Федеральное государственное бюджетное образовательное учреждение высшего образования «Московский государственный университет имени М.В.Ломоносова»

Механико-математический факультет

Кафедра газовой и волновой динамики

Курсовая работа

Задача о двух трещинах, расположенных на одной прямой

The problem of two cracks located on one straight line

Научный руководитель:

д.ф.-м.н., проф. Звягин А.В.

Выполнил:

студент 3-го курса

Гафуров Руслан Ирекович

Москва

2018

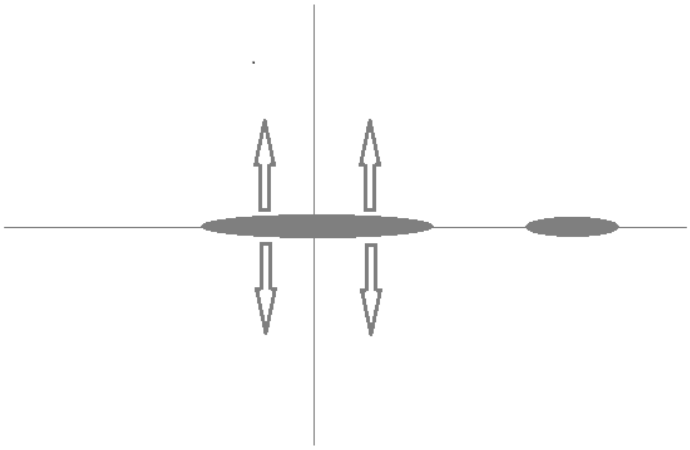
Аннотация

В данной работе рассматривается задача о двух трещинах, расположенных на одной прямой. Полученное решение позволяет наглядно увидеть, как изменяются смещения и напряжения в нашей линейно-упругой среде. Также посчитан коэффициент интенсивности напряжений на краях большей трещины. Приведены соответствующие графики.

Оглавление

Постановка задачи

Рассмотрим задачу о двух трещинах расположенных на одной прямой. Предполагаем, что трещины прямолинейные. Длина первый трещины 2L1, второй 2a. Расстояние между трещинами L2. На границе первой трещины известное напряжение –p, действующее по нормали к границе трещины. Массовых сил нет. Среда линейно-упругая. Задача двумерная.



На первой трещине напряжения по оси y равны –p. Касательные напряжения равны нулю

$$y = 0, |x| < L_1 : \sigma_{yy} = -p, \sigma_{xy} = 0$$

Вне трещины вторая компонента вектора перемещений и касательные напряжения равны нулю

$$y = 0, |x| > L_1 : u_y = 0, \sigma_{xy} = 0$$

На бесконечности

$$\sigma = 0$$

Основные уравнения

В основной системе уравнений используются уравнения равновесия

$$\begin{cases} \sigma_{xx,x} + \sigma_{xy,y} = 0, \\ \sigma_{xy,x} + \sigma_{yy,y} = 0. \end{cases}$$

условие совместности деформаций

$$2\varepsilon_{xy,xy} = \varepsilon_{xx,yy} + \varepsilon_{yy,xx}$$

выражение компонент тензора деформаций через перемещения

$$\varepsilon_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i}))$$

закон Гука

$$\varepsilon_{ij} = \frac{1+\nu}{E}\sigma_{ij} - \frac{\nu}{E}\sigma_{kk}\delta_{ij}$$

Зная выражения для деформаций, из закона Гука, можем преобразовать условие совместности деформаций

$$2\sigma_{xy,xy}\frac{1+\nu}{E} = \frac{1+\nu}{E}(\sigma_{xx,yy} + \sigma_{yy,xx}) - \frac{\nu}{E}(\sigma_{xx,yy} + \sigma_{yy,yy} + \sigma_{xx,xx} + \sigma_{yy,xx})$$

Далее, используя уравнения равновесия, получаем выражение

$$\Delta(\sigma_{xx} + \sigma_{yy}) = 0$$

Введем функцию напряжений U

$$U_{,yy} = \sigma_{xx}$$

$$U_{,xx} = \sigma_{yy}$$

$$-U_{,xy} = \sigma_{xy}$$

Тогда предыдущее выражение примет вид

$$\Delta\Delta U = 0$$

В полученном уравнении произведем замену переменных

$$z = x + iy \quad \bar{z} = x - iy$$

Операторы дифференцирования примут следующий вид

$$\frac{\partial}{\partial x} = \frac{\partial}{\partial z} + \frac{\partial}{\partial \bar{z}}; \quad \frac{\partial}{\partial y} = i(\frac{\partial}{\partial z} - \frac{\partial}{\partial \bar{z}})$$

$$\frac{\partial^2}{\partial x^2} = \frac{\partial^2}{\partial z^2} + 2\frac{\partial^2}{\partial z \partial \bar{z}} + \frac{\partial^2}{\partial \bar{z}^2}; \quad \frac{\partial^2}{\partial y^2} = -(\frac{\partial^2}{\partial z^2} - \frac{\partial^2}{\partial z \partial \bar{z}} + \frac{\partial^2}{\partial \bar{z}^2})$$

$$\Delta = 4\frac{\partial^2}{\partial z \partial \bar{z}} \qquad \Delta\Delta = 16\frac{\partial^4}{\partial z^2 \partial \bar{z}^2}$$

Получим бигармоническое уравнение в новых координатах

$$\frac{\partial^4 U}{\partial z^2 \partial \bar{z}^2} = 0$$

Общее решение этого уравнения

$$U(z, \bar{z}) = F_1(z)\bar{z} + F_2(z) + G_1(\bar{z})z + G_2(\bar{z})$$

$$U(x, y) = \frac{1}{2}[\varphi(z)\bar{z} + \bar{\varphi}(\bar{z})z + \psi(\bar{z}) + \psi(z)]$$

Тогда из определения функции напряжений следует

$$\sigma_{xx} + \sigma_{yy} = \Delta U = 2\frac{\partial}{\partial \bar{z}}[\varphi'(z)\bar{z} + \bar{\varphi}(\bar{z}) + \psi'(z)] = 2[\varphi'(z) + \bar{\varphi}'(\bar{z})] = 4Re\varphi'(z)$$

$$\sigma_{yy} - \sigma_{xx} - 2i\sigma_{xy} = 2[\bar{\varphi}''(\bar{z})z + \bar{\psi}''(\bar{z})]$$

Применим комплексное сопряжение ко второму уравнению и получим формулы Колосова-Мусхелишвили

$$\sigma_{xx} + \sigma_{yy} = 4Re\varphi'(z)$$

$$\sigma_{yy} - \sigma_{xx} + 2i\sigma_{xy} = 2[\varphi''(z)\bar{z} + \psi''(z)]$$

