

# Решение плоской задачи механики деформируемого твердого тела методом конечных элементов в пакете Mathcad



В статье излагается способ решения плоской задачи механики деформируемого твердого тела методом конечных элементов (МКЭ) в пакете Mathcad. Задача рассматривается в упругой постановке, поскольку нелинейно-упругие и упругопластические решения в МКЭ обычно достигаются путем решения ряда упругих задач. Преимущества МКЭ не подлежат сомнению, когда возникает необходимость учета в расчетах сложных свойств материала, разнообразных граничных условий для областей со сложной конфигурацией. Программы, реализующие МКЭ, как правило, относятся к классу «тяжелых» САД-систем и обычно являются дорогостоящими. Поэтому для исследователя с небольшим опытом представляют интерес более доступные средства. Для инженера, не являющегося в то же время опытным

программистом, удобным инструментом может стать пакет Mathcad. Стандартная математическая форма записи выражений и возможность задания программных блоков делает прозрачной теоретическую основу метода и позволяет достаточно просто реализовать алгоритм МКЭ. Средства Mathcad в данном случае позволяют избежать необходимости решать задачи обращения и транспонирования матриц, нахождения корней системы алгебраических линейных уравнений и др., поскольку для этого предусмотрены специальные функции и инструменты. Изложенная методика может быть интересна инженерам горных и строительных специальностей при изучении методов расчета напряженно-деформированного состояния среды и конструкций, а также в учебных целях.

## Задача

Расчет напряженно-деформированного состояния массива горных пород, включающий процедуру решения краевой задачи теории упругости в постановке плоской деформации.

В геомеханике и при проектировании подземных сооружений приходится выполнять расчет напряженно-деформированного состояния массива горных пород, который в большинстве случаев включает поиск решения краевой задачи теории упругости в постановке плоской деформации. При этом наличие полостей неправильной формы, сложная конфигурация расчетной обла-

сти, физическая неоднородность и анизотропия свойств материала исключительно затрудняют либо делают невозможным нахождение решения в аналитическом виде. Поэтому при решении таких задач получили распространение численные методы, среди которых наиболее совершенным является метод конечных элементов (МКЭ).

## Программные средства

Mathcad  
Excel

## Результаты

Реализован метод конечных элементов и получены результаты расчета напряженно-деформированного состояния массива горных пород.

## 1. Суть метода конечных элементов

Некоторое сплошное тело, объем которого равен  $V$ , разбивается на элементы (рис. 1). Пусть на границе заданы поверхностные силы  $P$  и на тело действуют объемные силы  $Q$ . На основе принципа минимума потенциальной энергии поле перемещений узлов конечно-элементной сетки, удовлетворяющее уравнениям равновесия и совместности для двумерной задачи упругости, минимизирует функционал, заданный уравнением [1, 2]

$$\chi = \int_V \frac{1}{2} \{\epsilon\}^T \{Q\} dV - \int_V \{\delta\}^T \{Q\} dV - \int_S \{\delta\}^T \{P\} dS. \quad (1.1)$$

Векторы деформации  $\epsilon$  и напряжений  $\sigma$  определяются матрицами

$$\{\epsilon\} = \begin{bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{bmatrix}, \quad \{\sigma\} = \begin{bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{bmatrix}, \quad (1.2)$$

силы  $\{Q\} = \begin{bmatrix} q_x \\ q_y \end{bmatrix}$ ,  $\{P\} = \begin{bmatrix} p_x \\ p_y \end{bmatrix}$  действуют на

объем  $V$  и внешнюю поверхность  $S$ ,  $\{\delta\} = \begin{bmatrix} u \\ v \end{bmatrix}$  — вектор узловых перемещений.

Потенциальная энергия деформации тела также определяется перемещениями, или

$$\int_V \frac{1}{2} \{\epsilon\}^T \{\sigma\} dV = \int_V \frac{1}{2} \{\delta\}^T [K_c] \{\delta\} dV,$$

где  $[K_c]$  — матрица жесткости системы (МЖС) размерности  $f \times n$  ( $f$  — показатель степени свободы для узлов элементов, для двумерной задачи  $f = 2$ , для трехмерной  $f = 3$ ;  $n$  — количество элементов).

