**ДОМАШНЯЯ ЗАДАЧА**

**ПОСТРОЕНИЕ ЛИНИЙ ТОКА ПРИ ПОТЕНЦИАЛЬНОМ ТЕЧЕНИИ**

**ФИО студентов:** Елонов Е. В., Черняев А. В.

**Группа:** 22-КФ

**Условие задачи**

Рассчитайте установившееся потенциальное течение идеальной жидкости по трубе прямоугольного сечения и постройте линии тока. Внутри трубы имеются препятствия заданной формы.

**Теория и решение к задаче**

Существует два метода описания движения жидкостей и газов. В методе Лагранжа течение жидкости рассматривается как движение N частиц, имеющих определенные координаты и скорости. Его развитием являются метод крупных частиц, метод частиц в ячейках, метод частиц-маркеров, метод сглаженных частиц и др. Основным в гидродинамике является метод Эйлера, требующий расчета поля скоростей V(r, t), поля плотностей p(r,t) и давлений р(r, t) путем решения системы дифференциальных уравнений. Познакомимся с этим методом на примере расчета установившегося потенциального течения идеальной жидкости.

Потенциальный характер течения означает, что линии тока незамкнуты, то есть течение невихревое (во всех точках rot V = 0). При этом поле скоростей является потенциальным; можно ввести функцию φ(x,y,z), называемую потенциалом скорости, такую, что: .

Отсюда следует div(grad φ) = 0, или:

Итак, потенциал скорости удовлетворяет уравнению Лапласа.

Ограничимся рассмотрением плоского течения несжимаемой жидкости, при котором все ее частицы перемещаются параллельно некоторой плоскости хОу. Частицы, лежащие на одном перпендикуляре к этой плоскости, имеют равные скорости. Их проекции Введем проекцию тока такую, что: .

Множество точек, для которых = const при фиксированном в силу стационарности течения, образует линию тока. При этом семейство линий равного потенциала скорости φ(x, y,) = const ортогонально (перпендикулярно) совокупности линий тока = const. Для функции тока также можно записать уравнение Лапласа:

При расчете течения следует учитывать граничные условия, определяющие значения рассчитываемых функций и их производных на границах твердых тел, обтекаемых жидкостью. Для идеальной жидкости проекция vn скорости жидкости на нормаль к поверхности тела, в системе отсчета, связанной с телом, равна 0: . Запишем уравнение Лапласа в конечных разностях:

Эта стационарная задача решается методом последовательных приближений: задается исходное распределение искомой функции (х, у), соответствующее граничным условиям, а затем осуществляется последовательность итераций, в ходе которых пересчитываются значения функции (х, у) в узлах сетки.

**Алгоритм моделирования с программным кодом**

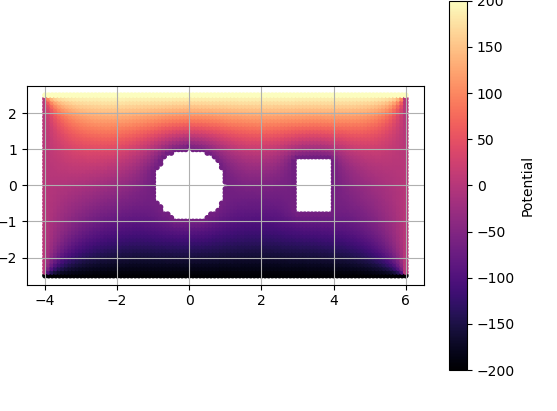
Программа состоит из 5 файлов:

1. figure.py – модели фигур-препятствий;
2. grid.py – размерная сетка;
3. main.py – запуск программы, задание констант, вывод результатов;
4. point.py – описание точек расчета;
5. solvers.py – решение уравнения, результат программы.

Листинг файла solvers.py:

from point import Point  
from grid import RectangleNet  
  
class MKE :  
 def \_\_init\_\_(self, grid, func, left=None, right=None, top=None, lower=None, border=None):   
 self.grid = grid  
 self.left = left  
 self.right = right  
 self.top = top  
 self.lower = lower  
 self.border = border  
 self.func = func  
  
  
 def solver2D(self, num\_iteration):  
  
  
 if self.left != None:  
 for pl in self.grid.left\_border:  
 pl.set\_potential(self.left(pl.x, pl.y))  
  
  
  
 if self.right != None:  
 for pr in self.grid.right\_border:  
 pr.set\_potential(self.right(pr.x, pr.y))  
  
  
 if self.top != None:  
 for pt in self.grid.top\_border:  
 pt.set\_potential(self.top(pt.x, pt.y))  
  
 if self.lower != None:  
 for plow in self.grid.lower\_border:  
 plow.set\_potential(self.lower(plow.x, plow.y))  
  
  
  
  
 for it in range(num\_iteration):  
 print(it)  
 border\_not\_set = True  
 for i in range(1, len(self.grid.net) - 1):  
 for j in range(1, len(self.grid.net[i]) - 1):  
 p = self.grid.net[i][j]  
 hx = self.grid.hx  
 hy = self.grid.hy  
  
  
 if not p.active:  
 continue  
  
 pxm = self.grid.net[i - 1][j]  
 pxp = self.grid.net[i + 1][j]  
 pym = self.grid.net[i][j - 1]  
 pyp = self.grid.net[i][j + 1]  
  
 if not(pxm.active and pxp.active and pym.active and pyp.active):  
 top = { (round(pb.x, 5), round(pb.y, 5)) for pb in self.grid.top\_border}  
lower = { (round(pb.x, 5), round(pb.y, 5)) for pb in self.grid.right\_border}  
  
  
  
if (round(p.x, 5), round(p.y, 5)) in top:  
p.psi = pxm.psi  
 elif(round(p.x, 5), round(p.y, 5)) in lower:  
p.psi = pxm.psi  
 elif border\_not\_set:  
 border\_not\_set = False  
 p.psi = pxm.psi  
  
 for pb in self.grid.border\_figure:  
 pb.psi = self.border(p.psi)  
  
  
 continue  
  
  
 p.psi = ((pxp.psi + pxm.psi) \* hy \* \*2 + (pyp.psi + pym.psi) \* hx \* \*2 - self.func(p.x, p.y) \* (hx \* hy) \* \*2) / (2 \* (hx \* \*2 + hy \* \*2))

**Результат программы**

****

**Контрольные вопросы**

1. С помощью какого уравнения мы решили задачу?
2. Назовите два метода описания движения жидкостей и газов.
3. Что означает потенциальный характер течения?
4. Где на картинке видны линии тока?
5. Какой метод используется для расчета значений в узлах сетки?
6. Какие граничные условия в задаче?