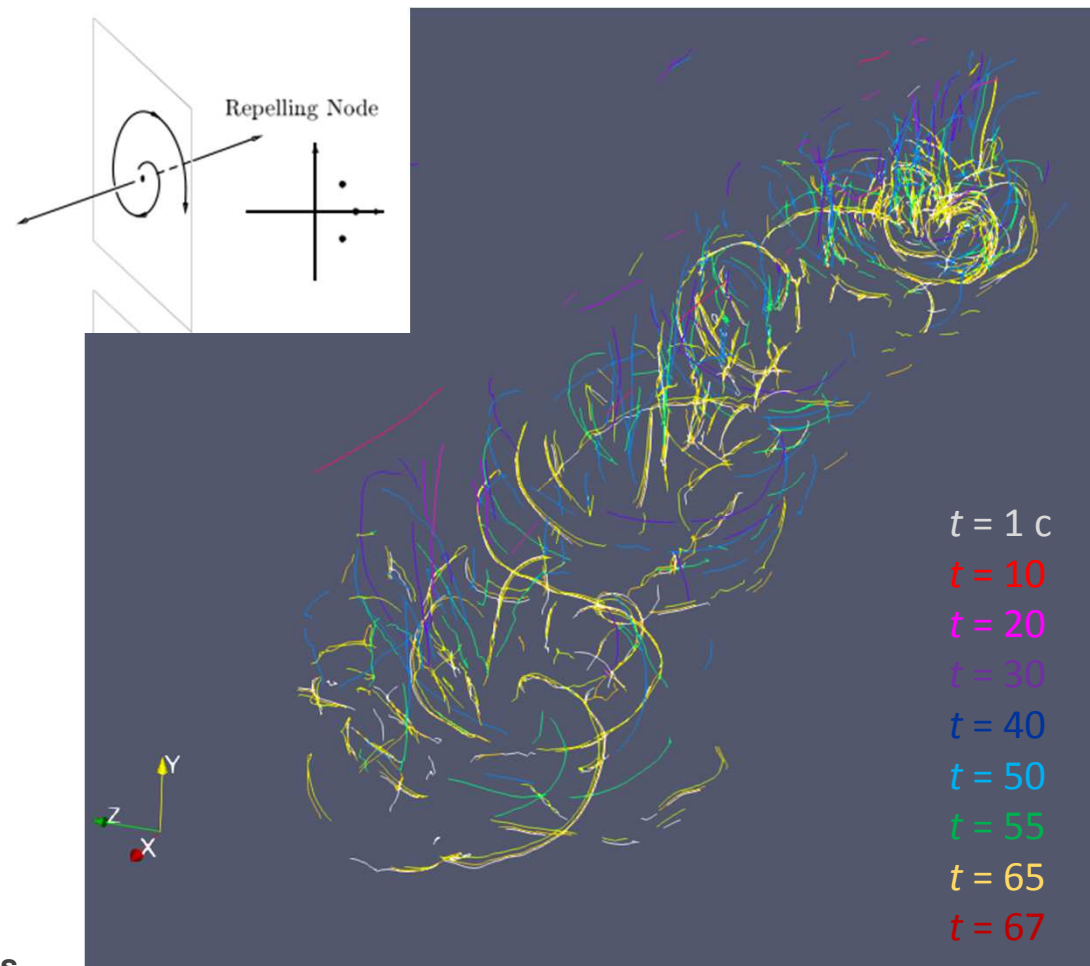
λ_2 и Q критерии

- λ_2 критерий вычисляется по собственным числам матрицы $S^2 + \Omega^2$
- Три собственных числа $\lambda_1 \geq \lambda_2 \geq \lambda_3$,
 - Область, где два собственных числа меньше нуля относятся к вихрю
- Q-критерий определяет область вихря как $Q = \|\Omega^2\| - \|S^2\| > 0$