Теперь задача для отыскания напряжений сводится к поиску функций $\varphi(z), \ \psi(z)$. Осталось выразить перемещения через эти же комплексные функции. Для этого воспользуемся законом Гука

$$\frac{E}{1+\nu}\varepsilon_{xx} = \sigma_{xx} - \nu(\sigma_{xx} + \sigma_{yy})$$

$$\frac{E}{1+\nu}\varepsilon_{yy} = \sigma_{yy} - \nu(\sigma_{xx} + \sigma_{yy})$$

$$\frac{E}{1+\nu}\varepsilon_{xy} = \sigma_{xy}$$

В эти уравнения подставляем выражение напряжений, через функцию напряжений, и выражение деформаций через перемещения

$$\frac{E}{1+\nu}\frac{\partial u_x}{\partial x} = (1-\nu)\Delta U - \frac{\partial^2 U}{\partial x^2}$$

$$\frac{E}{1+\nu}\frac{\partial u_y}{\partial y} = (1-\nu)\Delta U - \frac{\partial^2 U}{\partial y^2}$$

$$\frac{E}{2(1+\nu)}(\frac{\partial u_y}{\partial x}+\frac{\partial u_x}{\partial y})=-\frac{\partial^2 U}{\partial x\partial y}$$

Подставим выражения для оператора лапласа от функции напряжений в первые два уравнения и проинтегрируем их

$$\frac{E}{1+\nu}u_x=4(1-\nu)Re\varphi(z)-\frac{\partial U}{\partial x}+f(y)$$
$$\frac{E}{1+\nu}u_y=4(1-\nu)Im\varphi(z)-\frac{\partial U}{\partial y}+g(x)$$

Из третьего уравнения предыдущей системы следует $f'(y)+g'(x)=0$. Значит $f(y)=ky+b$, $g(x)=-ky+c$. Эти функции определяют перенос и поворот тела, как жесткого целого, поэтому можем положить их равными нулю. Далее получим уравнение

$$\frac{E}{1+\nu}(u_x+iu_y)=4(1-\nu)[Re\varphi(z)+iIm\varphi(z)]-\frac{\partial U}{\partial x}-i\frac{\partial U}{\partial y}$$

Так как

$$\frac{\partial U}{\partial x}+i\frac{\partial U}{\partial y}=2\frac{\partial U}{\partial z}=\varphi(z)+\bar{\varphi}'(\bar{z})z+\bar{\psi}'(\bar{z})$$

В конечном виде выражение для перемещений имеет вид

$$\frac{E}{1+\nu}(u_x+iu_y)=(3-4\nu)\varphi(z)-\bar{\varphi}'(\bar{z})z-\bar{\psi}'(\bar{z})$$

Аналогично для плосконапряженного состояния

$$\frac{E}{1+\nu}(u_x+iu_y)=\frac{3-\nu}{1+\nu}\varphi(z)-\bar{\varphi}'(\bar{z})z-\bar{\psi}'(\bar{z})$$

Получили систему для вычисления напряжений и деформаций

$$\sigma_{xx}+\sigma_{yy}=4Re\varphi'(z)$$
$$\sigma_{yy}-\sigma_{xx}+2i\sigma_{xy}=2[\varphi''(z)\bar{z}+\psi''(z)]$$
$$2\mu(u_x+iu_y)=k\varphi(z)-\bar{\varphi}'(\bar{z})z-\bar{\psi}'(\bar{z})$$

μ - модуль сдвига, $k=3-4\nu$ - в случае плоскодеформированного состояния, $k=\frac{3-\nu}{1+\nu}$ в случае плосконапряженного состояния.

Теперь рассмотрим следующую краевую задачу. Трещина длины $2L_1$, с действующим по нормали давлением p.

$$y = 0, |x| < L_1 : \sigma_{yy} = -p, \sigma_{xy} = 0$$
$$y = 0, |x| > L_1 : u_y = 0, \sigma_{xy} = 0$$

На бесконечности

$$\sigma = 0$$

Так как касательные напряжения равны нулю, из формул Колосова-Мусхелишвили получаем

$$Im(x\varphi''(x) + \psi'(x)) = 0$$

Введем обозначения

$$\varphi(z) = \frac{1}{2}T(z), \quad \psi(z) = -\frac{1}{2}zT'(z) + \frac{1}{2}T(z)$$

Тогда формулы Колосова-Мусхелишвили примут следующий вид

$$\sigma_{xx} = ReT'(z) - yImT''(z)$$
$$\sigma_{yy} = ReT'(z) + yImT''(z)$$
$$\sigma_{xy} = -yReT''(z)$$

А выражение для перемещений

$$u_x = \frac{1}{2\mu}(\frac{k-1}{2}ReT(z) - yImT'(z))$$
$$u_y = \frac{1}{2\mu}(\frac{k+1}{2}Im(z) - yReT'(z))$$

Получим следующую краевую задачу для функции $T(z)$

$$y = 0, |x| < L_1, \quad ReT'(z) = -p$$
$$y = 0, |x| > L_1, \quad ImT'(z) = 0$$

Введем функцию (1) $Q(z) = -i\sqrt{(z - L_1)(z + L_1)}\, T'(z)$

$$y = 0, |x| < L_1, \quad ReQ(z) = \sqrt{(L_1)^2 - x^2}\, ReT(z) = -\sqrt{(L_1)^2 - x^2}\, p$$
$$y = 0, |x| > L_1, \quad ReQ(z) = \sqrt{x^2 - (L_1)^2}\, ImT(z) = 0$$

Решением этой задачи является интеграл типа Коши

$$Q(z) = \frac{1}{2\pi i}\int_{-L_1}^{L_1} \frac{-2p\sqrt{(L_1)^2 - x^2}}{x - z}dx$$

