

Лабораторная работа №5

«Течение в пограничном слое»

Установившееся двумерное течение несжимаемой жидкости в ламинарном пограничном слое (ПС) описывается системой уравнений:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \quad u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = U \frac{dU}{dx} + \nu \frac{\partial^2 u}{\partial y^2}, \quad 0 < x \leq x_{max}, 0 < y \leq y_{max} \quad (1)$$

с дополнительными условиями

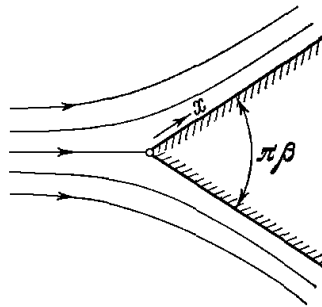
$$\begin{aligned} u &= U(0), \quad v = 0 \quad \text{при } x = 0, \\ u &= 0, \quad v = V_w \quad \text{при } y = 0, \\ u &= U(x) \quad \text{при } y = y_{max}. \end{aligned} \quad (2)$$

Здесь x и y – связанные с обтекаемой стенкой координаты, u и v – продольная и поперечная компоненты скорости, $U(x)$ – скорость внешнего потока, V_w – скорость вдува на стенке, $\nu = \mu/\rho$ – кинематическая вязкость жидкости.

Силовое воздействие потока на обтекаемое тело определяется напряжением трения на стенке

$$\tau_w = \mu \left(\frac{\partial u}{\partial y} \right)_{y=0}. \quad (3)$$

В частном случае обтекания клина с углом $\pi\beta$



скорость внешнего потока $U(x) = U_1 x^m$, $\frac{2m}{m+1} = \beta$, и задача имеет точное решение ([1], стр. 546-547):

$$\tau_{we}(x) = \Phi''(0) \sqrt{\frac{\mu \rho U^2(x) U'(x)}{\beta}},$$

где $\Phi''(0)$ дается таблицей

β	0	0.1	0.2	0.3	0.4	0.5	0.6	0.8	1.0	1.2	1.6	2
$\Phi''(0)$	0.4696	0.5870	0.6869	0.7748	0.8542	0.9277	0.9960	1.120	1.2326	1.336	1.521	1.687

Задание. С помощью метода конечных разностей (КР) провести численное моделирование течения в ПС над клином.

Использовать равномерную расчетную сетку. Исследовать влияние шагов сетки Δx (при постоянном Δy) и Δy (при постоянном Δx) на погрешность Δc_f (в %) определения местного коэффициента трения для непроницаемой стенки

$$c_f(x) = \frac{\tau_w(x)}{\rho U_1^2/2}, \quad \Delta c_f = |c_{fe} - c_f|/c_{fe}.$$

Провести графическое и табличное сравнение численного и точного решения. Определить действующую на тело силу сопротивления трения X_a и ее

коэффициент $c_{\chi a}$. Исследовать влияние на $c_{\chi a}$ относительной скорости вдува $V_w/U=0.005\dots 0.025$.

Варианты исходных данных приведены в таблице:

Вариант	Жидкая среда	$t, ^\circ\text{C}$	Угол клина	$U_1, \text{м/с}$	Вариант	Жидкая среда	$t, ^\circ\text{C}$	Угол клина	$U_1, \text{м/с}$
1	вода	10	$\pi/16$	0.50	2	азот	19	$\pi/2$	0.90
3	спирт	20	$\pi/15$	0.55	4	аммиак	21	$2\pi/3$	0.88
5	нефть	30	$\pi/14$	0.60	6	аргон	23	$3\pi/4$	0.86
7	бензин	28	$\pi/13$	0.65	8	ацетилен	25	$4\pi/5$	0.84
9	керосин	18	$\pi/12$	0.70	10	гелий	27	$5\pi/6$	0.82
11	бензол	8	$\pi/11$	0.75	12	кислород	29	$2\pi/15$	0.80
13	ацетон	12	$3\pi/14$	0.62	14	воздух	9	$5\pi/16$	0.78
15	толуол	22	$2\pi/13$	0.53	16	водород	11	$5\pi/12$	0.72
17	гексан	32	$2\pi/11$	0.47	18	этан	13	$4\pi/15$	0.42
19	масло	17	$5\pi/14$	0.32	20	пропан	15	$3\pi/8$	0.37