Критерий Sujudi-Haimes

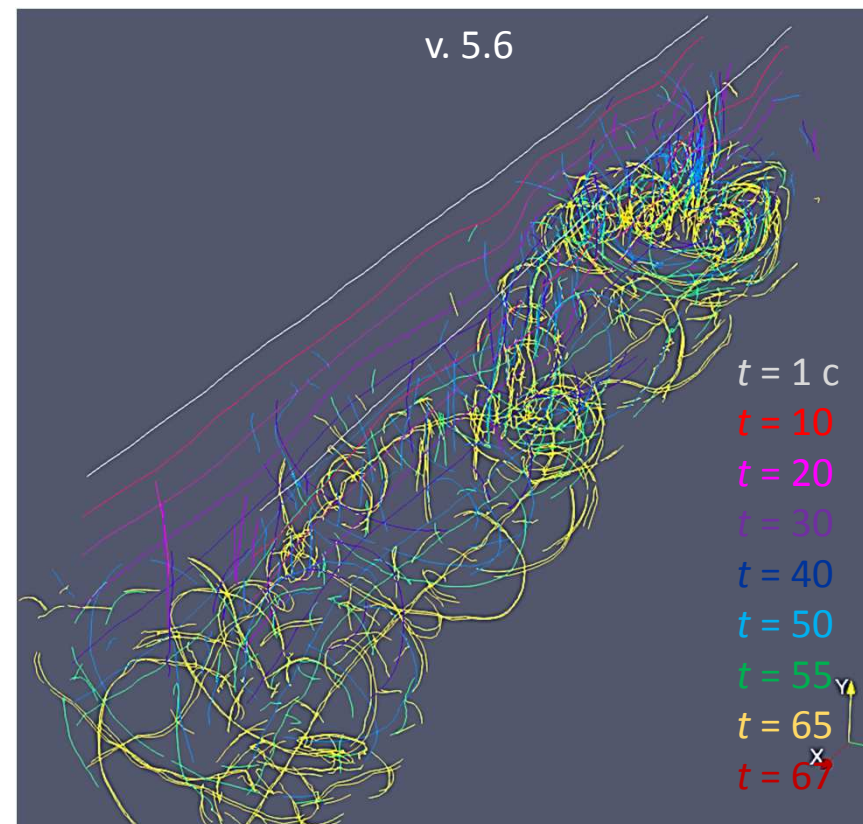
- Проводится поиск линий, где направление вектора скорости ($\dot{\vec{x}}$) совпадает с вектором ускорения ($\ddot{\vec{x}}$)
 - Область присутствия двух комплексно-сопряженных собственных чисел D
 - Собственный вектор от действительного числа должен быть параллелен вектору скорости
 - Эквивалентно $D\dot{\vec{x}} - \lambda_{\text{Re}}\dot{\vec{x}} = 0$
- - Не видит основные вихри



Критерий Леви

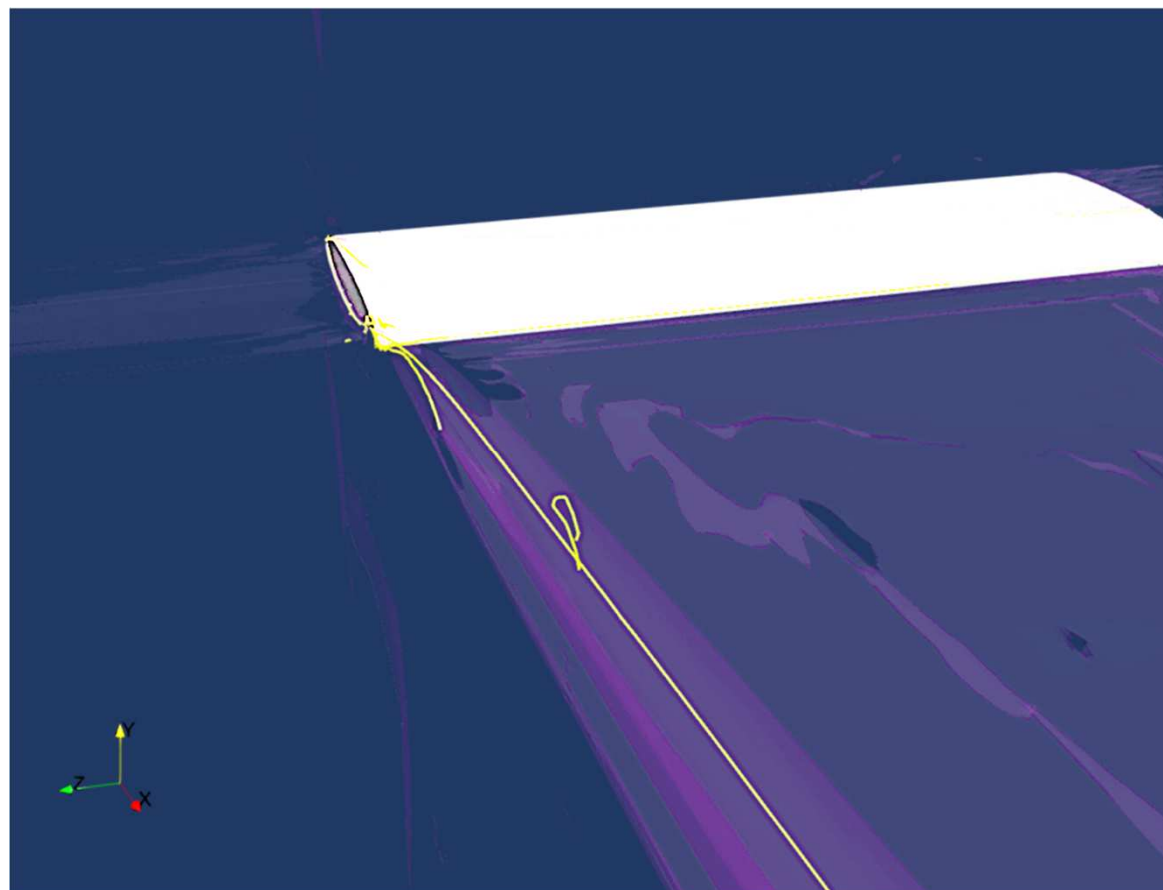
- Проводится поиск линий, где направление скорости ($\dot{\vec{x}}$) совпадает с вектором рывка ($\ddot{\vec{x}}$)

