Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования
«Московский Государственный Технический Университет имени Н.Э. Баумана»
Национальный исследовательский университет техники и технологий
(МГТУ им. Н.Э.Баумана)

Факультет «Робототехника и комплексная автоматизация»
Кафедра «Системы автоматизированного проектирования»
(РК-6)

Отчет по лабораторной работе №2
По дисциплине «Прикладная механика»
На тему «Расчет статически-неопределимой балки методом конечных элементов»

Выполнил студент группы РК6-34Б
Новиков В.П.

Москва, 2019 г.

Задача

Составить конечно-элементную программу для расчета статически неопределимой балки и проверить корректность ее работы с использованием Siemens NX.

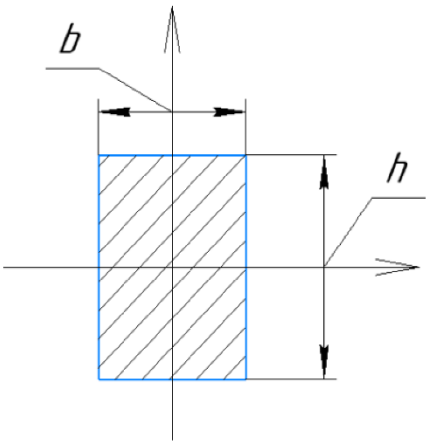


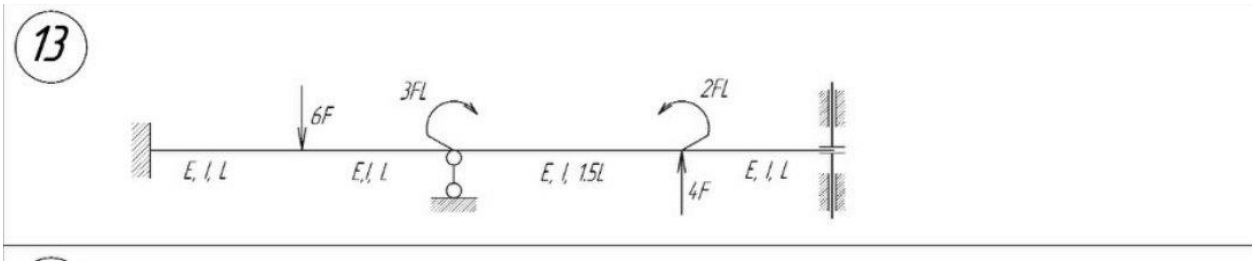
Рисунок 1. Поперечное сечение балки

Исходные данные: Материал балки: сталь (модуль Юнга $E = 2e11$ Па).

Сечение балки: прямоугольное (Рисунок 1).

Геометрические параметры балки: $l = 0.1$ м, $b = 10$ мм, $h = 20$ мм Величина нагрузки: $F = 10$ Н.

Вариант



Описание алгоритма решения

1) Задаем исходные данные:

b, h – ширина и высота сечения;

Jy – момент инерции;

l – длина единичного отрезка;

E – модуль Юнга;

F – величина силы;

N_el – кол-во КЭ в системе;

E_sys – вектор упругих свойств системы;

L_sys – вектор длин конечных элементов;

N_dofs – кол-во степеней свободы системы;

K_global – глобальная матрица жёсткости;

U_node – вектор граничных условий;

F_node – вектор внешних сил и моментов;

K_local – локальная матрица жесткости;

Index_M – матрица индексов.

2) Создаем матрицу индексов Index_M, хранящую номера узлов в глобальной и локальной системах координат.

3) С помощью вложенных циклов заполняем глобальную матрицу жесткости K_global на основе матриц жесткости конечных элементов K_e, которые считаются функцией K_loc_calc.

4) Накладываем кинематические uграничные условия, модифицируем матрицу жесткости. Для этого задается вектор U_node количеством элементов N_dofs. В ячейке, где узел закреплен, ставим 1, в свободных узлах ставим 0. Обнуляем столбцы и строки с номерами, соответствующими номерам элементов, равных 1, в векторе U_node. На пересечении этих столбцов и строк ставим 1.

5) Вычисляем вектор перемещений U = pinv(K_glob)*F_node';.

6) Выводим на экран вектор перемещений U, переводя значения в нужные единицы измерения (мм для перемещений и градусы для углов поворота).

