

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ  
Учреждение образования  
«Витебский государственный технологический университет»

## **КОМПЬЮТЕРНЫЕ СИСТЕМЫ КОНЕЧНОЭЛЕМЕНТНЫХ РАСЧЕТОВ**

**Методические указания по выполнению лабораторных работ  
для студентов специальности 1-40 05 01-01 «Информационные системы и  
технологии (в проектировании и производстве)»  
для дневной и заочной на базе ссуз форм обучения**

УДК 519.62(076.5)

Составители:

Е. Б. Дунина, А. С. Соколова, А. Н. Бизюк

Рекомендовано к изданию редакционно-издательским советом УО «ВГТУ», протокол № 8 от 29.04.2022.

**Компьютерные системы конечноэлементных расчетов:** методические указания по выполнению лабораторных работ / сост. Е. Б. Дунина, А. С. Соколова, А. Н. Бизюк. – Витебск : УО «ВГТУ», 2022. – 55 с.

В методических указаниях изложены теоретические сведения и задания к лабораторным работам по дисциплине «Компьютерные системы конечноэлементных расчетов». Издание предназначено для студентов специальности 1-40 05 01-01 «Информационные системы и технологии (в проектировании и производстве)» дневной и заочной на базе ссуз форм обучения.

УДК 519.62(076.5)

© УО «ВГТУ», 2022

## СОДЕРЖАНИЕ

<b>1 ОБЩИЕ СВЕДЕНИЯ О COMSOL MULTIPHYSICS</b> .....	4
ЛАБОРАТОРНАЯ РАБОТА 1. ЭЛЕКТРИЧЕСКАЯ ШИНА – МУЛЬТИФИЗИЧЕСКАЯ МОДЕЛЬ .....	5
ЛАБОРАТОРНАЯ РАБОТА 2. СОЗДАНИЕ ПРИЛОЖЕНИЯ В APPLICATION BUILDER .....	23
<b>2 РЕШЕНИЕ СИСТЕМ ЛИНЕЙНЫХ УРАВНЕНИЙ</b> .....	26
ЛАБОРАТОРНАЯ РАБОТА 3. СОЗДАНИЕ ПРИЛОЖЕНИЯ РЕШЕНИЯ СИСТЕМ ЛИНЕЙНЫХ УРАВНЕНИЙ .....	28
<b>3 МЕТОД КОНЕЧНЫХ РАЗНОСТЕЙ (МКР)</b> .....	34
ЛАБОРАТОРНАЯ РАБОТА 4. СОЗДАНИЕ ПРИЛОЖЕНИЯ РЕШЕНИЯ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ ВТОРОГО ПОРЯДКА С ПОСТОЯННЫМИ КОЭФФИЦИЕНТАМИ МЕТОДОМ КОНЕЧНЫХ РАЗНОСТЕЙ.....	35
ЛАБОРАТОРНАЯ РАБОТА 5. СОЗДАНИЕ ПРИЛОЖЕНИЯ РЕШЕНИЯ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ ВТОРОГО ПОРЯДКА С ПРОИЗВОЛЬНЫМИ КОЭФФИЦИЕНТАМИ МЕТОДОМ КОНЕЧНЫХ РАЗНОСТЕЙ.....	39
<b>4 МЕТОД СЕТОК ДЛЯ РЕШЕНИЯ УРАВНЕНИЙ С ЧАСТНЫМИ ПРОИЗВОДНЫМИ</b> .....	42
ЛАБОРАТОРНАЯ РАБОТА 6. СОЗДАНИЕ ПРИЛОЖЕНИЯ РЕШЕНИЯ НЕСТАЦИОНАРНОГО УРАВНЕНИЯ ТЕПЛОПРОВОДНОСТИ.....	43
ЛАБОРАТОРНАЯ РАБОТА 7. РАСЧЕТ КОЛЕБАНИЯ СТРУНЫ.....	47
<b>ЛИТЕРАТУРА</b> .....	52
<b>ПРИЛОЖЕНИЕ А</b> .....	53
<b>ПРИЛОЖЕНИЕ Б</b> .....	54

# 1 ОБЩИЕ СВЕДЕНИЯ О COMSOL MULTIPHYSICS

В современном мире численное моделирование ложится в основу разработки практически любых продуктов и позволяет уменьшить время и стоимость вывода конечного продукта на рынок. COMSOL Multiphysics – это комплексная интегрированная среда для численного моделирования физических явлений и разработки приложений. Одним из важных преимуществ пакета COMSOL Multiphysics является возможность решения связанных междисциплинарных задач в единой среде. Например, задача расчета напряженно-деформированного состояния может быть сопряжена с моделированием сопутствующих физических процессов, таких как деформация тела в результате нагрева, или взаимодействие потока среды с упруго-деформируемым телом. Уникальные возможности моделирования, предлагаемые COMSOL Multiphysics, позволяют инженерам оптимизировать характеристики, сократить время расчетов и снизить затраты на разработку конечного продукта.

Графическая оболочка COMSOL Multiphysics позволяет выполнить все операции по построению расчетных моделей, т. е. для решения задачи не потребуется вспомогательное программное обеспечение. Также данная оболочка реализует все этапы решения задачи, начиная от построения геометрических моделей и заканчивая обработкой и визуализацией результатов. Таким образом интегрированная среда решает сложные системы дифференциальных, алгебраических и интегральных уравнений, которые являются математической моделью того или иного физического процесса.

Особенности архитектуры пакета COMSOL Multiphysics:

1. Имеет множество встроенных математических моделей, в том числе мультифизических. В самой программе запрограммированы уравнения, описывающие разные физические процессы.
2. Позволяет решать произвольные уравнения, заданные пользователем.
3. Превращает сложную модель в простое приложение, которым может воспользоваться широкий круг специалистов.

