ПОЛИТЕХ
Санкт-Петербургский
Политехнический Университет
Петра Великого

Использование современных программных комплексов в расчете строительных конструкций

Яваров Александр Валерьевич, к. т.н., доцент СПбПУ (Политех)

Санкт-Петербург

2017 г.

Получение матриц элементов:

1. Запись функционала
2. Введение функций формы
3. Интегрирование (как правило, численное)

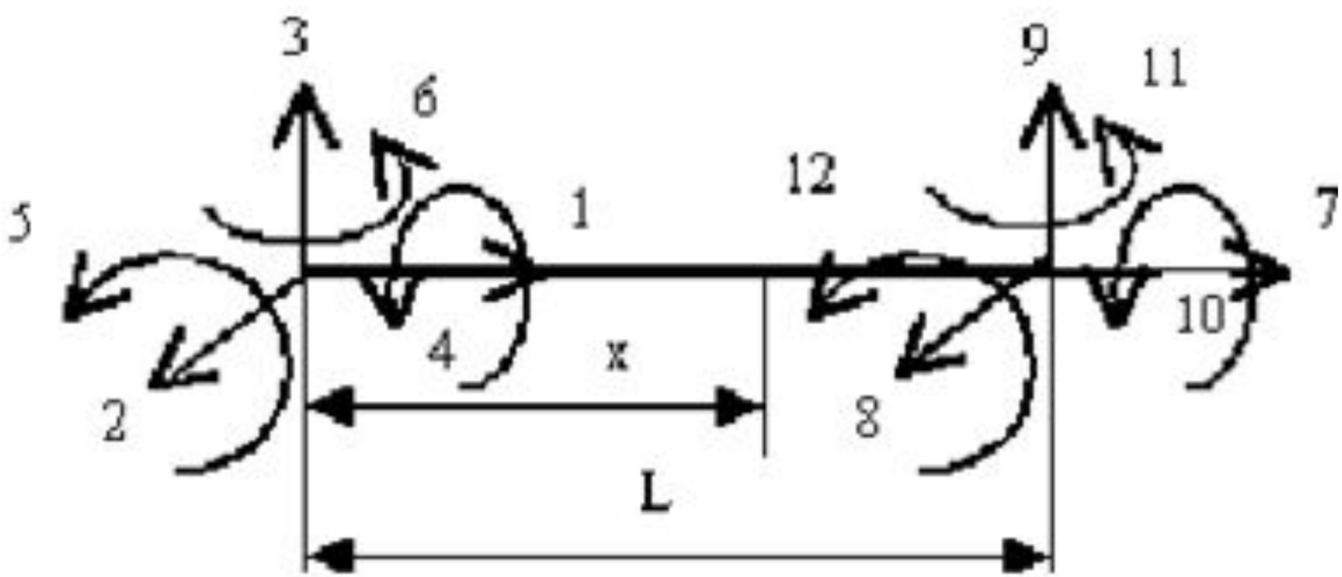
Конечный элемент пространственного стержня

Общее соотношение МКЭ для статических задач:

$$\mathbf{K\,U=P},$$

где K – глобальная матрица жесткости, U – вектор-столбец перемещений, P – вектор-столбец внешних нагрузок.

Степени свободы узлов пространственного стержневого элемента



$$U=\begin{pmatrix}1\\2\\3\\4\\5\\6\\7\\8\\9\\10\\11\\12\end{pmatrix}=\begin{pmatrix}u_1\\u_2\\u_3\\\phi_4\\u_3'\\u_2'\\u_7\\u_8\\u_9\\\phi_{10}\\u_9'\\u_8'\end{pmatrix}$$