Указания. Используем неявную двухслойную КР-схему с линеаризацией на равномерной расчетной сетке $\{(x^n, y_j)\}$:

$$\frac{1}{2} \left[\frac{u_j^{n+1} - u_j^n}{\Delta x} + \frac{u_{j-1}^{n+1} - u_{j-1}^n}{\Delta x} \right] + \frac{v_j^{n+1} - v_{j-1}^{n+1}}{\Delta y} = 0 \quad (4)$$

$$u_j^n \frac{u_j^{n+1} - u_j^n}{\Delta x} + v_j^n \frac{u_j^{n+1} - u_{j-1}^{n+1}}{\Delta y} = U^{n+1} \frac{U^{n+1} - U^n}{\Delta x} + v \frac{u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}}{\Delta y^2} \quad (5)$$

Вначале из (5), записанного для внутренних узлов ($1 \leq j \leq N_y - 2$), и граничных условий (ГУ)

$$u_0^{n+1} = 0 \quad (j = 0), \quad u_{N_y-1}^{n+1} = U^{n+1} \quad (j = N_y - 1)$$

находим приближенные значения $\{u_j^{n+1}\}$. Возникающую при этом СЛАУ решаем методом прогонки.

Затем из ГУ и (4) определяем

$$v_0^{n+1} = V_w \quad (j = 0),$$

$$v_j^{n+1} = v_{j-1}^{n+1} - \frac{\Delta y}{2\Delta x} [u_j^{n+1} - u_j^n + u_{j-1}^{n+1} - u_{j-1}^n] \quad (1 \leq j \leq N_y - 1).$$

Приводимая ниже программа рассчитывает течение воздуха в ПС на клине ($\beta = 0.3$) с текстовым и графическим выводом результатов:

```
# Расчет течения в ПС
import numpy as np
import matplotlib.pyplot as plt

def Uefun(x_val):
    return Uext * x_val ** am

def dUefun(x_val):
    return am * Uext * x_val ** (am - 1)

def BLIMPL1(x1, x2, Ny, u1, v1, u2, v2):
    # Неявная двухслойная схема
    c = np.zeros(Ny)
    d = np.zeros(Ny)
    e = np.zeros(Ny)
    f = np.zeros(Ny)
```

```

c[0] = 0
d[0] = 1
e[0] = 0
f[0] = 0
for j in range(1, Ny - 1):
    c[j] = -dxy1 * v1[j] - dxy2
    d[j] = u1[j] + 2 * dxy2
    e[j] = dxy1 * v1[j] - dxy2
    f[j] = u1[j] ** 2 + Uefun(x2) * (Uefun(x2) - Uefun(x1))
c[Ny-1] = 0
d[Ny-1] = 1
e[Ny-1] = 0
f[Ny-1] = Uefun(x2)
u2[:] = PROGON(Ny, c, d, e, f)
v2[0] = Vw
for j in range(1, Ny):
    v2[j] = v2[j-1] - dxy3 * (u2[j] - u1[j] + u2[j-1] - u1[j-1])

def PROGON(N, c, d, e, f):
    # Метод прогонки для СЛАУ (трёхдиагональная матрица)
    alf = np.zeros(N)
    bet = np.zeros(N)
    x_sol = np.zeros(N)
    # Прямой ход
    //////////////////////////////////
    # Обратный ход
    //////////////////////////////////
    return x_sol

def OUTPUT(x_val, Ny, y, u_arr, v_arr, x_vals, u_norms, v_norms):
    # Вывод данных в сечении x
    tau0 = amu * (-u_arr[2] + 4 * u_arr[1] - 3 * u_arr[0]) / (2 * dy)
    cf = tau0 / (0.5 * ro * Uext ** 2)
    Ux = Uefun(x_val)
    Rex = Ux * x_val / anu
    tau0e = phi20 * np.sqrt(amu * ro * Ux * Ux * dUefun(x_val) / bet)
    cfe = tau0e / (0.5 * ro * Uext ** 2)
    print(f"{x_val:>10.4e} {Ux:>10.4e} {Rex:>10.4e} {cf:>10.4e} {cfe:>10.4e}")
    with open('bolay1.txt', 'a') as f:
        f.write(f"{x_val:>10.4e} {Ux:>10.4e} {Rex:>10.4e} {cf:>10.4e} {cfe:>10.4e}\n")
    # Сохранение данных для графиков
    u_norm = u_arr / Ux
    v_norm = v_arr / Ux
    x_vals.append(x_val)
    u_norms.append(u_norm.copy())
    v_norms.append(v_norm.copy())

# Параметры
ro = 1.225 # плотность
amu = 1.7894e-5 # динам. вязкость
anu = amu / ro
Uext = 1
bet = 0.3
phi20 = 0.7748 # Phi''(0)
am = bet / (2 - bet)
Vw = 0.0 * Uext # Скорость вдува
xmin = 1e-1
Recr = 1e5
xmax = Recr * anu / Uext
Remax = Uext * xmax / anu
ymax = 15 * xmax / np.sqrt(Remax)
print(f'xmax={xmax:.3f}')
print(f'ymax={ymax:.3f}')

