

**Anes Team**



***ANES20XE: Код для численного моделирования процессов гидродинамики и теплообмена***

Версия 2.24

LES модели турбулентности

Москва 2019 г.

## Оглавление.

<b>ВВЕДЕНИЕ .....</b>	<b>4</b>
<b>1. ОПИСАНИЕ LES МОДЕЛЕЙ .....</b>	<b>5</b>
1.1 DNS уравнения для мгновенных переменных .....	5
1.2 RANS уравнения .....	6
1.3 LES уравнения .....	8
1.4 Связь LES и RANS уравнений .....	9
1.5 Задачи с Struct зонами .....	10
1.6 LES модели ANES.....	10
1.6.1 Модель Смагоринского-Лилли .....	11
1.6.2 Пристеночное демпфирование .....	11
1.6.3 Модель WALE .....	12
1.6.4 Модель CSM Кобаяши .....	12
1.6.5 DNS модель.....	13
<b>2. РАБОТА С LES МОДЕЛЮ.....</b>	<b>14</b>
2.1 Настройка .....	14
2.2 Особенности проведения расчетов .....	14
2.3 LES-модели пользователя .....	15
2.4 Особенности работы с уравнением энергии.....	17
2.5 Начальные условия.....	17
2.6 Возможности LES обработки.....	19
2.7 Мониторинг в процессе расчета. Сенсоры .....	20
2.8 Мониторинг в процессе расчета. Средние параметры по PO .....	20
2.9 Мониторинг в процессе расчета. Средние параметры на патчах .....	21
2.10 Мониторинг в процессе расчета. 3d поля .....	22
2.11 Режим осреднения .....	22
2.11.1 Режим осреднения: псевдополярная СК .....	25
2.11.2 Режим осреднения: F(z) распределения по граничным патчам .....	26
2.11.3 Режим осреднения: F(z,xy) распределения по граничным патчам .....	26
2.11.4 Режим осреднения: F(z) распределения интегральных параметров .....	27
2.12 Модель заданного расхода .....	28
2.12.1 Алгоритм V. Deschamps .....	29
2.12.2 Алгоритм MVM .....	29
2.12.3 Интеграция модели заданного расхода и PIG-модели.....	30

2.13	Расчет безразмерного расстояния до стенки для модели Пиомелли .....	30
2.14	Утилита переинтерполяции .....	31
<b>3.</b>	<b>ГИБРИДНЫЕ LES МОДЕЛИ .....</b>	<b>33</b>
3.1	IDDES: гибридная LES модель .....	33
3.1.1	Гибридный масштаб турбулентных вихрей .....	34
3.1.2	Активация модели .....	35
3.2	KDES модель .....	36
3.2.1	Описание модели .....	36
3.2.2	Настройка модели .....	38
<b>4.</b>	<b>ПАКЕТНЫЙ ПОСТПРОЦЕССОР LESPOST20 .....</b>	<b>40</b>
4.1	Файл скрипта .....	41
4.2	Общие операторы .....	41
4.3	Операторы для обработки временного сигнала .....	42
4.3.1	Оператор Signal .....	43
4.3.2	Оператор PDF .....	44
4.3.3	Оператор AutoCorr .....	45
4.3.4	Оператор SPECTRA .....	46
4.3.5	Оператор SMOOTH .....	48
4.4	Операторы для обработки файла результатов .....	50
4.4.1	Операторы статистической обработки .....	50
4.4.2	Оператор трехмерного спектра .....	51
4.4.3	Оператор осреднения в декартовой СК .....	54
4.4.4	Оператор осреднения в псевдополярной СК .....	55
4.4.5	Оператор 2D осреднения .....	57
4.5	Операторы для обработки граничных и FS патчей .....	59
4.5.1	Создание VTK файла .....	59
4.5.2	Одномерные распределения и осреднение .....	59

## **Введение**

В данном документе описаны различные версии LES (Large Eddy Simulation) моделей турбулентности кода ANES. Приведены описание настройки математических моделей, реализованных в Решателе, и описание пакетного постпроцессора для обработки результатов моделирования.

## 1. Описание LES моделей

### 1.1 DNS уравнения для мгновенных переменных

Рассмотрим определяющие уравнения баланса, которые используются в LES моделях. Рассматривается однофазная среда с переменными свойствами при наличии теплообмена. Для описания будем использовать декартовую системы координат (СК) и тензорные обозначения (по повторяющимся индексам подразумевается суммирование от 1 до 3). В этом случае уравнения сохранения массы, импульса и энергии можно записать в виде

$$\begin{aligned} \frac{\partial \rho}{\partial \tau} + \frac{\partial}{\partial x_k} (\rho u_k) &= 0, \\ \frac{\partial (\rho u_i)}{\partial \tau} + \frac{\partial}{\partial x_k} (\rho u_k u_i + \sigma_{ik}) &= S_i, \\ \frac{\partial (\rho h)}{\partial \tau} + \frac{\partial}{\partial x_k} (\rho u_k h - \lambda \nabla T) &= S_h \end{aligned} \quad (1.1)$$

Здесь  $h$  - энтальпия,  $\sigma_{ik}$  - составляющая тензора плотности потока импульса, связанная с молекулярными эффектами (тензор внутренних напряжений).  $S_i$ ,  $S_h$  - внутренние удельные источники.

Тензор плотности потока импульса можно представить в виде суммы изотропной и не-изотропной частей:

$$\sigma_{ik} = p \delta_{ik} - \tau_{ik}, \quad (1.2)$$

где  $p$  - статическое давление,  $\tau_{ik}$  - тензор вязких напряжений. Для его расчета используется закон Ньютона:

$$\tau_{ik} = 2\mu \left[ S_{ik} - \frac{1}{3} \delta_{ik} S_{mm} \right], \quad S_{ik} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_k} + \frac{\partial u_k}{\partial x_i} \right) \quad (1.3)$$

Появление второго члена (дивергенции скорости) связано с выполнением условия  $\tau_{mm} = 0$ . Объединяя (1.1) - (1.2) уравнения движения можно записать в виде стандартного уравнения переноса:

$$\frac{\partial (\rho u_i)}{\partial \tau} + \frac{\partial}{\partial x_k} \left( \rho u_k u_i - \mu \frac{\partial u_i}{\partial x_k} \right) = S_i - \frac{\partial P}{\partial x_i} + \frac{\partial}{\partial x_k} \left( \mu \frac{\partial u_k}{\partial x_i} \right), \quad (1.4)$$

$$P = p + \frac{2}{3} \mu \left( \frac{\partial u_m}{\partial x_m} \right)$$

Последний член в первом уравнении представляет собой дополнительный вязкий источник, связанный с отличием закона Ньютона (1.3) от стандартного градиентного закона молекулярного переноса. Если вязкость и плотность постоянны, то этот член равен нулю.

Для расчета этого члена можно использовать следующие соотношения:

$$-\frac{\partial}{\partial x_i} \left[ p + \frac{2}{3} \mu \left( \frac{\partial u_m}{\partial x_m} \right) \right] + \frac{\partial}{\partial x_k} \left( \mu \frac{\partial u_k}{\partial x_i} \right) = -\frac{\partial}{\partial x_i} \left[ p - \frac{1}{3} \mu \left( \frac{\partial u_m}{\partial x_m} \right) \right] - \frac{\partial}{\partial x_i} \left[ \mu \left( \frac{\partial u_m}{\partial x_m} \right) \right] + \frac{\partial}{\partial x_k} \left( \mu \frac{\partial u_k}{\partial x_i} \right)$$

Если ввести новое приведенное давление

$$P = p - \frac{1}{3} \mu \left( \frac{\partial u_m}{\partial x_m} \right), \quad (1.5)$$

то дополнительный источник можно записать в виде:

$$S_{\mu,i} = \frac{\partial \mu}{\partial x_k} \frac{\partial u_k}{\partial x_i} - \frac{\partial \mu}{\partial x_i} \frac{\partial u_k}{\partial x_k} \quad (1.6)$$

Обычно этот член мал по сравнению с другими членами. По умолчанию в Решателе этот член игнорируется. Если необходимо его рассчитывать, то нужно активировать флаг секции [Special Data]:

L("IsIntStressSource") = .True.

Окончательно систему уравнений для прямого DNS численного моделирования турбулентности можно записать в виде:

$$\begin{aligned} \frac{\partial \rho}{\partial \tau} + \frac{\partial}{\partial x_k} (\rho u_k) &= 0, \\ \frac{\partial (\rho u_i)}{\partial \tau} + \frac{\partial}{\partial x_k} \left( \rho u_k u_i - \mu \frac{\partial u_i}{\partial x_k} \right) &= -\frac{\partial P}{\partial x_k} + S_{\mu,i} + S_i \\ \frac{\partial (\rho h)}{\partial \tau} + \frac{\partial}{\partial x_k} (\rho u_k h - \lambda \nabla T) &= S_h \end{aligned} \quad (1.5)$$

Для корректного DNS моделирования необходимо использовать схемы высокого порядка (выше второго) по пространству и времени, поэтому DNS моделирование не предусмотрено в коде ANES.

## 1.2 RANS уравнения

Для инженерных расчетов необходимы поля, осредненные по периоду времени значительно большему, чем характерное время пульсаций мгновенных полей -  $T_\tau$ . Для этого можно использовать два алгоритма усреднения:

$$\bar{F} = \frac{1}{T_\tau} \int_0^{T_\tau} F(\mathbf{x}, \tau) \cdot d\tau, \quad \tilde{F} = \frac{\overline{\rho \cdot F}}{\bar{\rho}} \quad (1.6)$$

Второе осреднение - осреднение по Фавре - используется для задач, в которых плотность переменна и ее пульсации достаточно велики. Зная средние значения можно ввести понятия пульсаций:

$$F' = F - \bar{F}, \quad F'' = F - \tilde{F} \quad (1.7)$$

Эти пульсации удовлетворяют очевидным соотношениям:

$$\bar{F}' = 0, \quad \overline{\rho F''} = 0, \quad \overline{F''} = \bar{F} - \tilde{F} = -\frac{\overline{\rho' F'}}{\bar{\rho}} = -\frac{\overline{\rho' F''}}{\bar{\rho}}, \quad (1.8)$$

Рассмотрим случай, когда *плотность, вязкость и теплопроводность постоянны*. Применяя осреднение (1.6) к системе (1.5) можно получить осредненные уравнения Навье-Стокса (RANS):

$$\begin{aligned} \frac{\partial}{\partial x_k} (\rho \bar{u}_k) &= 0, \\ \frac{\partial}{\partial x_k} \left( \rho \bar{u}_k \bar{u}_i - \tau_{ik}^t - \mu \frac{\partial \bar{u}_i}{\partial x_k} \right) &= -\frac{\partial \bar{p}}{\partial x_k} + S_{\mu,i} + S_i, \\ \frac{\partial}{\partial x_k} (\rho \bar{u}_k \bar{h} + \mathbf{q}_k^t - \lambda \nabla \bar{T}) &= S_h, \\ \tau_{ik}^t &= -\overline{\rho u_i' u_k'}, \quad \mathbf{q}_k^t = +\overline{\rho h' \cdot u_k'} = \rho c_p \overline{T' \cdot u_k'} \end{aligned} \quad (1.9)$$

Здесь  $\tau_{ik}^t$ ,  $\mathbf{q}^t$  - тензор турбулентных напряжений и вектор турбулентного переноса тепла. В большинстве RANS моделях турбулентности для расчета этих переменных используется модель вихревой вязкости. В этой модели используется аналогия с (1.3):

$$\tau_{ik}^t - \frac{1}{3} \delta_{ik} \tau_{mm}^t = 2\mu_t \left( S_{ik} - \frac{1}{3} \delta_{ik} S_{mm} \right), \quad (1.10)$$

$$S_{mm} = \frac{\partial u_m}{\partial x_m} = \text{div}(\mathbf{u}), \quad \tau_{mm}^t = -\rho \overline{u'_m u'_m} = -2\rho k$$

Аналогичная гипотеза используется и для турбулентного теплового потока:

$$\mathbf{q}_k^t = \rho \overline{h' \cdot \mathbf{u}'_k} = -\frac{\mu_t c_p}{Pr_t} \nabla \bar{T} \quad (1.11)$$

Здесь  $k$  - удельная энергия турбулентности,  $Pr_t$  - турбулентное число Прандтля. В итоге систему (1.9) можно привести к стандартному для RANS виду:

$$\begin{aligned} \frac{\partial}{\partial x_k} (\rho \bar{u}_k) &= 0, \\ \frac{\partial}{\partial x_k} \left( \rho \bar{u}_k \bar{u}_i - \mu_{\text{eff}} \frac{\partial \bar{u}_i}{\partial x_k} \right) &= -\frac{\partial \bar{P}}{\partial x_k} + S_{\mu,i} + S_i, \\ \frac{\partial}{\partial x_k} (\rho \bar{u}_k \bar{h} - \lambda_{\text{eff}} \nabla \bar{T}) &= S_h, \\ \mu_{\text{eff}} &= \mu + \mu_t, \quad \lambda_{\text{eff}} = \lambda + \frac{\mu_t c_p}{Pr_t}, \quad \bar{P} = \bar{p} + \frac{2}{3} \rho k \end{aligned} \quad (1.12)$$

Имея мгновенные поля DNS моделирования можно «рассчитать» коэффициенты турбулентной вязкости и турбулентное число Прандтля. Например, для случая, когда ось  $x$  направлена вдоль стенки, а ось  $y$  - перпендикулярно стенке, то:

$$\mu_t = -\frac{\rho \overline{u'_x u'_y}}{\frac{\partial \bar{u}_x}{\partial y}}, \quad \lambda_t = \frac{\rho \overline{T u'_y}}{\frac{\partial \bar{T}}{\partial y}}, \quad Pr_t = \frac{\mu_t c_p}{\lambda_t} \quad (1.13)$$

Если имеются мгновенные поля, то для расчета тензора турбулентных напряжений и вектора турбулентного потока тепла можно использовать непосредственно определения этих величин, а не пульсации полей:

$$\tau_{ik}^t = \rho \bar{u}_i \bar{u}_k - \rho \bar{u}_i u_k, \quad q_k^t = \rho \bar{u}_k \bar{h} - \rho \bar{u}_k h \quad (1.14a)$$

Именно эти соотношения используются в LES модели ANES для получения осредненных величин для любых корреляций:

$$\overline{f'_1 \cdot f'_2} = \overline{f_1 \cdot f_2} - \bar{f}_1 \cdot \bar{f}_2 \quad (1.14b)$$

Если свойства (особенно плотность) *переменны*, то вид осредненных уравнений существенно усложняется. Обычно используется приближенная форма уравнений, содержащая оба типа осреднений. Поля для двух осреднений связаны между собой через корреляцию поля с пульсацией плотности.

$$\bar{F} - \tilde{F} = -\frac{\overline{\rho' F'}}{\bar{\rho}} = -\frac{\overline{\rho' F''}}{\bar{\rho}} \quad (1.15)$$

Чаще всего используется следующая система уравнений:

$$\begin{aligned} \frac{\partial \bar{p}}{\partial \tau} + \frac{\partial}{\partial x_k} (\bar{\rho} \tilde{u}_k) &= 0, \\ \frac{\partial (\bar{\rho} \tilde{u}_i)}{\partial \tau} + \frac{\partial}{\partial x_k} \left( \bar{\rho} \tilde{u}_k \tilde{u}_i - \bar{\mu} \frac{\partial \tilde{u}_i}{\partial x_k} - \tau_{ik}^t \right) &= -\frac{\partial \bar{P}}{\partial x_k} + S_{\mu,i} + S_i \\ \frac{\partial (\bar{\rho} \tilde{h})}{\partial \tau} + \frac{\partial}{\partial x_k} (\bar{\rho} \tilde{u}_k \tilde{h} - \lambda \nabla \tilde{T} + \mathbf{q}_k^t) &= S_h \end{aligned} \quad (1.16)$$

Турбулентные величины в этом случае определяются по соотношениям (1.14a):

$$\begin{aligned} \tau_{ik}^t &= \bar{\rho} \tilde{u}_i \tilde{u}_k - \overline{\rho u_i u_k} = \bar{\rho} (\tilde{u}_i \tilde{u}_k - \widetilde{u_i u_k}) = -\overline{\rho u_i'' u_k''} = -\overline{\rho u_i'' u_k''}, \\ \mathbf{q}_k^t &= \overline{\rho u_k h} - \bar{\rho} \tilde{u}_k \tilde{h} = \bar{\rho} (\widetilde{u_k h} - \tilde{u}_k \tilde{h}) = -\overline{\rho h_i'' u_k''} = -\overline{\rho u_k'' h''} \end{aligned} \quad (1.17)$$

### 1.3 LES уравнения

В LES подходе комбинируются DNS и RANS подходы. Для этого вводится понятие пространственной фильтрации. В данном пункте верхние символы «осреднения» будут использоваться для *отфильтрованных* значений:

$$\bar{F}(\mathbf{x}) = \int F(\mathbf{x}_1) \cdot G(\mathbf{x}_1, \mathbf{x}) dV_1, \quad \tilde{F}(\mathbf{x}) = \frac{1}{\bar{\rho}} \int \rho F(\mathbf{x}_1) \cdot G(\mathbf{x}_1, \mathbf{x}) dV_1 \quad (1.18)$$

Здесь  $G$  - это фильтр, удовлетворяющий нормировке:

$$\int G(\mathbf{x}_1, \mathbf{x}) dV_1 = 1$$

При использовании численных схем, основанных на МКО, фильтрация явно присутствует в дискретных уравнениях:

$$\bar{F}_p = \frac{1}{\Delta V_p} \int F dV \quad (1.19)$$

В этом случае роль фильтра играет единичная функция, отличная от нуля только в КО:

$$G(\mathbf{x}_p, \mathbf{x}) = \begin{cases} 1/\Delta V_p, & \mathbf{x} - \mathbf{x}_p \leq \Delta V_p \\ 0, & \mathbf{x} - \mathbf{x}_p > \Delta V_p \end{cases}$$

Применяя (1.18) к системе (1.5) можно получить нестационарные уравнения LES модели:

$$\begin{aligned} \frac{\partial \bar{p}}{\partial \tau} + \frac{\partial}{\partial x_k} (\bar{\rho} \bar{u}_k) &= 0, \\ \frac{\partial (\bar{\rho} \bar{u}_i)}{\partial \tau} + \frac{\partial}{\partial x_k} \left( \bar{\rho} \bar{u}_k \bar{u}_i - \bar{\mu} \frac{\partial \bar{u}_i}{\partial x_k} - \tau_{ik}^{sgs} \right) &= -\frac{\partial \bar{P}}{\partial x_k} + S_{\mu,i} + S_i \\ \frac{\partial (\bar{\rho} \bar{h})}{\partial \tau} + \frac{\partial}{\partial x_k} (\bar{\rho} \bar{u}_k \bar{h} - \lambda \nabla \bar{T} - \mathbf{q}_k^{sgs}) &= S_h \end{aligned} \quad (1.20)$$

Подсеточные напряжения и подсеточный тепловой поток равны:

$$\tau_{ik}^{sgs} = \bar{\rho} \bar{u}_i \bar{u}_k - \overline{\rho u_i u_k}, \quad \mathbf{q}_k^{sgs} = \overline{\rho u_k h} - \bar{\rho} \bar{u}_k \bar{h} \quad (1.21)$$

Для этих величин в теории LES моделей предложено много соотношений [1]. В коде ANES используется модель подсеточной турбулентной вязкости:



$$\tau_{ik}^{sgs} - \frac{1}{3} \delta_{ik} \tau_{mm}^{sgs} = 2\mu_{sgs} \left( S_{ik} - \frac{1}{3} \delta_{ik} S_{mm} \right), \quad (1.22)$$

$$q_k^{sgs} = -\frac{\mu_{sgs} c_p}{Pr_{sgs}} \nabla \bar{T}$$

Будем далее опускать символы фильтрации. В результате окончательную систему LES-уравнений можно записать в виде:

$$\frac{\partial \rho}{\partial \tau} + \frac{\partial}{\partial x_k} (\rho u_k) = 0,$$

$$\frac{\partial (\rho u_i)}{\partial \tau} + \frac{\partial}{\partial x_k} \left( \rho u_k u_i - \mu \frac{\partial u_i}{\partial x_k} - \mu_{sgs} \frac{\partial u_i}{\partial x_k} \right) = -\frac{\partial P}{\partial x_k} + S_{\mu,i} + S_i \quad (1.23)$$

$$\frac{\partial (\rho h)}{\partial \tau} + \frac{\partial}{\partial x_k} (\rho u_k h - \lambda \nabla T - \lambda_{sgs} \nabla T) = S_h$$

Здесь

$$P = p - \frac{1}{3} (\mu + \mu_{sgs}) \left( \frac{\partial u_m}{\partial x_m} \right) + \frac{1}{3} \tau_{mm}^{sgs}, \quad (1.24)$$

$$S_{\mu,i} = \frac{\partial (\mu + \mu_{sgs})}{\partial x_k} \frac{\partial u_k}{\partial x_i} - \frac{\partial (\mu + \mu_{sgs})}{\partial x_i} \frac{\partial u_k}{\partial x_k}$$

Сумму диагональных членов тензора подсеточных напряжений в «новом» давлении можно записать в виде:

$$\tau_{mm}^{sgs} = -2\rho k_{sgs} \quad (1.25)$$

где  $k_{sgs}$  - это энергия подсеточных флуктуаций скорости. В рамках модели подсеточной вязкости этот член *не моделируется*. Он просто включается в «новое» давление.

