**3. Сравнение полученных результатов с результатами современных расчетных программных комплексов.**

2.1 Метод конечных элементов

Суть метода следует из его названия. Область, в которой ищется решение дифференциальных уравнений, разбивается на конечное количество подобластей (элементов). В каждом из элементов произвольно выбирается вид аппроксимирующей функции. В простейшем случае это полином первой степени. Вне своего элемента аппроксимирующая функция равна нулю. Значения функций на границах элементов (в узлах) являются решением задачи и заранее неизвестны. Коэффициенты аппроксимирующих функций обычно ищутся из условия равенства значения соседних функций на границах между элементами (в узлах). Затем эти коэффициенты выражаются через значения функций в узлах элементов. Составляется система линейных алгебраических уравнений. Количество уравнений равно количеству неизвестных значений в узлах, на которых ищется решение исходной системы, прямо пропорционально количеству элементов и ограничивается только возможностями ЭВМ. Так как каждый из элементов связан с ограниченным количеством соседних, система линейных алгебраических уравнений имеет разрежённый вид, что существенно упрощает её решение.

Если говорить в матричных терминах, то собираются так называемые матрицы жёсткости (или матрица Дирихле) и масс. Далее на эти матрицы накладываются граничные условия (например, при условиях Неймана в матрицах не меняется ничего, а при условиях Дирихле из матриц вычёркиваются строки и столбцы, соответствующие граничным узлам, так как в силу краевых условий значение соответствующих компонент решения известно). Затем собирается система линейных уравнений и решается одним из известных методов.

Матрица жёсткости (матрица Дирихле) — матрица особого вида, использующаяся в методе конечных элементов для решения дифференциальных уравнений в частных производных. Она применяется при решениях задач электродинамики и механики.

Обычно матрица жёсткости получается разреженной, то есть содержащая большое количество нулей. Для работы с подобным типом матриц созданы специальные библиотеки (mtl4, SparseLib++, SPARSPAK и другие)

Определение

Элементы матрицы жёсткости в общем случае равны

|  |  |
| --- | --- |
|  | (1) |

Например, если дано [уравнение Пуассона](https://ru.wikipedia.org/wiki/%D0%A3%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D0%B5_%D0%9F%D1%83%D0%B0%D1%81%D1%81%D0%BE%D0%BD%D0%B0)

|  |  |
| --- | --- |
| -\nabla^2 u = f | (2) |

в пространстве \Omega и граничные условия — это u = 0 . 

Представим функцию как ряд:

|  |  |
| --- | --- |
| u \approx u^h = u_1\varphi_1+\cdots+u_n\varphi_n. | (3) |

u_i — это известные значения функции в узлах, а \varphi — некие базисные функции, то:

|  |  |
| --- | --- |
| A^{[k]}_{ij} = \int_{triangle}\nabla\varphi_i\cdot\nabla\varphi_j\, dx. | (4) |

Создание матрицы для одного треугольника

Пусть дан один конечный элемент, для простоты — треугольный. Матрица жёсткости, по сути, задаёт связи между узлами. Так как у элемента три узла (в локальной нумерации — 0, 1 и 2), то матрица будет иметь вид

|  |  |
| --- | --- |
| \begin{bmatrix}   S_{00} & S_{01} & S_{02}\\   S_{10} & S_{11} & S_{12}\\   S_{20} & S_{21} & S_{22}  \end{bmatrix} | (5) |

В дальнейшем матрицу для одного треугольника будем называть локальной, для всей сетки сразу — глобальной.

В общем случае, элементы S_{ij} определяются через линейные функции

|  |  |
| --- | --- |
| \alpha_1 = \cfrac {1} {4A} \big( (x_1y_2 + x_2y_1) \; + \; (y_1 - y_2)x \; + \; (x_2 - x_1)y \big) . | (6) |

где

A — площадь треугольного элемента.

\alpha_2 и \alpha_3 получаются из \alpha_1 цикличной перестановкой индексов. Удобно искать A как определитель матрицы

|  |  |
| --- | --- |
| A = \det \begin{bmatrix}   1 & x_1 & y_1 \\   1 & x_2 & y_2 \\   1 & x_3 & y_3  \end{bmatrix} | (7) |

Сами

|  |  |
| --- | --- |
| S_{ij} = \int (\nabla \alpha_i) (\nabla \alpha_j) dS \;\;\;\;\; i, j = 0, 1, 2 | (8) |