$$T = \{\partial D_{ij} / \partial x_k\} \quad DD\dot{\vec{x}} + T\dot{\vec{x}} = \kappa\dot{\vec{x}}$$
- Две реализации:
 - Плагин VCG для ParaView 5.6.0
 - Фильтр VortexCore в PV 5.9.0
- + Наилучшая идентификация ядер вихрей
- - Теория в целом годится для стационарных течений
- - Тривиальное решение также является решением (проблема с турбулентностью малой интенсивности)
- - Требуется дополнительная фильтрация (например по критерию Q)



Пример на крыле

- Поле по критерию λ_2
- Ядро вихря по критерию Леви



Декомпозиция по динамическим модам DMD

DMD, пролог

- Пусть \vec{u}_i - вектора состояний (размера M) через равные промежутки времени (Δt)
- Многомерные измерения сворачиваются в линейный массив (запоминаем закон свёртки, чтобы использовать его для развёртки)
- Формируются матрицы $U_I = \{\vec{u}_1, \vec{u}_2, \dots, \vec{u}_{N-1}\}$ $U_{II} = \{\vec{u}_2, \vec{u}_3, \dots, \vec{u}_N\}$
- Предполагается линейный переход между состояниями $\vec{u}_i = A\vec{u}_{i-1}$, $U_{II} = AU_I$
- Размер матрицы $A - M \times M$
- Все вектора состояний представляются через собственные вектора (моды) матрицы A

$$\vec{u}_i = \sum_j \vec{\varphi}_j b_j \lambda_j^{i-1} = \sum_j \vec{\varphi}_j b_j e^{(i-1)\omega_j \Delta t} = \Phi \text{diag}(\vec{b}) e^{(i-1)\vec{\omega} \Delta t}$$

$\Phi = \{\vec{\varphi}_j\}$ - матрица собственных векторов (динамических мод) $\vec{b} = \{b_j\}$ - вектор амплитуд $\vec{\omega} = \left\{ \omega_j = \frac{\ln(\lambda_j)}{\Delta t} \right\}$ - вектор частот

DMD, аппарат

- Используется сингулярное разложение матрицы

- через две унитарные матрицы и диагональную

$$U_I = U \Sigma V^*; \quad UU^* = I;$$

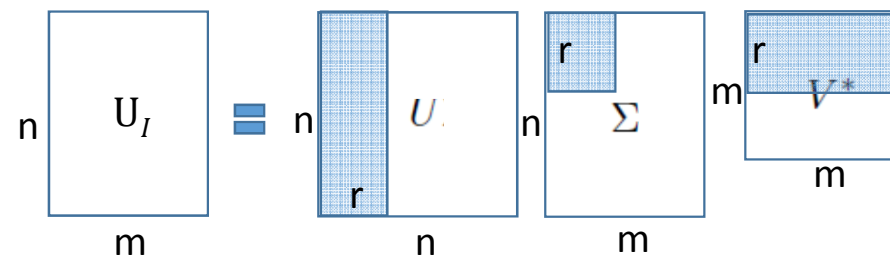
Σ – диагональная матрица сингулярных (убывающих) чисел

- Применяется аппарат python: **numpy.linalg.svd**

$$A = U_{II} U_I^{-1} = U_{II} V \Sigma^{-1} U^*$$

- Строится приближение матрицы A для облегчения поиска собственных чисел

- Из матриц U и V берется первые r столбцов (строк)



$$U^* U_{II} = \widehat{U}_{II} \quad \widetilde{A} = U^* A U$$

- Для \widetilde{A} находятся собственные числа λ_j и вектора W
- Собственные числа у A и \widetilde{A} совпадают
- $\Phi = U_{II} V \Sigma^{-1} W$, Φ – искомая матрица динамических мод