Функционал потенциальной энергии деформации

$$\Pi = \frac{1}{2}\int_0^l \left(EI_z\,(u''_2)^2 + EI_y\,(u''_3)^2 + EA\,(u'_1)^2 + G_{кр}\,I_{кр}\,(\phi'_1)^2\right) dx =$$

$$= \frac{1}{2}\int_0^l \left(EI_z\left(\frac{d^2 u_2}{dx^2}\right)^2 + EI_y\left(\frac{d^2 u_3}{dx^2}\right)^2 + EA\left(\frac{d u_1}{dx}\right)^2 + G_{кр}\,I_{кр}\left(\frac{d\phi}{dx}\right)^2\right) dx$$

Данное выражение получено с учетом известного *соотношения* :

$$M = EIw''$$

например, для первого члена подинтегрального *выражения* :

$$EI_z\,(u''_2)^2 = EIu''_2\,u''_2 = M\,u''_2$$

Функции формы

Зададим функции формы для продольных перемещений и *кручения* :

$$u[x] = Э_1[x]\,u_1 + Э_2[x]\,u_7$$

$$\phi[x] = Э_1[x]\,\phi_4 + Э_2[x]\,\phi_{10}$$

Для продольных перемещений и кручения полиномы следует использовать полиномы 1 - го порядка.

$$Э_1[x] = 1 - \frac{x}{l}$$

$$Э_2[x] = \frac{x}{l}$$

Функции формы

Запишем функции формы для вертикальных перемещений v :

$$v[x] = H_1[x]\, u_2 + H_2[x]\, u_2' + H_3[x]\, u_8 + H_4[x]\, u_8'$$

Вместо кооэффициентов подставим полиномы Эрмита (полиномы 3 - го порядка) :

$$H_1[x\_] = 1 - 3\frac{x^2}{L^2} + 2\frac{x^3}{L^3}$$

$$H_2[x\_] = x - 2\frac{x^2}{L} + \frac{x^3}{L^2}$$

$$H_3[x\_] = 3\frac{x^2}{L^2} - 2\frac{x^3}{L^3}$$

$$H_4[x\_] = -\frac{x^2}{L} + \frac{x^3}{L^2}$$

Подставим значение $x = 0$ и L

$v[x]\ /.\ x \to 0$

$v[x]\ /.\ x \to L$

Функции формы

Далее запишем функции формы для изгиба

в плоскости xz (z - верт ось, x - вдоль оси стержня) :

$$v'[x] = H_1'[x]\, u_2 + H_2'[x]\, u_2' + H_3'[x]\, u_8 + H_4'[x]\, u_8'$$

Подставим значение x = 0 и L

$v'[x]$ /. $x \to 0$

$v'[x]$ /. $x \to L$

Получается система из 16 коэффициентов и 16 условий,

представленная в виде таблицы ниже. Из таблицы видно,

что прямая подстановка значений x,

позволяет проверить правильность выбора полиномов.

	H1	H2	H3	H4
x=0	1	0	0	0
x=l	0	0	1	0

	H'1	H'2	H'3	H'4
x=0	0	1	0	0
x=l	0	0	0	1

;

Для горизонтально перемещения рассуждения аналогичны

$$w[x] = H_1[x]\, u_2 - H_2[x]\, u_2' + H_3[x]\, u_8 - H_4[x]\, u_8'$$

$$w'[x] = H_1'[x]\, u_2 - H_2'[x]\, u_2' + H_3'[x]\, u_8 - H_4'[x]\, u_8'$$

Функции формы

Составим матрицу функций форм H. Каждой строке соответствует своя функция

$$H = \begin{pmatrix} Э_1[x] & 0 & 0 & 0 & 0 & 0 & Э_2[x] & 0 & 0 & 0 & 0 & 0 \\ 0 & H_1[x] & 0 & 0 & 0 & H_2[x] & 0 & H_3[x] & 0 & 0 & 0 & H_4[x] \\ 0 & 0 & H_1[x] & 0 & -H_2[x] & 0 & 0 & 0 & H_3[x] & 0 & -H_4[x] & 0 \\ 0 & 0 & 0 & Э_1[x] & 0 & 0 & 0 & 0 & 0 & Э_2[x] & 0 & 0 \\ 0 & 0 & H_1'[x] & 0 & -H_2'[x] & 0 & 0 & 0 & H_3'[x] & 0 & -H_4'[x] & 0 \\ 0 & H_1'[x] & 0 & 0 & 0 & H_2'[x] & 0 & H_3'[x] & 0 & 0 & 0 & H_4'[x] \end{pmatrix}$$