Первый вид обобщения на несколько треугольников - сшивание

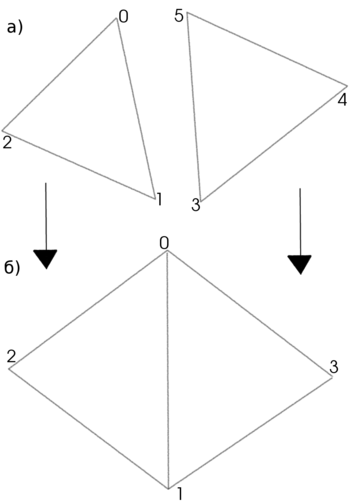
[](https://ru.wikibooks.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB:Stiffness_matrix_-_adding_triangles.png)

Рисунок 2.1 ‒ Сшивание

Для того, чтобы сделать из многих раздельных матриц, полученных выше, одну большую матрицу, описывающую отношения между узлами всей области расчёта, необходимо произвести процедуру объединения матриц. Пусть символ d обозначает разделённые элементы рис. 3.1, а символ c — объединённые элементы (рис. б).

Обозначим

|  |  |
| --- | --- |
| u_d^T = \begin{bmatrix} u_1 & u_2 & u_3 & u_4 & u_5 & u_6 \end{bmatrix} | (9) |

— вектор-строку значений функции в вершинах двух треугольников рис. 3.2. Символ u^T обозначает [транспонирование](https://ru.wikibooks.org/w/index.php?title=%D0%A2%D1%80%D0%B0%D0%BD%D1%81%D0%BF%D0%BE%D0%BD%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%B0%D1%8F_%D0%BC%D0%B0%D1%82%D1%80%D0%B8%D1%86%D0%B0&action=edit&redlink=1) матрицы u .  То есть, это вектор значений функции в шести узлах треугольников. Очевидно, что при объединении оных получится вектор u_d, содержащий только четыре компоненты.

Преобразование происходит по схеме

|  |  |
| --- | --- |
| u_d = Cu_c \; \Leftrightarrow \;   \begin{bmatrix}   u_1 \\   u_2 \\   u_3 \\   u_4 \\   u_5 \\   u_6  \end{bmatrix} \; = \;         \begin{bmatrix}   1 &   &   & \\     & 1 &   & \\     &   & 1 & \\     & 1 &   & \\     &   &   & 1 \\   1 &   &   &  \end{bmatrix} \cdot  \begin{bmatrix}   u_1 \\   u_2 \\   u_3 \\   u_4 \\  \end{bmatrix} | (10) |

Нумерация, конечно же, произвольная: необходимо равенство функции в соответствующих вершинах. Матрицу C называют матрицей преобразования, а само уравнение называют связанной системой.

Запишем теперь матрицу жёсткости для двух треугольников:

|  |  |
| --- | --- |
| S_d =     \begin{bmatrix}         S^{(1)} & 0 \\         0 & S^{(2)}     \end{bmatrix} \; \Leftrightarrow \;     \begin{bmatrix}       S_{00} &  S_{01} & S_{02} &        &        &        \\       S_{10} &  S_{11} & S_{12} &        &        &        \\       S_{20} &  S_{21} & S_{22} &        &        &        \\              &         &        & S_{33} & S_{34} & S_{35} \\              &         &        & S_{43} & S_{44} & S_{45} \\              &         &        & S_{53} & S_{54} & S_{55} \\     \end{bmatrix} | (11) |

Результирующая матрица S_{global} = C^T S_d C = 

|  |  |
| --- | --- |
| =     \begin{bmatrix}         S_{00}^{(1)} + S_{55}^{(2)} & S_{01}^{(1)} + S_{53}^{(2)} & S_{02}^{(1)} & S_{54}^{(2)} \\         S_{10}^{(1)} + S_{35}^{(2)} & S_{11}^{(1)} + S_{33}^{(2)} & S_{12}^{(1)} & S_{34}^{(2)} \\         S_{20}^{(1)}                & S_{20}^{(1)}                & S_{22}^{(1)} & 0            \\         S_{45}^{(2)}                & S_{43}^{(2)}                & 0            & S_{44}^{(2)} \\     \end{bmatrix} | (12) |

То есть, на каждом следующем шаге необходимо добавлять новые элементы к уже существующим.

Второй вид обобщения на несколько треугольников - дозаписывание

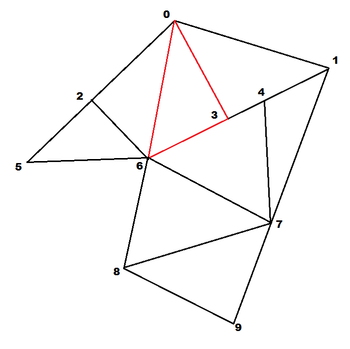
[](https://ru.wikibooks.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB:Mesh_FEM.png)

Рисунок 2.2 ‒ Сетка FEM

Пусть есть область, представленная и разбитая на треугольники так, как представлено на рисунке. Пусть данная сетка содержит N узлов. Создадим глобальную матрицу \mathfrak{S} (размера, очевидно, N \times N) и заполним её нулями. Начнём строить локальные матрицы S для треугольников, например, для \Delta 036 . 