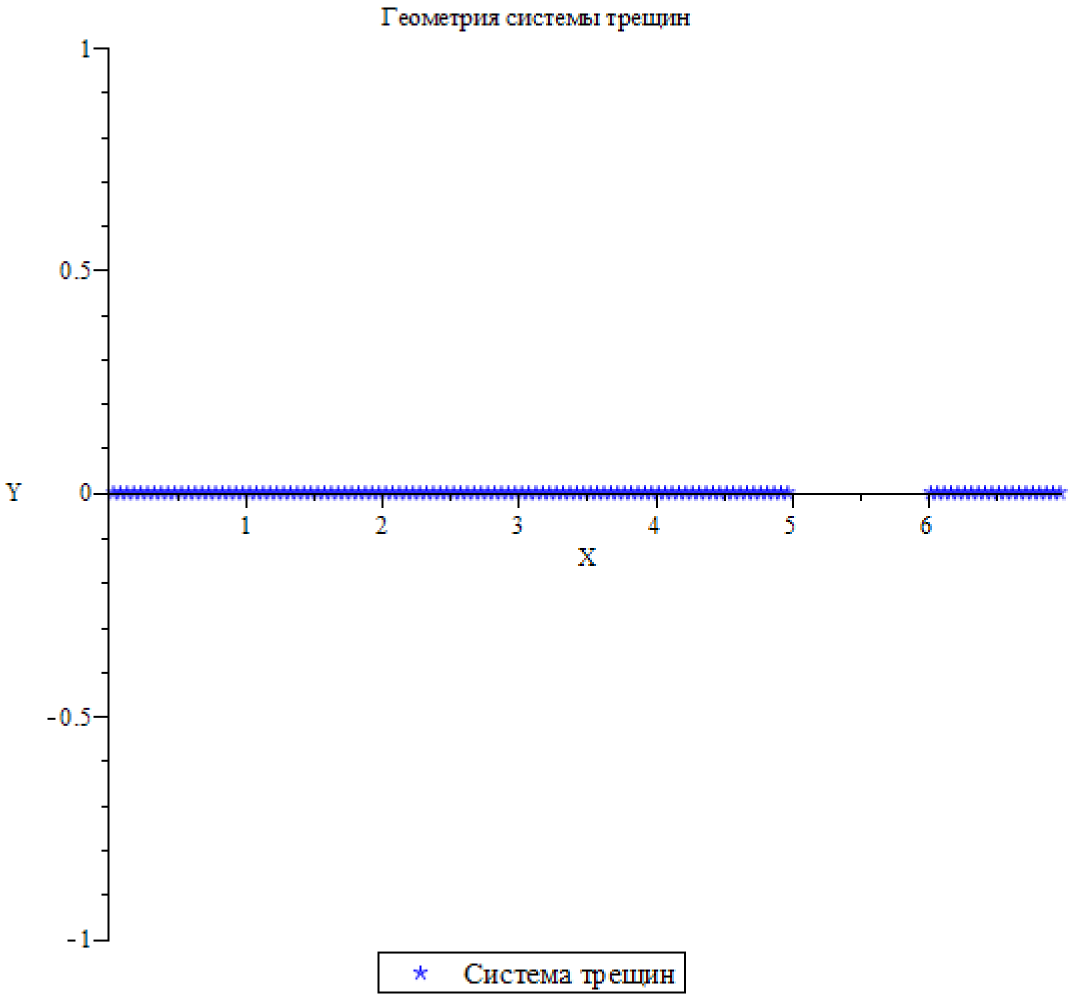
Из (1) получаем значение $T'(z)$

$$T'(z) = \frac{1}{\pi\sqrt{z^2 - (L_1)^2}} \int_{-L_1}^{L_1} \frac{p\sqrt{(L_1)^2 - x^2}}{z - x} dx$$

Методы решения

Для решения нашей задачи пользуемся методом граничных элементов. Идея заключается в том, что мы делим трещины на некоторое число элементов, длина которых должна быть много меньше характерных размеров в задаче.

В нашем случае N элементов на первой трещине, M – на второй. Также разрыв смещений на каждом элементе полагаем постоянным.



Воспользуемся выражениями для напряжений и перемещений полученными Краучем [1]

$$F(x,y) = \frac{1}{4\pi(1-\nu)}[(x-L)log\sqrt{(x-L)^2+y^2}-(x+L)log(\sqrt{(x+L)^2+y^2})-$$
$$y(arctan(y,x-L)-arctan(y,x+L))]$$

$$\sigma_{xx} = 2D_x(2F_{xy}(x,y,L)+yF_{xyy}(x,y,L))+2D_y2(F_{yy}(x,y,L)+yF_{yyy}(x,y,L))$$

(2) $$\sigma_{yy} = -2D_x yF_{xyy}(x,y,L)+2D_y(F_{yy}(x,y,L)-yF_{yyy}(x,y,L))$$

$$\sigma_{xy} = 2D_x(F_{yy}(x,y,L) + yF_{yyy}(x,y,L)) - 2D_y(F_{yy}(x,y,L) + yF_{yyy}(x,y,L))$$

$$u_x = 2D_x((1-\nu)F_y(x,y,L) + yF_{yy}(x,y,L)) - D_y((1-2\nu)F_x(x,y,L) - yF_{xy}(x,y,L))$$

$$u_y = D_x((1-2\nu)F_x(x,y,L) - yF_{xy}(x,y,L)) + D_y(2(1-\nu))F_y(x,y,L) - yF_{yy}(x,y,L)$$

Все выражения отнесены к модулю сдвига μ.

Далее составляем матрицу влияния граничных элементов друг на друга.

$$A_{ij} = \begin{pmatrix} A_{11} & \dots & A_{1\,2(N+M)} \\ \dots & \dots & \dots \\ A_{2(N+M)\,1} & \dots & A_{2(N+M)\,2(N+M)} \end{pmatrix}$$

Где A_{ij} влияние j-го элемента на i-ый. Выражения для вычисления A_{ij} описаны в [1].

Также составляем столбец правых частей

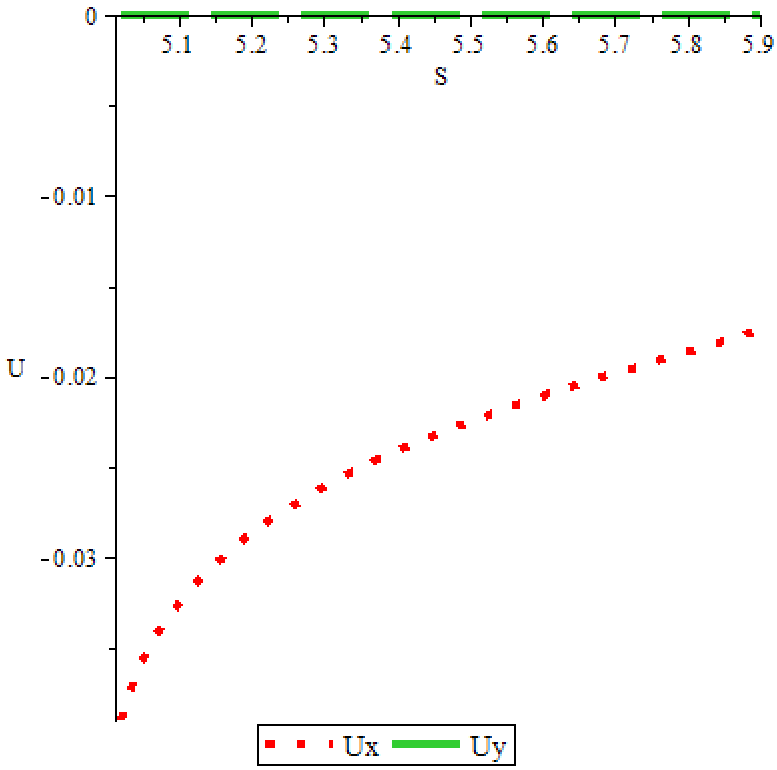
$$\vec{b} = \begin{pmatrix} -p_0 \\ \dots \\ -p_0 \\ 0 \\ \dots \\ 0 \end{pmatrix}$$

Первые N значений равны $-p_0$, что соответствует значениям σ_{yy} на первой трещине. Начиная с N+1 до N+M нули, так как σ_{yy} на второй трещине отсутствует. Следующие N+M значений также нулевые, потому что σ_{xy} равно нулю на всей прямой $y = 0$.

