МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ

Лабораторная работа
«Расчет равновесной поверхности капли жидкости»
Вариант 2

Бобовоза Владислава Сергеевича
студента 3 курса, 6 группы
специальность «прикладная математика»

Преподаватель:
Будник А. М.

Минск, 2024

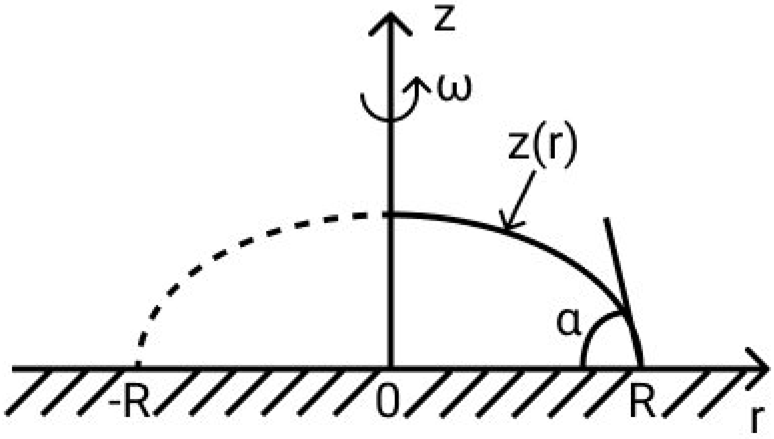
Условие задачи

Определить форму осесимметричной равновесной поверхности жидкости объема V, находящейся на горизонтальной поверхности. На жидкость действует поле центробежных сил, направленное вдоль вертикальной оси. На линии контакта жидкости с поверхностью задан угол смачивания α.

Необходимо:

1) Получить безразмерную математическую модель задачи при условии, что уравнение равновесной линии представляется в виде $z(r)$, взяв в качестве характерного размера расстояние от оси OZ до линии контакта.
2) Решить полученную дифференциальную задачу итерационно-разностным методом.
3) Найти решение при следующих значениях физических параметров:
$$\rho = 1\,\frac{\text{г}}{\text{см}^3},\ \sigma = 72.75\,\frac{\text{дин}}{\text{см}},\ \omega = 0.75\ \text{сек}^{-1},\ V = 1.02\ \text{см}^3,\ \alpha = 60°$$
4) Методом продолжения по параметру исследовать влияние на равновесную поверхность действующей на жидкость силы. Результаты представить графически.

Размерная постановка



Будем искать форму равновесной линии в виде $z(r), 0 \le r \le R$. Для рассматриваемой задачи имеем: объем V, $\overrightarrow{F_\text{ц}} = -\omega^2 r\overrightarrow{N}$. Тогда получим, что $\left(\omega^2 r; 0\right) = -\left(\frac{\partial \Pi}{\partial r}; \frac{\partial \Pi}{\partial z}\right)$. Отсюда имеем что $\frac{\partial \Pi}{\partial r} = -\omega^2 r$, тогда $\Pi = -\frac{\omega^2 r^2}{2}$.

Теперь найдем сумму кривизн для рассматриваемой задачи:

$$k_1 + k_2 = \pm\frac{1}{r}\left(\frac{rz'}{\sqrt{1+(z')^2}}\right)',$$

где $\left[\prime = \frac{d}{dr}\right]$. Но так как при увеличении r, область жидкости остается справа, берем знак «плюс», тогда

$$k_1 + k_2 = \frac{1}{r}\left(\frac{rz'}{\sqrt{1 + (z')^2}}\right)'.$$

Таким образом, условие Лапласа будет иметь вид:

$$\sigma\left(\frac{1}{r}\left(\frac{rz'}{\sqrt{1 + (z')^2}}\right)'\right) = -\rho\frac{\omega^2 r^2}{2} + C$$

где $C = const$, которая может быть найдена из следующего условия:

$$V = 2\pi\int_0^R z(r)\, r\, dr.$$

Найдем $\vec{n}$ и $\overrightarrow{n_\Sigma}$:

$$\overrightarrow{n_\Sigma} = (0;\ -1),\quad \vec{n} = \left(-\frac{z'}{\sqrt{1 + (z')^2}};\ \frac{1}{\sqrt{1 + (z')^2}}\right).$$