```

```

print(f'Remax={Remax:.1f}')
# Сетка
Nx = 1000
Ny = 801
nout = Nx // 10
dx = (xmax - xmin) / Nx
dy = ymax / (Ny - 1)
dxy1 = dx / (2 * dy)
dxy2 = anu * dx / (dy ** 2)
dxy3 = dy / (2 * dx)
x = np.zeros(Nx + 1)
y = np.zeros(Ny)
u = np.zeros((Nx + 1, Ny))
v = np.zeros((Nx + 1, Ny))
for n in range(Nx + 1):
    x[n] = xmin + n * dx
for j in range(Ny):
    y[j] = j * dy
# Начальные условия для u на входе (n=0)
u[0, 0] = 0
for j in range(1, Ny):
    u[0, j] = Uefun(x[0])
# Списки для сохранения данных графиков
x_vals = []
u_norms = []
v_norms = []
# Открытие файла для вывода
with open('bolay1.txt', 'w') as f:
    f.write('') # Инициализация файла
# Заголовок для вывода
print(f"{'x':>10} {'U(x)':>10} {'Rex':>10} {'cf':>10} {'cfe':>10}")
with open('bolay1.txt', 'a') as f:
    f.write(f"{'x':>10} {'U(x)':>10} {'Rex':>10} {'cf':>10} {'cfe':>10}\n")

# Основной цикл по слоям
for n in range(Nx):
    # Решение на новом слое
    BLIMPL1(x[n], x[n+1], Ny, u[n, :], v[n, :], u[n+1, :], v[n+1, :])
    if (n + 1) % nout == 0:
        # Текстовый вывод
        OUTPUT(x[n+1], Ny, y, u[n+1, :], v[n+1, :], x_vals, u_norms, v_norms)

# Графический вывод
dpi = 200
fig1, ax1 = plt.subplots(dpi=dpi)
ax1.set_xlabel('u/U, x')
ax1.set_ylabel('y')
ax1.grid(True)
fig2, ax2 = plt.subplots(dpi=dpi)
ax2.set_xlabel('v/U, x')
ax2.set_ylabel('y')
ax2.grid(True)
# Профили u и v
for i in range(len(x_vals)):
    x_val = x_vals[i]
    u_norm = u_norms[i]
    v_norm = v_norms[i]
    ku = 0.08
    ax1.plot(x_val + ku * u_norm, y, 'b-', linewidth=1)
    ax1.plot([x_val, x_val], [y[0], y[-1]], 'k-', linewidth=1)
    for j in range(1, Ny, 10):
        ax1.plot([x_val, x_val + ku * u_norm[j]], [y[j], y[j]], 'b-', linewidth=1)
    kv = 4
    ax2.plot(x_val + kv * v_norm, y, 'r-', linewidth=1)

```

```

ax2.plot([x_val, x_val], [y[0], y[-1]], 'k-', linewidth=1)
for j in range(1, Ny, 10):
    ax2.plot([x_val, x_val+kv*v_norm[j]], [y[j], y[j]], 'r-', linewidth=1)
fig1.savefig('u(y)', dpi=dpi, bbox_inches='tight')
fig2.savefig('v(y)', dpi=dpi, bbox_inches='tight')
# Линии тока
fig3, ax3 = plt.subplots(dpi=dpi)
X, Y = np.meshgrid(x, y)
U, V = u.T, v.T
plt.streamplot(X, Y, U, V, density=2, color=np.hypot(U, V), cmap='jet')
ax3.set_xlabel('x')
ax3.set_ylabel('y')
plt.colorbar()
fig3.savefig('streamlines', dpi=dpi, bbox_inches='tight', pad_inches=0.1)
plt.show()

```

Параметры функции «BLIMPL1», реализующей КР-схему: входные «x1», «u1», «v1» и «Ny» – x-координата и массивы компонент скорости в сечении $x=x1$, а также число узлов сетки по y; выходные «x2», «u2» и «v2» – x-координата и массивы компонент скорости в новом сечении $x=x2$.