Третья и пятая строка включают члены с минусам, что является следствием гипотезы Бернулли - Эйлера, по которой деформации выражаются следующим *образом* :

$$\epsilon[x] = \frac{du}{dx}$$

$$\kappa_1[x] = \frac{d^2 v}{dx^2}$$

$$\kappa_2[x] = -\frac{d^2 w}{dx^2}$$

$$\epsilon_{кр}[x] = \frac{d\phi}{dx}$$

Функционал

Запишем функционал в матричной *форме* :

$$\Pi = \frac{1}{2}\int_0^L \left(U^T \cdot B^T \cdot D \cdot B \cdot U\right) dx = U^T \left(\frac{1}{2}\int_0^L \left(B^T \cdot D \cdot B\right) dx\right) U$$

Матрица B

$$B = \begin{pmatrix} Э_1'[x] & 0 & 0 & 0 & 0 & 0 & Э_2'[x] & 0 & 0 & 0 & 0 & 0 \\ 0 & H_1''[x] & 0 & 0 & 0 & H_2''[x] & 0 & H_3''[x] & 0 & 0 & 0 & H_4''[x] \\ 0 & 0 & H_1''[x] & 0 & -H_2''[x] & 0 & 0 & 0 & H_3''[x] & 0 & -H_4''[x] & 0 \\ 0 & 0 & 0 & Э_1'[x] & 0 & 0 & 0 & 0 & 0 & Э_2'[x] & 0 & 0 \end{pmatrix}$$

Матрица D - матрица с жесткотями стержня.

$$DD = \begin{pmatrix} EA & 0 & 0 & 0 \\ 0 & EI_z & 0 & 0 \\ 0 & 0 & EI_y & 0 \\ 0 & 0 & 0 & GI_{кр} \end{pmatrix}$$

Интегрирование

$$\text{Beam} = \int_0^L \text{Dot}[\text{Bt}.\text{DD}.\text{B}]\, dx$$

Матрица жесткости:

$$\begin{pmatrix}
\frac{EA}{L} & 0 & 0 & 0 & 0 & 0 & -\frac{EA}{L} & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{12\,EI_z}{L^3} & 0 & 0 & 0 & \frac{6\,EI_z}{L^2} & 0 & -\frac{12\,EI_z}{L^3} & 0 & 0 & 0 & \frac{6\,EI_z}{L^2} \\
0 & 0 & \frac{12\,EI_y}{L^3} & 0 & -\frac{6\,EI_y}{L^2} & 0 & 0 & 0 & -\frac{12\,EI_y}{L^3} & 0 & -\frac{6\,EI_y}{L^2} & 0 \\
0 & 0 & 0 & \frac{GI_{\text{кр}}}{L} & 0 & 0 & 0 & 0 & 0 & -\frac{GI_{\text{кр}}}{L} & 0 & 0 \\
0 & 0 & -\frac{6\,EI_y}{L^2} & 0 & \frac{4\,EI_y}{L} & 0 & 0 & 0 & \frac{6\,EI_y}{L^2} & 0 & \frac{2\,EI_y}{L} & 0 \\
0 & \frac{6\,EI_z}{L^2} & 0 & 0 & 0 & \frac{4\,EI_z}{L} & 0 & -\frac{6\,EI_z}{L^2} & 0 & 0 & 0 & \frac{2\,EI_z}{L} \\
-\frac{EA}{L} & 0 & 0 & 0 & 0 & 0 & \frac{EA}{L} & 0 & 0 & 0 & 0 & 0 \\
0 & -\frac{12\,EI_z}{L^3} & 0 & 0 & 0 & -\frac{6\,EI_z}{L^2} & 0 & \frac{12\,EI_z}{L^3} & 0 & 0 & 0 & -\frac{6\,EI_z}{L^2} \\
0 & 0 & -\frac{12\,EI_y}{L^3} & 0 & \frac{6\,EI_y}{L^2} & 0 & 0 & 0 & \frac{12\,EI_y}{L^3} & 0 & \frac{6\,EI_y}{L^2} & 0 \\
0 & 0 & 0 & -\frac{GI_{\text{кр}}}{L} & 0 & 0 & 0 & 0 & 0 & \frac{GI_{\text{кр}}}{L} & 0 & 0 \\
0 & 0 & -\frac{6\,EI_y}{L^2} & 0 & \frac{2\,EI_y}{L} & 0 & 0 & 0 & \frac{6\,EI_y}{L^2} & 0 & \frac{4\,EI_y}{L} & 0 \\
0 & \frac{6\,EI_z}{L^2} & 0 & 0 & 0 & \frac{2\,EI_z}{L} & 0 & -\frac{6\,EI_z}{L^2} & 0 & 0 & 0 & \frac{4\,EI_z}{L}
\end{pmatrix}$$

```
"Экспорт координат узлов из Exel. Десятичные с запятой";
DXF = Import["A:/PHD work/Алгоритмы/Стержень Линейный/Координаты узлов.xlsx"];
"Размерность экспортированного массива данных";
d := Dimensions[DXF];
"Число узлов";
Nузлов := Part[d, 3]
"Приведение DXF к списку, а затем к матричному виду";
MatrixImport = Partition[Flatten[DXF], Nузлов];
"Список узлов с координатами";
(*Будем считать, что 1 строка - x, 2 строка - y, 3 строка z*)
Do[Node[n] = {MatrixImport[[1, n]], MatrixImport[[2, n]], MatrixImport[[3, n]]}, {n, 1, Nузлов}]
"Число элементов";
Nэлементов = Nузлов - 1;
"Вычисляем длину всех стержневых элементов. n - нумерация";

Do[L[n] = √((Node[n + 1][[1]] - Node[n][[1]])^2 +
    (Node[n + 1][[2]] - Node[n][[2]])^2 + (Node[n + 1][[3]] - Node[n][[3]])^2), {n, 1, Nэлементов}]

"Вычисляем проекции орта i1 всех элементов. Ось x направлена вдоль стержня";

(*i1[1]:=(Coords[2].x-Coords[1].x)/L;
i1[2]:=(Coords[2].y-Coords[1].y)/L;
i1[3]:=(Coords[2].z-Coords[1].z)/L;*)

Do[i1[1]n = (Node[n + 1][[1]] - Node[n][[1]]) / L[n], {n, 1, Nэлементов}];
Do[i1[2]n = (Node[n + 1][[2]] - Node[n][[2]]) / L[n], {n, 1, Nэлементов}];
Do[i1[3]n = (Node[n + 1][[3]] - Node[n][[3]]) / L[n], {n, 1, Nэлементов}];
```

```
(*vector multiplication to caclulate j1
                    |i       j      k  |
      j1=-i1 x k=  |i1[1] i1[2] i1[3]| =-(i*i1[2]*1-j*i1[1]*1);
                    | 0      0      1  |
   ort j1

  //ort j1
j1[1]:=-i1[2];
j1[2]:=i1[1];
j1[3]:=0;
L:=sqrt(sqr(j1[1])+sqr(j1[2]));
if L<1e-6 then
j1[2]:=1
else
begin
j1[1]:=j1[1]/L;
j1[2]:=j1[2]/L;
end;*)

(*Орт j1*)
```

Do[$locj1[1]_n$ = $i1[2]_n$, {n, 1, $N_{элементов}$}]

Do[$locj1[2]_n$ = -$i1[1]_n$, {n, $N_{элементов}$}]