Сформулируем граничное условие при $r = R$:

$$\cos\alpha = -\vec{n}\cdot\overrightarrow{n_\Sigma} = \frac{1}{\sqrt{1 + (z')^2}},\ \Rightarrow\ z'(R) = -\tan\alpha.$$

Сформулируем граничное условие при $r = 0$. Положим что $\alpha = 90°$, тогда получим:

$$-\vec{n}\cdot\overrightarrow{n_\Sigma} = 0 = \frac{-z'}{\sqrt{1 + (z')^2}},$$

а так как $\overrightarrow{n_\Sigma} = (-1; 0)$ и $\vec{n} = \left(-\frac{z'}{\sqrt{1+(z')^2}};\ \frac{1}{\sqrt{1+(z')^2}}\right)$, получим $z'(0) = 0$.

Также получим из графических соображений, что $z(R) = 0$.

Объединим в систему все полученные уравнения, тогда можно сказать, что размерная постановка имеет вид:

$$\begin{cases} \dfrac{1}{r}\left(\dfrac{rz'}{\sqrt{1+(z')^2}}\right)' = -\dfrac{\rho\omega^2 r^2}{2\sigma} + C \\ z'(0) = 0 \\ z'(R) = -\tan\alpha \\ z(R) = 0 \\ V = 2\pi\displaystyle\int_0^R z(r)\, r\, dr \end{cases}, \quad \begin{matrix} 0 \le r \le R \\ 0 < \alpha < \dfrac{\pi}{2} \end{matrix}$$

Безразмерная постановка

Построим безразмерную задачу из размерной, путем обезразмеривания по радиусу R. Тогда получим:

$$\bar{r} = \frac{r}{R}, \bar{z} = \frac{z}{R}, 0 \le \bar{r} \le 1,$$
$$q = C \cdot R,$$
$$\frac{dz}{dr} = \frac{d\bar{z}}{d\bar{r}}, \frac{d^2 z}{dr^2} = \frac{1}{R}\frac{d^2\bar{z}}{d\bar{r}^2}.$$

Переобозначим $\bar{r} = r$ и $\bar{z} = z$, тогда получим следующую систему:

$$\begin{cases} \dfrac{1}{r}\left(\dfrac{rz'}{\sqrt{1+(z')^2}}\right)' = -P\dfrac{r^2}{I} + q \\ z'(0) = 0 \\ z'(1) = -\tan\alpha \\ z(1) = 0 \\ V = 2\pi R^3\displaystyle\int_0^1 zr\, dr, \; => I = 2\pi\int_0^1 zr\, dr = \dfrac{V}{R^3}. \end{cases} \quad \begin{matrix} 0 \le r \le 1 \\ 0 < \alpha < \dfrac{\pi}{2}, \end{matrix}$$

где

$$P = \frac{\omega^2 \rho V}{2\sigma}.$$

Домножим первое уравнение системы на r и проинтегрируем от 0 до 1:

$$\int_0^1 \left(\frac{rz'}{\sqrt{1+(z')^2}}\right)' dr = -\int_0^1 P\frac{r^3}{I}dr + \int_0^1 qr\, dr, =>$$

$$\left.\frac{rz'}{\sqrt{1+(z')^2}}\right|_0^1 = -\frac{P}{I}\cdot\left.\frac{r^4}{4}\right|_0^1 + q\left.\frac{r^2}{2}\right|_0^1, =>$$

$$\frac{z'(1)}{\sqrt{1+(z'(1))^2}} = \frac{-\tan\alpha}{\sqrt{1+\tan^2\alpha}} = -\sin\alpha, =>$$

$$-\sin\alpha = -\frac{P}{4I} + \frac{q}{2}, =>$$

$$q = -2\sin\alpha + \frac{P}{2I}.$$

Подставим полученное значение q в систему и упростим, тогда получим следующую систему:

$$\begin{cases} \dfrac{1}{r}\left(\dfrac{rz'}{\sqrt{1+(z')^2}}\right)' = \dfrac{P\left(1-2r^2\right)}{2I} - 2\sin\alpha \\ z'(0) = 0 \\ z'(1) = -\tan\alpha \\ z(1) = 0 \\ I = 2\pi\displaystyle\int_0^1 zr\,dr. \end{cases} \quad \begin{matrix} 0 \le r \le 1 \\ 0 < \alpha < \dfrac{\pi}{2}, \end{matrix}$$

Таким образом была получена безразмерная постановка задачи.

