УДК 677-486.2:539.11

РАСЧЕТ ДВУМЕРНОЙ ДЕФОРМАЦИИ ТРИКОТАЖА

CALCULATION OF TWO-DIMENSIONAL DEFORMATION OF JERSEY

Л.А. КУДРЯВИН, О.Ф. БЕЛЯЕВ, В.А. ЗАВАРУЕВ, О.С. КОТОВИЧ
L.A.KUDRJAVIN, O.F.BELJAEV, V.A.ZAVARUEV, O.S.KOTOVIC

(Московский государственный текстильный университет им. А.Н. Косыгина)
(Moscow State Textile University "A.N. Kosygin")
E-mail: office@msta.ac.ru

Приведен с комментариями текст программы, основанной на математическом пакете MATLAB, позволяющей определить форму повторяющегося элемента после деформации и рассчитывать относительную деформацию такого элемента при деформации образца, а следовательно, и относительную деформацию самого образца.

The text of the program based on MATLAB mathematical suite is resulted with comments, allowing to define the form of a repeating element after deformation and to calculate a strain of such element by the sample deformation, and hence, a strain of the sample itself.

Ключевые слова: текст программы, комментарии, математический пакет MATLAB, трикотаж, двумерная деформация, форма элемента.

Keywords: the program text, comments, MATLAB mathematical suite, jersey, two-dimensional deformation, an element form.

Приведен несколько укороченный текст программы, позволяющей рассчитать относительную деформацию повторяющегося элемента, описание которой дано ранее в [1]. Форма этого элемента до деформации образца представлена на рис.1.

Текст программы приведен с комментариями (они в статье даны в скобках.). Комментарии поясняют команды и объясняют, как нужно изменить программу для того или иного случая. При расчете по программе принято R_{10}= 0.05, R_{20} = - 0.1, H=0.05, F_1=F_2=F=20; F_3=0 (все в выбранной системе единиц, например, в СИ), φ_1 = = 90°, φ_2 = 90°, $\delta_1 = \delta_2 = \delta$ = 90°.

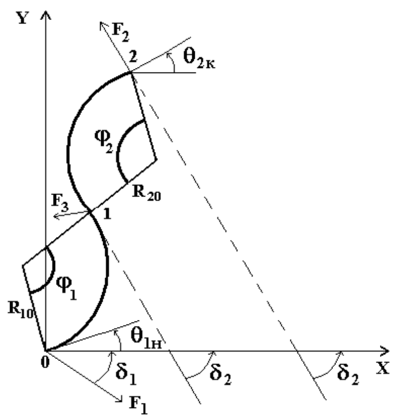


Рис. 1

Текст программы:

function rastdvuhokr (название программы);

global A1 A2 L1 L2 R10 R20 delta1 delta2 (объявление глобальных переменных, то есть переменных, значения которых одинаковы в основной программе и в подпрограммах);

F=10; H = 0.05; R10=0.05; R20=-0.1; (задание силы, действующей на нить; жесткости нити при изгибе и радиусов окружностей);

fi1g=90; fi2g=90; deltag=90 (задание углов φ_1, φ_2, δ в градусах);

fi1=fi1g*pi/180; fi2=fi2g*pi/180; delta=deltag*pi/180 (перевод углов в радианы);

L1=fi1*abs(R10); L2=fi2*abs(R20); L=L1+L2 (расчет длин окружностей и общей длины повторяющегося элемента);

A1=F1*L1^2/H; A2=F2*L2^2/H; (коэффициенты, используемые далее программой);

N=101 (N-1 – предварительное число участков, на которые разбивается каждая окружность, чем больше N, тем точнее результат. Как правило, вполне достаточно брать N=101. При этом каждая окружность разбивается на сто одинаковых участков, длина каждого участка $\Delta\tau$=0.01);

meshinit = linspace(0,1,N) (формирование сетки и начального приближения. Здесь первая и вторая цифры – значения τ в начале и конце каждого участка, τ – безразмерная длина участка, меняется от 0 до 1);

yinit=[1 L1/R10 1 L2/R20] (приводятся предположительные значения углов θ и производных $d\theta/d\tau$ для первой и второй окружностей. Обычно результат слабо зависит от этих значений);

initsol=bvpinit(meshinit, yinit) (эта функция задает сетку и начальные приближения).

Далее производится вызов солвера (решателя уравнения) bvp4c:

sol = bvp4c(@rside, @bound, initsol); (в sol.x содержатся значения τ, в sol.y – матрица);

tau=sol.x (таблица полученных программой значений τ);

n=length(tau); (число элементов в этой таблице. Обычно оно совпадает с N); dtau=1/(n-1) (n-1 – число участков, на которое разбита каждая окружность, dtau – длина одного участка). При этом sol.y(1,:), sol.y(2,:), sol.y(3,:), sol.y(4,:) – таблицы, определяющие значения θ_1, $d\theta_1/d\tau$, θ_2, $d\theta_2/d\tau$ при соответствующем значении τ. В программе θ_1, θ_2 обозначаются как teta1, teta2. Обозначим номер элемента в этих таблицах через j. Он меняется от 1 до n через 1. Подфункции rside и bound, определяющие матрицу sol, рассчитываются по функциям f1 и f2, приведенным в конце программы;

X1(1)=0; Y1(1)=0;(начальные координаты первого участка первой окружности);

dX1=dtau*L1*cos(teta1); dY1=dtau*L1*sin(teta1); (проекция длины первого участка первой окружности на оси X и Y).