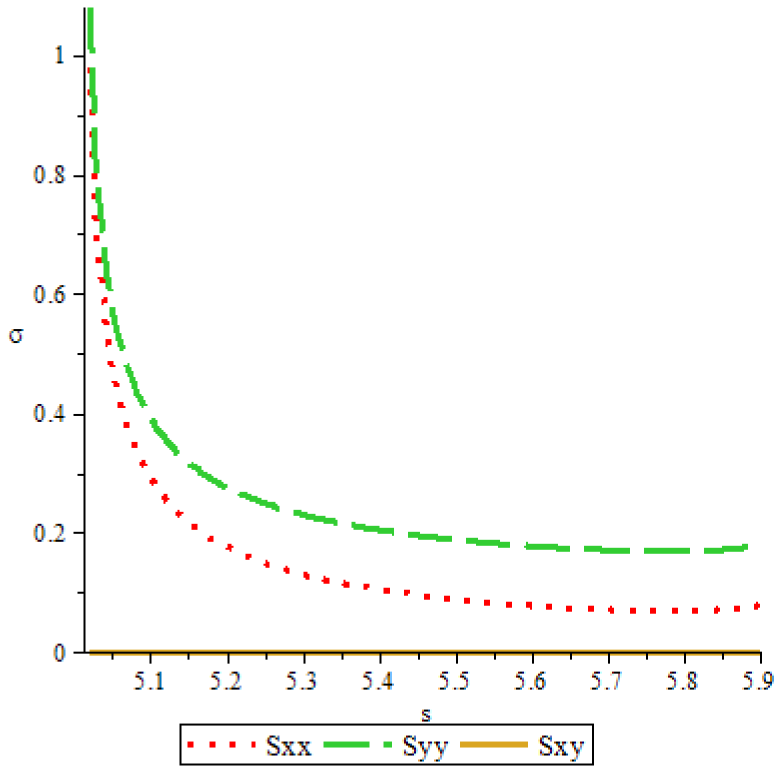
Решая систему линейных уравнений, получаем значения скачков перемещений D_x^i, D_y^i. Из формул (2) находим напряжения и перемещения.

Результаты

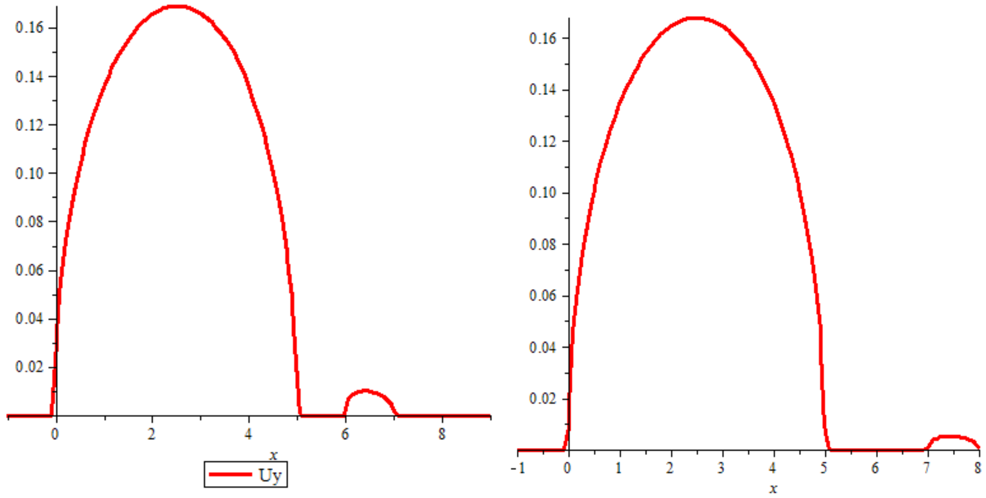
Перемещения на продолжении трещины



Напряжения на продолжении трещины



Геометрия верхнего берега трещины

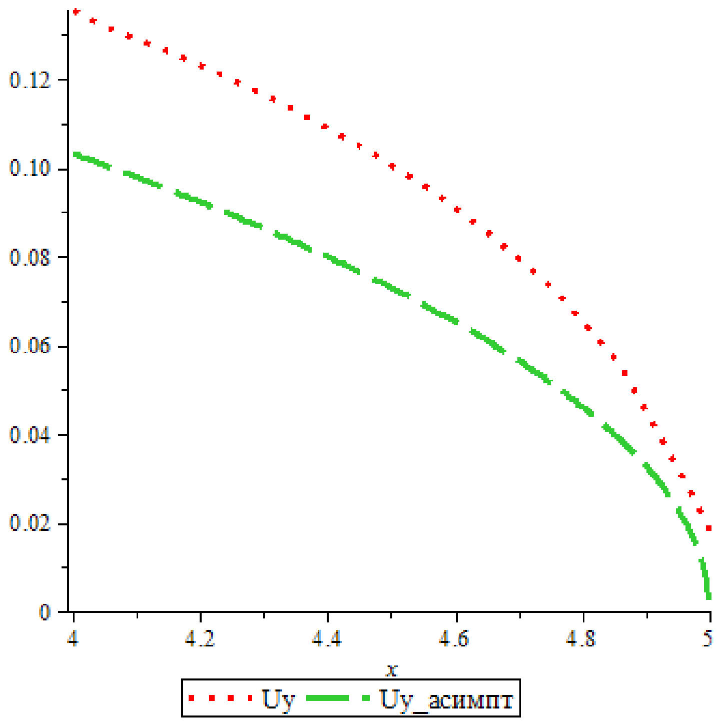


Из графиков видно, что чем дальше расположена маленькая трещина, тем меньше раскрытие ее берега.

Коэффициенты интенсивности напряжений посчитаны правом и левом краях трещины методом наименьших квадратов. Приведен график перемещений вблизи правого края трещины.

$K_1 = 0.193057498155531$ на правом крае,

$K_1 = 0.195362828653182$ на левом крае.



Итоги работы

В данной работе, используя основные уравнения теории упругости, с помощью численных методов (метода граничных элементов) была решена задача о двух трещинах, расположенных на одной прямой. Были посчитаны напряжения на продолжении трещины, перемещения, раскрытие верхнего берега трещины. Результаты наглядно проиллюстрированы на графиках. Был вычислен коэффициент интенсивности напряжений на правом и левом краях трещины. Из чего можно сделать вывод, наличие маленькой трещины делают большую менее устойчивой.

Список литературы

1. Крауч С., Старфилд А. Методы граничных элементов в механике твердого тела.: Пер. с англ. – М.:Мир, 1987
2. Бенерджи П., Баттерфилд Р. Метод граничных элементов в прикладных науках.: Пер. с англ. – М.: Мир, 1984

Приложения

Программа, используемая для расчетов в данной задаче.