Решение итерационно-разностным методом

Для начала зададим равномерную сетку $\omega_h = \left\{r_i = ih; h = \frac{1}{N}; i = \overline{0;N}\right\}$, где N – число разбиений отрезка [0; 1]. Интеграл I будем вычислять по квадратурной формуле трапеций:

$$I = 2\pi h\left(\frac{r_0z_0 + r_Nz_N}{2} + \sum_{i=1}^{N-1} r_iz_i\right) = \begin{bmatrix} r_0 = 0 \\ z_N = 0 \end{bmatrix} = 2\pi h\sum_{i=1}^{N-1} r_iz_i.$$

Теперь, рассмотрим первое уравнение безразмерной постановки.
Перепишем его в следующем виде:

$$\left(k(r)z'\right)' = -\frac{P}{I}r^3 + Cr,$$

где $k(r) = \frac{r}{\sqrt{1+(z')^2}}$, $C = \frac{P}{2I} - 2\sin\alpha$.

Распишем это уравнение следующим образом:

$$\frac{1}{h}\left(a_{i+1}\frac{z_{i+1}-z_i}{h} - a_i\frac{z_i - z_{i-1}}{h}\right) = -\frac{P}{I}r_i^3 + Cr_i, \; i = \overline{1, N-1}$$

$$a_i = \frac{\frac{1}{2}(r_{i-1} + r_i)}{\sqrt{1 + \left(\frac{z_i - z_{i-1}}{h}\right)^2}}, i = \overline{1,N-1}.$$

Аппроксимируем условия $z'(0) = 0, z'(1) = -\tan\alpha, z(1) = 0$ следующим образом:

$$0 = z'(0) \approx \frac{z_1 - z_0}{h} + \frac{h}{2} z''(0),$$

$$-\tan\alpha = z'(1) \approx \frac{z_N - z_{N-1}}{h} + \frac{h}{2} z''(1).$$

где для нахождения z'' раскроем производную в первом уравнении безразмерной постановки задачи, а после чего выразим необходимое и получим:

$$z''(r) = \left(1 + (z')^2\right)^{\frac{3}{2}} \left(-\frac{P}{I} r^2 + C\right) - \frac{z'}{r}\left(1 + (z')^2\right).$$

Получим $z''(0)$ и $z''(1)$:

$$z''(0) = C = \frac{P}{2I} - 2\sin\alpha$$

$$z''(1) = \left(1 + (\tan\alpha)^2\right)^{\frac{3}{2}} \left(-\frac{P}{I} + C\right) + \tan\alpha\left(1 + (\tan\alpha)^2\right) =$$

$$= \frac{1}{(\cos\alpha)^3}\left(-\frac{P}{I} + C\right) + \frac{\sin\alpha}{(\cos\alpha)^3} = \frac{1}{(\cos\alpha)^3}\left(-\frac{P}{I} + C + \sin\alpha\right) =$$

$$= \frac{1}{(\cos\alpha)^3}\left(-\frac{P}{I} + \frac{P}{2I} - 2\sin\alpha + \sin\alpha\right).$$

Тогда, получим следующее:

$$\frac{z_1 - z_0}{h} = \frac{h}{2}\left(\frac{P}{2I} - 2\sin\alpha\right)$$

$$z_N = 0$$

$$z_{N-1} = h\tan\alpha + \frac{h^2}{2}\left(\frac{1}{(\cos\alpha)^3}\left(-\frac{P}{I} + \frac{P}{2I} - 2\sin\alpha + \sin\alpha\right)\right).$$