Отметим еще один важный момент. Осреднение по времени и фильтрация принципиально отличаются. В отличие от осреднения для фильтрованных величин:

$$\overline{F'} = \overline{(F - \bar{F})} \neq 0, \quad \bar{F} \neq \overline{(\bar{F})}$$

Поэтому:

$$-\frac{\tau_{ik}^{sgs}}{\rho} = \overline{u_i u_k} - \bar{u}_i \bar{u}_k = \left\{ \overline{u_i u_k} - \bar{u}_i \bar{u}_k \right\} + \overline{u_i' \bar{u}_k + \bar{u}_i u_k'} + \overline{u_i' u_k'} = L_{ik} + C_{ik} + R_{ik}$$

где  $L_{ik}$  - тензор Леонарда,  $C_{ik}$  - перекрестный тензор,  $R_{ik}$  - тензор подсеточных напряжений Рейнольдса. Из этих трех тензоров точно рассчитать можно только  $L$ . Поэтому в практических LES моделях сразу моделируется сумма этих трех тензоров с помощью соотношения (1.22).

Практически все LES модели разработаны для несжимаемой среды. Если плотность переменна, то нужно использовать фильтрацию Фавре. Разумных строгих соотношений пока не получено. Поэтому для переменной плотности используется та же система (1.23).

#### 1.4 Связь LES и RANS уравнений

Использование LES модели аналогично DNS моделированию. Решается нестационарная система уравнений (1.23) после чего производится осреднение по времени по соотношениям (1.6) и (1.14). В итоге получаются поля осредненных переменных и их двойных корреляций. В этом разделе верхнюю черту будем использовать для отфильтрованных полей, а символы  $\langle \rangle$  - для осредненных по времени. Осредненные результаты моделирования можно (и нужно) сравнить с результатами DNS-моделирования или эксперименталь-

ными данными. Как это сделать правильно? Очевидно, что для средних переменных можно считать, что

$$\langle F \rangle_{\text{dns}} = \langle \bar{F} \rangle_{\text{les}}$$

Для двойных корреляций скоростей и температуры более правильным являются соотношения:

$$\langle u'_i u'_k \rangle_{\text{dns}} = \langle u'_i u'_k \rangle_{\text{les}} - \langle \frac{\tau_{ik}^{\text{sgs}}}{\rho} \rangle, \quad \langle u'_i h' \rangle_{\text{dns}} = \langle u'_i h' \rangle_{\text{les}} + \langle \frac{q_i^{\text{sgs}}}{\rho} \rangle \quad (1.26)$$

Или

$$\langle u'_i u'_k \rangle_{\text{dns}} = \langle u'_i u'_k \rangle_{\text{les}} - \langle 2v_{\text{sgs}} \bar{S}_{ik} \rangle, \quad \text{если } i \neq k$$

$$\langle u'_i h' \rangle_{\text{dns}} = \langle u'_i h' \rangle_{\text{les}} - \langle \frac{v_{\text{sgs}} c_p}{Pr_{\text{sgs}}} \frac{\partial \bar{T}}{\partial x_i} \rangle \quad (1.27)$$

Эти соотношения имеют прозрачный физический смысл. Например:

$\langle u'_i u'_k \rangle_{\text{les}}$  - это разрешаемая часть турбулентных напряжений,

$-\langle 2v_{\text{sgs}} \bar{S}_{ik} \rangle$  - это моделируемая (с помощью LES) часть турбулентных напряжений.

К сожалению, применить эти соотношения для диагональных членов тензора пульсаций скоростей нельзя, поскольку неизвестна величина  $k_{\text{sgs}}$ .

### 1.5 Задачи с Struct зонами

LES подсистема позволяет решать задачи, в которых могут присутствовать твердые стенки. В этом случае система (1.23) дополняется уравнением энергии для зон Struct для  $\Phi$ -переменной  $T_s$ :

$$\frac{\partial (\rho_s c_{ps} T_s)}{\partial \tau} = \frac{\partial}{\partial x_k} (\lambda_s \nabla T_s) + S_{ts}$$

На границе раздела F- и S- областей автоматически производится сшивка тепловых потоков. Для температуры  $T_s$  можно использовать все средства обработки, аналогичные для температуры F-фазы.

### 1.6 LES модели ANES

В текущей версии кода LES модель фактически сводится к модели алгебраической турбулентной вязкости, которая рассчитывает значения подсеточной вязкости  $v_{\text{sgs}}$ .

Уравнения сохранения для  $\Phi$ -переменных при этом не изменяются, а коэффициенты переноса рассчитываются по обычным соотношениям:

$$v_{\text{eff}} = v_g + v_{\text{sgs}}, \quad \lambda_{\text{eff}} = \lambda_g + \frac{\rho_g c_{pg} v_{\text{sgs}}}{Pr_{\text{sgs}}} \quad (1.28)$$

Здесь  $Pr_{\text{sgs}}$  - «подсеточное» турбулентное число Прандтля (по умолчанию = 0.85). Отметим, что значение подсеточной вязкости сохраняется в обычном массиве  $NuTUR(\cdot)$ .

Для расчета подсеточной вязкости используется две алгебраических модели: модель Смагоринского-Лилли (Smagorinsky-Lilly) [1] и модель WALE (Wall-Adapting Local Eddy-Viscosity) [1,3] Никоида-Дюкро. Наряду с этими моделями пользователь может реализовать свою собственную версию LES-модели.

### 1.6.1 Модель Смагоринского-Лилли

В этой модели подсеточная вязкость рассчитывается по соотношению:

$$v_{sgs} = L_s^2 \cdot \sqrt{G}, \quad G = 2 \cdot \sum_i \sum_j S_{ij} \cdot S_{ij}, \quad S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (1.29)$$

где  $L_s$  - масштаб длины. В текущей версии кода предусмотрено две модели расчета этого масштаба. В первой модели (назовем ее VOL13):

$$\Delta_{les} = (\Delta V_p)^{1/3}, \quad L_s = C_{sgs} \Delta_{les} \quad (1.30a)$$

где  $\Delta V_p$  - объем текущей ячейки сетки,  $C_{sgs}$  - постоянная модели,  $\Delta_{les}$  - подсеточный масштаб длины. Во второй модели (SHUR2008) [5]:

$$\Delta_{les} = \min(C_w \max(y_w, \Delta_{max}), \Delta_{max}), \quad L_s = \min(C_{sgs} \Delta_{les}, \kappa y_w), \quad (1.30b)$$

$$\Delta_{max} = \max(\Delta x_p, \Delta y_p, \Delta z_p)$$

где  $\kappa$  - постоянная Кармана ( $\kappa = 0.41$ ),  $y_w$  - расстояние до ближайшей стенки (она берется из переменной Решателя LDIS),  $C_w = 0.15$  - эмпирическая постоянная,  $\Delta_{max}$  - максимальный размер ячейки в направлении осей координат. Выбор модели расчета масштаба определяется оператором [Turbulence]:

ModelSGLS = SGLS\_VOL13 / SGLS\_SHUR2008

Квадратный корень из функции «вязкой диссипации»  $\sqrt{G}$  в Решателе сохраняется в переменной GFUN.

Интервал значений постоянной  $C_{sgs} = 0.065$  (плоский канал) - 0.23 (затухание изотропной турбулентности).

Если определено внутреннее поле пользователя с именем "iSGLS", то в нем сохраняется поле  $L_s$ .

### 1.6.2 Пристеночное демпфирование

При использовании модели Смагоринского для канальных течений возникает потребность введения локальных демпфирующих поправок, зависящих от  $y^+$ . В текущей версии кода для этого можно использовать демпфирующую функцию Пиомелли:

$$v_{sgs} = f_{damp} L_s^2 \cdot \sqrt{G}, \quad f_{damp} = 1 - \exp \left[ - \left( \frac{y^+}{25} \right)^3 \right] \quad (1.31)$$

Для расчета безразмерного расстояния до стенки используется соотношение:

$$y^+ = y_w \frac{\bar{u}_\tau}{\nu}, \quad \bar{u}_\tau = \sqrt{\frac{\bar{\tau}_w}{\rho}} \quad (1.31a)$$

Здесь  $\bar{\tau}_w$  - осредненное по участку стенки РО. Выбор такого участка зависит от используемой модели (см. пункт 2.11). При переменности свойств в качестве плотности и кинематическая вязкости в (1.31a) используются локальные значения!

Для активации демпфирующей функции используется оператор секции [Turbulence]:  
IsDumpPiomelli = .TRUE.

Для работы модели необходимо активировать переменную LDIS и:

1) либо алгоритм постоянного расхода  $G = \text{const}$  (пункт 2.9) для задач с периодическими граничными условиями (ПГУ),

2) либо для канальных течений без ПГУ необходимо для всех патчей стенок канала включить 1D режим осреднения (2.8.2).

Если определено внутреннее поле пользователя с именем "iYPLUS", то в нем сохраняется поле  $y^+$ .

### 1.6.3 Модель WALE

Для сложных задач возникает проблема корректного расчета  $y^+$ . В модели WALE (F. Nicoud, F. Ducros, 1999) для демпфирования используется модификация функции G. Новая G-функция при приближении к стенке быстро уменьшается ( $v_{sgs} \sim y^3$ ) и на стенке равна нулю. Подсеточная вязкость рассчитывается по соотношению:

$$v_{sgs} = L_s^2 \cdot \sqrt{G}, \quad G = \left[ \frac{S_d^{3/2}}{S^{5/2} + S_d^{5/4}} \right]^2, \quad L_s = C_w \Delta, \quad \Delta = (\Delta V_p)^{1/3}$$

$$S = \sum_i \sum_j S_{ij} \cdot S_{ij}, \quad S_d = \sum_i \sum_j S_{ij}^d \cdot S_{ij}^d, \quad (1.32a)$$

$$S_{ij}^d = \sum_k S_{ik} S_{kj} + \sum_k \Omega_{ik} \Omega_{kj} - \frac{1}{3} \delta_{ij} \left( \sum_k \sum_n S_{nk} S_{kn} + \sum_k \sum_n \Omega_{nk} \Omega_{kn} \right)$$

где :

$$S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right), \quad \Omega_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right) \quad (1.32b)$$

Характерное значение постоянной  $C_{sgs} = C_w = 0.5$  для изотропной турбулентности и  $C_w = 0.325$  (для канальных течений). Настройка модели:

```
[Macro sub]
ModelTur(trSgsLES)
....
[Turbulence]
LesMODEL = lesWALE
CtConstSGS = 0.325
.....
```

### 1.6.4 Модель CSM Кобаяши

Эта модель по сути аналогична модели WALE. Coherent structure Smagorinsky Model (CSM) предложена в работе Kobayashi [14]. Подсеточная вязкость рассчитывается по модели Смагоринского, в которой постоянная  $C_s$  заменяется функцией  $C_{CSM}$

$$v_{sgs} = (C_{csm} \Delta V^{1/3})^2 \cdot \sqrt{G},$$

$$C_{csm} = \sqrt{\frac{1}{22} |F_{CS}|^{3/2} (1 - F_{CS})}, \quad F_{CS} = \frac{Q}{E}, \quad (1.33)$$

$$Q = \frac{1}{2} [\Omega_{i,j} \Omega_{i,j} - S_{i,j} S_{i,j}], \quad E = \frac{1}{2} [\Omega_{i,j} \Omega_{i,j} + S_{i,j} S_{i,j}]$$

Настройка модели:

```
[Macro sub]
ModelTur(trSgsLES)
....
[Turbulence]
LesMODEL = lesCSM
```

### 1.6.5 DNS модель

В этой модели  $\nu_{sgs} = 0$ . Она позволяет проводить прямое численное моделирование .

Настройка модели:

```
[Macro sub]
```

```
ModelTur(trSgsLES)
```

```
....
```

```
[Turbulence]
```

```
LesMODEL = lesPDNS
```

## 2. Работа с LES моделью

### 2.1 Настройка

Для настройки модели используются две секции [Turbulence] и [Special Data] и макро-оператор:

ModelTur(trSgsLES)

или

ModelTur(trLesUserLES)

Первая форма оператора используется для активации внутренних моделей ANES, вторая - для реализации своей собственной LES модели.

Операторы секции [Turbulence]:

LesMODEL = lesSMAGORIN / lesWALE / lesCSM / lesPDNS

- оператор выбора модели подсеточной вязкости, по умолчанию используется модель Смагоринского.

CtConstSGS = <значения  $C_s$  или  $C_w$ >

- оператор задания постоянной  $C_s$  или  $C_w$ , по умолчанию  $C_s = 0.065$ ,  $C_w = 0.325$ .

CtPranTurSGS = <значение  $Pr_{sgs}$ >

- оператор задания подсеточного числа Прандтля, по умолчанию  $Pr_{sgs} = 0.85$ .

IsCalcRANS = .FALSE. / .TRUE.

- включение режима осреднения.

IsFAVRE = .FALSE. / .TRUE.

- использование осреднения по Фавре.

IsDumpPiomelli = .FALSE. / .TRUE.

- активация демпфирующей пристенной функции (1.31).

ModelSGLS = SGLS\_VOL13 / SGLS\_SHUR2008

- алгоритм расчета подсеточного масштаба длины (1.30).

### 2.2 Особенности проведения расчетов

При проведении LES моделирования необходимо:

- 1) использовать схемы второго порядка точности по пространству, например, схему TrueCD;
- 2) использовать схему второго порядка по времени - TTS\_3LEVEL;
- 3) для корректного использования пакетного постпроцессора при сохранении файлов результатов необходимо использовать сохранение в центрах ячеек: C("ARS.OUTCELL") = .TRUE.  
(если этот оператор не указан, он добавляется Решателем автоматически на этапе осреднения).

При использовании временной схемы TTS\_3LEVEL Решатель создает два файла рестарта с расширениями <Имя варианта>.аср и <Имя варианта>.аср00, чтобы обеспечить «безударный» расчет производных по времени.

В общем случае при переносе файлов для продолжения расчета на другом кластере или с другим числом ядер необходимы следующие файлы:

- < Имя варианта >.acr и < Имя варианта >.acr00 - файлы с полями Ф-переменных на двух шагах по времени,
- < Имя варианта >.sord - файл индексами ячеек для из пересчета на другое число ядер,
- < Имя варианта >.aldis - файл с сохраненными значениями поля LDIS,
- < Имя варианта >.tauasr - файл с распределение касательного напряжения вдоль оси z,
- < Имя варианта >.lesacr - файл с полями осреднения (используется , если IsCalcRans=.True).

Напомним две полезные возможности при работе на кластере:

1. Обычно при выполнении пакетного расчета пользователю доступен файл типа «output», в который накапливается консоль расчета. Для удобства контроля текущего времени расчета на кластере в строку шага по времени выводится значение этого времени в минутах. Для этого нужно включить оператор `L("IsCurTimeRUN")=.TRUE.`
2. Для уменьшения количества информации, выдаваемой в консоль можно указать шаг выдачи:  
`I("NoOutTime")=1 / <шаг по iTime>`

### 2.3 LES-модели пользователя

Как уже отмечалось, основой LES - модели является использование коэффициента подсеточной турбулентной вязкости вместо стандартной RANS турбулентной вязкости. В коде ANES реализован гибкий интерфейс для самостоятельного расчета подсеточной вязкости. Пользователь может задать любой алгоритм ее расчета, в том числе и на основе решения дополнительных уравнений переноса (DES модели). Важной особенностью такого подхода является то, что все средства обработки LES расчета, описанные ниже, доступны.

Для использования LES модели пользователя нужно задать оператор секции [Macro SUB]:

```
ModelTur(trLesUserLES)
```

и в секции [Turbulence] указать имя виртуальной функции LES модели:

```
TKEfun = Virtual(myLesModel)
```

Сама виртуальная функция имеет следующую структуру:

```
!=====
! Шаблон для LES модели   trLesUSER
!=====

!***** Секция [USER Fortran Variables] *****
! Здесь можно описать любые свои переменные и подпрограммы (после оператора CONTAINS)
!*****

!***** Секция [USER DATA] *****
! Здесь можно присвоить значения операторам:
!   RUDat(:) и IUDat(:)
! которые доступны в виртуальной функции
!*****

!***** Секция [Special DATA] *****
! Здесь можно использовать операторы этой секции:
!   L("Имя"), R("Имя"), C("Имя") и V("Имя")
! которые доступны в виртуальной функции через функции logical GetSpeDAT()
!*****

!***** Секция [Turbulence] *****
! Оператор описания виртуальной функции пользователя к модели:
```

```

!  TKEfun = Virtual(myLesUserVF)
!*****

!***** Сама Виртуальная Функция *****
![vf myLesUserVF]
!***** Сама Виртуальная Функция *****

subroutine myLesUserVF(RetDUMMY)
  USE acDATA
  USE acTurbDATA
  USE acLesTDATA
  USE usaDATA
  USE acIO
  USE acSUB   ! Это системные функции доступа (если они нужны)
  USE acPRINT

! ----- Интерфейс с Решателем -----
COMMON /CVF_FWALL/ iEntry,iPHI,Param1, Param2, RetCoef, RetVal
INTEGER iPHI, iEntry
real (4) RetVal, retCoef,Param1,Param2
! ----- Интерфейс с Решателем -----

  SELECT CASE(iEntry)
!=====
  CASE(ULES_INIT)
!=====
! Вход-Событие: Чтение, выделение памяти, инициализация любых данных пользователя
! Вся инициализация Решателя уже проведена (включая чтение файлов рестарта)
!=====

!=====
  CASE(ULES_PRINT)
!=====
! Вход-Событие: Любая Печать
! Можно использовать либо функции LWriteTitle/LWrite модуля acPrint
! либо write(LU_LST,*) , LU_LST = LUnit(U_LST)
!=====
!~ CALL LWriteTitle(40,'Модель KDES Абрамова')
!~ CALL LWrite('Ceps','vC_EPS,')

!=====
  CASE(ULES_BeforeSWEEP)
!=====
! Вход-Событие: Перед выполнением итерации, но после всех остальных расчетов Решателя.
! перед вызовом этой точки входа рассчитывается поле usGFUN = 2 *Sij * Sij
! Можно проводить любые расчеты. Для перебора всех ячеек, включая HALO
! нужно использовать следующий код:
!=====
  do iCL = 1,ABS(NoclFlowList)
    if(LGetCVID(iCL,.TRUE.,idCV)) exit ! Плюс HALO
! ----- Расчет для ячейки с индексом idCV -----
! .....
  end do

!=====
  CASE(ULES_NUSGS)
!=====
! Виртуальная Функция: расчет подсеточной вязкости Для текущей ячейки с индексом idCV
! Значение кинематической вязкости присваивается переменной RetVal
! Эта функция вызывается ПОСЛЕ точки входа ULES_BeforeSWEEP
! Здесь можно использовать много функций - см. исходный код acTurbLES.f90 !!!!
!=====

  RetVal = ???

```



```

=====
CASE(ULES_BeforeTime)
=====
! Вход-Событие: Перед началом нового шага по времени.
=====

=====
case DEFAULT
=====
  RetVal = 0.0
  RetCoef = 0.0
  END SELECT

end subroutine

```

Примеры написания своих LES моделей можно найти в библиотеке примеров ANES.

## 2.4 Особенности работы с уравнением энергии

При решении задач с периодическими граничными условиями (стабилизированные течения в каналах) физический смысл имеет не сама температура  $T_g$ , а разница  $\theta_g = T_g - T_m$ , где  $T_m$  - это среднемассовая температура, рассчитанная на *периодической* границе:

$$T_m|_{z=0} = \frac{1}{\int_A \rho u_n dA} \int_A \rho u_n T_g dA$$

Для работы с переменной  $\theta$  в *сенсорах* и в *осредненных переменных* используется оператор секции [Special Data]:

```
L("LES.IsMinusTBULK") = .TRUE.
```

Если этот флаг активирован, то вместо  $T_g$  будет использоваться  $\theta_g$  в:

- 1) в сенсорах,
- 2) в значении граничного патча (оператор "Tw"),
- 3) в *осредненной*  $\Phi$ -переменной "TG" и ее корреляциях.

При этом в файле результатов всегда сохраняется мгновенное поле  $\Phi$ -переменной  $T_g$ .

В новой версии кода добавлена возможность решения уравнения энергии, в которой  $\Phi$ -переменной является энтальпия  $h$ , а температура становится свойством  $T = f(h)$ . В этом случае уравнение энергии записывается в виде:

$$\frac{\partial(\rho \bar{h})}{\partial \tau} + \frac{\partial}{\partial x_k} \left( \rho \bar{u}_k \bar{h} - \frac{\lambda}{c_p} \nabla \bar{h} - \mathbf{q}_k^{sgs} \right) = S_h,$$

$$\mathbf{q}_k^{sgs} = -\frac{\mu_{sgs}}{Pr_{sgs}} \nabla \bar{h}$$

Эту форму уравнения нельзя использовать для стабилизированных задач. Основная область применения - задачи с сильно переменными свойствами.