DMD, примеры

• Успешный пример:

- $e^{0.1t} \sin(2\pi(x - t)) + \cos(3\pi(x - t))$
- Данные в виде 2D массива со 100 измерениями в области $[0; 1] \times [0; 1]$
- Результат DMD: $\omega_{1,2} = \pm 9.425i$, $\omega_{3,4} = 0,1 \pm 6.283i$

❖ Контр пример:

$$\diamond f(x, t) = \begin{cases} \sin(20\pi x) \sin(2\pi t), & x < 0,5 \\ \sin(20\pi x) \sin(2\pi t) + \sin(5\pi t), & x \geq 0,5 \end{cases}$$

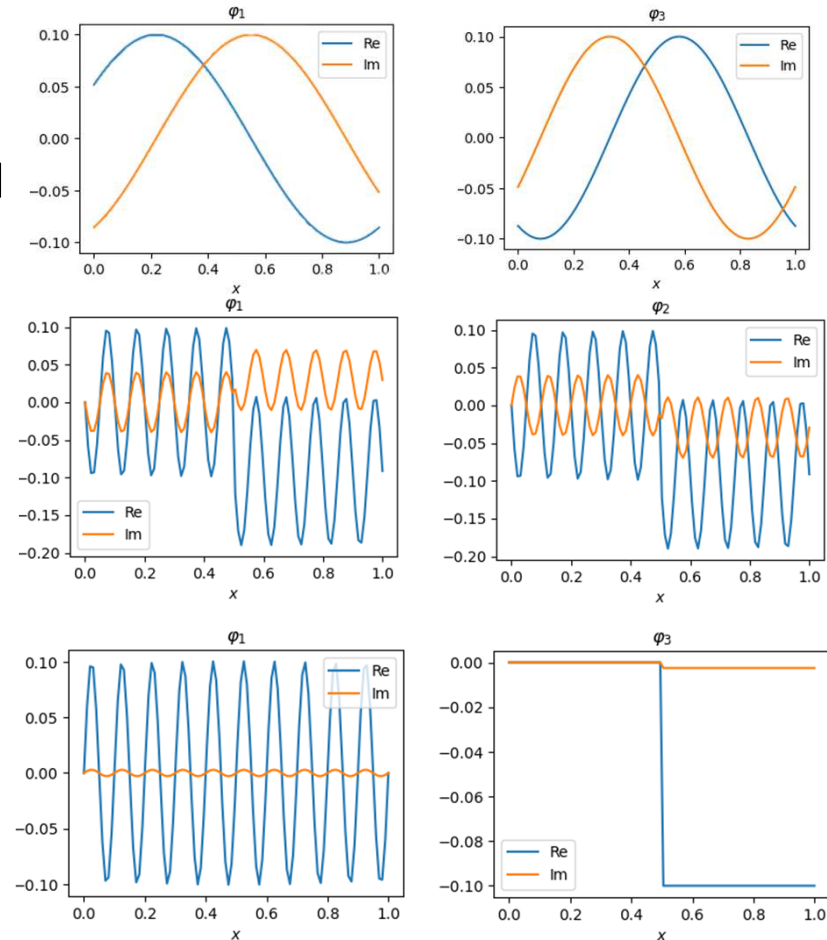
- ❖ Результат DMD: $\omega_{1,2} = 1.235 \pm 5,06i$

■ Разрешение контр примера:

- Бегущее окно



- Оптимальный размер окна для одномерного случая – 8-45% от количества измерений¹
- $\omega_{1,2} = \pm 6,283i$, $\omega_{3,4} = \pm 5i$