Из (1.1) и (1.2) можно получить:

$$[K_c] \{\delta\} = \{F\}, \quad (1.3)$$

где  $\{F\}$  — вектор узловых сил. Это основное матричное уравнение МКЭ. Компоненты матрицы жесткости определяются координатами узлов элементов и их упругими характеристиками: модулем упругости и коэффициентом Пуассона. Поэтому достаточным условием является определенность вектора нагрузки (узловых сил) или вектора смешанных сил, а также смешанные, взаимно уравновешивающие условия [1, 2].

## 2. Реализация МКЭ в пакете Mathcad

Рассмотрим простейший и в то же время наиболее распространенный вариант с разбиением области на треугольные элементы [3]. При дальнейшем изложении будем полагать, что читатель знаком с особенностями постановки задач и основными соотношениями теории упругости (см., например, [4]).

### 2.1. Подготовка исходных данных

Поскольку необходимо задать информацию о каждом из элементов и узлов расчетной области, удобнее всего использовать для подготовки данных табличный редактор Excel, тем более, что в Mathcad предусмотрена возможность импорта данных из файлов формата **.prn**. Итак, создадим в Excel два файла с таблицами, содержащими сведения об элементах и узлах. Структура таблиц и размерности величин в них приведены на рис. 2 и 3. В таблице данных об узлах имеются два столбца специальных переменных  $P_x$  и  $P_y$ , которым присваивается признак фиксации перемещения вдоль оси  $x$  или  $y$  соответственно (принимает значение 1, если задано нулевое перемещение, и 0 — если перемещение неизвестно).

Для сохранения таблиц в нужном формате выбираем пункт меню «Файл/Сохранить как...», указываем в соответствующих полях имя файла и тип — Форматированный текст (разделитель — пробел). После нажатия кнопки **Сохранить** нужно нажать в появившемся окне кнопку **Да**. В результате получаем файлы с именами, например, EL\_1.prn и KN\_1.prn.

### 2.2. Загрузка данных в Mathcad. Подготовка переменных

Для удобства нумерации элементов массивов далее в документе Mathcad определим индекс первых элементов в массивах равным единице:

ORIGIN := 1

Для получения данных из файлов в Mathcad используется функция **READPRN**(“filename.prn”) (возможно указание полного пути к файлу, в противном случае используется текущая папка, путь к которой можно узнать с помощью функции **CWD**).

Предположим, что ранее созданные файлы находятся в папке DATA на диске D. Присвоим их содержимое матрицам DEL и DKN:

Загрузка параметров элементов

DEL := READPRN("D:\DATA\EL\_1.prn")  
M := rows(DEL)      M = 231      - количество элементов

Загрузка параметров узлов

DKN := READPRN("D:\DATA\KN\_1.prn")  
R := rows(DKN)      R = 138      - количество узлов

Присвоим соответствующим переменным значения из матриц:

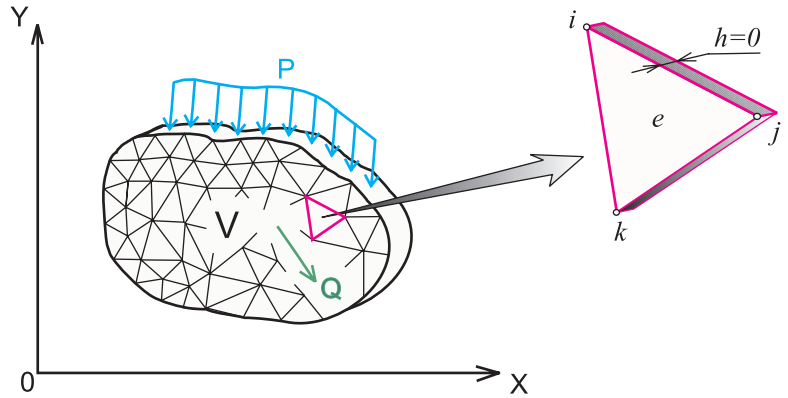
EL := augment(DEL<sup>{1}</sup>, DEL<sup>{2}</sup>, DEL<sup>{3}</sup>) - номера узлов, окружающих элементы

E := DEL<sup>{4}</sup> - модуль упругости

v := DEL<sup>{5}</sup> - коэффициент Пуассона

γ := DEL<sup>{6}</sup> - плотность

XY := augment(DKN<sup>{1}</sup>, DKN<sup>{2}</sup>) - координаты узлов



▲ Рис. 1. Конечно-элементное представление тела.