Таким образом, можно выписать следующую разностную схему второго порядка аппроксимации:

$$
\begin{cases}
\frac{1}{h^2}\left(a_{i+1}\left(z_{i+1}-z_i\right)-a_i\left(z_i-z_{i-1}\right)\right)=-\frac{P}{I}r_i^3+Cr_i,\ i=\overline{1,N-1} \\
a_i=\dfrac{\frac{1}{2}\left(r_{i-1}+r_i\right)}{\sqrt{1+\left(\frac{z_i-z_{i-1}}{h}\right)^2}},\ i=\overline{1,N-1} \\
\frac{z_1-z_0}{h}=\frac{h}{2}C \\
z_{N-1}=h\tan\alpha+\frac{h^2}{2}\left(\frac{1}{(cos\,\alpha)^3}\left(-\frac{P}{I}+C+\sin\alpha\right)\right) \\
z_N=0 \\
I=2\pi h\sum_{i=1}^{N-1}r_iz_i \\
C^{(k)}=\frac{P}{2I^{(k)}}-2\sin\alpha
\end{cases}
$$

Данная схема получилась нелинейной, поэтому расчеты нужно вести по следующему алгоритму:

1) Зададим начальное приближение. В нашем случае, за начальное приближение удобно взять $z_i^{(0)}=r_i,\ i=\overline{0,N}$;
2) Вычисляем $C,\ P,\ a_i,\ i=\overline{1,N-1}$;
3) Составляем трехдиагональную матрицу и решаем ее методом прогонки;
4) После решения системы, получим следующее приближение и проверяем следующий критерий остановки:

$$\left\|z^{(k+1)}-z^{(k)}\right\|\leq\varepsilon,\ <=>\ \max_i\left|z_i^{(k+1)}-z_i^{(k)}\right|\leq\varepsilon.$$

Построим для данной схемы метод последовательных приближений:

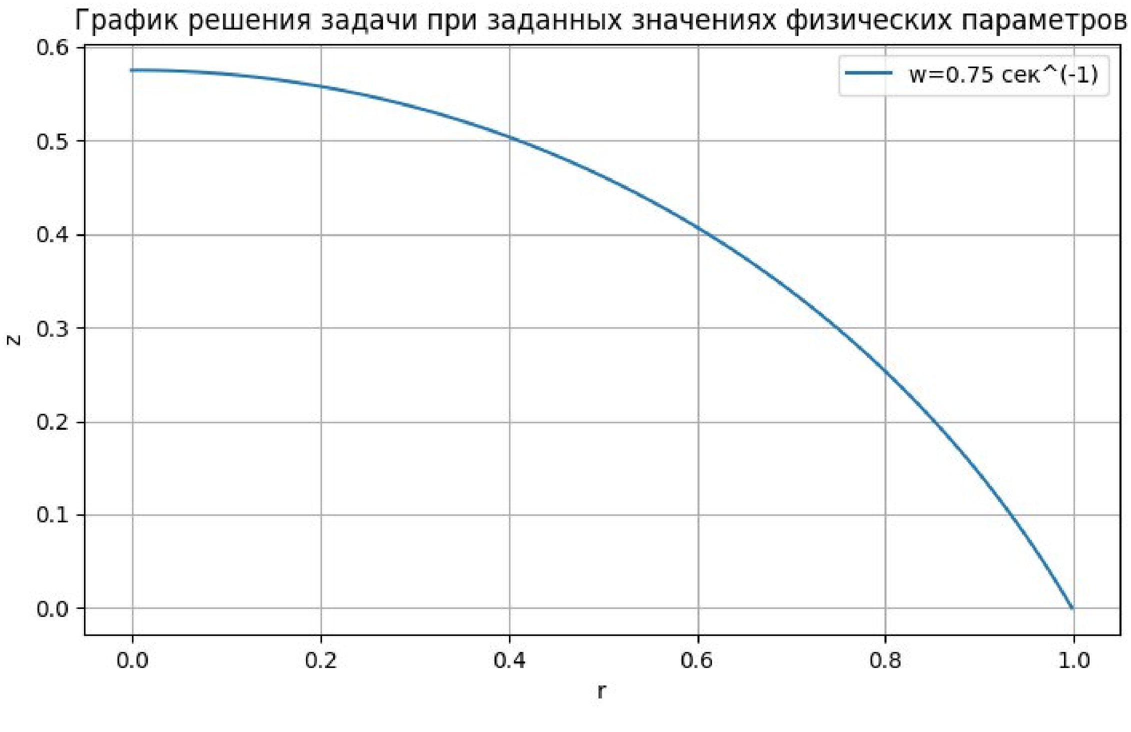
$$
\begin{cases}
z_{i-1}^{(k+1)}\dfrac{a_i^{(k)}}{h^2}-z_i^{(k+1)}\dfrac{a_{i+1}^{(k)}+a_i^{(k)}}{h^2}+z_{i+1}^{(k+1)}\dfrac{a_{i+1}^{(k)}}{h^2}=-\dfrac{P}{I^{(k)}}r_i^3+C^{(k)}r_i,\ i=\overline{1,N-1}\\
a_i^{(k)}=\dfrac{\frac{1}{2}(r_{i-1}+r_i)}{\sqrt{1+\left(\dfrac{z_i^{(k)}-z_{i-1}^{(k)}}{h}\right)^2}},\ i=\overline{1,N-1}\\
\dfrac{z_1^{(k+1)}-z_0^{(k)}}{h}=\dfrac{h}{2}C^{(k)}\\
z_{N-1}^{(k+1)}=h\tan\alpha+\dfrac{h^2}{2}\left(\dfrac{1}{(cos\,\alpha)^3}\left(-\dfrac{P}{I^{(k)}}+C^{(k)}+\sin\alpha\right)\right)\\
z_N=0\\
I=2\pi h\sum_{i=1}^{N-1}r_i z_i^{(k)}\\
C^{(k)}=\dfrac{P}{2I^{(k)}}-2\sin\alpha
\end{cases}
$$