Программа расчета напряженно-деформированного состояния прямоугольного тела с большой системой случайных равномерно распределенных трещин
Глобальные переменные программы:
k1-количество граничных элементов для первой трещины;
k2-количество граничных элементов для второй трещины;
ds1-длина граничного элемента для первой трещины;
ds2-длина граничного элемента для второй трещины;
alf1-угол наклона первой трещины;
alf2-угол наклона второй трещины;
X01,Y01-координаты начала первой трещины;
X02,Y02-координаты начала второй трещины;
N-общее количество вводимых граничных элементов (N=4n+k1+k2);
X[i],Y[i] - векторы координат середин каждого граничного элемента (N);
alfa[i] - угол наклона соответствующего граничного элемента к оси 0X глобальной системы координат;
A[i,j] - матрица коэффициентов влияния (2N*2N);
z[i] - неизвестный вектор коэффициентов, который определяется граничными условиями (2N);
F,Fx,Fy,Fxy,Fyy,Fxyy,Fyyy - производящая функция и её частные производные;

> $restart;$

> $PPi := convert(\mathrm{Pi}, float, 12);$

> $dl := 0.3;$

$$k1 := 300;\ k2 := 60;\ ds1 := \frac{1.}{k2};\ ds2 := ds1;\ alf1 := 0;\ alf2 := 0.;$$
$$X01 := 0.;\ Y01 := 0;\ X02 := 5.5;\ Y02 := 0.;$$

$$N := k1 + k2;$$

> $\nu := 0.33;$

> $$F := \mathbf{proc}(x, y, L) :: float; \mathbf{global}\ \nu, PPi; \mathbf{return}\left(\frac{1}{4 \cdot PPi \cdot (1-\nu)}\left((x - L)\cdot\log\left(\sqrt{(x-L)^2+y^2}\right) - (x+L)\cdot\log\left(\sqrt{(x+L)^2+y^2}\right) - y\cdot(\arctan(y, x-L) - \arctan(y, x+L))\right)\right); \mathbf{end};$$

$$Fx := \mathbf{proc}(x, y, L) :: float;\ \mathbf{global}\ \nu, PPi; \mathbf{return}\left(\frac{1.}{4 \cdot PPi \cdot (1-\nu)} \cdot \left(\log\left(\sqrt{\frac{(x-L)^2+y^2}{(x+L)^2+y^2}}\right)\right)\right); \mathbf{end};$$

$$Fy := \mathbf{proc}(x, y, L) :: \mathrm{float};\ \mathbf{global}\ \nu, \mathrm{PPi}; \mathbf{return}\left(-\frac{1.}{4 \cdot PPi \cdot (1-\nu)} \cdot (\arctan(y, x-L) - \arctan(y, x+L))\right); \mathbf{end};$$

$$Fyy := \mathbf{proc}(x, y, L) :: float;\ \mathbf{global}\ \nu, PPi; \mathbf{return} \left(-\frac{1.}{4 \cdot PPi \cdot (1-\nu)} \cdot \left(\frac{x-L}{(x-L)^2+y^2} - \frac{\mathrm{x}+\mathrm{L}}{(\mathrm{x}+\mathrm{L})^2+y^2} \right) \right);\ \mathbf{end};$$

$$Fxy := \mathbf{proc}(x, y, L) :: float; \mathbf{global}\ \nu, PPi; \mathbf{return} \left(\frac{1.}{4 \cdot PPi \cdot (1-\nu)} \cdot \left(\frac{y}{(x-L)^2+y^2} - \frac{\mathrm{y}}{(\mathrm{x}+\mathrm{L})^2+y^2} \right) \right); \mathbf{end};$$

$$Fyyy := \mathbf{proc}(x, y, L) :: float; \mathbf{global}\ \nu, PPi; \mathbf{return} \left(\frac{2 \cdot y}{4 \cdot PPi \cdot (1-\nu)} \cdot \left(\frac{x-L}{\left((x-L)^2+y^2\right)^2} - \frac{\mathrm{x}+\mathrm{L}}{\left((\mathrm{x}+\mathrm{L})^2+y^2\right)^2} \right) \right); \mathbf{end};$$

$$Fxyy := \mathbf{proc}(x, y, L) :: float; \mathbf{global}\ \nu, PPi;\ \mathbf{return} \left(\frac{1}{4 \cdot PPi \cdot (1-\nu)} \cdot \left(\frac{(x-L)^2-y^2}{\left((x-L)^2+y^2\right)^2} - \frac{(\mathrm{x}+\mathrm{L})^2-y^2}{\left((\mathrm{x}+\mathrm{L})^2+y^2\right)^2} \right) \right); \mathbf{end};$$

$$snxx := \mathbf{proc}(x, y, L) :: float; \mathbf{return}(2 \cdot (Fyy(x, y, L) + y \cdot Fyyy(x, y, L))); \mathbf{end};$$

$$snyy := \mathbf{proc}(x, y, L) :: float; \mathbf{return}(2 \cdot (Fyy(x, y, L) - \mathrm{y} \cdot Fyyy(x, y, L)))\ \mathbf{end};$$

$$snxy := \mathbf{proc}(x, y, L) :: float; \mathbf{return}(-2 \cdot y \cdot Fxyy(x, y, L)); \mathbf{end};$$

$$stxx := \mathbf{proc}(x, y, L) :: float; \mathbf{return}(2 \cdot (2 \cdot Fxy(x, y, L) + y \cdot Fxyy(x, y, L))); \mathbf{end};$$

$$styy := \mathbf{proc}(x, y, L) :: float; \mathbf{return}(-2 \cdot y \cdot Fxyy(x, y, L)); \mathbf{end};$$

$$stxy := \mathbf{proc}(x, y, L) :: float; \mathbf{return}(2 \cdot (Fyy(x, y, L) + y \cdot Fyyy(x, y, L))); \mathbf{end};$$

$$fnxx := \mathbf{proc}(x, y, L) :: float; \mathbf{global}\ \nu; \mathbf{return}\left(2 \cdot \nu \cdot Fy(x, y, L) + y \cdot Fyy(x, y, L)\right); \mathbf{end};$$

$$fnyy := \mathbf{proc}(x, y, L) :: float; \mathbf{global}\ \nu; \mathbf{return}\left(2 \cdot (1-\nu) \cdot Fy(x, y, L) - \mathrm{y} \cdot Fyy(x, y, L)\right); \mathbf{end};$$

$$fnxy := \mathbf{proc}(x, y, L) :: float; \mathbf{global}\ \nu; \mathbf{return}\left((1-2 \cdot \nu) \cdot Fx(x, y, L) - y \cdot Fxy(x, y, L)\right); \mathbf{end};$$

$$ftxx := \mathbf{proc}(x, y, L) :: float; \mathbf{global}\ \nu; \mathbf{return}\left((3-2 \cdot \nu) \cdot Fx(x, y, L) + \mathrm{y} \cdot Fxy(x, y, L)\right); \mathbf{end};$$

$$ftyy := \mathbf{proc}(x, y, L) :: float; \mathbf{global}\ \nu; \mathbf{return}\left(-(1-2 \cdot \nu) \cdot Fx(x, y, L) - y \cdot Fxy(x, y, L)\right); \mathbf{end};$$

$$ftxy := \mathbf{proc}(x, y, L) :: float; \mathbf{global}\ \nu; \mathbf{return}\left(2 \cdot (1-\nu) \cdot Fy(x, y, L) + y \cdot Fyy(x, y, L)\right); \mathbf{end};$$

$$X0 := \mathbf{proc}(x0, y0, x1, y1, alfa1) :: float; \mathbf{return}\ ((x0 - x1) \cdot \cos(alfa1) + (y0 - y1) \cdot \sin(alfa1));\ \mathbf{end};$$

$$Y0 := \mathbf{proc}(x0, y0, x1, y1, alfa1) :: float; \mathbf{return}\ (-(x0 - x1) \cdot \sin(alfa1) + (y0 - y1) \cdot \cos(alfa1)); \mathbf{end};$$