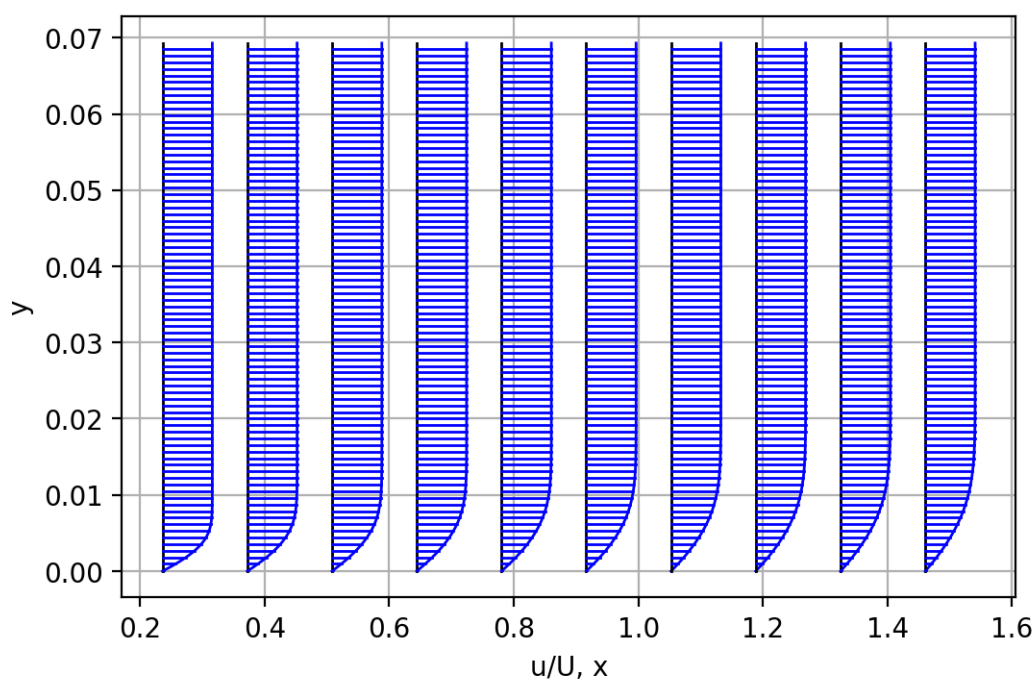
Результаты запуска программы:

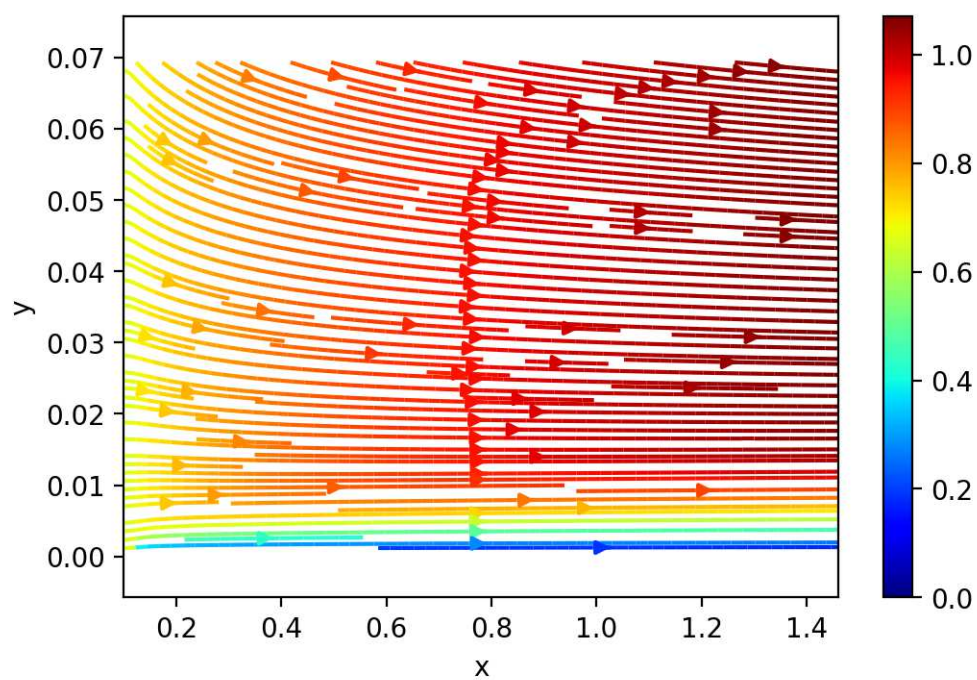
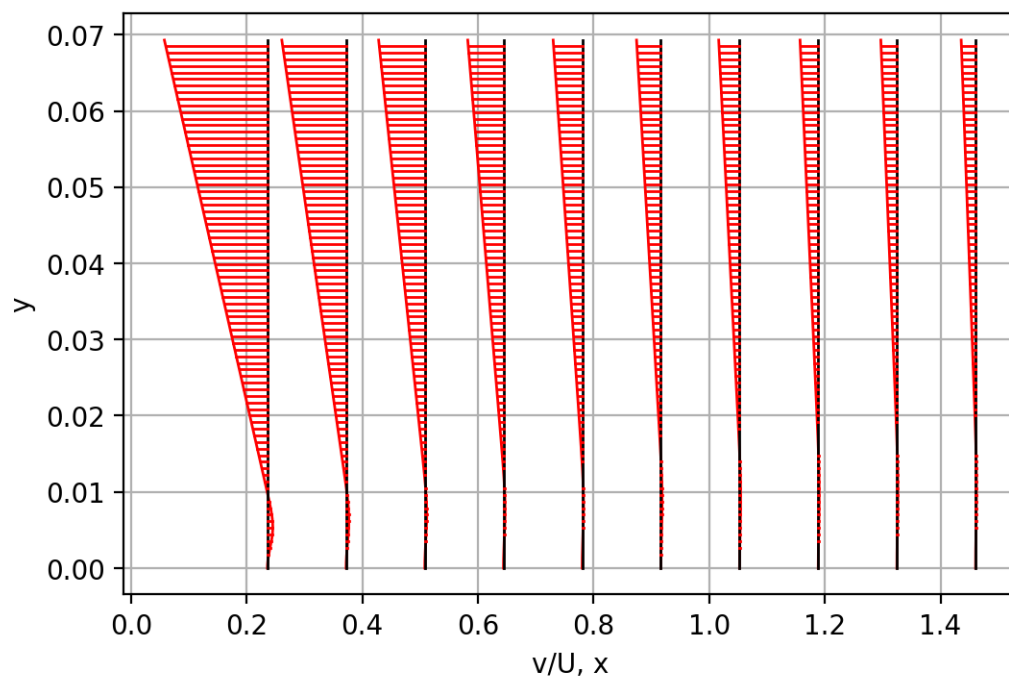
```

xmax=1.461
ymax=0.069
Remax=100000.0

```

x	U(x)	Rex	cf	cfe
2.3607e-01	7.7511e-01	1.2527e+04	6.7401e-03	6.3797e-03
3.7215e-01	8.3993e-01	2.1399e+04	5.8640e-03	5.7318e-03
5.0822e-01	8.8742e-01	3.0875e+04	5.3969e-03	5.3265e-03
6.4429e-01	9.2536e-01	4.0815e+04	5.0818e-03	5.0374e-03
7.8037e-01	9.5718e-01	5.1135e+04	4.8462e-03	4.8153e-03
9.1644e-01	9.8472e-01	6.1780e+04	4.6595e-03	4.6366e-03
1.0525e+00	1.0091e+00	7.2708e+04	4.5057e-03	4.4880e-03
1.1886e+00	1.0310e+00	8.3888e+04	4.3756e-03	4.3614e-03
1.3247e+00	1.0509e+00	9.5297e+04	4.2633e-03	4.2516e-03
1.4607e+00	1.0692e+00	1.0692e+05	4.1647e-03	4.1549e-03





Литература

1. Лойцянский Л.Г. Механика жидкости и газа. М. -Л.: Гостехиздат, 1950. - 676 с.