При наличии в расчетной области Struct-зон также нельзя использовать уравнение энергии, записанное через энтальпию.

## 2.5 Начальные условия

При решении задач с ПГУ необходимо задать возмущенные начальные поля. В ANES для этого предусмотрено два алгоритма:

В первом алгоритме нужно на первом этапе нужно задать с помощью операторов Init() секции [PHI Variables] распределения полей (если оператор не задан - то нулевые значения). На втором этапе на эти поля накладываются случайные возмущения. Для этого используются операторы

R("LESFLUCT.<Ф-имя>") = E

Если значение E положительно, то поле Ф-переменной модифицируется по соотношению

$$\Phi = \Phi \cdot (1 - E \cdot r_{01}()),$$

если E отрицательна, то

$$\Phi = \Phi + E \cdot (r_{01}() - 0.5)$$

где  $r_{01}()$  - случайная функция в интервале (0,1) с равномерным распределением вероятности.

Во втором алгоритме для задания начального распределения используются поля, полученные в независимом стационарном расчете. Такой расчет можно провести с использованием модели одномерного стабилизированного течения (в декартовой или цилиндрической геометрии) или двумерного стабилизированного течения в плоскости (x,y). Если при расчете указать оператор:

L("CReateDEVVEL") = .True.

то будут созданы одномерные или двумерные файлы \*\_ugz.dat и \*\_ktur.dat, которые и будут использоваться в данном алгоритме. Начальные поля задаются по соотношениям:

$$u_z = U_0 + C_k \sqrt{\frac{2k_0}{3}} r_g, \quad u_x = C_k \sqrt{\frac{2k_0}{3}} r_g, \quad u_y = C_k \sqrt{\frac{2k_0}{3}} r_g, \quad p = 0,$$

$U_0 = U_0(x)$ ,  $k_0 = k_0(x)$  – одномерный стационарный расчет,

$U_0 = U_0(x, y)$ ,  $k_0 = k_0(x, y)$  – двумерный стационарный расчет

Здесь  $r_g$  - случайная функция с гауссовым распределением (с средним значением = 0 и дисперсией = 1).

Если рассчитывается уравнение энергии (Ф- переменная TG) и имеется файл \*\_tg.dat, то поле  $T_g$  будет проинициализировано значениями из этого файла (без возмущений).

Допускается использование «псевдополярной» геометрии. В этом случае используется одномерное стационарное решение, но координата "x" трактуется как радиус относительно центра ( $x_c, y_c$ ):

$$U_0 = U_0(r), \quad k_0 = k_0(r), \quad r = \sqrt{(x - x_c)^2 + (y - y_c)^2}$$

Активация и настройка этого алгоритма осуществляется с помощью операторов:

C("LES.RANSIni") = "1DX" / "1DR" / "2DXY"

C("LES.PrefIniFiles") = <путь И префикс файлов dat>

V("PolarCenter") = vec(xC,yC,0)

R("LES.CoeffKTUR") = 5.0 / CoeffKTUR

V("LES.OffXY") = vec(xOff,yOff,0)

Оператор "LES.RANSIni" задает режим использования координат стационарного расчета. Префикс файлов со стационарными полями задается оператором "LES.PrefIniFiles".

Активация «псевдополярной» геометрии происходит, если задан вектор центра (оператор "PolarCenter").

Замечание: В имени центра ПсевдоПолярной СК нет символов "LES.", поскольку этот оператор используется и для расчета внутренних полей пользователя с именами «UGR» и «UGteta».

Коэффициент  $C_k$  задается оператором "LES.CoeffKTUR".

При использовании модели "2dXY" можно задать сдвиг начала прямоугольника Flow-зоны с помощью оператора V("LES.OffXY"). В этом случае

$$U_0 = U_0(x - x_{\text{off}}, y - y_{\text{off}}), k_0 = k_0(x - x_{\text{off}}, y - y_{\text{off}}).$$

Этот оператор удобно использовать при наличии Struct-стенок.

При использовании гибридной модели (trSST\_IDDES) необходимо задать распределения для турбулентной энергии  $k$  и удельной скорости диссипации  $\omega$ . В этом случае для создания «стационарных» распределений необходимо использовать  $k$ - $\omega$  или SST модель турбулентности и будет дополнительно создан файл \*\_omtur.dat. Распределение  $\omega_0$  из этого файла будет использоваться для инициализации диссипации  $\omega$  во всей РО (без возмущений). Для задания начального распределения  $k$  можно использовать два подхода. В первом подходе распределение  $k$  берется из стационарного распределения  $k_0$ . Во втором подходе поле  $k$  рассчитывается из соотношения:

$$k = \nu_{t0} \cdot \omega_0,$$

где  $\nu_{t0}$  - постоянное значение турбулентной кинематической вязкости, определяемое оператором:

R("LES.ViscTurlni") = <значение вязкости>

## 2.6 Возможности LES обработки

Для удобства представления результатов моделирования в Решатель добавлены различные модели обработки результатов расчета. В этом разделе приведено краткое описание этих моделей, более подробно они описаны ниже.

Модель	Оператор	Описание
Балансы	<b>Energy</b>	Вывод в текстовый файл *_energy.timeles основных интегральных параметров РО: $E_k$ , $U_m$ , $T_m$ , $T_v$ , $Re_\tau$ . Используется для оценки выхода на квазистационарный режим [п. 2.7].
Сенсоры	<b>Sensor</b>	Выбираются точки в РО, для которых на каждом шаге по времени создается текстовый файл *.timeles со значениями всех $\Phi$ -переменных [п. 2.6]. В LES-постпроцессоре можно использовать для статистического анализа.
Скаляры	<b>LESSCAL</b>	Сохранение <i>осредненных по поверхности</i> граничного или FS патча параметров $q_w$ , $T_w$ , $\tau_x$ , $\tau_y$ , $\tau_z$ . Сама привязка осуществляется через скалярную переменную пользователя. Мгновенные значения записываются в один текстовый файл *_scalarg.timeles на каждом шаге по времени. В режиме осреднения <i>осредненные по времени</i> значения записываются в файл результатов как значения скаляров пользователя. Заметим, что эти средние значения можно получить и усредняя мгновенные значения [п. 2.8].
Z-параметры	<b>RANS.IsBparamZ</b>	Режим является «развитием» режима Energy. В этом режиме рассчитываются <i>осредненные по Z-сечению</i> распределения $\rho(z)$ , $G(z)$ , $U_m(z)$ , $T_m(z)$ и $h_m(z)$ . В режиме расчета сохраняются мгновенные значения, в режиме осреднения мгновенные и осредненные значения. Указанные распределения сохраняются в текстовом файле с именем *_bparamz_XX.dat [п. 2.10.4].
Z-поля	<b>RANS.Patch1D</b>	Режим является развитием LESSCAL. В этом режиме для

		указанного граничного патча производится осреднение по граням патча для каждого z-сечения величин $\tau_z$ и $T_w$ . На шаге создания файлов результатов создается текстовый файл с зависимостями $\tau_z(z)$ и $T_w(z)$ и с именем *_patch1d_<ИмяПатча>_XX.dat. В режиме расчета сохраняются мгновенные значения, в режиме осреднения - осредненные по времени [п. 2.10.2]. При наличии Struct-областей и использовании FS-патча дополнительно рассчитывается плотность теплового потока в F-фазу - $q_w(z)$ .
ZXY-поля	RANS.Patch2D	Развитие RANS.Patch1D. В этом режиме сохраняются двумерные распределения по патчу. Эти поля ( $\tau_z(xu, z)$ и $T_w(xu, z)$ ) сохраняются в бинарном файле с именем *_patch2d_<ИмяПатча>_XX.dat [п. 2.10.3]. Для его обработки необходимо использовать LES-постпроцессор [п. 4.5]. При наличии Struct-областей и использовании FS-патча дополнительно рассчитывается плотность теплового потока в F-фазу - $q_w(xu, z)$ .

## 2.7 Мониторинг в процессе расчета. Сенсоры

Пользователь может задать до 16 точек в расчетной области, для которых будут выводиться значения всех Ф-переменных. Точки задаются оператором секции [Special Data]:

```
V("SENSOR.ИмяТочки") = vec(xP,yP,zP)
```

Здесь (xP,yP,zP) - координаты точки, <ИмяТочки> используется для задания имени файла в виде:

```
<ПрефиксРезультатов>_<ИмяТочки>.timeles
```

В новом расчете (StartMode = SM\_NEW) или расчете с другого варианта (StartMode = SM\_RES) файл создается заново и в первую строку записывается заголовок. При старте с КТ (StartMode = SM\_CP) файл дозаписывается. По умолчанию значения Ф-переменных записываются на каждом шаге. Шаг записи можно изменить оператором:

```
I("LES.StepLESOUT") = <шаг>
```

Значение температуры в точке сенсора зависит от оператора

```
L("LES.IsMinusTBULK") = .FALSE. / .TRUE.
```

если значение .False., то используется непосредственно  $T_g$  (и  $T_s$ ). Если значение оператора .True., то используется разность  $\theta_g = T_g - T_m$  (и  $T_s - T_m$ ).

## 2.8 Мониторинг в процессе расчета. Средние параметры по РО

Для анализа процесса выхода на квазистационар создается файл с именем:  
<ПрефиксРезультатов>\_energy.timeles

В этот файл записываются значения следующих величин:

$$E_k = \frac{1}{V_{PO}} \iiint_{PO} \frac{u_x^2 + u_y^2 + u_z^2}{2} dV \quad - \text{средняя механическая энергия,}$$

$$T_v = \frac{1}{V_{PO}} \iiint_{PO} T_g dV \quad - \text{средняя температура,}$$

$$U_m = U_b = \frac{1}{A} \int_A u_n dA \quad - \text{средняя скорость через поверхностный патч } A,$$

$$T_m = T_b = \frac{1}{\int_A \rho u_n dA} \int_A \rho u_n T_g dA \quad - \text{среднемассовая температура через поверхностный}$$

патч  $A$ ,

$$Re_\tau = \frac{1}{\nu} \sqrt{\frac{\bar{\tau}_w}{\rho}} \quad - \text{характерное значение числа Рейнольдса, рассчитанное}$$

по скорости трения. Эта величина рассчитывается, если используется модель заданного расхода (п. 2.9).

Здесь  $\bar{\tau}_w$  - среднее трение по всем поверхностям, описываемым патчами с флагом Wall.

Имя поверхностного патча  $A$  типа SURFACE необходимо задать оператором  $C("LES.UMPATCH") = \text{"Имя патча } A"$

Если используются ПГУ, то в качестве такого патча можно использовать патч, создаваемый Компилятором для реализации ПГУ, например, CYCLE\_BS\_Z".

Если заданы скалярные переменные пользователя с именами "EK", "UM", "TM", , "TV" то в них записываются значения  $E_k$ ,  $U_m$ ,  $T_m$ ,  $T_v$ . Можно использовать дополнительные имена-алиасы : "Ubulk" = "UM" и "Tbulk" = "TM" и "Tvol" = "TV". Кроме того значения  $T_m$  и  $T_v$  сохраняется в переменных myForm Решателя с именами DEV\_TBULK и DEV\_TVOL.

Для задач с замкнутой расчетной областью, в которой отсутствуют входные и выходные границы (или периодические границы), патч  $A$  можно не указывать. В этом случае рассчитываются только переменные  $E_k$  и  $T_v$ , а остальные переменные равны нулю.

**Если в уравнении энергии  $\Phi$ -переменной является энтальпия  $h$ , то в файл и в переменные пользователя вместо TM и TV записываются другие поля:**

$$h_v = \frac{1}{\iiint_{PO} \rho dV} \iiint_{PO} \rho h_g dV \quad - \text{средняя энтальпия (имя "HV" или "HVOL"),}$$

$$h_m = h_b = \frac{1}{\int_A \rho u_n dA} \int_A \rho u_n h_g dA \quad - \text{среднемассовая энтальпия (имя "HM" или "HBULK")}$$

## 2.9 Мониторинг в процессе расчета. Средние параметры на патчах

В файлах результатов можно сохранить некоторые интегральные параметры. Для этого необходимо:

- 1) завести скалярную переменную с произвольным именем, например <Uname>,
- 2) добавить оператор алгоритма расчета этого скаляра:  
 $C("LESSCAL.<Uname>") = \text{"Формула.ИмяПатча"}$

Здесь <ИмяПатча> - имя граничного или FS патча, на котором нужно рассчитывать интегральную величину. Параметр <Формула> определяет алгоритм расчета:

$$\text{"Qw"} \quad - \text{средний по поверхности тепловой поток } q_w = - \left( \lambda_g + \frac{\rho_g c_{pg} v_{sgs}}{Pr_{sgs}} \right) \nabla T_g \cdot \mathbf{n}_b ,$$

"Tw" - средняя по поверхности температура ( $T_g$  или  $\theta_g$ ),

"Hw" - средняя по поверхности энтальпия (если решается уравнение

энергии через энтальпию)

"TauX", "TauY", "TauZ" - средние x,y,z-компоненты вектора касательного напряжения :

$$\boldsymbol{\tau}_w = \rho (v_g + v_{sgs}) \frac{\mathbf{U}_\tau}{\delta_w}, \quad \mathbf{U}_\tau = \mathbf{u}_p - |\mathbf{u}_p \cdot \mathbf{n}_b| \cdot \mathbf{n}_b$$

Здесь  $\mathbf{u}_p$  - вектор скорости в приграничном узле,  $\mathbf{n}_b$  - нормаль к границе (направленная внутрь РО),  $\delta_w$  - расстояние от центра ячейки до граничной грани.

Для мониторинга интегральных параметров на граничных патчах в процессе расчета (и для более детального временного анализа в постпроцессоре) можно выводить их значения в файл на каждом шаге по времени. Для этого используется файл с именем

<ПрефиксРезультатов>\_scalar.timeles

Для создания такого файла необходимо использовать оператор:

L("LES.IsScalarTime") = .True.

Замечания:

1. При использовании режима осреднения в файл timeles записываются *мгновенные* значения интегральных параметров, а в файлы результатов - *осредненные!*

2. При решении уравнения энергии через температуру, то значение температуры на патче (формула "Tw") зависит от оператора L("LES.IsMinusTBULK") = .FALSE. / .TRUE.

если значение .False., то используется непосредственно  $T_g$ . Если значение оператора .True., то используется разность  $\theta_g = T_g - T_m$ . При решении уравнения энергии через энтальпию этот оператор игнорируется.

## 2.10 Мониторинг в процессе расчета. 3d поля

В Решателе предусмотрены встроенные алгоритмы для расчета некоторых сеточных переменных пользователя (они должны быть объявлены как внутренние):

VortX, VortY, VortZ - компоненты вектора завихренности ( $\boldsymbol{\omega} = \text{rot}(\mathbf{u})$ ),

VortMOD - модуль вектора завихренности,

Qcrit - скалярное поле Q критерия.

Поле Qcrit рассчитывается по соотношению:

$$Q = \frac{1}{2} \left[ \sum_{i=1}^3 \sum_{k=1}^3 \Omega_{i,k}^2 - \sum_{i=1}^3 \sum_{k=1}^3 S_{i,k}^2 \right]$$

и представляет собой второй инвариант тензора градиента скорости.

## 2.11 Режим осреднения

Если необходимо провести осреднение *по времени* мгновенных полей, то необходимо проводить *новый расчет*, считав начальные распределения с предыдущего расчета (StartMode = SM\_RES) и установить флаг в секции [Turbulence]:

IsCalcRANS = .True.

Для сохранения осредненных полей используются *внутренние* сеточные поля пользователя. Их имена могут быть произвольными. Для задания алгоритма осреднения данной U-переменной используются операторы:

C("RANS.U-имя") = "Фк"

C("RANS.U-имя") = "Фк\*Фm"

C("RANS.U-имя") = "DENS"

C("RANS.U-имя") = "HG"

C("RANS.U-имя") = "TG"

C("RANS.U-имя") = "CP"

C("RANS.U-имя") = "DUX\_DX"

C("RANS.U-имя") = "DUX\_DY"

C("RANS.U-имя") = "DUX\_DZ"

C("RANS.U-имя") = "DUY\_DX"

C("RANS.U-имя") = "DUY\_DY"

C("RANS.U-имя") = "DUY\_DZ"

C("RANS.U-имя") = "DUZ\_DX"

C("RANS.U-имя") = "DUZ\_DY"

C("RANS.U-имя") = "DUZ\_DZ"

C("RANS.U-имя") = "DT\_DX"

C("RANS.U-имя") = "DT\_DY"

C("RANS.U-имя") = "DT\_DZ"

C("RANS.U-имя") = "SGS\_TXY"

C("RANS.U-имя") = "SGS\_TXZ"

C("RANS.U-имя") = "SGS\_TYZ"

C("RANS.U-имя") = "SGS\_TXX"

C("RANS.U-имя") = "SGS\_TYY"

C("RANS.U-имя") = "SGS\_TZZ"

C("RANS.U-имя") = "SGS\_QX"

C("RANS.U-имя") = "SGS\_QY"

C("RANS.U-имя") = "SGS\_QZ"

C("RANS.U-имя") = "DissTEM"

Здесь  $\Phi_k$  и  $\Phi_m$  - имена  $\Phi$ -переменных расчета. Формулы для осреднения зависят от флага IsFAVRE секции [Turbulence]. Если флаг отключен (IsFAVRE = .False.), то используется обычное осреднение по времени

$$\bar{\Phi} = \frac{\sum_{i=1}^{N_T} \Phi_i}{N_T} \quad (2.1)$$

Если флаг включен (IsFAVRE = .True.), то используется осреднение Фавре:

$$\tilde{\Phi} = \frac{\sum_{i=1}^{N_T} \rho_i \Phi_i}{\rho N_T} = \frac{\sum_{i=1}^{N_T} \rho_i \Phi_i}{\sum_{i=1}^{N_T} \rho_i} \quad (2.2)$$

Важное замечание: при использовании осреднения Фавре необходим оператор

C("RANS.U-имя") = "DENS"

с произвольным U-имя (этот оператор необходим для расчета средней плотности).

При использовании первой формы

C("RANS.U-имя") = "Фк"

вычисляется среднее значение (2.1) или (2.2)  $\Phi$ -переменной. При использовании второй формы

C("RANS.U-имя") = "Фк\*Фм"

рассчитывается корреляция произведения пульсаций  $\Phi$ -переменных:

$$\overline{\Phi'_i \Phi'_k} = \overline{(\Phi_i - \bar{\Phi}_i)(\Phi_k - \bar{\Phi}_k)} = \overline{\Phi_i \Phi_k} - \bar{\Phi}_i \bar{\Phi}_k, \quad \text{или} \quad (2.3)$$

$$\widetilde{\Phi''_i \Phi''_k} = \widetilde{\Phi_i \Phi_k} - \tilde{\Phi}_i \tilde{\Phi}_k$$

Смысл остальных форм операторов осреднения следующий:

C("RANS.U-имя") = "DENS"	:	$U = \bar{\rho},$
C("RANS.U-имя") = "HG"		$U = \tilde{h},$
C("RANS.U-имя") = "TG"		$U = \bar{T},$
C("RANS.U-имя") = "CP"		$U = \bar{c}_p$
C("RANS.U-имя") = "SGS_TXY"		$U = 2 \cdot \overline{v_{sgs} S_{xy}}$
C("RANS.U-имя") = "SGS_TXZ"		$U = 2 \cdot \overline{v_{sgs} S_{xz}}$
C("RANS.U-имя") = "SGS_TYZ"		$U = 2 \cdot \overline{v_{sgs} S_{yz}}$
C("RANS.U-имя") = "SGS_TXX"		$U = 2 \cdot \overline{v_{sgs} S_{xx}}$
.....		
C("RANS.U-имя") = "DUX_DK"		$U = \frac{\overline{\partial u_x}}{\partial x_k}, \quad k = x, y, z$
.....		
C("RANS.U-имя") = "DT_DK"		$U = \frac{\overline{\partial T}}{\partial x_k}, \quad k = x, y, z$
.....		
C("RANS.U-имя") = "SGS_QX"		$U = \frac{\overline{v_{sgs} c_p \partial T}}{\text{Pr}_{sgs} \partial x}$
C("RANS.U-имя") = "SGS_QY"		$U = \frac{\overline{v_{sgs} c_p \partial T}}{\text{Pr}_{sgs} \partial y}$
C("RANS.U-имя") = "SGS_QZ"		$U = \frac{\overline{v_{sgs} c_p \partial T}}{\text{Pr}_{sgs} \partial z}$
C("RANS.U-имя") = "DissTEM"		$U = \left( \frac{\overline{\partial T'}}{\partial x_k} \right)^2$

Смысл переменных "HG" и "TG" зависит от  $\Phi$ -переменной уравнения энергии. Переменная "HG" используется, если решается уравнение для TG, а переменная "TG" - если решается уравнение для энтальпии.