$X := DKN^{(3)}$	$Y := DKN^{(4)}$	- осевые компоненты узловых сил
$U := DKN^{(6)}$	$V := DKN^{(7)}$	- заданные осевые перемещения узлов
$PX := DKN^{(5)}$	$PY := DKN^{(8)}$	- признаки заданных узловых перемещений

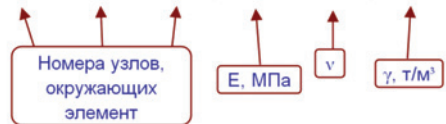
Для проверки правильности исходных данных и использования их в дальнейшем расчете необходимо сформировать вектор узловых сил с учетом действия массовых сил, векторы заданных перемещений, признаков и рассчитать площади элементов.

Площадь  $n$ -го элемента удобно задать в виде функции пользователя (в векторе  $V$  перечисляются глобальные номера узлов элемента):

```

A(n) := for i ∈ 1..3
        Vi ← ELn,i
        for i ∈ 1..3
            x1 ← XYVi,1
            y1 ← XYVi,2
        Δ ←  $\left| \frac{[(x_1 - x_2)(y_1 - y_3) - (x_1 - x_3)(y_1 - y_2)]}{2} \right|$ 
    
```

	A	B	C	D	E	F
1	1	12	13	1000	0.3	0
2	1	13	2	1000	0.3	0
3	2	13	14	1000	0.3	0
4	2	14	3	1000	0.3	0
5	3	14	15	1000	0.3	0
6	3	15	4	1000	0.3	0
7	4	15	16	1000	0.3	0
8	4	16	5	1000	0.3	0



▲ Рис. 2. Структура таблицы исходных данных с информацией об элементах.

	A	B	C	D	E	F	G	H	I
1	4.085	4.81	0	0	0	0	0	0	
2	4.785	5.63	0	0	0	0	0	0	
3	5.4	6.35	0	-1053	0	0	0	0	
4	6.315	6.35	-317.5	-1053	0	0	0	0	
5	0.57	1.02	0	0	0	0	0	0	
6	0.655	1.175	0	0	0	0	0	0	



▲ Рис. 3. Структура таблицы исходных данных с информацией об узлах и заданных узловых силах и перемещениях.

Общая площадь расчетной области:

$$\sum_{i=1}^M A(i) \cdot m^2 = 156.229 m^2$$

В массовые силы пересчитывается вес элементов, поровну на каждый из их узлов:

```

FM := for i in 1..R
      Yi ← 0
      for n in 1..M
        for i in 1..3
          Vi ← ELn,i
        for i in 1..3
          YVi ← YVi -  $\frac{A(n) \cdot \gamma_n \cdot g}{3}$ 
      Y
    
```

Узловые силы, перемещения и их признаки размещаются в векторах последовательными парами значений: на четных позициях вертикальные компоненты, на нечетных — горизонтальные:

<p>Вектор заданных узловых сил системы</p> $F_c := \begin{cases} \text{for } i \in 1..R \\ F_{c_{2i-1}} \leftarrow X_i \cdot 10^{-3} \\ F_{c_{2i}} \leftarrow (Y_i + FM_i) \cdot 10^{-3} \end{cases}$ $\sum_i F_{c_i} = -68.348$	<p>Вектор заданных узловых перемещений</p> $\delta_c := \begin{cases} \text{for } i \in 1..R \\ \delta_{c_{2i-1}} \leftarrow U_i \\ \delta_{c_{2i}} \leftarrow V_i \end{cases}$ $\sum_i \delta_{c_i} = 0$	<p>Вектор признаков заданных узловых перемещений</p> $S := \begin{cases} \text{for } i \in 1..R \\ S_{2i-1} \leftarrow PX_i \\ S_{2i} \leftarrow PY_i \end{cases}$ $\sum_i S_i = 22$
--	---	--