Найти решение при заданных значениях физических параметров

В данном случае, используя реализованную программу, расположенную в приложении, получим следующие результаты:

$$P=0.003943298,$$

а график имеет следующий вид:



Исследование влияния на равновесную поверхность действующей на жидкость силы

В первую очередь, вычислим значение P при заданных значениях. Оно было вычислено в пункте выше, поэтому просто выпишем его:

$$P_0 \approx 0.003942989.$$

Увеличим угловую скорость в 10 раз, тогда P_0 увеличится в 100 раз:

$$P_1 \approx 0.394329896.$$

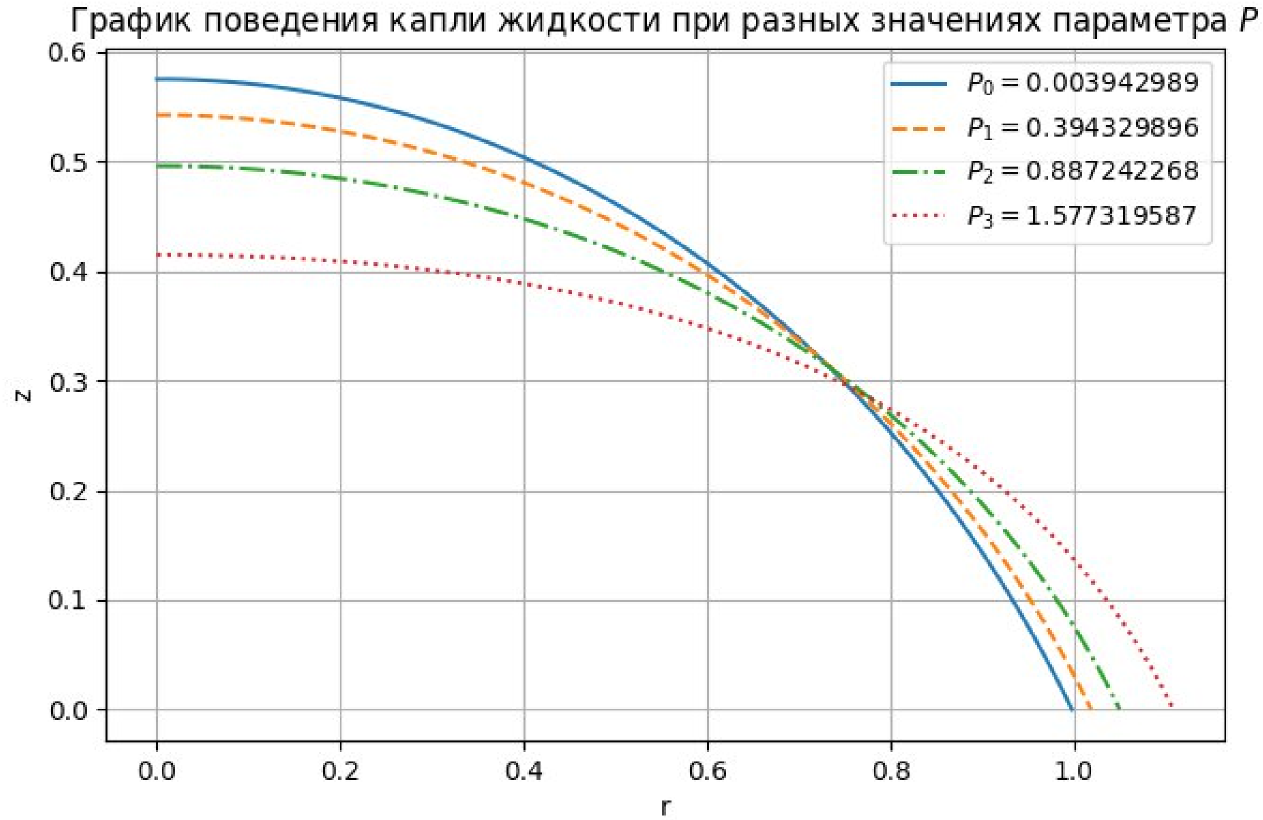
Еще раз увеличим угловую скорость в 15 раз, тогда P_0 увеличится в 225 раз:

$$P_2 \approx 0.887242268.$$

Снова увеличим угловую скорость, но уже в 20 раз, тогда P_0 увеличится в 400 раз:

$$P_3 \approx 1.577319587.$$

Исследуем поведение капли жидкости при этих параметрах P. Для построения графиков будет делить r, z на $I^{\frac{1}{3}}$, которое было получено на последней итерации. В итоге, получим следующий график:



ПРИЛОЖЕНИЕ

Программная реализация решения задачи на Python

```
import matplotlib.pyplot as plt
import numpy as np

I_value = 0.0

def solve(zPrev, P, alpha):
    r = np.linspace(0, 1, N + 1)
    h = 1.0 / N
    I = 2 * np.pi * h * np.sum(r[1:N] * zPrev[1:N])
    C = (P / (2 * I)) - 2 * np.sin(np.radians(alpha))

    a_buffer = np.zeros(N + 1)
    a_buffer[1:N] = (r[1:N] - (h / 2)) / np.sqrt(1 + ((zPrev[1:N] - zPrev[:N - 1]) / h)
** 2)

    a = np.zeros(N + 1)
    b = np.zeros(N + 1)
    c = np.zeros(N + 1)
    f = np.zeros(N + 1)

    a[0] = 0
    c[0] = 1 / h
    b[0] = 1 / h
    f[0] = C * h / 2

    a[N - 1] = 0
    c[N - 1] = 1
    b[N - 1] = 0
    f[N - 1] = (h * np.tan(np.radians(alpha)) +
                (h ** 2 / 2) * (1 / (np.cos(np.radians(alpha)) ** 3)) *
                (-(P / I) + C + np.sin(np.radians(alpha))))

    a[N] = 0
    c[N] = 1
    f[N] = 0

    for i in range(1, N - 1):
        a[i] = a_buffer[i] / (h ** 2)
```

```
        c[i] = (a_buffer[i + 1] + a_buffer[i]) / (h ** 2)
        b[i] = a_buffer[i + 1] / (h ** 2)
        f[i] = ((P * (r[i] ** 3)) / I) - (C * r[i])

    global I_value
    I_value = I

    return sweepMethod(N, a, b, c, f)

def sweepMethod(N, a, b, c, f):
    alpha = np.zeros(N + 1)
    beta = np.zeros(N + 2)

    alpha[1] = b[0] / c[0]
    for i in range(1, N):
        alpha[i + 1] = b[i] / (c[i] - alpha[i] * a[i])

    beta[1] = f[0] / c[0]
    for i in range(1, N + 1):
        beta[i + 1] = (f[i] + a[i] * beta[i]) / (c[i] - a[i] * alpha[i])

    y = np.zeros(N + 1)
    y[N] = beta[N + 1]
    for i in range(N - 1, -1, -1):
        y[i] = alpha[i + 1] * y[i + 1] + beta[i + 1]

    return y

if __name__ == "__main__":
    N = 100 # Число разбиений
    eps = 1e-6 # Точность
    rho = 1.0 # г/см^3
    sigma = 72.75 # дин/см
    omega = 0.75 # 1/сек
    V = 1.02 # см^3
    alpha = 60 # градусов

    # ---------------------------------------------------------------------
    P = omega**2 * rho * V / (2 * sigma)
    print(f"P0 = {P}")
```

```
zNext = np.linspace(0, 1, N + 1)
zNext = 1 - zNext
zPrev = np.zeros_like(zNext)