Для возможности рестарта вместе с файлом \*.acr сохраняется файл с расширением \*.lesacr.

Для полного рестарта с продолжением осреднения необходим этот файл. Если этого файла нет, то осреднение начнется сначала!

Замечание: при решении уравнения энергии через температуру смысл T в осредняемых переменных зависит от оператора:

$$L("LES.IsMinusTBULK") = .TRUE.$$



Если оператор активирован, то *при осреднении* вместо  $T_g$  используется разность  $\theta_g = T_g - T_m$ . При этом  $\Phi$ -переменная TG в файлах результатов естественно содержит рассчитанное мгновенное значение  $T_g$ .

При решении уравнения энергии через энтальпию этот оператор игнорируется.

Для температуры Ts (Struct-фаза) возможен расчет среднего значения и интенсивности ее пульсаций.

### 2.11.1 Режим осреднения: псевдополярная СК

Если активирована псевдополярная СК (задан оператор V("PolarCenter")), то можно провести осреднение дополнительных полей:

$$C("RANS.U-имя") = "UgR" \quad : \quad U = u_r,$$

$$C("RANS.U-имя") = "UgTeta" \quad : \quad U = u_\theta,$$

$$C("RANS.U-имя") = "DT_DR" \quad : \quad U = \frac{\overline{\partial T}}{\partial r},$$

$$C("RANS.U-имя") = "DT_DTeta" \quad : \quad U = \frac{1}{r} \frac{\partial T}{\partial \theta},$$

$$C("RANS.U-имя") = "DUR_DR" / "DUR_DTeta" / "DUR_DZ" : U = \frac{\overline{\partial u_r}}{\partial r}, \quad \frac{1}{r} \frac{\overline{\partial u_r}}{\partial \theta}, \quad \frac{\overline{\partial u_r}}{\partial z},$$

$$C("RANS.U-имя") = "DUTeta_DR" / "DUTeta_DTeta" / "DUTeta_DZ" : U = \frac{\overline{\partial u_\theta}}{\partial r}, \quad \frac{1}{r} \frac{\overline{\partial u_\theta}}{\partial \theta}, \quad \frac{\overline{\partial u_\theta}}{\partial z},$$

$$C("RANS.U-имя") = "DUZ_DR" / "DUZ_DTeta" / "DUZ_DZ" : U = \frac{\overline{\partial u_z}}{\partial r}, \quad \frac{1}{r} \frac{\overline{\partial u_z}}{\partial \theta}, \quad \frac{\overline{\partial u_z}}{\partial z}$$

$$C("RANS.U-имя") = "SGS_QR" \quad U = \frac{\overline{v_{sgs} c_p \partial T}}{Pr_{sgs} \partial r}$$

$$C("RANS.U-имя") = "SGS_QTETA" \quad U = \frac{\overline{v_{sgs} c_p}}{Pr_{sgs}} \frac{1}{r} \frac{\partial T}{\partial \theta}$$

$$C("RANS.U-имя") = "SGS_TRTETA" \quad U = 2 \cdot \overline{v_{sgs} S_{r\theta}}, \quad S_{r\theta} = \frac{1}{2} \left[ \frac{1}{r} \frac{\partial u_r}{\partial \theta} + \frac{\partial u_\theta}{\partial r} - \frac{u_\theta}{r} \right]$$

$$C("RANS.U-имя") = "SGS_TRZ" \quad U = 2 \cdot \overline{v_{sgs} S_{rz}}, \quad S_{rz} = \frac{1}{2} \left[ \frac{\partial u_r}{\partial z} + \frac{\partial u_z}{\partial r} \right]$$

$$C("RANS.U-имя") = "SGS_TZTETA" \quad U = 2 \cdot \overline{v_{sgs} S_{z\theta}}, \quad S_{z\theta} = \frac{1}{2} \left[ \frac{1}{r} \frac{\partial u_z}{\partial \theta} + \frac{\partial u_\theta}{\partial z} \right]$$

Одновременно имена "UgR" и "UgTeta" можно использовать в качестве имен  $\Phi$ -переменных в двойных корреляциях

$$C("RANS.U-имя") = "Фк*Фт"$$

вместе с остальными  $\Phi$ -переменными.

### 2.11.2 Режим осреднения: $F(z)$ распределения по граничным патчам

При решении нестабилизированных канальных задач (без периодических граничных условий) возникает проблема осреднения только по периметру сечения канала и времени. В текущей версии производится осреднение *касательного напряжения*  $\tau_z$  и *температуры*  $T_w$  на *поверхности граничного или FS патча*. Если уравнение энергии решается через энтальпию дополнительно рассчитывается и значение средней энтальпии  $h_w$ . Для FS-патча дополнительно рассчитывается плотность теплового потока  $q_w$ .

Этот режим можно использовать только, если нет дробления по оси Z (IsNoRefZ = .True.). Для активации осреднения на *граничном патче* с именем <Имя Патча> необходимо использовать оператор секции [Special Data]:

C("RANS.PATCH1D") = <Имя Патча>

В этом случае для каждого сечения по координате Z проводится поверхностное осреднение по граням патча с одинаковым значением Z. Если включен режим осреднения (IsCalcRANS = .True.) дополнительно производится *осреднение и по времени*. Распределения Z,  $\tau_z$ ,  $T_w$  и  $q_w$  выводятся на шаге *создания файлов результатов* в файл с именем (XX - номер шага по времени):

<ПрефиксРезультатов>\_patch1d\_<ИмяПатча>\_XX.dat

При расчете распределений всегда используется все точки по z (1 - NFZ). Сечения по z, которые не входят в патч будут заполнены нулями.

При моделировании канальных течений распределения мгновенных значений  $\tau_{z\_mid}(z)$  по всем граничным патчам используются для расчета  $Y^+$  в моделях пристенного демпфирования (например, в модели Пиомелли 1.5.2). Для расчета  $\tau_{z\_mid}(z)$  используется соотношение:

$$\tau_{w\_mid}(z) = \frac{\sum_k \tau_{w,k}(z) A_k(z)}{\sum_k A_k(z)}, \quad (2.5)$$

где  $\tau_{w,k}$  - мгновенное значение трения,  $A_k(z)$  - площадь «полоски» граней для k-ого патча. В режиме расчета (IsCalcRANS = .False.) это значение выводится во все dat-файлы четвертой колонкой.

Заметим, что этот режим осреднения можно использовать как для осесимметричной задачи (в этом случае патчи расположены друг за другом по оси z), так и в случае прямоугольного или треугольного канала. В последнем случае патчи будут располагаться как по оси z, так и в сечениях по z. Алгоритм суммирования позволяет для этих случаев получить вполне разумные значения  $\tau_{w\_mid}$  для  $Y^+$ .

### 2.11.3 Режим осреднения: $F(z,xy)$ распределения по граничным патчам

Для не осесимметричных задач (течение в прямоугольном канале, течение в горизонтальной трубе при наличии магнитного поля или сил плавучести) осредненные по времени параметры на *граничном патче* (трение на стенке и температура стенки) зависят не только от координаты z, но и от «перпендикулярных» координат (x, y или  $\theta$ ). Для канальных течений, отсутствия разбиения по оси z (IsNoRefZ = .True.) и патчей, созданных с помощью операторов Patch и 2dPatch, поверхность граничного патча представляет собой набор прямоугольных граней, расположенных на структурной прямоугольной сетке в «плоскости» (z,x), (z,y) или (z,  $\theta$ ). В этом случае можно вывести осредненные по времени значения касательного напряжения  $\tau_z$ , температуры стенки  $T_w$  (и энтальпии  $h_w$  при решении уравнения энергии для HG) в виде обычного двумерного массива. В качестве «поверхностных» координат используется координата z и одна из «перпендикулярных» координат.

Для активации этого режима используется оператор секции [Special Data]:

C("RANS.PATCH2D") = <Dir2.Имя Патча>

Здесь Dir2 = X,Y,Teta - вторая координата. Координату Teta можно использовать только, если задан режим псевдополярной системы (задан оператор V("PolarCenter")). В этом случае в качестве координаты используется угол  $\theta$ , отсчитываемый от оси x против часовой стрелки.

Распределения  $\tau_z(Z,Dir2)$  и  $T_w(Z,Dir2)$  выводятся на шаге создания файлов результатов в двоичный файл с именем (XX - номер шага по времени):

<ПрефиксРезультатов>\_patch2d\_<ИмяПатча>\_XX.dat

Этот файл имеет следующую структуру:

```
"$PATCH2D Z Dir2$"      ! первая строка -16 символов (Dir2 = "X ", "Y ", "TETA")
NDir2 , NZ               ! Число узлов сетки по Z и Ndir2 (integer)
Dir2(1 ... Ndir2)       ! Массив центров граней по «перпендикулярной» оси (real(4))
Z(1 ... NZ)             ! Массив центров граней по оси Z (real(4))
Do iz = 1,NZ
  Do ix = 1,NDir2
    TauWall(ix, iz)     ! Значения TauWall (real(4))
  End do
End do
Do iz = 1,NZ
  Do ix = 1,NDir2
    TWall(ix, iz)      ! Значения Tw в сечениях по Z
  End do
End do
IsHWALL = 0/1
Do iz = 1,NZ
  Do ix = 1,NDir2
    HWall(ix, iz)     ! Значения Hw в сечениях по Z
  End do
End do
IsQWALL = 0/1
Do iz = 1,NZ
  Do ix = 1,NDir2
    QWall(ix, iz)     ! Значения qw в сечениях по Z
  End do
End do
```

Для обработки этого файла можно использовать пакетный LES-постпроцессор (оператор Patch2D). Плотность теплового потока записывается в файл, если используется FS-патч.

#### 2.11.4 Режим осреднения: $F(z)$ распределения интегральных параметров

При моделировании нестабилизированных течений возникает необходимость в распределениях по z следующих основных интегральных параметров:

$$G(z) = \int_{S_0} \rho u_z dx dy \quad - \text{расход,}$$

$$\rho_m = \frac{1}{S_0} \int_{S_0} \rho dx dy \quad - \text{средняя плотность,}$$

$$U_m = \frac{G}{S_0 \rho_m} \quad - \text{среднемассовая скорость,}$$

$$h_m = \frac{\int_{S_0} \rho h_u dx dy}{G} \quad - \text{среднемассовая энтальпия,}$$

$$T_m = T(h_m) \quad - \text{среднемассовая температура.}$$

Здесь  $S_0(z)$  - площадь поперечного сечения канала. Эти соотношения используются в общем случае среды с произвольными переменными свойствами. Если теплоемкость  $c_p$  по-

стоянна, то температура является линейной функцией энтальпии  $T = h/c_p$  и колонка  $h_m$  не выводится в файл. Если плотность  $\rho$  постоянна, то среднемассовая скорость совпадает с определением среднемассовой скорости для несжимаемой среды:

$$U_m = \frac{1}{S_0} \int_{S_0} u_z dx dy$$

Для вывода этих параметров необходимо использовать оператор секции [Special Data]:

```
L("RANS.IsBparamZ") = .True.
```

В этом случае на шаге выдачи результатов создается текстовый файл с именем (XX - номер шага по времени):

```
<ПрефиксРезультатов>_bparamz_XX.dat
```

При выполнении начального расчета (IsCalcRANS = .False.) в файл выводятся мгновенные значения параметров на шаге выдачи:

```
"z" "Dens_INST" "G_INST", "UM_INST" "TM_INST" "HM_INST"
```

При выполнении осреднения (IsCalcRANS = .True.) наряду с мгновенными значениями выводятся осредненные по времени значения:

```
"z" "Dens_INST" "G_INST", "UM_INST" "TM_INST" "HM_INST" ...  
"Dens_AVE" "G_AVE", "UM_AVE" "TM_AVE" "HM_AVE"
```

## 2.12 Модель заданного расхода

При LES моделировании стабилизированных течений (для определенности поток направлен вдоль оси  $z$ ) используется алгоритм разбиения поля давления на периодическую составляющую  $p_f$  и постоянный градиент давления  $\beta_p$ :

$$p = p_f + \beta_p \cdot z, \quad \beta_p = - \left( \frac{dp}{dz} \right)_0 = \text{const} \quad (2.6)$$

Для моделирования необходимо задать либо сам постоянный градиент, либо полный расход через канал  $G_0$  (в кг/с) или его плотность  $m_0$  (кг/(м<sup>2</sup>·с)):

$$G_0 = \rho_0 \int_{S_c} u_z dS, \quad m_0 = \frac{G_0}{S_c} \quad (2.7)$$

где  $S_c$  - площадь сечения канала в входном или выходном сечении РО (площадь поверхностного патча, используемого в ПГУ),  $\rho_0$  - постоянная плотность среды.

Поскольку используются периодические граничные условия в направлении оси течения (это может быть направления  $x, y$  или  $z$ ), то расход постоянен в любом сечении по этому направлению (масса среды постоянна в РО).

В случае задания градиента нужно просто задать постоянный удельный источник в уравнении движения для  $u_z$ , равный  $\beta_p$ .

В случае постоянного расхода необходимо «подбирать» значение  $\beta_p$  так, чтобы расход сохранялся и был равен заданному расходу  $G_0$ . В текущей версии кода для этого используется два алгоритма: алгоритм V. Deschamps [4] и собственный MVM-алгоритм кода ANES.

Выбор модели определяется оператором секции [Special Data]:

```
C("LES.ModelGCONST") = Deschamps / MVM
```

### 2.12.1 Алгоритм V. Deschamps

Суть этого алгоритма следующая. Если проинтегрировать уравнение движения в направлении основного потока (по умолчанию - это z-направление) по объему РО, то можно получить следующее интегральное уравнение импульса:

$$L_z \frac{\partial G}{\partial \tau} = \beta_p V_0 - T_w + F_z V_0, \quad F_z = \frac{1}{V_0} \int f_z dV, \quad T_w = \int_{S_w} \mu \frac{\partial u_z}{\partial n_b} dS_w \quad (2.8)$$

Здесь:

$V_0$  - объем расчетной области,

$f_z$  - удельный силовой источник (силы плавучести, магнитная индукция и т.д.),

$S_w$  - граница РО, занятая твердыми стенками,

$n_b$  - нормаль к границе, направленная внутрь РО.

На дискретном уровне уравнение (2.8) можно записать в виде

$$G^{n+1} = G^n + g^n \Delta \tau, \quad g^n = \frac{V_0}{L_z} \left\{ \beta_p^n - \frac{T_w^n}{V_0} + F_z^n \right\}$$

Здесь верхние индексы  $n$  и  $(n+1)$  относятся к предыдущему и текущему шагу по времени. В модели V. Deschamps значение градиента давления на новом шаге по времени рассчитывается по соотношению:

$$\beta^{n+1} = \beta^n + \frac{L_z \Delta \tau}{V_0} \left[ \alpha_1 \frac{G_0 - G^{n+1}}{\Delta \tau} + \alpha_2 \frac{G_0 - G^n}{\Delta \tau} \right], \quad (2.9)$$

$$\alpha_1 = 2, \quad \alpha_2 = -0.2$$

Соотношение (2.9) хорошо и устойчиво работает для задач, записанных в безразмерной форме, когда в качестве масштаба скорости выбирается среднemasсовая скорость или скорость трения. При решении размерных задач могут возникать неустойчивости и колебания расхода  $G$  (и градиента  $\beta_p$ ) и в этом случае рекомендуется использовать MVM-модель.

Заметим, что расчет члена  $F_z$  производится автоматически, при этом в него включаются все внутренние источники и источники пользователя, заданные для компоненты скорости в направлении потока.

Алгоритм (2.9) выполняется перед расчетом нового шага по времени и все необходимые параметры рассчитываются по значениям, взятым с предыдущего шага по времени. При расчете члена  $T_w$  дополнительно рассчитывается величина  $Re_\tau$ :

$$Re_\tau = \frac{\bar{u}_\tau}{\nu}, \quad \bar{u}_\tau = \sqrt{\bar{\tau}_w}, \quad \bar{\tau}_w = \frac{T_w}{S_w}. \quad (2.10)$$

Эта переменная доступна пользователю в подсистеме myFORM и в виртуальных функциях по имени RETAU\_MID.

Эта модель используется по умолчанию.

### 2.12.2 Алгоритм MVM

Эта модель также основана на уравнении (2.8). В модели градиент давления рассчитывается по соотношению:

$$\beta_p^{(n+1)} = \frac{T_w^*}{V_0} - F_z^* + \alpha_{dp} \frac{L_z}{V_0} \frac{G_0 - G^{(n)}}{\Delta \tau}$$

Здесь символ  $*$  означает значения параметров на текущей итерации текущего  $(n+1)$  шага по времени,  $\alpha_{dp}$  - коэффициент релаксации (для большинства задач равен 1).

Этот алгоритм одинаково работоспособен и для безразмерных и размерных задач. Более того, он гораздо быстрее выходит на постоянный расход. Главный «минус» алгоритма

- необходимость расчета членов со звездочкой на каждом шаге по времени (это увеличивает время расчета шага по времени на 5-8%).

Для активации и настройки алгоритмов используются операторы секции [Special Data]:

```
C("LES.ModelGCONST") = Deschamps / MVM
R("LES.RelaxGCONST") = 1
L("LES.IsGCONST") = .TRUE.
C("LES.GDIR") = "X/Y/Z" <- направление потока
R("LES.G0") = <постоянный расход, кг/сек>
R("LES.M0") = <постоянная плотность расхода, кг/(м2·сек)>
R("LES.BetaDP_INI") = <Начальное значение градиента  $\beta_p$ >
```

Важные замечания:

- 1) Наряду с самым постоянным расходом  $G_0$  пользователь может задавать плотность массового расхода  $m_0$ ;
- 2) источник, связанный с  $\beta_p$ , задавать пользователю не нужно, он задается автоматически;
- 3) текущее значение градиента сохраняется в переменной решателя DEV\_DPDZ, ее можно использовать в операторах myFORM;
- 4) значение  $\beta_p$  сохраняется в файле <ПрефиксРезультатов>\_energy.timeles

При использовании режима заданного расхода на консоль выдается строка:

```
~LES:GCONST: BetaDP=1.903412E-03, G=12.566397, G0= 12.566400
```

Или

```
~LES:GCONST: BetaDP=1.903412E-03, M=1.0005, M0= 1.000000
```

где BetaDP - новое значение  $\beta_p$ , G (M) - текущий расход (плотность массового потока), G0 (M0) - заданный расход.

### 2.12.3 Интеграция модели заданного расхода и PIG-модели

Если в расчете используется PIG-модель (см. раздел 2.1 документа [6]), то расчетная область разбивается на две «независимые» зоны: PIG-зона с периодическими граничными условиями и зона основного течения. Если модель заданного расхода *активируется*, то она автоматически привязывается к PIG-зоне. Это позволяет реализовать модель периодического генератора турбулентности на входе в канал.

### 2.13 Расчет безразмерного расстояния до стенки для модели Пиомелли

Для различных моделей необходимо демпфирование турбулентной вязкости (или источников) вблизи стенки. Для этого необходимо рассчитать касательное напряжение на стенке  $\bar{\tau}_w$ , «соответствующее» данной ячейке:

$$y^+ = y_w \frac{\bar{u}_\tau}{\nu}, \quad \bar{u}_\tau = \sqrt{\frac{\bar{\tau}_w}{\rho}}$$

В текущей версии кода используется следующий подход для расчета этого касательного напряжения.

Если пользователь задал режим осреднения по 1D патчам (п. 2.9.2), то в качестве касательного напряжения используется элемент массива  $\tau_{w\_mid}(iz)$ , где  $iz$  - это индекс ячейки.

Если режим осреднения по 1D патчам не задан, то для расчета касательного напряжения для любых ячеек используется среднее по всей стенке касательное напряжение, рассчитываемое по соотношению:

$$\bar{\tau}_w = \rho (Re_\tau v)^2$$

где  $Re_\tau$  - число Рейнольдса трения, рассчитываемое в модели заданного расхода (2.10).

Если не заданы режимы осреднения по 1D патчам или постоянного расхода, то использовать  $y^+$  нельзя!

Значение  $Re_\tau$  и  $\tau_z(z)$  рассчитываются после выполнения первого шага по времени. Поэтому на самом первом шаге по времени эти значения неизвестны.

При использовании  $Re_\tau$  в случае продолжения расчета (с КТ или с другого шага по времени) на первом шаге используется значение, сохраненное в файле результатов. Для нового расчета (StartMode = SM\_NEW) необходимо задать начальное значение  $Re_\tau$ . Для этого используется оператор секции [Special Data]:

```
R("LES.RETAU_INI") = 1 / <Re $\tau$ >
```

При использовании  $\tau_{z\_mid}(z)$  на первом шаге по времени также используется  $Re_\tau$  и соотношение:

$$\bar{\tau}_w = \rho (Re_\tau v)^2,$$

где в качестве плотности и вязкости используются локальные значения.

Заметим, что если определено внутреннее поле пользователя с именем "iYPLUS", то в нем сохраняется поле  $y^+$ .

## 2.14 Утилита переинтерполяции

При проведении LES-расчетов оптимальным является следующий подход:

- 1) на первом этапе проводится расчет на грубой сетке,
- 2) на втором этапе проводится переинтерполяция полей на мелкую сетку,
- 3) на третьем этапе производится окончательный расчет на мелкой сетке.