### 2.3. Расчет матрицы жесткости системы

Матрица жесткости системы  $[K_c]$  получается путем объединения матриц жесткости элементов  $[K]$ , которые, в свою очередь, рассчитываются по следующему выражению:

$$[K] = \Delta [B]^T [D] [B],$$

где  $\Delta$  — площадь элемента,  $[B]$  — матрица производных функций формы (функция влияния узлов),  $[D]$  — матрица связи напряжений и деформаций,

$$\{\sigma\} = [D] \{\epsilon\}. \quad (2.1)$$

Площадь элемента вычисляется с помощью ранее заданной функции пользователя  $A(n)$ . Матрицу  $[D]$  также удобно задать в виде функции пользователя. Для условий плоской деформации она будет иметь вид:

$$D(E, \nu) := \frac{E}{(1-2 \cdot \nu) \cdot (1+\nu)} \cdot \begin{bmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & \frac{(1-2 \cdot \nu)}{2} \end{bmatrix}$$

Матрица  $[B]$  связывает между собой перемещения узлов элемента с его деформацией:

$$\{\epsilon\} = [B] \{\delta\}. \quad (2.2)$$

Здесь  $\{\delta\} = \{u_i \quad v_i \quad u_j \quad v_j \quad u_k \quad v_k\}^T$ ,  
 $\{\epsilon\} = \{\epsilon_x \quad \epsilon_y \quad \gamma_{xy}\}^T$ ,

$$[B] = \begin{bmatrix} \frac{\partial N_i}{\partial x} & 0 & \frac{\partial N_j}{\partial x} & 0 & \frac{\partial N_k}{\partial x} & 0 \\ 0 & \frac{\partial N_i}{\partial y} & 0 & \frac{\partial N_j}{\partial y} & 0 & \frac{\partial N_k}{\partial y} \\ \frac{\partial N_i}{\partial y} & \frac{\partial N_i}{\partial x} & \frac{\partial N_j}{\partial y} & \frac{\partial N_j}{\partial x} & \frac{\partial N_k}{\partial y} & \frac{\partial N_k}{\partial x} \end{bmatrix},$$

где  $N_i = \frac{1}{2\Delta} (a_i + b_i x + c_i y)$ ;  $a_i = x_j y_k - x_k y_j$ ;  
 $b_i = y_j - y_k$ ;  $c_i = x_k - x_j$  (выражения для функций формы  $N_j$ ,  $N_k$  получаются путем круговой подстановки индексов в порядке  $i, j, k$ );  $i, j, k$  — номера узлов элемента;  $x_{i,j,k}$ ,  $y_{i,j,k}$  — координаты узлов.

После несложных преобразований матрицу  $[B]$  можно записать в виде

$$[B] = \frac{1}{2 \cdot \Delta} \begin{bmatrix} b_i & 0 & b_j & 0 & b_k & 0 \\ 0 & c_i & 0 & c_j & 0 & c_k \\ c_i & b_i & c_j & b_j & c_k & b_k \end{bmatrix}.$$

Расчет матрицы  $[B]$  представим в виде функции пользователя, задав предварительно вспомогательную матрицу  $P$ , определяющую порядок перестановки индексов в функциях формы:

```

P :=  $\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix}$ 
B(n) := for i in 1..3
        Vi ← ELn,i
        for i in 1..3
          Xi ← XYVi,1
          Yi ← XYVi,2
        for i in 1..3
          bi ← YPi,2 - YPi,3
          ci ← XPi,3 - XPi,2
        Δ ←  $\left| \frac{[(X_1 - X_2)(Y_1 - Y_3) - (X_1 - X_3)(Y_1 - Y_2)]}{2} \right|$ 
        B ←  $\frac{1}{2 \cdot \Delta} \cdot \begin{pmatrix} b_1 & 0 & b_2 & 0 & b_3 & 0 \\ 0 & c_1 & 0 & c_2 & 0 & c_3 \\ c_1 & b_1 & c_2 & b_2 & c_3 & b_3 \end{pmatrix}$ 
    
```

Матрица жесткости системы вычисляется в следующем программном блоке:

```

i := 1..2 · R   j := 1..2 · R   MOi,j := 0
Kc := MJS ← MO
for n in 1..M
  for i in 1..3
    V2i-1 ← 2 · ELn,i - 1
    V2i ← 2 · ELn,i
  K ← A(n) · B(n)T · D(En, νn) · B(n)
  MJE ← MO
  for i in 1..6
    for j in 1..6
      MJSVi,Vj ← MJSVi,Vj + Ki,j
MJS
    
```

Объединение матриц жесткости элементов в МЖС производится по следующему правилу: член МЖС  $K_{c_{i,j}}$  является суммой членов  $K_{i,j}$  из матриц жесткости всех элементов, примыкающих к узлу с  $i$ -й степенью свободы.