$$Unnx := \mathbf{proc}(x, y, L) :: float;\ \mathbf{global}\ \nu; \mathbf{return}\left(-(1-2 \cdot \nu) \cdot Fx(x, y, L) - y \cdot Fxy(x, y, L)\right); \mathbf{end};$$

```
Unny := proc(x, y, L) :: float; global ν;return(2·(1 − ν)·Fy(x, y, L)
    − y·Fyy(x, y, L));end;

UnTx := proc(x, y, L) :: float; global ν;return(2·(1 − ν)·Fy(x, y, L)
    + y·Fyy(x, y, L));end;

UnTy := proc(x, y, L) :: float; global ν;return((1 − 2·ν)·Fx(x, y, L)
    − y·Fxy(x, y, L));end;

Ufnx := proc(x, y, L) :: float; global ν;return(−y·Fx(x, y, L));end;
Ufny := proc(x, y, L) :: float; global ν;return((3 − 4·ν)·F(x, y, L)
    − y·Fy(x, y, L));end;

UfTx := proc(x, y, L) :: float; global ν;return((3 − 4·ν)·F(x, y, L)
    + y·Fy(x, y, L));end;

> UfTy := proc(x, y, L) :: float; global ν;return(−y·Fx(x, y, L));end;
> X := Vector(N, datatype = float);
> Y := Vector(N, datatype = float);
> alfa := Vector(N, datatype = float);
> h := Vector(N, datatype = float);
>
  Geometry := proc( ) local i, j, s, nn; global X, Y, alfa, h, n, N, PPi; for i
      from 1 by 1 to k1 do h[i] := ds1/2.; Y[i] := 0.; X[i] := X01 + 2
      ·h[i]·(i − 0.5); alfa[i] := alf1;end do;for i from k1 + 1 by 1 to N
      do h[i] := ds2/2; Y[i] := 0.; X[i] := X02 + h[i]·2·(i − k1
      − 0.5); alfa[i] := alf2;end do;return(1);end proc;

> Geometry( );


> plot(X, Y, style = point, symbol = asterisk, color = blue, labels = ["X",
    "Y"]);

> with(LinearAlgebra);
> A := Matrix(2·N, 2·N, datatype = float);
> b := Vector(2·N, datatype = float);
```

```
> 
FormmatA := proc( ) local i, j, xx0, yy0, Gam, SSxx, SSyy, SSxy;
    global A, X, Y, alfa, h, N; for i from 1 by 1 to N do for j from 1 by 1
    to N do xx0 := X0(X[i], Y[i], X[j], Y[j], alfa[j]); yy0
     := Y0(X[i], Y[i], X[j], Y[j], alfa[j]); Gam := alfa[j]; SSxx
     := snxx(xx0, yy0, h[j]); SSyy := snyy(xx0, yy0, h[j]); SSxy
     := snxy(xx0, yy0, h[j]); A[i, j] := SSxx·(1. − cos(2·Gam))/2
     + SSyy·(1. + cos(2·Gam))/2 − SSxy·sin(2·Gam); xx0
     := X0(X[i], Y[i], X[j], Y[j], alfa[j]); yy0 := Y0(X[i], Y[i],
    X[j], Y[j], alfa[j]); Gam := alfa[j] − alfa[i]; SSxx := stxx(xx0,
    yy0, h[j]); SSyy := styy(xx0, yy0, h[j]); SSxy := stxy(xx0, yy0,
    h[j]); A[i, N + j] := SSxx·(1. − cos(2·Gam))/2
     + SSyy·(1. + cos(2·Gam))/2 − SSxy·sin(2·Gam); end do; end do;
    for i from 1 by 1 to N do for j from 1 by 1 to N do xx0 := X0(X[i],
    Y[i], X[j], Y[j], alfa[j]); yy0 := Y0(X[i], Y[i], X[j], Y[j],
    alfa[j]); Gam := alfa[j]; SSxx := snxx(xx0, yy0, h[j]); SSyy
     := snyy(xx0, yy0, h[j]); SSxy := snxy(xx0, yy0, h[j]); A[N + i, j]
     := (SSyy − SSxx)/2·sin(2·Gam) + SSxy·cos(2·Gam); xx0
     := X0(X[i], Y[i], X[j], Y[j], alfa[j]); yy0 := Y0(X[i], Y[i],
    X[j], Y[j], alfa[j]); Gam := alfa[j] − alfa[i]; SSxx := stxx(xx0,
    yy0, h[j]); SSyy := styy(xx0, yy0, h[j]); SSxy := stxy(xx0, yy0,
    h[j]); A[N + i, N + j] := (SSyy − SSxx)/2·sin(2·Gam) + SSxy
    ·cos(2·Gam); end do; end do; return(1); end proc;

> FormmatA( );
> Formvectb := proc( ) local i; global b, N; for i from 1 by 1 to N do b[i]
     := -0.1; end; for i from N + 1 by 1 to 2·N do b[i] := 0.; end; for i
    from k1 + 1 by 1 to N do b[i] := 0.; end; return(1); end proc;
> Formvectb( ); for i from 1 by 1 to 2·N do eval(b[i]); end do;
> for i from 1 by 1 to 2·N do eval(A[i, i]); end do;
> z := LinearSolve(A, b);
> for i from 1 by 1 to 2·N do eval(z[i]); end;
> 
SIGMAxx := proc(xx, yy) :: float; local i, j, xx0, yy0, Gam, SSxx, SSyy,
    SSxy, σxx, SSSxx; global N, h, X, Y, alfa, z; SSSxx := Vector(2
    ·N, datatype = float); for j from 1 by 1 to N do xx0 := X0(xx, yy,
    X[j], Y[j], alfa[j]); yy0 := Y0(xx, yy, X[j], Y[j], alfa[j]); Gam
     := alfa[j]; SSxx := snxx(xx0, yy0, h[j]); SSyy := snyy(xx0, yy0,
    h[j]); SSxy := snxy(xx0, yy0, h[j]); SSSxx[j]
     := SSxx·(1. + cos(2·Gam))/2 + SSyy·(1.-cos(2·Gam))/2
     − SSxy·sin(2·Gam); end; for j from 1 by 1 to N do xx0 := X0(xx,
    yy, X[j], Y[j], alfa[j]); yy0 := Y0(xx, yy, X[j], Y[j], alfa[j]);
    Gam := alfa[j]; SSxx := stxx(xx0, yy0, h[j]); SSyy := styy(xx0,
    yy0, h[j]); SSxy := stxy(xx0, yy0, h[j]); SSSxx[N + j]
     := SSxx·(1. + cos(2·Gam))/2 + SSyy·(1.-cos(2·Gam))/2
     − SSxy·sin(2·Gam); end; σxx := 0.; for i from 1 by 1 to 2·N
    do σxx := σxx + SSSxx[i]·z[i]; end; return(σxx); end;
```

```
> SIGMAyy := proc(xx, yy) :: float; local i, j, xx0, yy0, Gam, SSxx, SSyy,
     SSxy, σyy, SSSyy; global N, h, X, Y, alfa, z; SSSyy := Vector(2
     ·N, datatype = float); for j from 1 by 1 to N do xx0 := X0(xx, yy,
     X[j], Y[j], alfa[j]); yy0 := Y0(xx, yy, X[j], Y[j], alfa[j]); Gam
     := alfa[j]; SSxx := snxx(xx0, yy0, h[j]); SSyy := snyy(xx0, yy0,
     h[j]); SSxy := snxy(xx0, yy0, h[j]); SSSyy[j]
     := SSxx·(1.-cos(2·Gam))/2 + SSyy·(1. + cos(2·Gam))/2
     + SSxy·sin(2·Gam); end; for j from 1 by 1 to N do xx0 := X0(xx,
     yy, X[j], Y[j], alfa[j]); yy0 := Y0(xx, yy, X[j], Y[j], alfa[j]);
     Gam := alfa[j]; SSxx := stxx(xx0, yy0, h[j]); SSyy := styy(xx0,
     yy0, h[j]); SSxy := stxy(xx0, yy0, h[j]); SSSyy[N + j]
     := SSxx·(1.-cos(2·Gam))/2 + SSyy·(1. + cos(2·Gam))/2
     + SSxy·sin(2·Gam); end; σyy := 0.; for i from 1 by 1 to 2·N
     do σyy := σyy + SSSyy[i]·z[i]; end; return(σyy); end;

> SIGMAxx(0.5, 0.501); SIGMAyy(0.5, 0.501);
> SIGMAxy := proc(xx, yy) :: float; local i, j, xx0, yy0, Gam, SSxx, SSyy,
     SSxy, σxy, SSSxy; global N, h, X, Y, alfa, z; SSSxy := Vector(2
     ·N, datatype = float); for j from 1 by 1 to N do xx0 := X0(xx, yy,
     X[j], Y[j], alfa[j]); yy0 := Y0(xx, yy, X[j], Y[j], alfa[j]); Gam
     := alfa[j]; SSxx := snxx(xx0, yy0, h[j]); SSyy := snyy(xx0, yy0,
     h[j]); SSxy := snxy(xx0, yy0, h[j]); SSSxy[j] := (SSyy - SSxx)/2
     ·sin(2·Gam) + SSxy·cos(2·Gam); end; for j from 1 by 1 to N do xx(
     := X0(xx, yy, X[j], Y[j], alfa[j]); yy0 := Y0(xx, yy, X[j], Y[j],
     alfa[j]); Gam := alfa[j]; SSxx := stxx(xx0, yy0, h[j]); SSyy
     := styy(xx0, yy0, h[j]); SSxy := stxy(xx0, yy0, h[j]); SSSxy[N
     + j] := (SSyy - SSxx)/2 ·sin(2·Gam) + SSxy·cos(2·Gam); end;
     σxy := 0.; for i from 1 by 1 to 2·N do σxy := σxy + SSSxy[i]·z[i];
     end; return(σxy); end;

> SIGMAxy(0.5, 0.51);
> DisplisamentUx := proc(xx, yy) :: float; local i, j, xx0, yy0, ux0, Ux,
     Uxx, Uyy, Gam; global N, h, X, Y, alfa, z; Ux := Vector(2·N,
     datatype = float); for j from 1 by 1 to N do xx0 := X0(xx, yy, X[j],
     Y[j], alfa[j]); yy0 := Y0(xx, yy, X[j], Y[j], alfa[j]); Gam
     := alfa[j]; Uxx := Unnx(xx0, yy0, h[j]); Uyy := Unny(xx0, yy0,
     h[j]); Ux[j] := Uxx·cos(Gam) - Uyy·sin(Gam); end; for j
     from 1 by 1 to N do xx0 := X0(xx, yy, X[j], Y[j], alfa[j]); yy0
     := Y0(xx, yy, X[j], Y[j], alfa[j]); Gam := alfa[j]; Uxx
     := UnTx(xx0, yy0, h[j]); Uyy := UnTy(xx0, yy0, h[j]); Ux[N
     + j] := Uxx·cos(Gam) - Uyy·sin(Gam); end; ux0 := 0.; for i
     from 1 by 1 to 2·N do ux0 := ux0 + Ux[i]·z[i]; end; return(ux0);
     end;

> DisplisamentUx(0.9, 0.5);
```