Для переинтерполяции используется утилита aInterACP, входящая в состав кода ANES. В текущей версии утилиты есть два ограничения:

- 1) геометрия РО вариантов с грубой и мелкой сетками должна быть одинаковая,
- 2) в проекте с грубой сеткой должно отсутствовать дробление по оси Z.

Для запуска утилиты можно использовать пункт меню оболочки ANES:

«Инструменты/Утилита переинтерполяции»

или командную строку:

```
%ANES%\bin\alInterACP.exe -i:<In-префикс> -o:<Out-префикс>
```

Здесь <In-префикс > - префикс файлов результатов с грубой сеткой, <Out-префикс > - префикс файлов результатов с мелкой сеткой. Для работы утилиты необходимы следующие файлы *проекта с грубой сеткой*:

- <In-префикс>.agr - файл сеточной информации.
- <In-префикс>.asr - файл рестарта,
- <In-префикс>.arhi - файл с именами Ф-переменных,
- <In-префикс>.asrgrad - файл с градиентами Ф-переменных.

и три файла проекта с *мелкой сеткой*:

- <Out-префикс>.agr - файл сеточной информации,
- <Out-префикс>.arhi - файл с именами Ф-переменных,

<Out-префикс>.cord - файл Компилятора с порядком расположения ячеек.

Для создания файла с расширением \*.acprgrad, содержащего градиенты всех Ф-переменных, необходимо включить оператор секции [Special Data]:

```
L("SaveGrad") = .True.
```

Результатом работы утилиты являются два выходных файла:

<Out-префикс>\_inter.acpr - файл полями мелкой сетки,

<Out-префикс>\_inter.sord - файл с нумерацией ячеек мелкой сетки.

Типичный алгоритм работы с утилитой:

- 1) если в варианте с грубой сеткой не был создан файл \*.acprgrad нужно включить оператор создания этого файла в проект, сделать его рестарт и выполнить несколько шагов по времени;
- 2) Запустить *компиляцию* проекта с мелкой сеткой для создания необходимых файлов;
- 3) Добавить в проект операторы:  
StartMode = SM\_RES  
NameStart = "<Out-префикс>\_inter.acpr "
- 4) запустить расчет с мелкой сеткой.

Утилита для переинтерполяции использует соотношение:

$$F_{iFine} = F_{iCoarse} + [\nabla F]_{iCoarse} \{ \mathbf{x}_{iFine} - \mathbf{x}_{iCoarse} \},$$

где  $iFine$ ,  $iCoarse$  - индексы ячеек мелкой и грубой сеток,  $\mathbf{x}$  - координаты центров ячеек. Для ускорения поиска ячеек утилита использует алгоритм, связанный с *отсутствием дробления по оси z* варианта с грубой сеткой! Поэтому в варианте с грубой сеткой должен присутствовать оператор секции [Unstructured Cartesian Grid]:

```
IsNoRefZ = .True.
```

При поиске индексов  $iCoarse$  возможны две ситуации, связанные с небольшими отличиями положения граничных дробных ячеек:

- 1) ячейка  $iFine$  находится вне РО грубой сетки, но можно найти соседнюю грубую ячейку,
  - 2) ячейка  $iFine$  находится вне РО грубой сетки, но соседнюю ячейку найти нельзя.
- В первом случае соседняя ячейка используется для интерполяции, во втором случае значение  $F_{iFine}$  Заполняется нулем. Информация о таких ячейках выдается на консоль утилиты.



### 3. Гибридные LES модели

В гибридных DES моделях (Detached Eddy Simulation) используется комбинация LES модели с одной из RANS моделей в пристенных зонах.

#### 3.1 IDDES: гибридная LES модель

В этой модели [7,8] в качестве RANS модели используется модифицированная SST модель Ментера. Заметим, что SST модель, реализованная в IDDES, отличается от RANS SST модели кода ANES. В этой модели отсутствуют низкорейнольдсовские поправки, реализованные в RANS-версии.

RANS модель турбулентности состоит из уравнений для кинетической энергии  $k$  и удельной скорости диссипации  $\omega$ . Эта модель представляет собой комбинацию двух моделей:  $k$ - $\varepsilon$  в ядре потока и  $k$ - $\omega$  вблизи твердых стенок. (для обозначения обеих диссипаций используется одна переменная  $\omega$ ). Сам такой переход осуществляется с помощью специальной сглаживающей функции  $F_1$ . Там где  $F_1 = 1$  используется «чистая»  $k$ - $\omega$  модель, при  $F_1 = 0$  используется  $k$ - $\varepsilon$  модель. Все коэффициенты уравнений рассчитываются с помощью линейной интерполяции

$$C = C^{(1)}F_1 + C^{(2)}(1 - F_1)$$

Система уравнений SST модели, используемая в IDDES, имеет вид:

$$\frac{\partial(\rho k)}{\partial \tau} + \text{div} \left( \rho \mathbf{U}k - \rho \left( \mathbf{v} + \frac{\mathbf{v}_t}{\sigma_k} \right) \nabla k \right) = \rho \left[ P_t - k \frac{\sqrt{k}}{L_{des}} \right], \quad (3.1)$$

$$\frac{\partial(\rho \omega)}{\partial \tau} + \text{div} \left( \rho \mathbf{U}\omega - \rho \left( \mathbf{v} + \frac{\mathbf{v}_t}{\sigma_\omega} \right) \nabla \omega \right) = \rho \left[ C_{\omega 1} \frac{P_t}{v_t} - C_{\omega 2} \omega^2 \right] + 2(1 - F_1) \frac{\rho}{\omega \sigma_\omega^{(2)}} (\nabla k \cdot \nabla \omega)$$

Здесь  $P_t$  - порождение турбулентности:

$$P_t = \min \{ v_t G, 10 \cdot C_\mu \cdot k \omega \},$$

$$G = S^2 = 2S_{ik} \cdot S_{ik}, \quad S_{ik} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_k} + \frac{\partial u_k}{\partial x_i} \right) \quad (3.2)$$

Коэффициент турбулентной вязкости рассчитывается по соотношению:

$$v_t = \frac{k}{\omega} \frac{1}{\max \left\{ 1, \frac{\sqrt{G} F_2}{0.31 \omega} \right\}}, \quad (3.3)$$

где  $F_2$  - вторая функция-переключатель.

Турбулентные числа Прандтля:

$$\sigma_k = \frac{1}{\frac{F_1}{\sigma_k^{(1)}} + \frac{1 - F_1}{\sigma_k^{(2)}}}, \quad \sigma_\omega = \frac{1}{\frac{F_1}{\sigma_\omega^{(1)}} + \frac{1 - F_1}{\sigma_\omega^{(2)}}}, \quad (3.4)$$

$$\sigma_k^{(1)} = 1.176, \sigma_k^{(2)} = 1, \quad \sigma_\omega^{(1)} = 2, \sigma_\omega^{(2)} = 1.168$$

Коэффициенты уравнения для  $\omega$  -  $\varepsilon$  :

$$C_{\omega 1} = \gamma = C_{\omega 1}^{(1)}F_1 + C_{\omega 1}^{(2)}(1 - F_1), \quad C_{\omega 1}^{(1)} = 0.556, \quad C_{\omega 1}^{(2)} = 0.44,$$

$$C_{\omega 2} = \beta = C_{\omega 2}^{(1)}F_1 + C_{\omega 2}^{(2)}(1 - F_1), \quad C_{\omega 2}^{(1)} = 0.075, \quad C_{\omega 2}^{(2)} = 0.0828$$

(красным цветом обозначены имена констант в «родных» обозначениях Ментера). Напомним, на всякий случай, связь между  $C_{\omega 1}$  и  $C_{\omega 2}$ :

$$C_{\omega 1} = \frac{C_{\omega 2}}{C_{\mu}} - \frac{\kappa^2}{\sigma_{\omega} \sqrt{C_{\mu}}}$$

Главная функция-переключатель  $F_1$  является основным сглаживающим фактором перехода от одной RANS модели к другой:

$$F_1 = \tanh(A^4),$$

$$A = \min \left[ \max \left( \frac{\sqrt{k}}{C_{\mu} \omega \delta_w}, \frac{500\nu}{\delta_w^2 \omega} \right), \frac{4\rho k}{\sigma_w^{(2)} D_{k\omega} \delta_w^2} \right], \quad (3.5)$$

$$D_{k\omega} = \max \left( \frac{2\rho}{\sigma_w^{(2)} \omega} (\nabla k \cdot \nabla \omega), 10^{-10} \right),$$

Функция - переключатель в соотношении для турбулентной вязкости  $F_2$ :

$$F_2 = \tanh \left[ \left[ \max \left( \frac{2\sqrt{k}}{C_{\mu} \omega \delta_w}, \frac{500\nu}{\delta_w^2 \omega} \right) \right]^2 \right]. \quad (3.6)$$

Здесь  $\delta_w$  – расстояние до ближайшей твердой стенки.

На твердых стенках используются следующие граничные условия:

$$k_{\text{wall}} = 0, \quad \omega_p = C_{\text{wall}} \frac{6\nu}{C_{\mu} \delta_w^2} \quad (3.7)$$

Значение константы  $C_{\text{wall}}$  в разных версиях SST меняется в широких пределах. В ANES в RANS версии  $C_{\text{wall}} = 1$ , в IDDES версии используется другое значение  $C_{\text{wall}} = 10$ .

### 3.1.1 Гибридный масштаб турбулентных вихрей

Для замыкания (3.1) необходимо рассчитать масштаб  $L_{\text{des}}$  в члене диссипации турбулентной энергии (3.1).

В текущей версии используется соотношение:

$$L_{\text{des}} = L_{\text{rans}} f_d + L_{\text{les}} (1 - f_d) \quad (3.8)$$

RANS масштаб рассчитывается по соотношению SST модели:

$$L_{\text{rans}} = \frac{\sqrt{k}}{C_{\mu} \omega} \quad (3.10)$$

LES масштаб рассчитывается по модели Шура:

$$L_{\text{les}} = C_{\text{des}} \cdot \Delta_{\text{les}},$$

$$\Delta_{\text{les}} = \min(C_w \max(\delta_w, \Delta_{\text{max}}), \Delta_{\text{max}}),$$

$$\Delta_{\text{max}} = \max(\Delta x_p, \Delta y_p, \Delta z_p), \quad (3.11)$$

$$C_{\text{des}} = C_{\text{des1}} F_1 + C_{\text{des2}} (1 - F_1), \quad C_{\text{des1}} = 0.78, \quad C_{\text{des2}} = 0.61$$

Функция  $f_d$  определяет переход от RANS к LES модели:

$$f_d = \max(f_b, 1 - f_{dt})$$

Функция  $f_b$  является «защитной» функцией, гарантирующей, что у стенки всегда будет RANS модель (хотя может быть и в узкой зоне):

$$f_b = \min(1, 2 \exp(-9\alpha^2), \alpha = 0.25 - \frac{\delta_w}{\Delta_{\text{max}}}) \quad (3.12)$$

Заметим, что эта функция является чисто геометрической величиной, связанной с геометрией РО.

Функция  $f_{dt}$  имеет вид:

$$f_{dt} = 1 - \tanh\left\{(C_{d1} \cdot r_{dt})^{C_{d2}}\right\}, \quad r_{dt} = \frac{v_t}{(\kappa\delta_w)^2 D} \quad (3.13)$$

Член  $D$  имеет следующий вид:

$$D = \sqrt{\frac{1}{2}(S^2 + \Omega^2)} = \sqrt{\frac{1}{4}\{(g_{ik} + g_{ki})^2 + (g_{ik} - g_{ki})^2\}} = \\ = \sqrt{\frac{1}{8}\{g_{ik}^2 + g_{ki}^2 + 2g_{ik}g_{ki} + g_{ik}^2 + g_{ki}^2 - 2g_{ik}g_{ki}\}} = \sqrt{g_{ik}^2},$$

$$g_{ik} = \frac{\partial u_i}{\partial x_k}, \quad S^2 = 2S_{ik}S_{ik}, \quad \Omega^2 = 2\Omega_{ik}\Omega_{ik},$$

$$S_{ik} = \frac{1}{2}(g_{ik} + g_{ki}), \quad \Omega_{ik} = \frac{1}{2}(\Omega_{ik} - \Omega_{ki})$$

Значение постоянных  $C_{d1}$  и  $C_{d2}$  можно изменять в файле проекта в секции [Turbulence]:

```
R("CtCDES1") = 0.78 ! Cdes1
R("CtCDES2") = 0.61 ! Cdes2
R("CtCWALL") = 0.15 ! Cw
R("CtCDT1") = 20 ! Cdt1
R("CtCDT2") = 3 ! Cdt2
```

### 3.1.2 Активация модели

Для активации модели используется макрофункция секции [Macro Sub]:

```
modeltur(trSST_IDDES)
```

При работе с IDDES можно в файл результатов вывести поля, связанные с алгоритмом. Для этого нужно ввести *внутренние* поля пользователя (для них не нужно задавать операторы расчета значений этих полей - оператор calc):

iF1SST - это функция смешения  $F_1$ ,

iF2SST - это функция смешения  $F_2$ ,

iLenTUR - «истинный» масштаб энергосодержащих вихрей  $l_e = \frac{\sqrt{k}}{C_\mu^{1/4}\omega}$ ,

iFDT - функция  $f_{dt}$ ,

iRDT - функция  $r_{dt}$ ,

iFD - основная функция смешения  $f_d$ ,

iL\_RANS - масштаб  $L_{rans} = \frac{\sqrt{k}}{C_\mu\omega}$ ,

iL\_LES - масштаб  $L_{les} = C_{des} \cdot \Delta_{les}$ ,

iL\_DES - масштаб  $L_{des}$ .

Как показал опыт использования IDDES модели, наиболее хорошие результаты дают следующие настройки:

1. Для всех Ф-переменных, кроме турбулентной энергии  $k$  и удельной диссипации  $\omega$  необходимо использовать центрально-разностную схему TrueCD (на начальном этапе расчета можно использовать схему SuperBee).

2. Для турбулентной энергии  $k$  и удельной диссипации  $\omega$  нужно использовать стандартную степенную схему ANES.
3. Для задания начального распределения нужно использовать «стандартный» подход, описанный в п. 2.7. Для создания «стационарных» распределений необходимо использовать  $k$ - $\omega$  или SST модель турбулентности, в этом случае будет дополнительно создан файл \*\_omtur.dat. Распределение  $\omega_0$  из этого файла будет использоваться для инициализации диссипации  $\omega$  во всей РО (без возмущений). Для задания начального распределения  $k$  можно использовать два подхода. В первом подходе распределение  $k$  берется из стационарного распределения  $k_0$ . Во втором подходе поле  $k$  рассчитывается из соотношения:

$$k = v_{t0} \cdot \omega_0,$$

где  $v_{t0}$  - постоянное значение турбулентной кинематической вязкости, определяемое оператором:

$$R("LES.ViscTurIni") = \langle \text{значение вязкости} \rangle$$

4. При моделировании канальных течений и использовании PIG-модели (см. раздел 2.1 документа [6]) для турбулентной энергии и удельной диссипации нужно задать граничные условия:

$$\begin{aligned} \text{COEF}(\text{"ИмяIN.kTUR"}) &= \text{BC\_MASSONLY} \\ \text{VAL}(\text{"IN.kTUR"}) &= \text{BC\_PIG} \end{aligned}$$

$$\begin{aligned} \text{COEF}(\text{"ИмяIN.omTUR"}) &= \text{BC\_MASSONLY} \\ \text{VAL}(\text{"IN.omTUR"}) &= \text{BC\_PIG} \end{aligned}$$

## 3.2 KDES модель

### 3.2.1 Описание модели

В ANES реализована модификация гибридной модели, получившая название в литературе KDES.

В основе используемой гибридной модели лежит двухслойная  $k$ - $\varepsilon$  модель турбулентности [10,11], согласно которой в области  $R_k < 200$  решается только одно уравнение для турбулентной кинетической энергии  $k$ , а скорость диссипации  $\varepsilon$  и турбулентная вязкость  $\nu_t$  рассчитываются по алгебраическим соотношениям

$$\frac{\partial(\rho k)}{\partial \tau} + \nabla \cdot \left( \rho \mathbf{u} k - \rho \left( \nu + \frac{\nu_t}{\sigma_k} \right) \nabla k \right) = \rho \nu_t G - \rho \varepsilon, \quad (3.14)$$

$$G = 2S_{ij} \cdot S_{ij}, \quad S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right),$$

$$\nu_t = C_\mu^{1/4} f_\mu (R_k) \sqrt{k} L_{\text{rans}}, \quad L_{\text{rans}} = \kappa y_w, \quad \kappa = 0.41;$$

$$\varepsilon = C_\mu^{3/4} f_\varepsilon (R_k) \frac{k^{3/2}}{L_{\text{rans}}}, \quad R_k = \frac{\sqrt{k} y_w}{\nu}, \quad C_\mu = 0.09,$$

Здесь  $y_w$  – расстояние до ближайшей твердой стенки.

Для демпфирующих функций  $f_\mu$  и  $f_\varepsilon$ , зависящих от модифицированного турбулентного числа Рейнольдса  $R_k$ , существуют различные выражения. В настоящее время в коде ANES реализованы две модели:

Вольфштейна [10]:

$$f_\mu = 1 - \exp\left(\frac{-R_k}{A_\mu}\right), \quad f_\varepsilon = \left(1 - \exp\left(\frac{-R_k}{A_\varepsilon}\right)\right)^{-1}, \quad (3.15)$$

$$A_\mu = 70, \quad A_\varepsilon = 5.1$$

и Норриса–Рейнольдса [12]:

$$f_\mu = 1 - \exp\left(\frac{-R_k}{A_\mu}\right), \quad f_\varepsilon = 1 + \frac{A_\varepsilon}{R_k}, \quad (3.16)$$

$$A_\mu = 50.5, \quad A_\varepsilon = 5.3$$

Гибридная модель, сочетающая RANS- и LES-подходы, построена на основе идей, изложенных в работе [13]. Моделирование подсеточной турбулентной энергии  $k_{sgs}$  проводится с помощью уравнения, аналогичного (3.14), в котором соответствующим образом выбирается масштаб длины.

Общий вид уравнений модели:

$$\begin{aligned} \frac{\partial(\rho \mathbf{u})}{\partial \tau} + \nabla \cdot (\rho \mathbf{u} \mathbf{u} - \rho (v + v_{sgs}) \nabla \mathbf{u}) &= -\nabla p + \mathbf{S}_u, \\ \frac{\partial(\rho k_{sgs})}{\partial \tau} + \nabla \cdot (\rho \mathbf{u} k_{sgs} - \rho (v + v_{sgs}) \nabla k_{sgs}) &= \rho v_{sgs} G - \rho \varepsilon, \\ \frac{\partial(\rho h)}{\partial \tau} + \nabla \cdot \left( \rho \mathbf{u} h - \rho \left( \frac{\lambda}{\rho c_p} + \frac{v_{sgs}}{Pr_{sgs}} \right) \nabla h \right) &= 0, \end{aligned} \quad (3.17)$$

где  $h$  – удельная энтальпия,  $Pr_{sgs} = 0.85$ .

В коде реализованы две модификации KDES модели, отличающиеся алгоритмами сглаживания масштабов длины.

В первой модели (KSES\_SMO\_LLES) скорость диссипации и подсеточная и турбулентная вязкость рассчитываются по соотношениям

$$v_t = C_\mu^{1/4} f_\mu (R_k) \sqrt{k_{sgs}} L_{des}, \quad \varepsilon = C_\mu^{3/4} f_\varepsilon (R_k) \frac{k_{sgs}^{3/2}}{L_{des}}, \quad (3.18a)$$

$$L_{des} = \min\{L_{trans}, L_{les}\}, \quad L_{les} = C_{sgs} \Delta_{les} \Psi_L, \quad \Psi_L = f_\varepsilon^{1/4} / f_\mu^{3/4}. \quad (3.19a)$$

Во второй модели (KDES\_SMO\_LRANS) используются другие соотношения:

$$v_t = C_\mu^{1/4} F_\mu (R_k) \sqrt{k_{sgs}} L_{des}, \quad \varepsilon = C_\mu^{3/4} \frac{k_{sgs}^{3/2}}{L_{des}}, \quad F_\mu = \begin{cases} f_\mu & \text{for RANS zone} \\ 1 & \text{for LES zone} \end{cases} \quad (3.18b)$$

$$L_{des} = \min\{L_{trans}, L_{les}\}, \quad L_{trans} = \kappa y_w / f_\varepsilon (R_k), \quad L_{les} = C_s \Delta_{les} \quad (3.19b)$$

Для вычисления  $\Delta_{les}$  (масштаба фильтрации в методе LES) существует несколько способов. В коде ANES реализовано два варианта расчета этой величины:

$$\Delta_{les} = \sqrt[3]{\Delta V}, \quad (3.20)$$

$$\Delta_{les} = \min\{C_w \max\{y_w, h_m\}, h_m\}, \quad (3.21)$$

$$C_w = 0.15, \quad h_m = \max\{\Delta_x, \Delta_y, \Delta_z\}$$

где  $\Delta V$  – объем ячейки расчетной сетки,  $\Delta_x, \Delta_y, \Delta_z$  – размеры ячейки в направлении осей  $x, y, z$ .