Элементы, стоящие на нечетных позициях, отвечают за вертикальные перемещения узлов, а на четных - за угол поворота.

Текст программы

```
function prik_meich_lab2
    format long
    h = 20; % Высота сечения, мм
    b = 10; % Ширина сечения, мм
    Jy = (b*h^3)/12; % Момент инерции сечения
    l = 100; % Длина, мм
    E = 2e5; % Модуль Юнга
    F = 10; % Сила, Н
    NumberOfElements = 4; % Кол-во КЭ в системе
    E_System = [E, E, E, E]; % Вектор упругих своиств системы
    LenghtSystem = [l, l, 1.5*l, l]; % Вектор длин конечных элеметов
    DegreesOfFreedomNumber = 2*(NumberOfElements + 1); % Кол-во степенй
свободы системы
    K_GlobalMatrix = zeros(DegreesOfFreedomNumber); % Глобальная матрица
жёсткости
    Vector_U = [1,1,  0,0, 1,0, 0,0, 0,1]; % Вектор граничных условий
    Vector_F = [0,0,  -6*F,0,  0,-3*F*l, 4*F,2*F*l, 0,0]; % Вектор внешних
сил и ментов
    K_LocalMatrix = zeros(4); % Локальная матрица жёсткости
    IndexMatrix = [1:4; % Матрица индексов
              3:6;
              5:8;
              7:10;];

    for i = 1:NumberOfElements % Вычисление матрицы жёсткости для каждого КЭ
и ансамблирование
        K_LocalMatrix = K_LocalMatirxCalc(LenghtSystem(i), E_System(i), Jy);
        for j = 1:4
            for k = 1:4
              K_GlobalMatrix(IndexMatrix(i, j),IndexMatrix(i, k)) =
K_GlobalMatrix(IndexMatrix(i, j),IndexMatrix(i, k))+K_LocalMatrix(j,k);
            end
        end
    end

    for i = 1:DegreesOfFreedomNumber % Наложение кинематических граничных
условий
       if(Vector_U(i) == 1)
          K_GlobalMatrix(i, :) = 0;
          K_GlobalMatrix(:, i) = 0;
          K_GlobalMatrix(i, i) = 1;
       end
    end

    U = pinv(K_GlobalMatrix)*Vector_F';

    for i = 1:DegreesOfFreedomNumber % Вывод вектора перемещений на экран
     if(rem(i,2) == 1)
        fprintf('%f mm\n',U(i)) % Вывод перемещений в мм
     else
         fprintf('%f deg\n',U(i)*180/pi)  % Вывод угла поворота в градусах
     end
    end
end
```

```
function K = K_LocalMatirxCalc(L, E, J)  % Вычсление матрицы жесткости для
плоского балочного элемента
    K = [12*E*J/(L^3),  6*E*J/(L^2), -12*E*J/(L^3),  6*E*J/(L^2);
          6*E*J/(L^2),      4*E*J/L,  -6*E*J/(L^2),      2*E*J/L;
        -12*E*J/(L^3), -6*E*J/(L^2),  12*E*J/(L^3), -6*E*J/(L^2);
          6*E*J/(L^2),      2*E*J/L,  -6*E*J/(L^2),      4*E*J/L];
end
```