#### 2.4. Решение системы уравнений

Основное уравнение МКЭ (1.3) представляет собой систему линейных алгебраических уравнений (СЛАУ). Если какое-то  $i$ -е узловое перемещение задано, то число неизвестных уменьшается на единицу. При этом элементы  $i$ -го столбца МЖС следует умножить на заданное перемещение и результаты умножения вычесть из заданных узловых сил:

$$F_i := \begin{cases} \text{for } i \in 1..n \\ F_i \leftarrow F_{c_i} - \sum_{j=1}^n (K_{c_{i,j}} \cdot \delta_{c_j} \cdot S_j) \\ F \end{cases}$$

После этого  $i$ -й столбец и  $i$ -я строка МЖС, а также  $i$ -й неизвестный член в векторе сил могут быть удалены. Для удаления строк и столбцов из МЖС используем подматрицы, заданные функциями пользователя; M11 — удаляет первую строку и столбец, Mnn — последние, MI\_IV — промежуточные.

```
M11(A,p,n) := submatrix(A,p+1,n,p+1,n)
Mnn(A,p,n) := submatrix(A,1,p-1,1,p-1)
MI_IV(A,p,n) := { I ← submatrix(A,1,p-1,1,p-1)
                  II ← submatrix(A,1,p-1,p+1,n)
                  III ← submatrix(A,p+1,n,p+1,n)
                  IV ← submatrix(A,p+1,n,1,p-1)
                  M ← stack(augment(I,II),augment(IV,III)) }
K := { vc ← 0
      M ← Kc
      for i ∈ 1..n
          { M ← M11(M,i-vc,n-vc) if (i-vc = 1) ∧ {S1 = 1}
            M ← Mnn(M,i-vc,n-vc) if (i = n) ∧ {S1 = 1}
            M ← MI_IV(M,i-vc,n-vc) if (i-vc ≠ 1) ∧ (i ≠ n) ∧ {S1 = 1}
            vc ← vc + 1 if S1 = 1 }
      M }
F := { vc ← 0
      V ← F1
      for i ∈ 1..n
          { V ← submatrix(V,i+1-vc,n-vc,1,1) if {S1 = 1} ∧ (i-vc = 1)
            V ← submatrix(V,1,i-1-vc,1,1) if {S1 = 1} ∧ (i = n)
            V ← stack(submatrix(V,1,i-1-vc,1,1),submatrix(V,i+1-vc,n-vc,1,1)) if {S1 = 1} ∧ (i-vc ≠ 1) ∧ (i ≠ n)
            vc ← vc + 1 if S1 = 1 }
      V }
```

Таким образом, нужно решить СЛАУ  $[K]\{\delta\} = [F]$ . В данном случае возможности пакета Mathcad позволяют сильно упростить задачу. Для этого предусмотрена функция **Isolve**(M,V), которая использует итерационный метод Левенберга–Маркардта (Levenberg–Marquardt) для нахождения вектора решения СЛАУ, коэффициенты которой содержатся в массиве M, а свободные члены — в векторе V.

```
δ1 := Isolve(K,F)
δ := { vc ← 1
      for i ∈ 1..n
          { δi ← δci if S1 = 1
            δi ← δ1vc if S1 = 0
            vc ← vc + 1 if S1 = 0 }
      δ }
(u
 v) := { for i ∈ 1..R
        { u1 ← δ2i-1
          v1 ← δ2i }
        (u
 v) }
```

Программный модуль слева возвращает «на места» в общем векторе заданные узловые перемещения, ранее из него удаленные. Второй блок создает два вектора с осевыми компонентами узловых перемещений.