Серия тестовых расчетов показала, что:

- использование формул (3.20) и (3.21) для линейного масштаба дало похожие результаты, однако полученные с использованием (3.21) профили скорости имели небольшой изгиб в буферной области;
- для вариантов с пристенными  $y^+ \approx 1$  и  $y^+ \approx 2$  отличие результатов оказалось незначительным;
- число точек по оси  $z$  оказывает существенное влияние на точность результатов.
- отличие результатов расчетов с  $C_{sgs} = 0.23$  от данных DNS весьма заметное и значительно большее по сравнению с вариантом  $C_{sgs} = 0.1$ .
- расчеты по модели Норриса–Рейнольдса дали чуть лучшие результаты по сравнению с моделью Вольфштейна.

### 3.2.2 Настройка модели

Для активации модели используется макрофункция секции [Macro Sub]:

```
modeltur(trSgsKDES)
```

Для настройки параметров модели используются операторы секции [Turbulence]:

```
VerKDES = KDES_SMO_LLES / KDES_SMO_LRANS
```

- версия модели;

```
CtConstSGS = <значения  $C_s$  или  $C_w$ >
```

- оператор задания постоянной  $C_{sgs}$ , по умолчанию  $C_{sgs} = 0.1$ ;

```
CtPranTurSGS = <значение  $Pr_{sgs}$ >
```

- оператор задания подсеточного числа Прандтля, по умолчанию  $Pr_{sgs} = 0.85$ ;

```
ModelSGLS = SGLS_VOL13 / SGLS_SHUR2008
```

- алгоритм расчета подсеточного масштаба длины (3.20) или (3.21). По умолчанию используется модель SGLS\_VOL13 (3.20).

и секции [Special Data]:

```
C("KDES.DumpFUN") = "Wolf"/"Nor-Ren" : Модель демпфирующих функций для K
```

```
R("KDES.AMU") = 70 / 50.5 : Постоянные демпфирующих функций
```

```
R("KDES.AEPS") = 5.1 / 5.3
```

По умолчанию используется модель Вольфштейна и версия KDES\_SMO\_LLES.

При работе с KDES можно в файл результатов вывести поля, связанные с алгоритмом. Для этого нужно ввести *внутренние* поля пользователя (для них не нужно задавать операторы расчета значений этих полей - оператор Calc):

iLLES, iLRANS, iLDES - масштабы  $L_{les}$ ,  $L_{rans}$ ,  $L_{des}$ ,

iFLES - 1 при  $L_{les} < L_{rans}$  и 0 при  $L_{les} > L_{rans}$ ,

iF\_PSI - значение функции  $\Psi_L$ .

iRK - значение  $R_k$ .

Как показал опыт использования KDES модели, наиболее хорошие результаты дают следующие настройки:

1. Для всех Ф-переменных, кроме турбулентной энергии  $k_{sgs}$  необходимо использовать центрально-разностную схему TrueCD (на начальном этапе расчета можно использовать схему SuperBee).
2. Для турбулентной энергии  $k_{sgs}$  нужно использовать стандартную степенную схему ANES (это устанавливается по умолчанию).

3. Для задания начального распределения нужно использовать «стандартный» подход, описанный в п. 2.7. В качестве начального распределения  $k_{sgs}$  нужно использовать малое постоянное значение (по умолчанию задается значение  $10^{-5}$ ).
4. При моделировании канальных течений и использовании RIG-модели (см. раздел 2.1 документа [6]) для турбулентной энергии нужно задать граничные условия:  
COEF("ИмяIN.KDES") = BC\_MASSONLY  
VAL("IN.KDES") = BC\_RIG

#### 4. Пакетный постпроцессор lesPost20

Пакетный постпроцессор предназначен для статистической обработки результатов моделирования. Можно использовать три типа данных:

- 1) временной сигнал,
- 2) файл результатов для одиночного шага по времени,
- 3) файл с полями на граничном патче для одиночного шага по времени.

В *первом случае* для анализа используется текстовый файл следующей структуры:

```
<Имя Time> <ИмяФ1> <ИмяФ2>...
Time1 ЗначениеФ1 ЗначениеФ2 .....
.....
```

Первая колонка постпроцессором трактуется как независимая переменная - время, следующие колонки - это значения  $\Phi$ -переменных. Имена переменных описываются в первой строке. В имени нельзя использовать символ "\*", но можно использовать пробелы. В последнем случае имена нужно заключить в двойные кавычки.

Во *втором случае* для анализа используется файл результатов конкретного шага по времени (NNN - номер шага) вместе с файлами Компилятора:

```
<Имя варианта>.agr      - файл сеток,
<Имя варианта>.aphi     - файл с именами  $\Phi$ -переменных,
<Имя варианта>NNN.ars   - файл с полями  $\Phi$ -переменных.
```

Отметим, что для «правильной» обработки результатов необходимы поля  $\Phi$ -переменных в центрах ячеек сетки. При сохранении значений в вершинах ячеек (этот вариант используется по умолчанию в Решателе) при загрузке файлов результатов производится перерасчет значений в центрах ячеек и об этом выдается предупреждающее сообщение. С математической точки зрения такой пересчет эквивалентен фильтрации полей с помощью коробочного фильтра шириной в два контрольных объема! Для сохранения полей в центрах ячеек необходимо использовать оператор секции [Special Data]:

```
L("ARS.OUTCELL") = .True.
```

Постпроцессор позволяет для временного сигнала позволяет:

- 1) построить графики как самой  $\Phi$ -переменной, так и ее пульсации (отклонения от среднего значения); при этом можно использовать «составные»  $\Phi$ -переменные в виде произведения « $\Phi_1 * \Phi_2$ »;
- 2) рассчитать распределение плотности вероятности сигнала  $\Phi$ -переменной;
- 3) рассчитать автокорреляционную функцию;
- 4) построить спектр  $\Phi$ -переменной;
- 5) получить отфильтрованные профили  $\Phi$ -переменной.

При использовании файлов результатов одиночного шага по времени эти операции применяются к зависимости  $\Phi(x)/\Phi(y)/\Phi(z)$ , проходящей через любую точку расчетной области. Дополнительно реализованы две обработки:

- 1) расчет трехмерного энергетического спектра для расчетной области с изотропными пульсациями полей скорости;
- 2) получение одномерных полей, усредненных по двум другим направлениям.

В *третьем случае* для анализа используется файл с именем

```
<ПрефиксРезультатов>_patch2d_<ИмяПатча>_NNN.dat
```

созданный с помощью оператора C("RANS.PATCH2D") (см. пункт 2.9.3). В этом случае можно вывести значения касательного напряжения и температуры стенки по од-



ному направлению, осреднить значения по одному направлению или вывести эти поля в виде поверхностного VTK - файла.

Для каждой обработки постпроцессор показывает графики рассчитанной зависимости. Для этого используется диалоговая утилита кода ANES laForPlot.exe. Дополнительно создается текстовый файл с таблицей рассчитанной зависимости. Эту таблицу можно использовать для построения графика сторонними продуктами. Лучше всего для этого использовать подсистему matplotlib языка программирования Python. Постпроцессор создает вместе с текстовым файлом файл Python для построения графика. Главное достоинство такого способа построения графиков заключается в возможности простого редактирования параметров построения графиков.

Язык программирования Python вместе с библиотекой matplotlib можно получить, установив дистрибутив WinPython или Anaconda.

Для вызова постпроцессора используется команда:

```
<anes>\bin\gfor64\arLESpst20.exe -i:<Имя скрипта> [-od:<Каталог графиков>]
```

здесь

<Каталог графиков> - каталог для выходных файлов \*.dat и \*.py; если каталог не указан, то используется текущий каталог,

:<Имя скрипта> -- имя файла с операторами построения.

#### 4.1 Файл скрипта

Файл с командами построения представляет собой обычный текстовый файл с расширением \*.apsc. Синтаксис команд построения аналогичен командам утилиты верификации laVerify и пакетного постпроцессора arVpost20:

1. Каждый оператор записывается на отдельной строке.
2. Все что расположено правее знака “!” считается комментарием.
3. При записи операторов регистр букв не учитывается.

Файл скрипта разбит на блоки, начало блока определяется оператором

Oper = TimeValues / ARS / Patch2d

Блок, начинающийся с оператора TimeValues, позволяет провести статистическую обработку одного временного сигнала, ARS – файлов результатов одного шага по времени, Patch2d - файлов с полями на граничном патче на одном шаге по времени.

#### 4.2 Общие операторы

Эти операторы используются во всех блоках. Они действуют на все операторы, расположенные *ниже них*.

Операторы

IsPlot = True / False

IsOut = False / True

IsPython = False / True

активируют построение графиков (IsPlot), создание текстовых файлов с результатом расчетов (IsOut) и создание скрипта на Python (IsOUT и IsPython) для построения графиков. Файлы \*.dat и \*.py создаются в каталоге <Каталог графиков>, указанного в строке запуска.

Оператор

PrefixOutFiles = <Префикс выходных файлов>

задает общую часть выходных файлов. Выходной файл имеет следующую структуру:

<Каталог графиков>\<Префикс выходных файлов>-<Режим>-<ИмяФ>.dat

<Каталог графиков>\<Префикс выходных файлов>-<Режим>-<ИмяФ>.py

Здесь <Режим> определяет тип статистической обработки:

signal - сама временная зависимость  $\Phi$ -переменной или ее пульсации,

pdf - плотность вероятности,

auto - корреляционная функция,

spectrum - спектр,

smooth - сглаженная зависимость.

## Операторы

PlotLineColor = <Цвет>

PlotLineColor1 = <Цвет>

PlotMarkerColor = <Цвет>

определяют цвет линий графиков и гистограмм. Параметр <Цвет> задается строками:

Black, Maroon, Green, Olive, Navy, Purple, Teal, Gray, Silver,

Red, Lime, Yellow, Blue, Fuchsia, Aqua, White,

ScrollBar, Background, ActiveCaption, InactiveCaption, Menu,

Window, WindowFrame, MenuText, WindowText, CaptionText,

ActiveBorder, InactiveBorder, AppWorkSpace, Highlight,

HighlightText, BtnFace, BtnShadow

## Оператор

PlotLineWidth = 2

задает толщину линии графика.

## Операторы

PlotFontSize = 12

PlotMSGFontSize = 12

задают размеры фонтов надписей графиков и заголовка окна laForPlot.

## Оператор

IsFluct = False/True

определяет, как необходимо использовать зависимости  $\Phi(t)$ ,  $\Phi(x)$ ,  $\Phi(y)$  и  $\Phi(z)$ . Если значение этого оператора True, то при вызове любой команды обработки вместо сигнала используется переменная

$$\Phi' = \Phi - \bar{\Phi}$$

где  $\bar{\Phi}$  - среднее значение.

### 4.3 Операторы для обработки временного сигнала

Эти операторы начинаются со строки

Oper = TimeValues

Они предназначены для работы с временными сигналами, заданным файлом:

File = "<путь к файлу>"

Например,

```
File = "data\u-rans-64_32-12-32.timeles"
```

При выполнении этого оператора производится загрузка значений  $\Phi$ -переменных, при этом в листинг постпроцессора выводится информация о параметрах сигнала:

```
=====
===   Средние значения и дисперсии временного сигнала
=====
1. Data file.....= data\u-rans-64_32-12-32.timeles
2. No. of Point.....= 3354
3. TimeBeg / TimeEnd.....= 3.00000E-03 10.062
4. dtimestep.....= 2.99997E-03

--- Name  ----- Mean  ----- Dispersion  ----
UX          13.23          2.760
UY         -5.2464E-02     0.3373
UZ          -0.1095          1.042
P           0.6016          2.406
```

Здесь перечислены доступные имена  $\Phi$ -переменных и их средние значения и дисперсии. В постпроцессоре каждая  $\Phi$ -переменная рассматривается как массив значений

$\Phi_i, i = 1 \dots N_s,$

определенных во временных точках  $t_i$ . Среднее значение и дисперсия рассчитывается по соотношениям, справедливым для произвольной неравномерной сетке по времени  $t_i$ :

$$\bar{\Phi} = \frac{1}{t_N - t_1} \sum_{k=2}^{N_s} \frac{\Phi_k + \Phi_{k-1}}{2} (t_k - t_{k-1}),$$

$$\sigma_{\Phi} = \sqrt{\frac{1}{t_N - t_1} \sum_{k=2}^{N_s} \frac{(\Phi_k - \bar{\Phi})^2 + (\Phi_{k-1} - \bar{\Phi})^2}{2} (t_k - t_{k-1})} \quad (4.1)$$

Для однозначности анализа при работе с временным сигналом производится сдвиг по времени так, чтобы  $t_1 = 0$ , а  $t(N_s) = T_s = t(N_s) - t(1)$ .

Во всех операторах, описанных ниже, кроме  $\Phi$ -переменных, заданных в файле сигнала, можно использовать новые переменные, которые являются их произведениями. В этом случае их имя задается в виде "<Имя $\Phi_1$ >\*<Имя $\Phi_2$ >". Ниже такая переменная будет называться *составной*  $\Phi$ -переменной.

### 4.3.1 Оператор Signal

Этот оператор строит график текущего сигнала (если `IsPlot = True`) и создает файлы (если `IsOUT = True` и `IsPython = True`):

```
<Каталог графиков>\<Префикс выходных файлов>-signal-<ИмяФ>.dat
```

```
<Каталог графиков>\<Префикс выходных файлов>- signal -<ИмяФ>.py
```

Формат оператора

```
Signal("ИмяФ")
```

Для *обычной*  $\Phi$ -переменной оператор строит графики:

$\Phi$  - для режима функции `IsFluct = False`,

$\Phi - \bar{\Phi}$  - для режима пульсации `IsFluct = True`.

Для *составной*  $\Phi$ -переменной используются аналогичные зависимости:

$\Phi = \Phi_1 * \Phi_2$  - для режима функции `IsFluct = False`,

$\Phi = (\Phi_1 - \bar{\Phi}_1) \cdot (\Phi_2 - \bar{\Phi}_2)$  - для режима пульсации `IsFluct = True`.

Пример сигнала, построенного с помощью ру-скрипта, показан на рисунке 4.1.

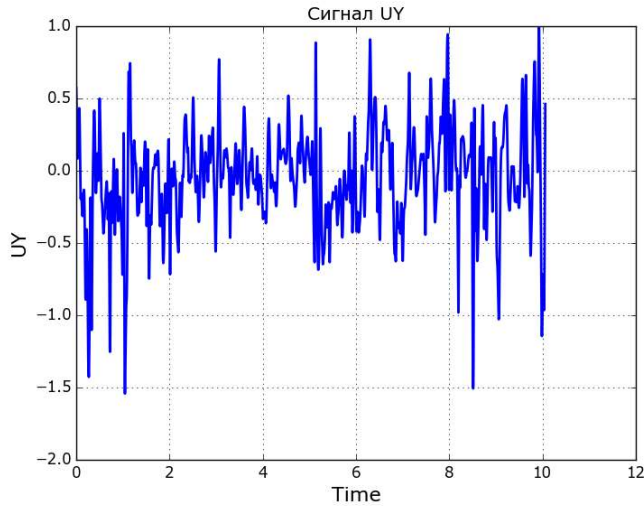


Рисунок 4.1 - Пример работы оператора Signal

### 4.3.2 Оператор PDF

Оператор предназначен для создания гистограммы плотности вероятности (Probability Density Function) сигнала. Плотность вероятности рассчитывается следующим образом. Интервал изменения ( $\Phi_{\max} - \Phi_{\min}$ ) разбивается на  $N_p$  зон с шириной  $\Delta\Phi = (\Phi_{\max} - \Phi_{\min}) / N_p$ . Плотность вероятности  $P$  рассчитывается по соотношению:

$$P(\Phi_k)\Delta\Phi = P\left\{\Phi_k - \frac{\Delta\Phi}{2} < \Phi < \Phi_k + \frac{\Delta\Phi}{2}\right\}\Delta\Phi = \frac{N_k}{N_S} \quad (4.2)$$

где  $N_k$  - число значений  $\Phi$ , попавших в указанный интервал значений. Для построения используется оператор:

PDF("ИмяФ",[ $N_p$ ])

Если число разбиений интервала изменения  $\Phi$  не указан, то полагается  $N_p = 100$ . Правила использования ключа IsFluct те же, что и для Signal.

Пример плотности вероятности, построенной с помощью ру-скрипта, показан на рисунке 4.2.

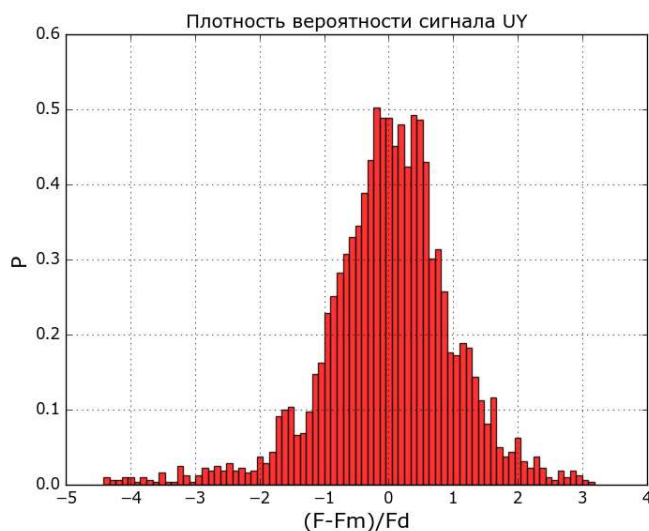


Рисунок 4.2 - Пример работы оператора PDF

### 4.3.3 Оператор AutoCorr

Оператор предназначен для расчета автокорреляционной или корреляционной функции. Для *обычной*  $\Phi$ -переменной рассчитывается безразмерный коэффициент автокорреляции (для IsFluct = False и True соответственно) :

$$R(\tau) = \frac{\int_0^{T_{\max}} \Phi(t) \Phi(t + \tau) \cdot dt}{\int_0^{T_{\max}} \Phi^2(t) \cdot dt}, \quad (4.3)$$

$$R(\tau) = \frac{\int_0^{T_{\max}} [\Phi(t) - \bar{\Phi}][\Phi(t + \tau) - \bar{\Phi}] \cdot dt}{\int_0^{T_{\max}} [\Phi(t) - \bar{\Phi}]^2 \cdot dt}$$

Для *составной*  $\Phi$ -переменной рассчитывается сама автокорреляционная функция:

$$B(\tau) = \frac{1}{T_{\max}} \int_0^{T_{\max}} \Phi_1(t) \Phi_2(t + \tau) \cdot dt, \quad (4.4)$$

$$B(\tau) = \frac{1}{T_{\max}} \int_0^{T_{\max}} [\Phi_1(t) - \bar{\Phi}_1][\Phi_2(t + \tau) - \bar{\Phi}_2] \cdot dt$$

Время интегрирования  $T_{\max}$  определяется пользователем, но оно не может быть больше  $T_S/2$ . В постпроцессоре используется алгоритм, основанный на постоянном шаге по времени ( $\Delta t_k = t_{k+1} - t_k = \text{const}$ ). Если шаги по времени переменные, то на консоль выдается предупреждение. Формулы расчета автокорреляционной функции и ее коэффициента:

$$B_k = B(\tau_k) = \frac{1}{N_{\max} - 1} \sum_{i=1}^{N_{\max}-1} \frac{\Phi_i \Phi_{i+k-1} + \Phi_{i+1} \Phi_{i+k}}{2}, \quad \tau_k = (k-1) \Delta \tau,$$

$$R_k = \frac{B_k}{B_1}$$

Для построения используется оператор:

```
AutoCorr("ИмяФ" [,Tmax])
```

Если параметр  $T_{\max}$  не указан, то в качестве времени интегрирования используется  $T_{\max} = T_S/2$ .

Пример корреляционной функции, построенной с помощью ру-скрипта, показан на рисунке 4.3.

Если используется обычная  $\Phi$ -переменная, то в листинг постпроцессора выводится информация о микро и макро масштабах Тейлора:

```
-----
--- Автокорреляция пульсации сигнала
-----
1. Signal..... = P
2. MaxTime(Signal)..... = 10.059
3. MaxTime(AutoCorr)..... = 0.50100
4. Time(Taylor Micro)..... = 1.38724E-02
5. Time(Taylor Macro)..... = 3.01312E-02
```

Для расчета микро  $\lambda_t$  и макро  $\Lambda_t$  масштабов Тейлора используются соотношения:

$$\lambda_t = \frac{\Delta\tau}{2} \left\{ \frac{1}{\sqrt{1-R_2}} + \frac{2}{\sqrt{1-R_3}} \right\}, \quad \Lambda_t = \Delta\tau \sum_{i=1}^{N_{\max}} R_i \quad (4.5)$$



Рисунок 4.3 - Пример работы оператора AutoCORR

#### 4.3.4 Оператор СПЕКТРА

Оператор предназначен для построения дискретного «энергетического» спектра одномерного сигнала.