> $DisplisamentUy := \mathbf{proc}(xx, yy) :: \text{float}; \mathbf{local}\ \text{i, j, xx0, yy0, uy0, Uy, Uxx, Uyy, Gam}; \mathbf{global}\ \text{N, h, X, Y, alfa, z}; \text{Uy} := \text{Vector}(2 \cdot N, datatype = float); \mathbf{for}\ j\ \mathbf{from}\ 1\ \mathbf{by}\ 1\ \mathbf{to}\ N\ \mathbf{do}\ xx0 := X0(xx, yy, X[j], Y[j], alfa[j]); yy0 := Y0(xx, yy, X[j], Y[j], alfa[j]); Gam := alfa[j]; Uxx := Unnx(xx0, yy0, h[j]); Uyy := Unny(xx0, yy0, h[j]); Uy[j] := Uxx \cdot \sin(Gam) + \text{Uyy} \cdot \cos(Gam); \mathbf{end}; \mathbf{for}\ j\ \mathbf{from}\ 1\ \mathbf{by}\ 1\ \mathbf{to}\ N\ \mathbf{do}\ xx0 := X0(xx, yy, X[j], Y[j], alfa[j]); yy0 := Y0(xx, yy, X[j], Y[j], alfa[j]); Gam := alfa[j]; Uxx := UnTx(xx0, yy0, h[j]); Uyy := UnTy(xx0, yy0, h[j]); Uy[N + j] := Uxx \cdot \sin(Gam) + Uyy \cdot \cos(Gam); \mathbf{end}; uy0 := 0.; \mathbf{for}\ i\ \mathbf{from}\ 1\ \mathbf{by}\ 1\ \mathbf{to}\ 2 \cdot N\ \mathbf{do}\ uy0 := uy0 + Uy[i] \cdot z[i]; \mathbf{end}; \mathbf{return}(uy0); \mathbf{end};$

> $Nprob1 := 100; Xprob1 := vector[row](Nprob1); Yprob1 := vector[row](Nprob1);$

> $dss := \dfrac{10}{Nprob1 - 1}; Xprob1[1] := -1.; Yprob1[1] := 0.0000001;$

> $\mathbf{for}\ i\ \mathbf{from}\ 2\ \mathbf{by}\ 1\ \mathbf{to}\ Nprob1\ \mathbf{do}\ Xprob1[i] := Xprob1[i-1] + dss; Yprob1[i] := Yprob1[1]; \mathbf{end\ do};$

> $Uyprob1 := vector[row](Nprob1);$

> $\mathbf{for}\ i\ \mathbf{from}\ 1\ \mathbf{by}\ 1\ \mathbf{to}\ Nprob1\ \mathbf{do}\ Uyprob1[i] := DisplisamentUy(Xprob1[i], Yprob1[i]); \mathbf{end\ do};$

> $with(CurveFitting);$

> $Uyyyy(qq) := spline(Xprob1, Uyprob1, qq);$

> $Uyyyy(2.5);$

> $plot(Uyyyy(x), x = -1 ..9);$

> $Nprob2 := 10; p := vector[row](Nprob2); step := \dfrac{1.}{Nprob2}; p[1] := 4.1; c1 := 0; c2 := 0;$