#### 2.5. Поиски осевых деформаций и напряжений в элементах

Зная полученные узловые перемещения, можно рассчитать деформации и напряжения для каждого элемента по соотношениям (2.1) и (2.2):

```
{εx εy γxy σx σy τxy} := { for i ∈ 1..M
                              { δe ← submatrix(M0,1,6,1,1)
                                for j ∈ 1..3
                                    { v2j-1 ← 2 · EL1,j - 1
                                      v2j ← 2 · EL1,j }
                                for j ∈ 1..6
                                    { δej ← δvj
                                      DE ← B(i) · δe
                                      DS ← D(E1,vi) · DE
                                      εx1 ← DE1
                                      εy1 ← DE2
                                      γxy1 ← DE3
                                      σx1 ← DS1
                                      σy1 ← DS2
                                      τxy1 ← DS3 }
                              {εx εy γxy σx σy τxy }
```

В каждом элементе также подсчитываются главные напряжения и угол между осью  $y$  и вектором максимального главного напряжения  $\sigma_1$ . Чтобы избежать деления на ноль, в строке вычисления угла использовано условное выражение, которое в случае равенства нулю знаменателя дроби присваивает углу  $\alpha_i$  значение  $\pi/2$ .

$$(\sigma_1 \ \sigma_3 \ \alpha) := \begin{cases} \text{for } i \in 1..M \\ \sigma_{1i} \leftarrow \frac{1}{2} \cdot \left[ \sigma_{xi} + \sigma_{yi} + \sqrt{(\sigma_{xi} - \sigma_{yi})^2 + 4 \cdot (\tau_{xyi})^2} \right] \\ \sigma_{3i} \leftarrow \frac{1}{2} \cdot \left[ \sigma_{xi} + \sigma_{yi} - \sqrt{(\sigma_{xi} - \sigma_{yi})^2 + 4 \cdot (\tau_{xyi})^2} \right] \\ \alpha_i \leftarrow \text{if } \left( \tau_{xyi} = 0, \frac{\pi}{2} \cdot \text{sign}(\sigma_{1i} - \sigma_{3i}), \text{atan} \left( \frac{\sigma_{1i} - \sigma_{3i}}{\tau_{xyi}} \right) \right) \\ \alpha_i \leftarrow \alpha_i \cdot \frac{180}{\pi} \\ (\sigma_1 \ \sigma_3 \ \alpha) \end{cases}$$

2.6. Сохранение результатов

Расчет по вышеизложенной процедуре занимает довольно непродолжительное время (например, на ПК с процессором Pentium IV 1300 MHz; RAM 128 MB время расчета 95 узлов для области из 119 элементов составляет около 3 с). Тем не менее, желательно сохранить результаты для последующего анализа.

Для этого сформируем матрицы, характеризующие напряженно-деформированное состояние (НДС) и поле перемещений, записав в них также координаты центров элементов и узлов:

```

EXY := for i ∈ 1..M
      for j ∈ 1..3
        V_j ← EL_{i,j}
      for j ∈ 1..3
        x_j ← XY_{V_j,1}
        y_j ← XY_{V_j,2}
      EX_i ← mean(x)
      EY_i ← mean(y)
    EXY ← augment(EX,EY)

XE := EXY^{(1)}    YE := EXY^{(2)}
    
```

(для нахождения центров элементов использована функция **mean()**, возвращающая среднее значение элементов вектора)

```

σ := augment(EXY, σ_x, σ_y, τ_xy, ε_x, ε_y, γ_xy, σ_1, σ_3, α)
uv := augment(XY, u, v)
    
```

Для записи данных в файл в пакете Mathcad предусмотрена функция **WRITEPRN** ("filename.prn"). Перед ее использованием можно задать количество знаков после запятой с помощью переменной PRNPRECISION и ширину столбца в файле с помощью переменной PRNCOLWIDTH:

```

PRNPRECISION := 4    PRNCOLWIDTH := 12
WRITEPRN("D:\RESULTUVV_1.prn") := uv
PRNPRECISION := 8    PRNCOLWIDTH := 16
WRITEPRN("D:\RESULTWDS_1.prn") := σ
    
```

Дальнейший анализ данных наиболее удобен в графическом виде.

2.7. Пример расчета

Выполним расчет НДС вокруг выработки круглого сечения, залегающей на глубине  $H$  от поверхности [4]. В целях упрощения принимаем расчетную область в виде квадрата со сторонами  $6 \cdot D$  ( $D$  — диаметр выработки); нагрузка от веса вышележащих горных пород задается в виде распределенных напряжений на границах области:  $p_y = \gamma H = 1000 \text{ кПа}$ ,  $p_x = 0, 3\gamma H = 300 \text{ кПа}$ . В силу симметрии достаточно рассмотреть четверть области (рис. 4).