Если на интервале  $[0, T_S]$  задана функция  $\Phi(t)$ , то ее можно разложить в ряд Фурье:

$$\Phi(t) = \sum_{k=-\infty}^{+\infty} \hat{\Phi}_k e^{i \cdot 2\pi \cdot \nu_k \cdot t} \quad (4.6)$$

Здесь  $k$  - номер гармоники,  $k = 0, \pm 1, \pm 2, \dots$ ,  $i$  - мнимая единица. Частота дискретной гармоники (в герцах) равна:

$$\nu_0 = k \cdot \Delta\nu_0, \quad \Delta\nu_0 = \frac{1}{T_S} \quad (4.7)$$

Амплитуды гармоник являются комплексными числами и могут быть рассчитаны из соотношений прямого преобразования Фурье:

$$\hat{\Phi}_k = a_k - ib_k = \frac{1}{T_S} \int_0^{T_S} \Phi(t) \cdot e^{-i \cdot 2\pi \cdot \nu_k \cdot t} dt \quad (4.8)$$

Комплексные амплитуды удовлетворяют важной теореме - теореме Парсеваля:

$$\overline{\Phi^2} = \frac{1}{T_S} \int_0^{T_S} \Phi^2 dt = \sum_{k=-\infty}^{+\infty} F_k, \quad (4.9)$$

$$F_k = \hat{\Phi}_k \cdot \overline{\hat{\Phi}_k} = |\hat{\Phi}_k|^2 = a_k^2 + b_k^2$$

Величину  $\overline{\Phi^2}$  можно рассматривать как «энергию» сигнала  $\Phi$ -переменной, тогда квадрат амплитуды гармоники можно рассматривать как часть «энергии», связанной с этой гармоникой.

Если сигнал задан дискретными значениями  $\Phi_i$ ,  $i = 1 \dots N_S$  с *постоянным* шагом  $\Delta t$ , то возникают две особенности:

1. Ряд (4.6) становится конечным. Суммирование по  $k$  ограничивается гармоникой с номером  $k = N_S/2$ . Соответствующая максимальная дискретная частота

$$v_n = \frac{N_S}{2} \Delta v_0 \quad (4.10)$$

называется частотой «отсечения» или частотой Найквиста. Если  $\Phi$ -переменная действительная, то амплитуды гармоник удовлетворяют соотношениям:

$$\Phi_{-k} = \Phi_k^*, \quad F_{-k} = F_k,$$

поэтому гармоники можно представить следующим образом:  $k=0$  - нулевая гармоника с амплитудой равной среднему значению  $\Phi$ ,  $N_S/2$  гармоник с положительной частотой и  $N_S/2$  гармоник с отрицательной частотой.

2. Выражение для амплитуд гармоник (4.8) при постоянном шаге по времени можно свести к сумме:

$$\hat{\Phi}_k = \frac{1}{N_S} \sum_{m=0}^{N_S-1} \Phi_{m+1} e^{-i \frac{2\pi}{N_S} k \cdot m} = \frac{Z_k}{N_S}, \quad F_k = \frac{Z_k \cdot Z_k^*}{N_S^2} \quad (4.11)$$

$$Z_k = \sum_{m=0}^{N_S-1} \Phi_{m+1} e^{-i \frac{2\pi}{N_S} k \cdot m}$$

Новая амплитуда  $Z_k$  введена потому, что именно эта сумма рассчитывается во всех алгоритмах дискретного преобразования Фурье. В постпроцессоре для расчета спектра используется открытая библиотека FFTW версии 3.3.3 (авторы М. Frigo и S. G. Johnson).

В дискретном виде теорему Парсеваля можно записать в виде:

$$E_{\text{val}} = \overline{\Phi^2} = \frac{1}{N_S} \sum_{m=1}^{N_S} \Phi_m^2, \quad E_K = \frac{1}{N_S^2} \left\{ |Z_0|^2 + 2 \cdot \sum_{k=1}^{N_S/2} |Z_k|^2 \right\} \quad (4.12)$$

$$E_{\text{val}} = E_K$$

Эта теорема всегда проверяется. Величины  $E_{\text{val}}$  и  $E_K$  выводятся в листинг постпроцессора:

```
-----
---   Спектр пульсации сигнала
-----
1. Signal..... = P
2. Time(end)..... = 10.059
3. UseTime(end)..... = 10.059
4. NoTimeData..... = 3354
5. NoUseData..... = 3354
6. NoFreqData..... = 1678
7. dFrequency..... = 9.94144E-02
8. Freq(Nyquist)..... = 166.72
9. Mean(Signal)..... = -1.38490E-04
10. Dispersion(Signal)^2..... = 5.7948
11. Sum(EVAL)..... = 5.7948
12. Sum(EK)..... = 5.7948
```

В качестве графика в постпроцессоре строится зависимость безразмерной «энергии» гармоники от ее номера  $i\text{Freq}$ :

$$Fk_{i\text{Freq}} = \frac{F_{i\text{Freq}}}{\Phi^2} = f(i\text{Freq}), \quad i\text{Freq} = 1, \dots, \frac{N_S}{2}$$

Нулевая гармоника не включается в график, чтобы можно было использовать логарифмические оси графика.

Для этого используется оператор

```
SPECTRA("ИмяФ",[T_max],[iFreqKolm, EKolm])
```

Здесь  $T_{\max}$  - время участка сигнала для спектрального анализа, если параметр не задан, или он равен нулю, то  $T_{\max} = T_S$ . Для «наглядности» выводится второй график следующего вида:

$$F_k = E_{\text{kolm}} \left( \frac{i\text{Freq}}{i\text{Freq}_{\text{Kolm}}} \right)^{(-5/3)},$$

где  $i\text{Freq}_{\text{Kolm}}$ ,  $E_{\text{kolm}}$  задаются в операторе SPECTRA. Если они не указаны, то второй график не строится.

Пример работы оператора показан на рисунке 4.4.

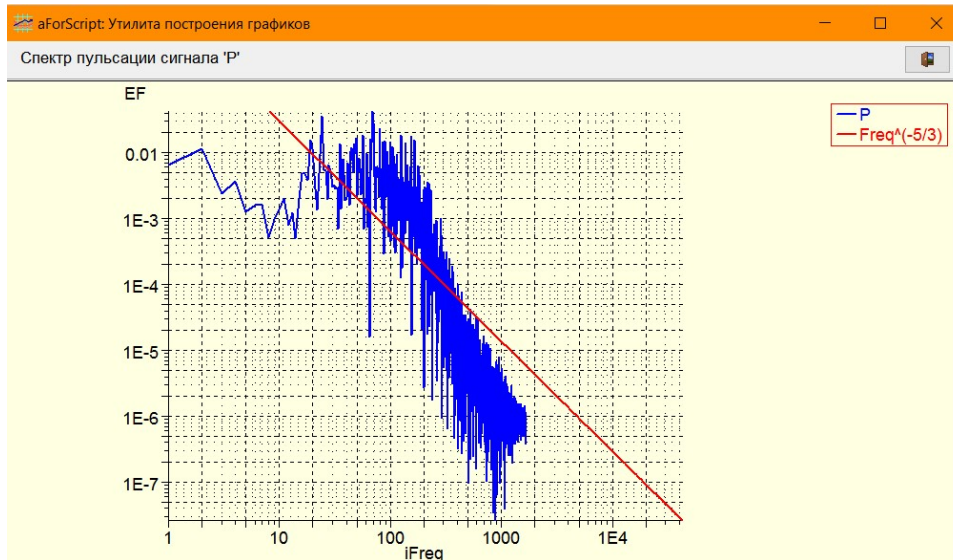


Рисунок 4.4 - Пример работы оператора SPECTRA

### 4.3.5 Оператор SMOOTH

Оператор предназначен для сглаживания одномерного сигнала с помощью «коробочного» фильтра. Сглаженное значение в момент времени "k" рассчитывается по соотношению:

$$\tilde{\Phi}_k = \frac{1}{\Delta f} \sum_{i=k-\Delta f_2}^{k+\Delta f_2} \Phi_i$$

Здесь

$\Delta f$  - ширина фильтра в точках (нечетная целая величина),

$\Delta f_2 = \text{int}(\Delta f/2)$  - полуширина фильтра (int - функция целой части).

Форма оператора:

SMOOTH("ИмяФ",[DeltaF])

Здесь  $\Delta f$  - ширина фильтра в «точках», если задано четное значение, то оно округляется до нечетного. Если значение не указано, используется значение  $\Delta f = 3$ .

Пример работы оператора показан на рисунке 4.5.



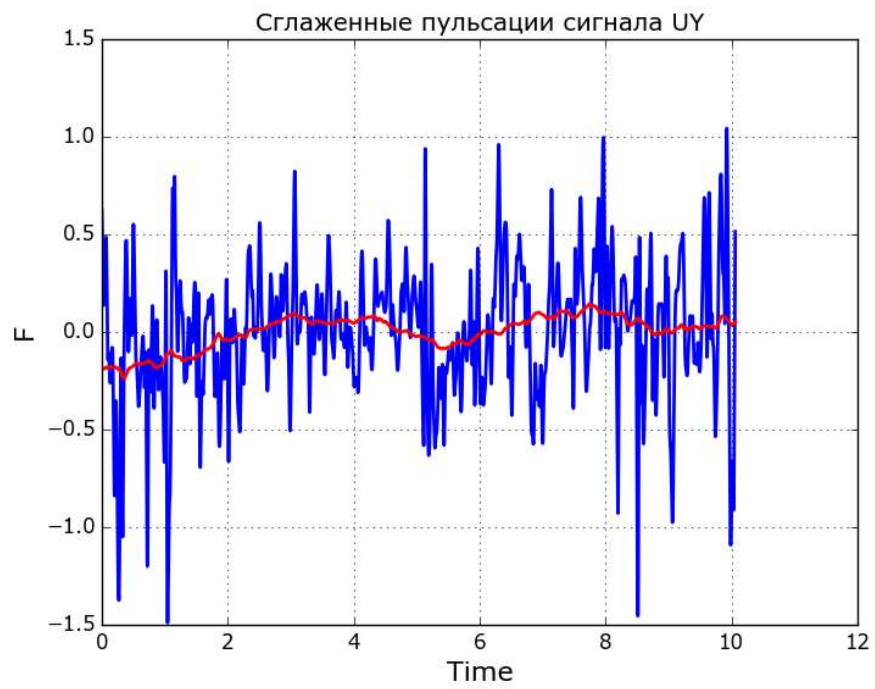


Рисунок 4.5 - Пример работы оператора SMOOTH

#### 4.4 Операторы для обработки файла результатов

Эти операторы начинаются со строки

Oper = ARS

Они предназначены для работы с полями файла результата расчета одного шага по времени. Нужные файлы задаются оператором

```
LoadARS(<Имя варианта>,"NNN")
```

Для работы оператора нужны три файла:

```
<Имя варианта>.agr      - файл сеток,
<Имя варианта>.aphi    - файл с именами Ф-переменных,
<Имя варианта>NNN.ars  - файл с полями Ф-переменных.
```

где

```
<Имя варианта> - путь и начало имен файлов,
NNN             - символьный номер шага по времени.
```

При использовании SVL ячеек необходим еще файл

```
<Имя варианта>.acbl
```

содержащий индексы SVL-ячеек для их правильной обработки.

При выполнении этого оператора производится загрузка результатов, при этом в листинг постпроцессора выводится информация о параметрах файла и именах доступных Ф-переменных:

```
=====
===      обработка файла Результаты (ARS)
=====
1. Prefix RES.....= data\u-les-taylor-green-128_
2. NoResStep.....= 9
3. TimeRES.....= 9.0401
4. SizeX.....= 6.2832
5. SizeY.....= 6.2832
6. SizeZ.....= 6.2832
7. NFX / NFY / NFZ.....= 128 128 128
8. NoCells.....= 2097152
9. NoPHI.....= 4
10. NoUSER.....= 5
11. Name(PHI/User).....:
    UGX, UGY, UGZ, PF, ViscSGS, VortX, VortY, VortZ,
    VortMod
```

##### 4.4.1 Операторы статистической обработки

Для полей из файлов результатов можно применять все операторы статистической обработки временного сигнала из пункта 2.3, за исключением оператора SMOOTH. Одномерный сигнал формируется как одномерное распределение не по времени, а вдоль одной из координатных осей. Положение этой линии в операторах задается 4 параметрами:

```
"X/Y/Z", x0,y0,z0
```

Первый параметр - это направление линии, вторые три параметра - координаты точки, через которую проходит линия.

Сами операторы:

Signal("ИмяФ",X, x0,y0,z0)  
 PDF("ИмяФ",Z, x0,y0,z0 [,Np])  
 AutoCorr("ИмяФ",Z, x0,y0,z0 [,Tmax])  
 SPECTRA(Z, x0,y0,z0 [,Tmax],[iFreqKolm, EKolm])

#### 4.4.2 Оператор трехмерного спектра

Рассмотрим область в форме куба с размером  $L$ , в которой задана периодическая по трем координатам функция  $\Phi(x_1, x_2, x_3) = \Phi(\mathbf{x})$ . Эту функцию можно разложить в ряд Фурье:

$$\Phi(\mathbf{x}) = \sum_{k_1=-\infty}^{k_1=+\infty} \sum_{k_2=-\infty}^{k_2=+\infty} \sum_{k_3=-\infty}^{k_3=+\infty} \hat{\Phi}_{k_1, k_2, k_3} \exp\left\{\frac{2\pi}{L}i(k_1 \cdot x_1 + k_2 \cdot x_2 + k_3 \cdot x_3)\right\} \quad (4.13)$$

Здесь  $\hat{\Phi}_{k_1, k_2, k_3}$  - амплитуда дискретной пространственной гармоники с волновым вектором (измеряется в  $m^{-1}$ ):

$$\tilde{\mathbf{k}} = \left(\frac{2\pi}{L}k_1, \frac{2\pi}{L}k_2, \frac{2\pi}{L}k_3\right) \quad (4.14)$$

Координаты волнового вектора дискретные с шагом  $\Delta\tilde{\mathbf{k}} = \frac{2\pi}{L}$ . Амплитуды рассчитываются из интегралов:

$$\hat{\Phi}_{k_1, k_2, k_3} = \frac{1}{L^3} \iiint_L \Phi(\mathbf{x}) \exp\left\{-\frac{2\pi}{L}i(k_1 \cdot x_1 + k_2 \cdot x_2 + k_3 \cdot x_3)\right\} dx_1 dx_2 dx_3 \quad (2.15)$$

Для трехмерного случая также справедлива теорема Парсеваля

$$\overline{\Phi^2} = \frac{1}{L^3} \iiint_L \Phi^2(\mathbf{x}) dx_1 dx_2 dx_3 = \sum_{k_1=-\infty}^{k_1=+\infty} \sum_{k_2=-\infty}^{k_2=+\infty} \sum_{k_3=-\infty}^{k_3=+\infty} F_{k_1, k_2, k_3} \quad (4.16)$$

$$F_{k_1, k_2, k_3} = \hat{\Phi}_{k_1, k_2, k_3} \cdot \hat{\Phi}_{k_1, k_2, k_3}^* = \left|\hat{\Phi}_{k_1, k_2, k_3}\right|^2$$

Если функция  $\Phi$  дискретная -  $\Phi_{j_1, j_2, j_3}$ ,  $j_1, j_2, j_3 = 1 \dots N$ , то:

- 1) число гармоник ограничено волновым вектором Найквиста:

$$k_n = \frac{N}{2}, \quad \tilde{k}_n = \frac{N}{2} \Delta\tilde{k}_0, \quad \Delta\tilde{k}_0 = \frac{2\pi}{L}$$

- 2) выражение для амплитуд гармоник (4.16) при постоянном шаге по времени можно свести к сумме:

$$\hat{\Phi}_{k_1, k_2, k_3} = \frac{1}{N^3} \sum_{j_1=0}^{N-1} \sum_{j_2=0}^{N-1} \sum_{j_3=-\infty}^{N-1} \Phi_{j_1+1, j_2+1, j_3+1} \exp\left\{-\frac{2\pi}{N}i(k_1 \cdot j_1 + k_2 \cdot j_2 + k_3 \cdot j_3)\right\} = \frac{Z_{k_1, k_2, k_3}}{N^3}$$

Амплитуда  $Z$  рассчитывается в процедурах библиотеки FFTW. Комплексные амплитуды  $\hat{\Phi}_{k_1, k_2, k_3}$  представляют собой трехмерный массив с размерностью  $[0 \dots N/2]$  по всем осям. Для каждой  $\Phi$ -переменной справедливо соотношение Парсеваля (4.16), которое в дискретном случае можно записать в виде:

$$\begin{aligned}
E_{\text{val}} &= \overline{\Phi^2} = \frac{1}{N^3} \sum_{j_1=1}^N \sum_{j_2=1}^N \sum_{j_3=1}^N \Phi_{j_1, j_2, j_3}^2, \\
F_{k_1, k_2, k_3} &= \frac{Z_{k_1, k_2, k_3} Z_{k_1, k_2, k_3}^*}{N^6} \\
E_K &= F_{0,0,0} + 2 \cdot \sum_{k_1=1}^{N/2} F_{k_1,0,0} + 2 \cdot \sum_{k_2=1}^{N/2} F_{0,k_2,0} + 2 \cdot \sum_{k_3=1}^{N/2} F_{0,0,k_3} + \\
&+ 4 \cdot \sum_{k_1=1}^{N/2} \sum_{k_2=1}^{N/2} F_{k_1, k_2, 0} + 4 \cdot \sum_{k_1=1}^{N/2} \sum_{k_3=1}^{N/2} F_{k_1, 0, k_3} + 4 \cdot \sum_{k_2=1}^{N/2} \sum_{k_3=1}^{N/2} F_{0, k_2, k_3} + \\
&+ 8 \sum_{j_1=1}^N \sum_{j_2=1}^N \sum_{j_3=1}^N F_{k_1, k_2, k_3} \\
E_{\text{val}} &= E_K
\end{aligned} \tag{4.17}$$

В теории турбулентности при описании изотропного случая используется сферическая СК в волновом пространстве. Если учесть, что целый волновой вектор  $\mathbf{k} = (k_1, k_2, k_3)$  имеет шаги по координатам равные 1 ( $\Delta k_1 = \Delta k_2 = \Delta k_3 = 1$ ), то теорема Парсеваля (4.16) может быть записана в виде трехмерного численного интеграла в декартовой волновой системе:

$$\overline{\Phi^2} = \sum_{k_1=-N/2}^{k_1=+N/2} \sum_{k_2=-N/2}^{k_2=+N/2} \sum_{k_3=-N/2}^{k_3=+N/2} F_{k_1, k_2, k_3} \Delta k_1 \Delta k_2 \Delta k_3$$

Этот интеграл можно записать в сферической СК волновых векторов:

$$\overline{\Phi^2} = \sum_{k=0}^{N/2} E_k \Delta k, \quad E_k = 4\pi \cdot k^2 \cdot F_k, \tag{4.18}$$

$$k = \sqrt{k_{x1}^2 + k_{x2}^2 + k_{x3}^2}$$

Для расчета  $E_k$  можно использовать два подхода. В обоих подходах создаются «радиальные» массивы  $k(\cdot)$  и  $E_k(\cdot)$  с размерностью  $\sqrt{3} \cdot \frac{N}{2}$ . Массив  $k$  заполняется номерами сферических гармоник:

$$k(i) = i = 0, 1, 2, \dots, \sqrt{3} \frac{N}{2}$$

Для заполнения массива  $E_k$  можно использовать два подхода, назовем их моделями Sofiane и Keldysh. Первая модель предложена в [2], вторая - в работах института прикладной математики РАН.

Для заполнения массива  $E_k$  используется следующий алгоритм:

1. В трехмерном цикле перебираются все ячейки волновых векторов, включая и отрицательные:

```

do ix = -N/2, N/2
  do iy = -N/2, N/2
    do iz = -N/2, N/2
      k1 = abs(ix); k2 = abs(iy); k3 = abs(iz)
      kP = SQRT(k1**2 + k2**2 + k3**2)
      kW = INT(kP); kE = kW + 1
      FP = F(k1, k2, k3)
    end do
  end do
end do

```

2. Дробный «сферический» волновой вектор  $k_P$  располагается между двумя целыми «сферическими» индексами  $k_W = \text{int}(k_P)$  и  $k_E = k_W + 1$ . значение  $F_P$  необходимо распределить между значениями в элементах  $k_W$  и  $k_E$ .
3. В модели Sofiane значения в элементах  $k_W$  и  $k_E$  «накапливаются» по соотношениям:

$$E_k(k_W) = E_k(k_W) + 4\pi \cdot \beta_w k_P^2 \cdot F_P, \quad \beta_w = k_W + 1 - k_P,$$

$$E_k(k_E) = E_k(k_E) + 4\pi \cdot (1 - \beta_w) k_P^2 \cdot F_P,$$

$$D(k_W) = D(k_W) + \beta_w, \quad D(k_E) = D(k_E) + 1 - \beta_w$$

Одновременно заполняется массив-делитель  $D()$ . После перебора 3D ячеек значения  $E_k$  рассчитываются по соотношениям

$$E_k = E_k/D$$

4. В модели Keldysh используется другой принцип заполнения: значение  $F_P$  заносится либо в элемент  $E_k(k_W)$  если  $(k_P - k_W) < 0.5$ , в противном случае элемент добавляется к  $E_k(k_E)$ . В этом алгоритме естественно не производится умножение на  $4\pi k^2$ .