Введём локальную нумерацию для данного треугольника: пусть его верхняя вершина имеет локальный номер 0, далее по часовой стрелке 1 и 2. Иначе говоря, пусть глобальным номерам 0,3,6 соответствуют локальные номера 0,1,2 соответственно.

Составим матрицу для этого треугольника так, как описано выше, получив что-то типа

|  |  |
| --- | --- |
| S =    \begin{bmatrix}   S_{00} & S_{01} & S_{02} \\   S_{10} & S_{11} & S_{12} \\   S_{20} & S_{21} & S_{22}      \end{bmatrix} | (13) |

Теперь заменим локальную нумерацию на глобальную. То есть запишем локальное число S_{00} как глобальное число \mathfrak{S_{00}}, S_{01} — как \mathfrak{S_{03}}, — как \mathfrak{S_{06}} и так далее.

Получим

|  |  |
| --- | --- |
| \mathfrak{S} =   \begin{bmatrix}   S_{00} & 0 & 0 & S_{03} & 0 & 0 & S_{06} & 0 & 0 & 0 \\   0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\   0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\   S_{10} & 0 & 0 & S_{13} & 0 & 0 & S_{16} & 0 & 0 & 0 \\   0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\   0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\   S_{20} & 0 & 0 & S_{21} & 0 & 0 & S_{22} & 0 & 0 & 0 \\   0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\   0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\   0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0      \end{bmatrix} | (14) |

С остальными треугольниками поступаем аналогично. Необходимо помнить, что надо "дописать" число в глобальную ячейку, то есть прибавить к уже существующему.

Матрица масс

Матрица масс собирается по тем же правилам, но чуть по другим формулам. Создаётся матрица S размеров три на три, затем говорится, что

|  |  |
| --- | --- |
| C_1 = \cfrac {(x_2 - x_1) (x_3 - x_1) + (y_2 - y_1)(y_3 - y_1)} {4A} | (15) |
| C_2 = \cfrac {(x_3 - x_2) (x_1 - x_2) + (y_3 - y_2)(y_1 - y_2)} {4A} | (16) |
| C_3 = \cfrac {(x_1 - x_3) (x_2 - x_3) + (y_1 - y_3)(y_2 - y_3)} {4A} | (17) |
| S_{22} = S_{22} + C_1 | (18) |
| S_{23} = S_{23} - C_1 | (19) |
| S_{32} = S_{32} - C_1 | (20) |
| S_{33} = S_{33} + C_1 | (21) |
| S_{33} = S_{33} + C_2 | (22) |
| S_{31} = S_{31} - C_2 | (23) |
| S_{13} = S_{13} - C_2 | (24) |
| S_{11} = S_{11} + C_2 | (25) |
| S_{11} = S_{11} + C_3 | (26) |
| S_{12} = S_{12} - C_3 | (27) |
| S_{21} = S_{21} - C_3 | (28) |
| S_{22} = S_{22} + C_3 | (29) |

где

A — площадь данного треугольника, которая считается, как в предыдущей главе.

C_2 и C_3 получаются из C_1 циклической перестановкой, равно как второй и третий блок элементов матрицы из первого.

После чего полученная матрица S записывается в матрицу \mathfrak{S} любым известным читателю способом. В коде используется метод дозаписи, приведённый выше.

Учёт граничных условий

Условия Дирихле

В случае граничных условий первого рода необходимо изменить матрицу

Граничное условие гласит, что функция в узлах на границе равна нулю, для узла необходимо вычеркнуть столбец и n-ую строку в матрице, а так же вычеркнуть сам узел из массива узлов решётки.

Условия Нейнмана

В случае граничных условий второго рода глобальная матрица не меняется.

2.1 Сравнение с расчётным программным комплексом ANSYS

Современные расчетные комплексы позволяют рассчитывать большой спектр инженерных задач. ANSYS — универсальная программная система конечно-элементного (МКЭ) анализа, существующая и развивающаяся на протяжении последних 30 лет, является довольно популярной у специалистов в сфере автоматизированных инженерных расчётов (CAE, Computer-Aided Engineering) и КЭ решения линейных и нелинейных, стационарных и нестационарных пространственных задач механики деформируемого твёрдого тела и механики конструкций (включая нестационарные геометрически и физически нелинейные задачи контактного взаимодействия элементов конструкций), задач механики жидкости и газа, теплопередачи и теплообмена, электродинамики, акустики, а также механики связанных полей. Моделирование и анализ в некоторых областях промышленности позволяет избежать дорогостоящих и длительных циклов разработки типа «проектирование — изготовление — испытания». Система работает на основе геометрического ядра Parasolid.