while np.max(np.abs(zNext - zPrev)) > eps:
    zPrev = zNext.copy()
    zNext = solve(zPrev, P, alpha)

print(f"I0 = {I_value}")

r0 = [i * 1/N / (I_value ** (1/3)) for i in range(N + 1)]
z0 = [zNext[i] / (I_value ** (1/3)) for i in range(N + 1)]
# ----------------------------------------------------------------------
print()
# ----------------------------------------------------------------------
omega = 0.75*10
P = omega**2 * rho * V / (2 * sigma)
print(f"P1 = {P}")

zNext = np.linspace(0, 1, N + 1)
zPrev = np.zeros_like(zNext)

while np.max(np.abs(zNext - zPrev)) > eps:
    zPrev = zNext.copy()
    zNext = solve(zPrev, P, alpha)

print(f"I1 = {I_value}")

r1 = [i * 1/N / (I_value ** (1/3)) for i in range(N + 1)]
z1 = [zNext[i] / (I_value ** (1/3)) for i in range(N + 1)]
# ----------------------------------------------------------------------
print()
# ----------------------------------------------------------------------
omega = 0.75*15
P = omega**2 * rho * V / (2 * sigma)
print(f"P2 = {P}")

zNext = np.linspace(0, 1, N + 1)
zPrev = np.zeros_like(zNext)
```

```
    while np.max(np.abs(zNext - zPrev)) > eps:
        zPrev = zNext.copy()
        zNext = solve(zPrev, P, alpha)

    print(f"I2 = {I_value}")

    r2 = [i * 1/N / (I_value ** (1/3)) for i in range(N + 1)]
    z2 = [zNext[i] / (I_value ** (1/3)) for i in range(N + 1)]
    # ----------------------------------------------------------------------
    print()
    # ----------------------------------------------------------------------
    omega = 0.75*20
    P = omega**2 * rho * V / (2 * sigma)
    print(f"P3 = {P}")

    zNext = np.linspace(0, 1, N + 1)
    zPrev = np.zeros_like(zNext)

    while np.max(np.abs(zNext - zPrev)) > eps:
        zPrev = zNext.copy()
        zNext = solve(zPrev, P, alpha)

    print(f"I3 = {I_value}")

    r3 = [i * 1/N / (I_value ** (1/3)) for i in range(N + 1)]
    z3 = [zNext[i] / (I_value ** (1/3)) for i in range(N + 1)]
    # ----------------------------------------------------------------------

    plt.plot(r0, z0, linestyle='-', label='$P_0 = 0.003942989$')
    plt.plot(r1, z1, linestyle='--', label='$P_1 = 0.394329896$')
    plt.plot(r2, z2, linestyle='-.', label='$P_2 = 0.887242268$')
    plt.plot(r3, z3, linestyle=':', label='$P_3 = 1.577319587$')
    plt.title('График поведения капли жидкости при разных значениях 
параметра $P$')
    plt.xlabel('r')
    plt.ylabel('z')
    plt.grid(True)
    plt.legend()
    plt.show()
```