Результаты работы моделей для задачи о затухании изотропной турбулентности показаны на рисунке 4.6. В постпроцессоре реализованы обе модели, их выбор осуществляется оператором:

```
ModelSpectra3D = Sofiane/Keldysh
```

По умолчанию используется Sofiane модель.

Для построения спектра используется оператор

```
SPECTRA3D("ИмяUX", "ИмяUY", "ИмяUZ", [iWaveKolm, EKolm])
```

Здесь

"ИмяUX", "ИмяUY", "ИмяUZ" - имена  $\Phi$ -переменных для компонент вектора скорости,  
iFreqKolm, EKolm - параметры для построения «линии» Колмогорова-Обухова  
(см. пункт 2.3.4).

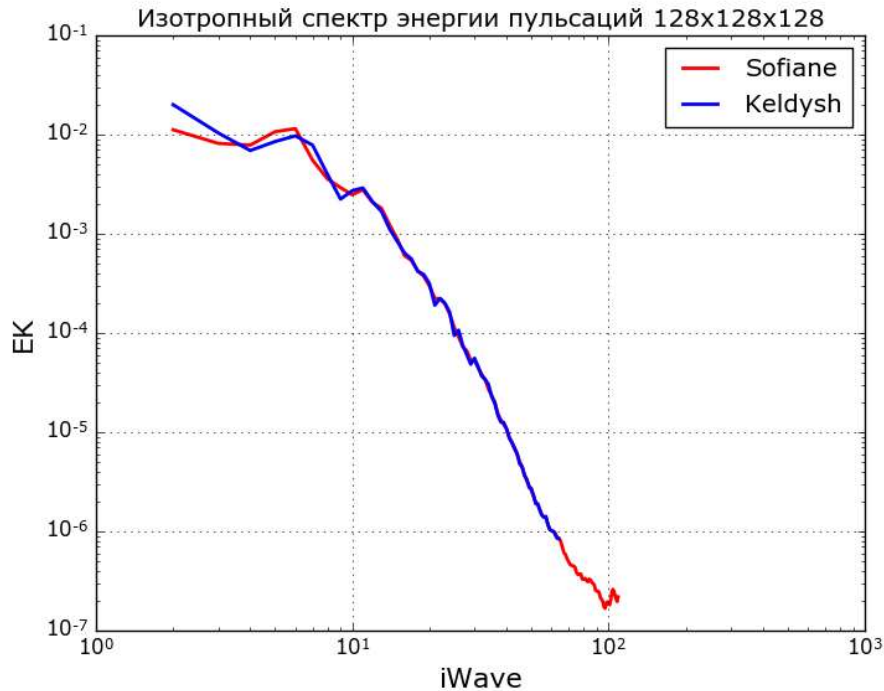


Рисунок 4.6 - Пример работы оператора SPECTRA3D

Оператор построения 3D спектра выводит в листинг информацию о интегральных параметрах, аналогичных одномерному спектру:

```
=====
===   Трехмерный спектр
=====
1. Sum(EVAL).....= 8.48312E-02
2. Sum(EK_3D).....= 8.48312E-02
4. Model(EK_Sphere).....= keldysh
5. Sum(EK_Sphere).....= 8.48247E-02
```

#### 4.4.3 Оператор осреднения в декартовой СК

Этот оператор предназначен для получения одномерных зависимостей вдоль одной из осей. Для этого производится осреднение по двум другим осям:

$$F_{1d}(y) = \frac{1}{A_{xz}} \int_{A_{xz}} F(x, y, z) dx dz$$

Вызов оператора:

```
AverageSPACE("ИмяФ",Dir_1D,[Dir2a_beg, Dir2a_end, DirZ_beg, DirZ_end])
```

Здесь Dir\_1D = x, y, z - направление одномерного распределения (осреднение производится по двум другим направлениям).

Если указаны только два параметра, то осреднение по другим осям производится по всей РО. Если указаны еще четыре параметра, то значение Dir\_1D должно быть равно "x" или "y" и:

```
Dir2a_beg, Dir2a_end - границы осреднения по дополнительной оси в сечении по z,
DirZ_beg, DirZ_end  - границы осреднения по оси z.
```

В этом случае осреднение проводится по ячейкам, попавшим в площадь  $A_{xz}$  прямоугольник

```
(Dir2a_beg, DirZ_beg) - (Dir2a_end, DirZ_end).
```

Имя выходного dat-файла зависит от прямоугольника осреднения. Если осреднение проводится по всей РО, то имя файла:

<Каталог графиков>\<Префикс выходных файлов>-<ИмяФ>.dat

Если задан интервал по z, то:

<Каталог графиков>\<Префикс выходных файлов>\_z<ZC>\_<ИмяФ>.dat

где <ZC> - средняя координата z  $((DirZ\_beg + DirZ\_end)/2)$ . Если задан прямоугольник по x (или y), то:

<Каталог графиков>\<Префикс выходных файлов>\_z<ZC>\_x<XC>\_<ИмяФ>.dat

<Каталог графиков>\<Префикс выходных файлов>\_z<ZC>\_y<YC>\_<ИмяФ>.dat

Пример работы оператора при осреднении по всей РО показан на рисунке 4.7.

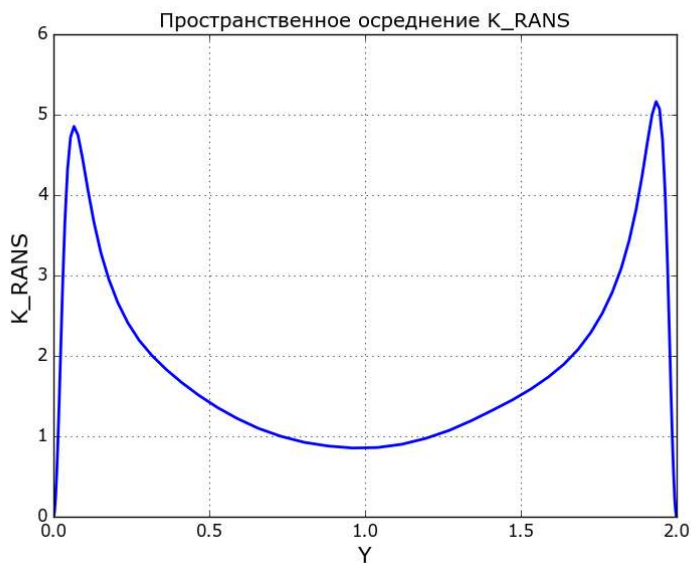


Рисунок 4.7 - Пример работы оператора AverageSpace

#### 4.4.4 Оператор осреднения в псевдополярной СК

При использовании псевдополярной системы координат можно провести осреднение по оси z и  $\theta$ . Для этого используется оператор:

AverageSPACE("ИмяФ", "Dir1D", xC, yC)

Здесь (xC, yC) - координаты центра полярной СК. Параметр Dir1D может принимать два значения: "R" и "L". В первом случае в качестве абсциссы используется радиус, во втором - расстояние до ближайшей твердой границы. Для использования второго режима необходимо в файл проекта добавить в секцию [User Variables] операторы:

```
Used(iLDIS)
.....
Calc("iLDIS") = myFORM(LDIS)
```

а в секцию [Special Data] оператор:

```
L("USE_LDIS") = .true.
```

В этом случае в файле результатов будет сохранено поле с расстоянием до ближайшей стенки (с именем iLDIS).

Заметим, что второй вариант является более удобным, поскольку при использовании в качестве абсциссы графика радиуса могут возникать искажения полей вблизи стенки. Особенно это заметно при использовании координат стенки  $y$  или  $y^+$ . Во втором варианте поле  $iLDis$  - это и есть координата  $y$ .

При использовании оператора осреднения по всей РО имя dat-файла имеет следующую структуру:

<Каталог графиков>\<Префикс выходных файлов>\_tetaz\_<ИмяФ>.dat

Пример осреднения по радиусу показан на рисунке 2.8.

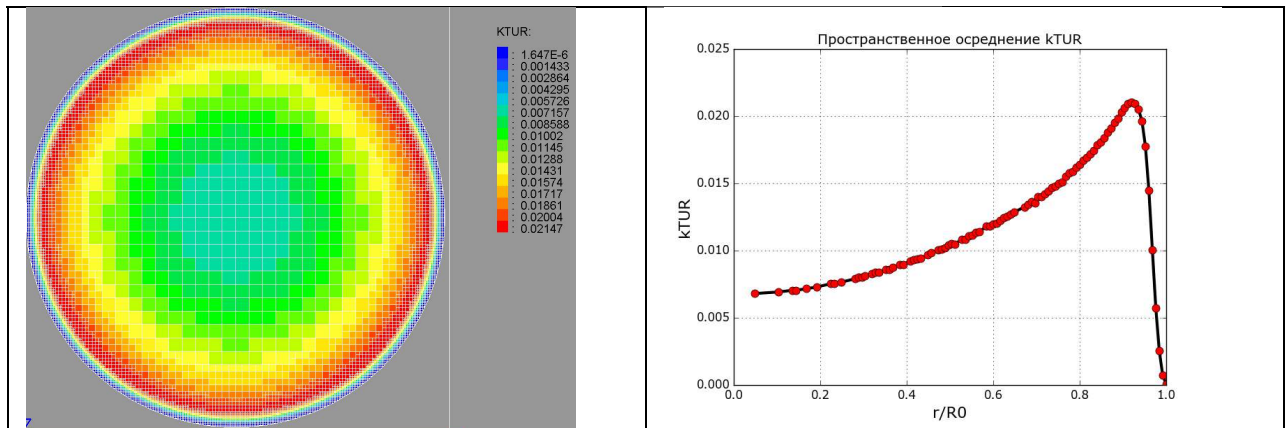


Рисунок 4.8 - Осреднение в псевдополярной СК

Можно ввести осреднение в заданном интервале сечений по  $z$ . В этом случае оператор имеет вид:

AverageSPACE("ИмяФ", "R/L", xC, yC, zBegAVE, zEndAVE )

где zBegAVE, zEndAVE - границы координаты  $z$  ячеек, используемых для осреднения.

В этом случае имя dat-файла имеет вид:

<Каталог графиков>\<Префикс выходных файлов>\_z<ZC>\_<ИмяФ>.dat

где <ZC> - средняя координата  $z$  ( $(zBegAVE + zEndAVE)/2$ ).

И, наконец, можно вести осреднение по интервалам как по оси  $z$ , так и угла  $\theta$ . Для этого используется следующая форма оператора:

AverageSPACE("ИмяФ", "R/L", xC, yC, zBegAVE, zEndAVE , TetaBegAVE, TetaEndAVE)

где TetaBegAVE, TetaEndAVE - границы изменения угла центра ячеек, используемых для осреднения. Угол измеряется в градусах в интервале  $0^\circ \dots 360^\circ$ .

В этом случае имя dat-файла имеет вид:

<Каталог графиков>\<Префикс выходных файлов>\_z<ZC>\_teta<TC>\_<ИмяФ>.dat

где <TC> - среднее значение угла ( $(TetaBegAVE + TetaEndAVE)/2$ ).

Для настройки осреднения можно использовать еще два оператора:

TolCompare = 1.e-3 / <относительная точность совпадения точек>

FracPoins = 0.02 / <параметр алгоритма LOWESS>

Для понимания смысла этих параметров опишем используемый алгоритм осреднения:

1. На первом этапе для каждой ячейки сечения  $(x, y)$  производится осреднение по  $z$  и набор этих точек переводится в зависимости  $F(r)$  или  $F(Ldis)$ . Число точек этих массивов равны числу ячеек в сечении.



2. На втором этапе производится сглаживание этих зависимостей с помощью алгоритма LOWESS (Smoothing by robust locally weighted regression) [9]. Степень сглаживания (осреднения) зависит от параметра FracPoints, значение которого необходимо подбирать экспериментально (разумный интервал 0.02 - 0.1).
3. На третьем этапе производится «сжатие» зависимости с помощью алгоритма сглаживания. В этом случае все точки по R или Ldis для которых выполняется условие  $R(i2) - R(i1) < TolCompare * (R(i2) + R(i1))$  заменяются одной точкой, а значения F в них осредняются.

#### 4.4.5 Оператор 2D осреднения

Для построения цветных контуров полей и полей вектора скоростей можно использовать файлы результатов, сохраненные в формате vtk для постпроцессора ParaView.

Для того, чтобы построить такие картинки средствами Python в постпроцессоре реализован режим вывода осредненных значений по части оси z в формате csv. В этом формате создается текстовый файл следующей структуры:

```
X,Y,<ИмяФ>
x1,y1,Ф1
.....
```

который можно считать в скрипте Python средствами пакета Panda. Для построения можно использовать функцию `tricontourf` пакета `matplotlib`.

Для создания такого файла используется оператор:

```
FieldXY("ИмяФ" [, zBegAVE,zEndAVE])
```

где `zBegAVE`, `zEndAVE` - границы координаты z ячеек, используемых для осреднения. Если эти параметры не указаны, то осреднение проводится по всем координатам z и создается файл с именем:

```
<Каталог графиков>\<Префикс выходных файлов>_fieldxy_<ИмяФ>.dat
```

Если границы осреднения по z указаны, то имя файла имеет следующую структуру:

```
<Каталог графиков>\<Префикс выходных файлов>_fieldxy_z<ZC>_<ИмяФ>.dat
```

где `<ZC>` - средняя координата z  $((ZbegAVE + ZendAVE)/2)$ .

Пример контура, построенного с помощью такого метода показан на рисунке 4.4.

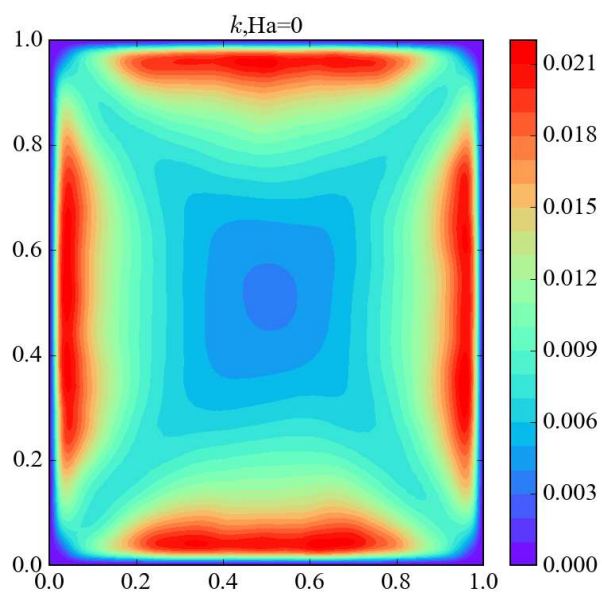


Рисунок 4.9 - Контур, созданный с помощью команды FieldXY

## 4.5 Операторы для обработки граничных и FS патчей

Эти операторы начинаются со строки

```
Oper = Patch2d
```

Они предназначены для работы с полями, сохраненными в файлах с именами:

```
<Имя варианта>_patch2d_<ИмяПатча>_NNN.dat
```

Эти файлы создаются в Решателе как на этапе расчета, так и на этапе осреднения (см. 2.9.3). Файлы содержат поля касательного напряжения  $\tau_w(i2,iz)$ , температуры стенки  $T_w(i2,iz)$  и энтальпии на стенке  $h_w(i2,iz)$  по граничному патчу прямоугольной формы. При использовании FS-патча дополнительно сохраняется поле  $q_w(i2,iz)$ . На этапе расчета записываются мгновенные значения, на этапе осреднения - поля, осредненные по времени. Смысл первой переменной зависят от типа патча: это либо индекс координаты по оси  $x$  ( $ix$ ),  $y$  ( $iy$ ) или угла  $\theta$  ( $iTeta$ ) для псевдополярной СК.

Для загрузки файла служит оператор:

```
LoadPatch2D("<Имя варианта>","ИмяПатча","NNN")
```

где ИмяПатча - имя патча в команде RANS.Patch2D, NNN - символьный номер шага по времени.

### 4.5.1 Создание VTK файла

В постпроцессоре ANES в режиме  $L("ARS.OUTCELL") = .True$ . (который обычно используется в режиме LES расчета) нельзя посмотреть распределения трения и температуры на патче. Это можно сделать только в режиме  $L("ARS.OUTCELL") = .False$ . Данная команда позволяет исправить данную неприятность.

Для создания VTK-файла с полями на Patch2D патче используются команды:

```
VTK(Y_off)
VTK(X_off)
VTK(XC,YC,Radius)
```

Первый вариант используется для распределений ( $ix,iz$ ), второй - для ( $iy,iz$ ), третий - для ( $iTeta,iz$ ). Параметры  $X\_off$  и  $Y\_off$  задают значение третьей декартовой координаты для плоского патча. Параметры  $XC, YC$  и  $Radius$  задают положение центра цилиндра патча и его радиус.

Имя VTK-файла формируется следующим образом:

```
<Каталог графиков>\<Префикс выходных файлов>_patch2d_<ИмяПатча>.vtk
```

### 4.5.2 Одномерные распределения и осреднение

Патч Patch2D представляет собой двумерные структурные распределения по осям ( $x/y/Teta, z$ ). Для создания одномерных распределений по координатам  $x/y/Teta$  используется оператор:

```
AverageSPACE("ИмяПоля", [ZbegAVE,ZendAVE])
```

Здесь

ИмяПоля - трение "TauW" , температура стенки "Twall" или "Hwall" ;

Если указано только имя поля, то по z - координате проводится осреднение. Если указаны параметры zBegAVE,zEndAVE, то по этому интервалу проводится осреднение. Если нужно вывести распределение в одном сечении, то нужно указать совпадающие значения zBegAVE и zEndAVE.

Для формирования имен dat-файлов используется следующий принцип:

<Каталог графиков>\<Префикс выходных файлов>\_patch2d\_<ИмяПатча>\_zall\_<ИмяПоля>.dat

<Каталог графиков>\<Префикс выходных файлов>\_patch2d\_<ИмяПатча>\_z<ZC>\_<ИмяПоля>.dat

## Литература

1. P. Sagaut. Large Eddy Simulation for Incompressible Flows// Springer, second edition, 2002, 575 p.
2. Sofiane Benhamadouche. Large Eddy Simulation with the unstructured collocated arrangement // Ph.D. thesis, School of Mechanical, Aerospace and Civil Engineering, Manchester, 2006.
3. F. Nicoud, F. Ducros. Subgrid-scale stress modelling based on the square of the velocity gradient tensor// *Turbulence and Combustion* 62(3), 1999, pp. 183–200.
4. V. Deschamps. "Simulation numerique de la turbulence inhomogene incompressible dans un ecoulement de canal plan"// Ph.D. Thesis INPT, 1988.
5. Mikhail.L. Shur, P. R. Spalart, M. K. Strelets, A. K. Travin. A hybrid RANS-LES approach with delayed-DES and wall-modelled LES capabilities// *International Journal of Heat and Fluid Flow*, 29 (2008) 1638–1649.
6. Код Anes20хе. «Дополнительные модели кода», версия 2.24, 2019.
7. Гарбарук А.В. Моделирование турбулентности в расчетах сложных течений: учебное пособие / А.В. Гарбарук, М.Х. Стрелец, М.Л. Шур – СПб: Изд-во Политехн. Ун-та, 2012. – 88 с.
8. М.С. Грицкевич. Расчет турбулентных пристенных течений с использованием зонного RANS-LES подхода с объемным источником турбулентных пульсаций// ДИССЕРТАЦИЯ на соискание ученой степени кандидата физико-математических наук, Санкт-Петербург, 2012, 120 с.
9. W.S. Cleveland. Robust Locally Weighted Regression and Smoothing Scatterplots // *Journal of American Statistical Association*, 1979, vol. 74, n. 368, pp. 829 - 936.
10. Wolfshtein M. The velocity and temperature distribution in one-dimensional flow with turbulence augmentation and pressure gradient // *Intern. J. Heat and Mass Transfer*. 1969. V. 12. P. 301–318.
11. Код Anes20хе. «Математические модели кода», версия 2.24, 2019.
12. Norris L., Reynolds W. Turbulent channel flow with a moving wavy boundary // Dept. of Mech. Eng. Stanford University, California, 1975. Report No. FM-10.
13. Abramov A, Ivanov N., Smirnov E. Numerical analysis of turbulent Rayleigh–Benard convection in confined enclosures Benard convection in confined enclosures using a hybrid RANS/LES approach [Электронный ресурс]. URL: <http://aero.spbstu.ru/> (дата обращения 10.04.2018).
14. H. Kobayashi. “The subgrid-scale models based on coherent structures for rotating homogeneous turbulence and turbulent channel flow,” *Phys. Fluids* 17, 045104, 2005.