В данном случае при разбиении на треугольные элементы получилась сеть из 95 узлов и 119 элементов. Нумерация — произвольная.

Все виды нагрузки, действующие на исследуемую область и формирующие в ней определенное напряженно-деформированное состояние, приводятся к статически эквивалентным силам, приложенным в узлах.

В силу симметрии граничные условия по перемещениям следующие: горизонтальные компоненты вдоль вертикальной ( $x = 0$ ) и вертикальные вдоль горизонтальной ( $y = 0$ ) сторон квадрата равны нулю. Неизвестны перемещения всех узловых точек внутри массива, на контуре выработки и на грани области.

Результаты расчета можно представить в виде эпюр (рис. 5), изолиний напряжений или перемещений (рис. 6, а), поверхностей уровня (рис. 6, б). Сохранение и представление результатов расчета в виде векторов (матриц) позволяет сделать это без затруднений (см., например, [5, 6]).

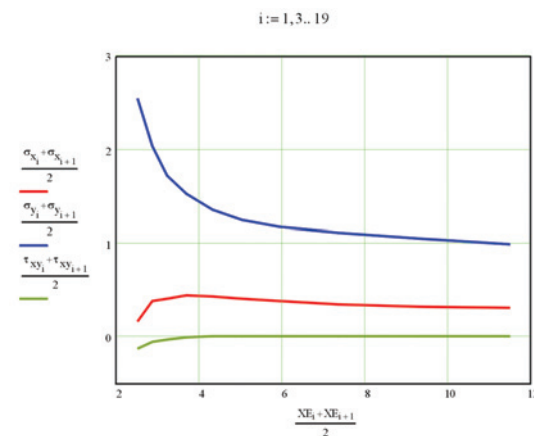
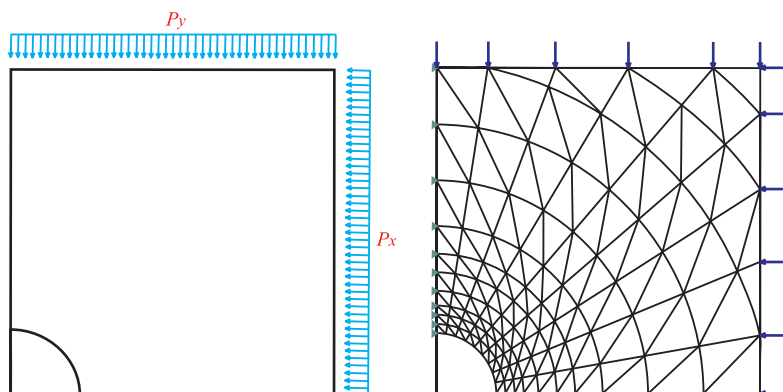


**Автор:**  
Бачурин Леонид Леонидович, заведующий лабораторией, ассистент кафедры разработки пластовых месторождений; Красноармейский филиал Донецкого национального технического университета, г. Красноармейск, Донецкая область, Украина

Заключение

Изложенная процедура решения задач теории упругости МКЭ в Mathcad позволяет избежать выполнения многих рутинных процедур: обращения матриц, решения СЛАУ, оптимизации исходных данных путем рациональной нумерации уз-

▼ Рис. 4. Расчетная схема и ее конечно-элементное представление.



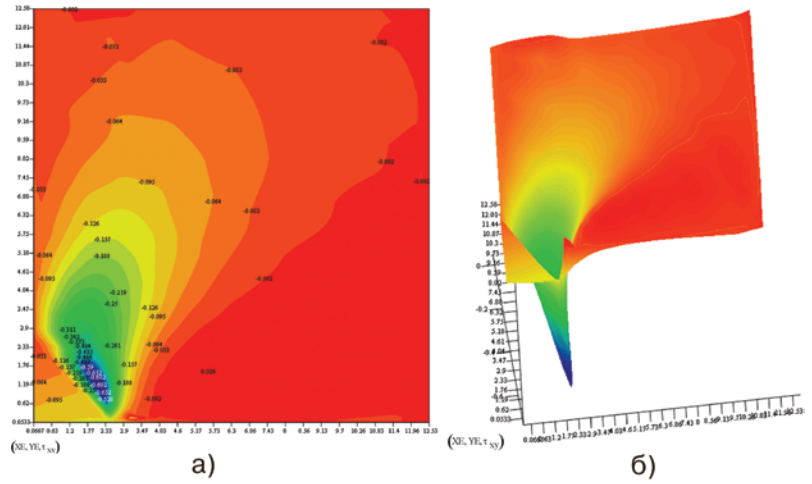
▲ Рис. 5. Эпюры напряжений вдоль горизонтальной оси (для сглаживания значения напряжений приводятся к центрам прямоугольников, составленных из соседних треугольников).