2 вариант программы

```
function main()
N_el = input('elements mount ');
l = input('nominal length ');
b = input('section length ');
h= input('section width ');
Jy=b*h^3/12;
f = input('nominal force ');
E=2e7;

N_dof_el = 4;
N_dof_sys = (N_el+1)*N_dof_el/2;


U = zeros(1, N_dof_sys);
for i = 1:N_dof_sys
  if (mod(i,2)==1)
    U(i) = input('first DOF of i element: ');
  end
  if (mod(i,2)==0)
    U(i) = input('fifth DOF of i element: ');
  end
end

F = zeros(1, N_dof_sys);
for i = 1:N_dof_sys
  if (mod(i,2)==1)
    F(i) = f*input('force of i element(coef only): ');
  end
  if (mod(i,2)==0)
    F(i) = f*l*input('moment of i element(coef only): ');
  end
end

L = zeros(1, N_el);
for i = 1:N_el
  L(i) = l*input('length i element: ');
end

Index_M = zeros(N_el,N_dof_el);
for i = 1:N_el
  Index_M(i, :) = [((i-1)*2+1):((i-1)*2+N_dof_el)];
end

K_el = zeros(N_dof_el);
```

```
K_g = zeros(N_dof_sys);
E_sys = [E, E, E];

for i = 1:N_el
  K_el = K_el_calc(E_sys(1), L(1), Jy)
  j = Index_M(i, 1);
  k = Index_M(i, N_dof_el);
  K_g(j:k, j:k) = K_g(j:k, j:k) + K_el;
end

for i = 1:N_dof_sys
  if(U(i) == 0)
     K_g(:, i) = 0;
     K_g(i, :) = 0;
     K_g(i, i) = 1;
  end
end

res=inv(K_g)*F'

end

function K = K_el_calc(E, L, J)
K = [12*E*J/(L^3), 6*E*J/(L^2), -12*E*J/(L^3), 6*E*J/(L^2);
    6*E*J/(L^2), 4*E*J/L, -6*E*J/(L^2), 2*E*J/L;
    -12*E*J/(L^3), -6*E*J/(L^2), 12*E*J/(L^3), -6*E*J/(L^2);
    6*E*J/(L^2), 2*E*J/L, -6*E*J/(L^2), 4*E*J/L];
end
```

Результат работы программы

0.000000 mm

0.000000 deg

-0.004609 mm

-0.001567 deg

0.000000 mm

0.006267 deg

0.036797 mm

0.015398 deg

0.050234 mm

0.000000 deg

Выполнение расчета исходной системы в программе Siemens NX

CAD-модуль, создание геометрической модели:

1) Создаем модель моделирования;
2) Создаем плоский эскиз (строим 4 соединённых линии вдоль оси x) и задаем размер;
3) Сохраняем файл.

CAE-модуль, создание конечно-элементной модели балочного элемента:

1) Создаем новый файл КЭ модели;
2) Связываем файл с моделью, применив отображение геометрии прямых ;
3) Создаем конечно-элементную сетку. Выбираем параметр "1D сетка" и применяем к модели. Число элементов выбираем равным 1. .
4) Создаем поперечное сечение прямоугольной формы с нужными шириной и высотой;
5) Выбираем тип материала "AISI_STEEL_1005";
6) Сохраняем файл.

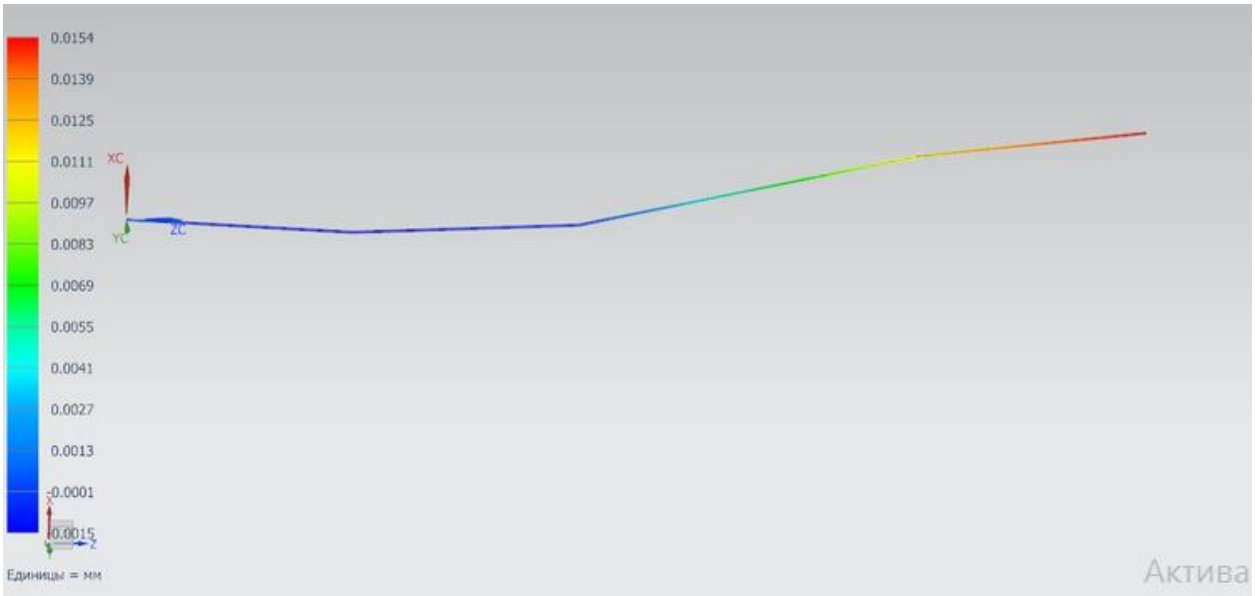
CAE-модуль, решение прочностной задачи:

1) Создаем новый файл симуляции;
2) Связываем с файлом конечно-элементной модели;
3) Задаем граничные условия. Выбираем в верхнем меню «Тип закрепления» -> «Ограничение задаваемое пользователем». В левом верхнем углу в фильтре выбора ставим «Узел». Выбираем нужный узел и ставим «фиксировано» в необходимом месте;
4) Задаем внешний силовой фактор. Выбираем в верхнем меню «Тип нагрузки» -> «Сила». Выбираем нужный узел и прикладываем к нему силу. Задаем вектор направления силы;

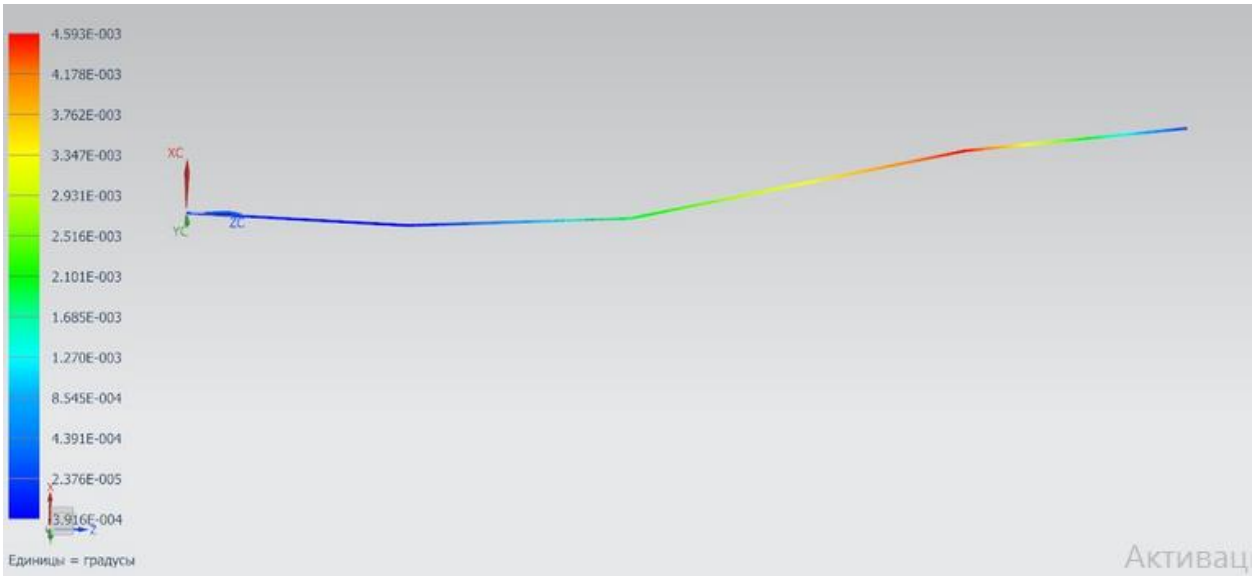
5) Задаем момент. Выбираем в верхнем меню «Тип нагрузки» -> «Момент». Выбираем нужный узел и прикладываем к нему момент. Задаем вектор направления момента;
6) Сохраняем файл;

Результат расчёта в Siemens NX

Перемещение в узлах



Поворот в узлах



	W_1, mm	θ_1, °	W_2, mm	θ_2, °	W_3, mm	θ_3, °	W_4, mm	θ_4, °	W_5, mm	θ_5, °
MatLab	0	0	-0.0046	-0.0015	0	0.006267	0.0367	0.0153	0.05	0
Siemens Nx	0	0	-0.0051	-0.0014	0	0.006263	0.0369	0.016	0.05	0
Отн. погрешность	0%	0 %	8%	4%	0%	0.3%	0.4%	6%	0.6%	0 %

Вывод

Результаты, полученные в NX и результаты, полученные в MatLab, равны с некой погрешностью. Это происходит потому что в обоих случаях использовался один и тот же метод расчёта(метод конечных элементов).