Моделирование использует метод конечных элементов. Суть метода следует из его названия. Область, в которой ищется решение дифференциальных уравнений, разбивается на конечное количество подобластей (элементов). В каждом из элементов произвольно выбирается вид аппроксимирующей функции. В простейшем случае это полином первой степени. Вне своего элемента аппроксимирующая функция равна нулю. Значения функций на границах элементов (в узлах) являются решением задачи и заранее неизвестны. Коэффициенты аппроксимирующих функций обычно ищутся из условия равенства значения соседних функций на границах между элементами (в узлах). Затем эти коэффициенты выражаются через значения функций в узлах элементов. Составляется система линейных алгебраических уравнений. Количество уравнений равно количеству неизвестных значений в узлах, на которых ищется решение исходной системы, прямо пропорционально количеству элементов и ограничивается только возможностями ЭВМ. Так как каждый из элементов связан с ограниченным количеством соседних, система линейных алгебраических уравнений имеет разрежённый вид, что существенно упрощает её решение.

Если говорить в матричных терминах, то собираются так называемые матрицы жёсткости (или матрица Дирихле) и масс. Далее на эти матрицы накладываются граничные условия (например, при условиях Неймана в матрицах не меняется ничего, а при условиях Дирихле из матриц вычёркиваются строки и столбцы, соответствующие граничным узлам, так как в силу краевых условий значение соответствующих компонент решения известно). Затем собирается система линейных уравнений и решается одним из известных методов.

С точки зрения вычислительной математики, идея метода конечных элементов заключается в том, что минимизация функционала вариационной задачи осуществляется на совокупности функций, каждая из которых определена на своей подобласти, для численного анализа системы позволяет рассматривать его как одну из конкретных ветвей диакоптики — общего метода исследования систем путём их расчленения.

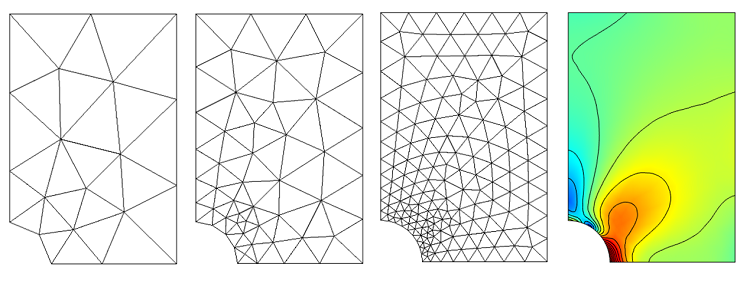


Рисунок 3.3 ‒ Пример сетки конечных элементов разной размерности

Для построения модели мною была выбрана система «Компас». Технология создания твердотельной модели сборки представлена в таблице1.1

таблица 1.1

|  |  |  |
| --- | --- | --- |
| 1 Создание элемента сборки | | |
| 1.1 Использована функция вытянуть | |  |
| 1.2 Для создания полости внутри цилиндра была использована функция вырезать | |  |
| 1.3 Для создания штока цилиндра была использована функция вытянуть | |  |
| 1.4 Для создания грундбуксы была использована операция выдавить | |  |
| 2. Сопряжения грундбуксы и штока | | |
| 2.1 Совмещаем грундбуксу и шток |  | |
| 2.2 Совмещаем шток с цилиндром |  | |

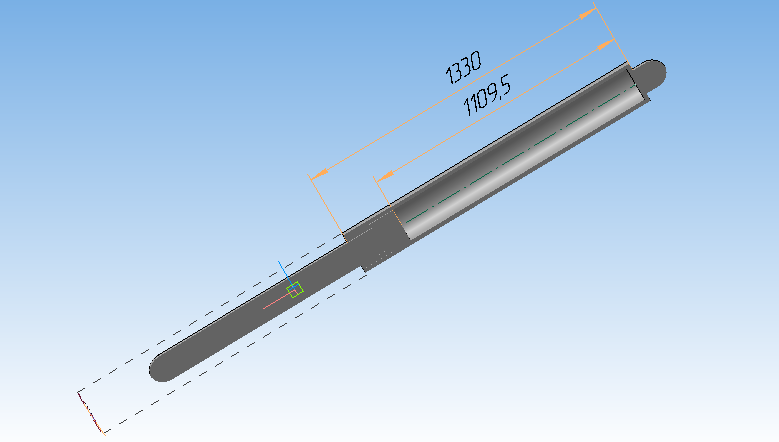


Рисунок 3.4 ‒ Сечение расчетной модели