Далее до end идет цикл, рассчитывающий координаты начала j-го участка первой окружности:

for j=2:n; X1(j)=X1(j-1)+dX1(j-1); Y1(j)=Y1(j-1)+dY1(j-1); end;

X1k=X1(n); Y1k=Y1(n); X2(1)=X1k; Y2(1)=Y1k; (это координаты конца последнего участка первой окружности и в то же время – координаты начала первого участка второй окружности);

dX2=dtau*L2*cos(teta2); dY2=dtau*L2*sin(teta2); (проекция длины первого участка второй окружности на оси X и Y).

Далее опять до end цикл, рассчитывающий координаты начала j-го участка второй окружности:

for j=2:n; X2(j)=X2(j-1)+dX2(j-1); Y2(j)=Y2(j-1)+dY2(j-1); end;

X2k=X2(n); Y2k=Y2(n) (координаты конца последнего участка второй окружности и в то же время координаты Xк, Yк конца всего повторяющегося элемента в деформированном состоянии);

plot(X1,Y1,'-b',X2,Y2,'-b') (построение графических зависимостей Y1(X1) и Y2(X2), то есть построение формы повторяющегося элемента в нагруженном образце);

xlabel('X'); ylabel('Y') (нанесение обозначений на осях X,Y);

grid on (нанесение сетки на график);

Функции f1 (подфункция rside) и f2 (подфункция bound):

```
function f1 = rside(x, y); global A1 A2 delta1 delta2;
f1 = [y(2); -A1*sin(y(1)+delta1);y(4); -A2*sin(y(3)+delta2)];
function f2 = bound(ya, yb); global L1 L2 R10 R20;
f2 = [ya(1); yb(1)-ya(3); yb(2)/L1-1/R10-ya(4)/L2+1/R20; yb(4)-L2/R20]
```

Функция f1 решает уравнение (2) из [1] для первой и второй окружностей, а в функции f2 задаются граничные условия для каждой окружности. Под x в функции f1 понимается величина τ, под y(1), y(2), y(3), y(4) – значения θ_1, $d\theta_1/d\tau$, θ_2, $d\theta_2/d\tau$.

В функции f2 ya(1), ya(2), yb(1), yb(2) – значения θ_1 и $d\theta_1/d\tau$ в начальной и конечной точках первой окружности; ya(3), ya(4), yb(3), yb(4) – то же самое для второй окружности. Функция f2 играет очень важную роль в определении формы повторяющегося элемента. Программа подбирает такое решение, чтобы каждое выражение в функции f2 равнялось бы нулю. Так, например, запись ya(1) означает, что в начале первой окружности угол θ_1 должен равняться нулю. Если вместо него записать ya(1)-pi/3, это означало бы, что угол θ_1 в начале первой окружности должен быть равен $\pi/3$ или 60°. Вместо этого условия можно записать, например, ya(2)-L1/R10 или ya(2)+L1/R10. Первая часть этой записи означает, что в начале первой окружности $d\theta_1/d\tau = L_1/R_{10}$, то есть производная в начале первой окружности не зависит от растягивающей силы и если $R_{10}>0$, угол θ_1 растет с увеличением τ, при второй части записи – уменьшается. В то же время обе записи означают, что в начале первой окружности момент сил равен нулю, то есть имеет место шарнир. Аналогично для конца второй окружности можно, например, записать: yb(3) или yb(4)-L2/R20. Первая запись показывает, что касательная к концу второй окружности (к концу повторяющегося элемента) направлена вдоль оси X, а вторая – изгибающий момент в конце второй окружности равен нулю (шарнир).

Теперь рассмотрим вопрос стыковки двух окружностей. Условий стыковки два. Первое условие – одинаковый угол наклона θ касательной к оси X в конце первой и в начале второй окружности, то есть должно выполняться условие yb(1)-ya(3) = 0 (равенство нулю второго выражения в функции f2).

Перейдем ко второму условию стыковки.