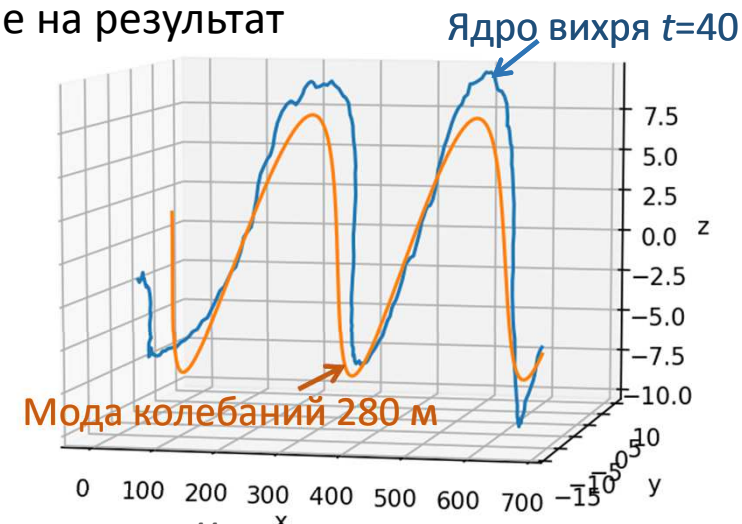
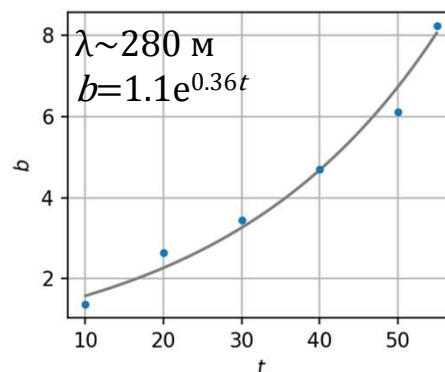
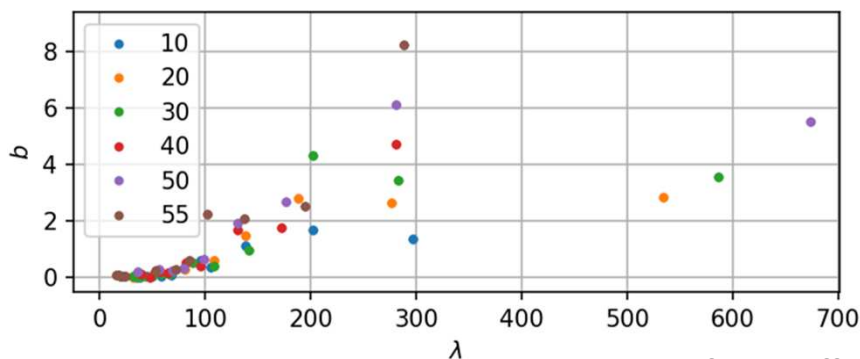
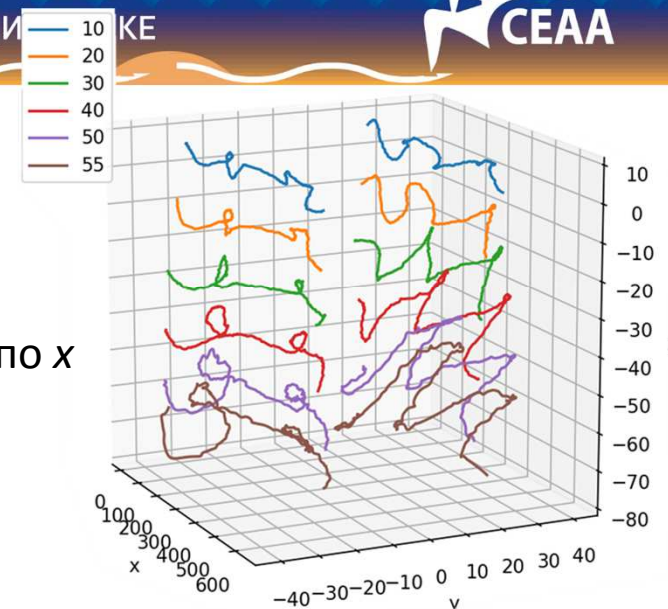
DMD, кратко

- Любое поле сворачивается в линейный вектор/набор линейных векторов
- Одна мода – это столбец из матрицы Φ , матрица Φ – это собственные вектора линейного оператора A динамической системы
 - A - Переход из одного состояния по t в другое
 - A - Переход от одного «окна» в векторе к соседнему
- Когда имеется эволюция по времени, мода может быть любой дискретной функцией $u(x)$
 - Например квадратный импульс



DMD вихревого следа

- Входной массив: линии минимального давления
 - Проведена интерполяция на равномерное распределение точек по x
- Подавление шумов с помощью tlsDMD^1
- Дискретизация по времени очень редкая
 - Проведен DMD с бегущим окном для каждого t
 - Размер окна варьировался для того, чтобы изучить его влияние на результат
 - Выделены волны ~ 100 , ~ 140 , ~ 280 м



1. Dawson S. T. M. et al. Characterizing and correcting for the effect of sensor noise in the dynamic mode decomposition // Experiments in Fluids. – 2016. – Т. 57. – №. 3. – С. 42.

Заключение

- При исследовании вихрей рекомендуется выделить из всей расчётной области зоны интереса
- Наиболее наглядный способ идентификации и визуализации – критерий параллельности местного вектора скорости и вектора рывка
- Метод DMD нашел моду неустойчивости с длиной волны сопоставимой с длиной Найквиста