лов. Весь математический аппарат представлен в исходном виде.

Все расчеты выполняются с приемлемой для практического применения скоростью. Точность результатов в данном случае определяется частотой разбиения расчетной области и качеством исходных данных.

**Литература**

1. Зенкевич О. Метод конечных элементов в технике: Пер. с англ.— М., Мир, 1975.— 542 с.
2. Норри Д., де Фриз Ж. Введение в метод конечных элементов: Пер. с англ.— М.: Мир, 1981.— 304 с.
3. Ержанов Ж. С., Каримбаев Т. Д. Метод конечных элементов в задачах механики горных пород.— Алма-Ата: Наука, 1975.— 239 с.
4. Фадеев А. Б. Метод конечных элементов в геомеханике.— М.: Недра, 1987.— 221 с.
5. Дьяконов В. Mathcad 8/2000: Специальный справочник.— СПб: Питер, 2000.— 592 с.
6. Очков В. Ф. Mathcad 7 Pro для студентов и инженеров.— М.: КомпьютерПресс, 1998.— 384 с.



▲ Рис. 6. Примеры визуализации результатов расчета.

**Консультационный Центр MATLAB  
компании SoftLine  
приглашает прослушать  
курсы лекций:**

- Введение в MATLAB (16 акад. часов)
- Введение в Simulink (24 акад. часа)
- Применение нейронно-сетевых алгоритмов на базе системы MATLAB в финансовом и экономическом прогнозировании (24 акад. часа)
- Использование System Identification Toolbox для моделирования систем (24 акад. часа)
- Анализ и синтез систем управления в среде MATLAB (24 акад. часа)
- Анализ и фильтрация временных рядов в среде MATLAB (24 акад. часа)
- Цифровая обработка сигналов (32 акад. часа)
- Решение двумерных задач математической физики с помощью PDE Toolbox системы MATLAB (24 акад. часа)
- Решение двумерных задач математической физики с помощью системы Femlab (24 акад. часа)
- Избранные разделы статистики с использованием MATLAB (24 акад. часа)
- Simulink для экономистов (24 акад. часа)

Базовые программы курсов могут быть изменены по Вашему желанию. Занятия проводятся индивидуально или в маленьких группах (2–3 человека).

Подробная информация: <http://www.matlab.ru/lection>.

Также проводятся индивидуальные консультации по работе с MATLAB и Simulink.

**Заявки по e-mail [info@matlab.ru](mailto:info@matlab.ru) и тел. (095) 232-0023.**

**Подписчикам журнала  
«Exponenta Pro»  
скидка на обучение  
10%.**

**Новые книги**

*Минин Л. С., Хроматов В. Е.,  
Самсонов Ю. П.*

**Расчетные и тестовые задания  
по сопротивлению материалов:  
Учеб. пособие для вузов.  
Под ред. В. Е. Хроматова.  
М.: Высшая школа, 2003**

В сборнике приводятся варианты 16 заданий по основным разделам сопротивления материалов.

Рассмотрены следующие темы: растяжение-сжатие стержней, кручение, изгиб, устойчивость, балка на упругом основании, расчет тонкостенных оболочек, расчет осесимметричных тел, выносливость, изгибные колебания. Варианты отличаются расчетной схемой и числовыми данными. Даны примеры выполнения заданий в системе Mathcad и 128 задач для текущего контроля знаний в тестовой форме. В Приложении содержатся справочные данные по системе Mathcad, основные расчетные формулы сопротивления материалов и сортаменты профилей.

Пособие подготовлено на основе многолетнего опыта преподавания в МЭИ на кафедре ДПМ.