Do[$j1[3]_n$ = 0, {n, 1, $N_{элементов}$}]

```
(*Длина L:=sqrt(sqr(j1[1])+sqr(j1[2]))*)
```

Do$\left[LJ_n = \sqrt{locj1[1]_n{}^2 + locj1[2]_n{}^2}, \{n, 1, N_{элементов}\}\right]$

```
(*vector multiplication to caclulate k1
                 | i      j      k    |
  k1=-i1 x j1=- | i1[1] i1[2] i1[3] | =
                 | j1[1] j1[2]   0    |
      =-(i*(-i1[3]*j1[2])+j*(i1[3]*j1[1])+k*(i1[1]*j1[2]-i1[2]*j1[1]));
  local ort k1

  k1[1]:=(-i1[3]*j1[2]);
k1[2]:=i1[3]*j1[1];
k1[3]:=(i1[1]*j1[2]-i1[2]*j1[1]);

L:=sqrt(sqr(k1[1])+sqr(k1[2])+sqr(k1[3]));
k1[1]:=k1[1]/L;
k1[2]:=k1[2]/L;
k1[3]:=k1[3]/L;*)

(*Орт k1*)
Do[Lock1[1]_n = -i1[3]_n j1[2]_n, {n, 1, N_элементов}]
Do[Lock1[2]_n = i1[3]_n j1[1]_n, {n, 1, N_элементов}]
Do[Lock1[3]_n = i1[1]_n j1[2]_n - i1[2]_n j1[1]_n, {n, 1, N_элементов}]

(*Длина L:=sqrt(sqr(k1[1])+sqr(k1[2])+sqr(k1[3]))*)
Do[LK_n = Sqrt[Lock1[1]_n^2 + Lock1[2]_n^2 + Lock1[3]_n^2], {n, 1, N_элементов}]
(*Проекции орта k1*)
Do[k1[1]_n = Lock1[1]_n / LK_n, {n, 1, N_элементов}]
Do[k1[2]_n = Lock1[2]_n / LK_n, {n, 1, N_элементов}]
Do[k1[3]_n = Lock1[3]_n / LK_n, {n, 1, N_элементов}]


 Do[C_n = {{i1[1]_n, i1[2]_n, i1[3]_n},
            {j1[1]_n, j1[2]_n, j1[3]_n},
            {k1[1]_n, k1[2]_n, k1[3]_n}}, {n, 1, N_элементов}]
(*C:=TMatrix.create(3,3);
```

```
Do[LKn = Sqrt[Lock1[1]n^2 + Lock1[2]n^2 + Lock1[3]n^2], {n, 1, Nэлементов}]
(*Проекции орта k1*)
Do[k1[1]n = Lock1[1]n / LKn, {n, 1, Nэлементов}]
Do[k1[2]n = Lock1[2]n / LKn, {n, 1, Nэлементов}]
Do[k1[3]n = Lock1[3]n / LKn, {n, 1, Nэлементов}]


Do[Cn = {{i1[1]n, i1[2]n, i1[3]n},
         {j1[1]n, j1[2]n, j1[3]n},
         {k1[1]n, k1[2]n, k1[3]n}}, {n, 1, Nэлементов}]
(*C:=TMatrix.create(3,3);
C[1,1]:=i1[1];C[1,2]:=j1[1];C[1,3]:=k1[1];
C[2,1]:=i1[2];C[2,2]:=j1[2];C[2,3]:=k1[2];
C[3,1]:=i1[3];C[3,2]:=j1[3];C[3,3]:=k1[3];*)
ZeroC = ConstantArray[0, {3, 3}];


Do[
 MatrixForm[Qn = ArrayFlatten[{{Cn, ZeroC, ZeroC, ZeroC}, {ZeroC, Cn, ZeroC, ZeroC},
 {ZeroC, ZeroC, Cn, ZeroC}, {ZeroC, ZeroC, ZeroC, Cn}}]],
 {n, 1, Nэлементов}]
```

Спасибо за внимание!