>

> $\mathbf{for}\ i\ \mathbf{from}\ 2\ \mathbf{by}\ 1\ \mathbf{to}\ Nprob2\ \mathbf{do}\ p[i] := p[1] + step \cdot (i-1); c2 := c2 + Uyyyy(p[i-1]) \cdot \text{sqrt}(5 - p[i-1]); c1 := c1 + \text{sqrt}(5 - p[i-1]); \mathbf{end\ do}: K_1 := \dfrac{c2}{c1 \cdot (1. - 0.33)} \cdot \text{sqrt}\left(\dfrac{PPi}{2}\right);$

> $p[1] := 0.1; \mathbf{for}\ i\ \mathbf{from}\ 2\ \mathbf{by}\ 1\ \mathbf{to}\ Nprob2\ \mathbf{do}\ p[i] := p[1] + step \cdot (i-1); c2 := c2 + Uyyyy(p[i-1]) \cdot \text{sqrt}(p[i-1]); c1 := c1 + \text{sqrt}(p[i-1]); \mathbf{end\ do}: K_1 := \dfrac{c2}{c1 \cdot (1. - 0.33)} \cdot \text{sqrt}\left(\dfrac{PPi}{2}\right);$

> $Nprob := 40; Xprob0 := 5.01; Yprob0 := 0.;$

> "Nprob-количество пробных точек сечения; Xprob0, Yprob0-начальные координаты пробной линии сечения"

> $DisplisamentUy(0.9, 0.5);$

> $xprob := Vector(\text{Nprob}, datatype = float);$
"массив координат x сечения "

> $yprob := Vector(\text{Nprob}, datatype = float);$
"массив координат y сечения "

> $ds := \dfrac{2.}{\text{Nprob}}$; "шаг вдоль линии сечения"

> $SIxx := Vector(Nprob, datatype = float);$

> $SIyy := Vector(Nprob, datatype = float);$

> $SIxy := Vector(\text{Nprob}, datatype = float);$

> $ux := Vector(\mathrm{Nprob}, datatype = float);$

> $uy := Vector(\mathrm{Nprob}, datatype = float);$

> **for** i **from** 1 **by** 1 **to** $Nprob$ **do** $yprob[i] := 0.; xprob[i] := Xprob0 + ds \cdot (i - 0.5); ux[i] := DisplisamentUx(xprob[i], yprob[i]); uy[i] := DisplisamentUy(xprob[i], yprob[i]); SIxx[i] := SIGMAxx(xprob[i], yprob[i]); SIyy[i] := SIGMAyy(xprob[i], yprob[i]); SIxy[i] := SIGMAxy(xprob[i], yprob[i]);$**end**;

> $with(CurveFitting);$

> $with(plots);$

> $fux(s) := spline(\mathrm{xprob, ux, s});$

> $fuy(s) := spline(\mathrm{xprob, uy, s});$

> $fsxx(s) := spline(\mathrm{xprob, SIxx, s});$

> $fsyy(s) := spline(\mathrm{xprob, SIyy, s});$

> $fsxy(s) := spline(\mathrm{xprob, SIxy, s});$

> $s1 := \dfrac{(xprob[1] + xprob[2])}{2.};$

> $s2 := \dfrac{(xprob[Nprob] + xprob[Nprob - 1])}{2.};$

> $plot([fux(s), \mathrm{fuy(s)}], \mathrm{s = s1 .. s2}, labels = [\text{"S", "U"}], title$;
= "Перемещения на продолжении трещины", $legend$ = ["Ux", "Uy"])
;

> $plot([fsxx(s), fsyy(s) + 0.1, fsxy(s)], s = s1 .. s2, labels = [\text{"s", "σ"}],$
$title$ = "Напряжения на продолжении трещины", $legend$ = ["Sxx", "Syy", "Sxy"]);

> $N1 := 20;$
"Количество точек для вычисления коэффициента интенсивности"

$$N1 := 20$$

"Количество точек для вычисления коэффициента интенсивности"

> "Вычисление КК для Асимптотики Syy=KK/scrt(x_i-1.) методом наименьших квадратов"

"Вычисление КК для Асимптотики Syy=KK/scrt(x_i-1.) методом наименьших квадратов"

> $SIGMAyy := 0; XXX := 0;$ **for** i **from** 1 **by** 1 **to** $N1$ **do** $SIGMAyy := SIGMAyy + \dfrac{SIyy[i]}{\sqrt{xprob[i] - 1.}}; XXX := XXX + \dfrac{1.}{\sqrt{xprob[i] - 1.}};$ **end do**;

> $KK := \dfrac{SIGMAyy}{XXX};$

> "Сравнение полученной асимптотики с численными значениями"

"Сравнение полученной асимптотики с численными значениями"

> $plot\left(\left[fsyy(s) + 0.1, \dfrac{KK}{\sqrt{s - 5.}}\right], s = s1 .. s2, labels = [\text{"s", "σ"}], title\right.$
= "Напряжения на продолжении трещины", $legend$ = ["Syy", "Syy_асимпт"]);

> "Определение перемещения берегов трещины"

"Определение перемещения берегов трещины"

> **for** i **from** 1 **by** 1 **to** $Nprob$ **do** $yprob[i] := 0.005;\ xprob[i] := 0.001 + \frac{1.}{Nprob} \cdot (i - 0.5);\ ux[i] := DisplisamentUx(xprob[i], yprob[i]);\ uy[i] := DisplisamentUy(xprob[i], yprob[i]);\ SIxx[i] := SIGMAxx(xprob[i], yprob[i]);\ SIyy[i] := SIGMAyy(xprob[i], yprob[i]);\ SIxy[i] := SIGMAxy(xprob[i], yprob[i]);$ **end**;

> $ffux(ss) := spline(\mathrm{xprob}, \mathrm{ux}, \mathrm{ss});$

> $ffux(0.78);$

> $ffuy(ss) := spline(\mathrm{xprob}, \mathrm{uy}, \mathrm{ss});$

> $ffsxx(ss) := spline(\mathrm{xprob}, \mathrm{SIxx}, \mathrm{ss});$

> $ffsyy(ss) := spline(\mathrm{xprob}, \mathrm{SIyy}, \mathrm{ss});$

> $ffsxy(ss) := spline(\mathrm{xprob}, \mathrm{SIxy}, \mathrm{ss});$

> $s1 := \frac{(xprob[1] + xprob[2])}{2.};$

> $s2 := \frac{(xprob[N1 - 1] + xprob[N1 - 2])}{2.};$

> **for** i **from** 1 **by** 1 **to** $Nprob$ **do** $xprob[i] := xprob[i] + ux[i];\ yprob[i] := yprob[i] + uy[i];$ **end do**;

> $plot$(xprob, yprob, style = point, symbol = asterisk, color = blue, $labels$ = ["X", "Y"], $title$ = "Геометрия верхнего берега трещины", $legend$ = ["Система трещин"]);

> $plot$([$ffsxx(ss)$, $ffsyy(ss) + 0.1$, $ffsxy(ss)$], $ss = s1..s2$, $labels$ = ["s", "σ"], $title$ = "Напряжения на верхнем берегу трещины", $legend$ = ["Sxx", "Syy", "Sxy"]);