Если упругая линия жесткостью на изгиб H в недеформированном состоянии представляет собой окружность радиуса R_0, то после действия на нее изгибающего момента M ее новый радиус R можно рассчитать по формуле [2]:

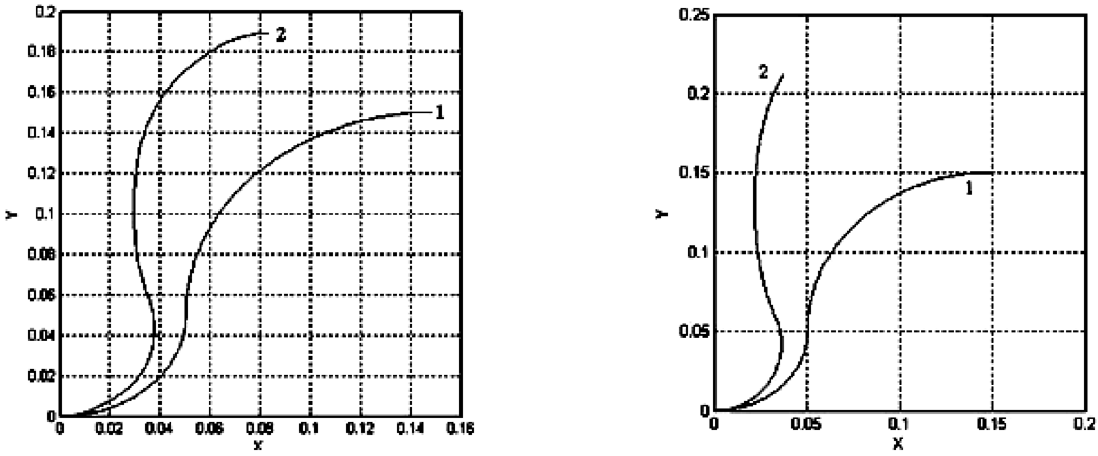
$1/R - 1/R_0 = M/H$ или $d\theta/ds-1/R_0=M/H$. Так как в точке стыковки двух окружностей нет сосредоточенного момента и жесткости обеих окружностей одинаковы, правые части этих формул в этой точке для обеих окружностей должны быть равны, поэтому $d\theta_1/ds_1-1/R_{10}= d\theta_2/ds_2-1/R_{20}$.

Поскольку на первом участке $s=\tau*L_1$, а на втором - $s=\tau*L_2+L_1$ (первое сообщение), то переходя от s к τ, вместо верхнего равенства получим $d\theta_1/(L_1*d\tau)-1/R_{10}= =d\theta_2/(L_2*d\tau)-1/R_{20}$ или

$(d\theta_1/d\tau)/L_1 -1/R_{10} – (d\theta_2/d\tau)/L_2 + 1/R_{20}=0$.

Учитывая все вышесказанное, третье выражение в функции f2 можно записать следующим образом: yb(2)/L1-1/R10-ya(4)/L2+1/R20.

Проводя далее расчеты при данном значении $F \neq 0$ и при F=0, определяем форму повторяющегося элемента в деформированном и недеформированном образце и координаты Xк, Yк и $Xк_0$, $Yк_0$ его конца. Отношения $\varepsilon_x=(Xк-Xк_0)/Xк_0$ и $\varepsilon_y=(Yк-Yк_0)/Yк_0$ дадут относительную деформацию повторяющегося элемента, а следовательно, и образца по осям X и Y. При этих расчетах полагаем, что начало повторяющегося элемента находится в начале координат. Формы повторяющегося элемента до деформации образца (1) и после деформации (2) показаны на рис.2-а ($\theta_{1н}=0°$, $\theta_{2к}=0°$, $\delta=90°$) и на рис. 2-б (($\theta_{1н}=0°$, $d\theta_{2к}/ds=1/R_{20}$, $\delta=-90°$). Для первого случая $\varepsilon_x=-44{,}3\%$, $\varepsilon_y=25{,}9\%$, для второго – $\varepsilon_x=-75\%$, $\varepsilon_y=41{,}1\%$.



а) б)

Рис. 2

Расчеты проводились с помощью математического пакета MATLAB 2007 (лицензия 360533).

По разработанной программе при небольших ее изменениях можно теоретически рассчитывать деформацию других видов трикотажа, изготовленных из других материалов, деформацию тканей, а также любых материалов, в которых можно выделить повторяющийся элемент. По этой программе можно решать и обратную задачу – зная деформацию повторяющегося элемента, определять силы, действующие на этот элемент. Кроме того, если из образца аккуратно выделить нить, из которой он изготовлен, и сравнить ее форму с ее формой в образце, то можно определить усилия между элементами структуры в данном образце.

Эту программу можно использовать также для расчета деформации модельных структур.

ВЫВОДЫ

Приведен с комментариями текст программы, основанной на математическом пакете MATLAB, позволяющей определить форму повторяющегося элемента после деформации и рассчитывать относительную деформацию такого элемента при деформации образца, а следовательно, и относительную деформацию самого образца. Комментарии поясняют команды и объясняют, как нужно изменить программу для того или иного случая.

ЛИТЕРАТУРА

1. *Кудрявин Л.А., Беляев О.Ф., Заваруев В.А., Котович О.С.* Применение нелинейной теории упругости к расчету двумерной деформации трикотажа // Изв. вузов. Технология текстильной промышленности. – 2010, №8. С.69...72.

2. *Попов Е.П.* // Теория и расчет упругих стержней. М: Наука, 1986, с. 294.

Рекомендована кафедрой технологии трикотажного производства. Поступила 09.10.10.