Пакет COMSOL имеет модульную структуру:

– модули электродинамики (AC/DC, радиочастоты, волновая оптика, геометрическая оптика, плазма, полупроводники);

– модули механики и акустики (механика конструкций, нелинейные материалы, композитные материалы, геомеханика, усталость материала, роторная динамика, динамика многотельных систем, акустика и др.);

– модули гидродинамики и теплопередачи (вычислительная гидродинамика, миксер, гидродинамика полимеров, микрогидродинамика, течения в пористых средах, течения в грунтах, течения в трубопроводах, молекулярные течения, металлургия, теплопередача);

– модули химии (химические реакции, электрохимические аккумуляторы, топливные ячейки и электролизеры, электроосаждение, коррозия, электрохимия);

– многоцелевые модули (модуль оптимизации, библиотека свойств материалов, модуль трассировки частиц, модуль термодинамики жидкостей и газов);

– продукты для интеграции, предназначенные для совместного использования COMSOL с другими средствами проектирования и математического моделирования.

## ЛАБОРАТОРНАЯ РАБОТА 1. ЭЛЕКТРИЧЕСКАЯ ШИНА – МУЛЬТИФИЗИЧЕСКАЯ МОДЕЛЬ

### *Краткие теоретические сведения*

#### 1.1 Постановка задачи

Требуется создать модель электрической шины, подводящей постоянный электрический ток большой величины к промышленному оборудованию (рис. 1.1). Электрическая шина изготовлена из меди, болты – из титанового сплава. Титановый сплав относится к сильнокоррозионной среде.



Рисунок 1.1 – Электрическая шина

Ток, протекающий по электрической шине от болта 1 к болтам 2а и 2б, нагревает ее (джоулевый нагрев). Так как электропроводность титана ниже, чем у меди, то плотность тока в болтах будет более высокой. Это приведет к нежелательной электрической нагрузке от болтов на шину.

Явление джоулевого нагрева описывается законами сохранения для электрического тока и энергии. После решения уравнений для этих двух законов получим поле температуры и электрическое поле соответственно. Все поверхности, кроме контактных поверхностей болтов, охлаждаются за счет естественной конвекции воздухом вокруг электрической шины.

Предположим, что выступающие части болта не влияют на охлаждение или нагрев устройства. Электрический потенциал на вертикальной поверхности верхнего болта составляет  $20\text{ мВ}$ , а потенциал на двух горизонтальных поверхностях нижних болтов равен  $0\text{ В}$ . В результате получается довольно высокая и потенциально небезопасная нагрузка на электрическую шину данного типа. Более сложные граничные условия для электродинамического анализа, в том числе возможность задавать полный ток через границу, доступны в модуле AC/DC.

*Задачи моделирования* – рассчитать тепловую деформацию элемента токоподводящей конструкции: найти плотность электрического тока и рассчитать нагрев медной шины; рассчитать механические деформации, вызванные термическим расширением.

Создадим геометрическую модель шины (рис. 1.2):

- длина шины – 9 см;
- ширина – 5 см;
- толщина – 5 мм;
- радиус токоподводов – 6 мм.

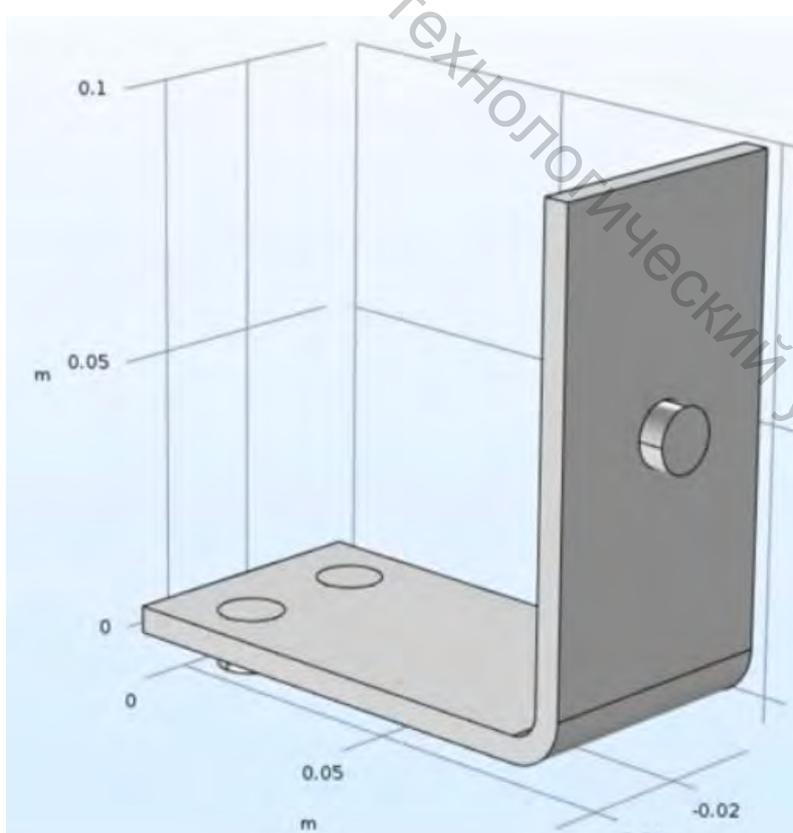


Рисунок 1.2 – Геометрическая модель медной шины

## 1.2 Создание новой модели

Рассмотрим последовательность действий при работе с пакетом COMSOL Multiphysics.

После запуска программы можно выбрать способ построения модели (рис. 1.3):

- Model Wizard – мастер построения модели, который позволяет за три предварительных шага выбрать основные базовые настройки будущей модели;
- Blank Model – шаблон пустой модели.

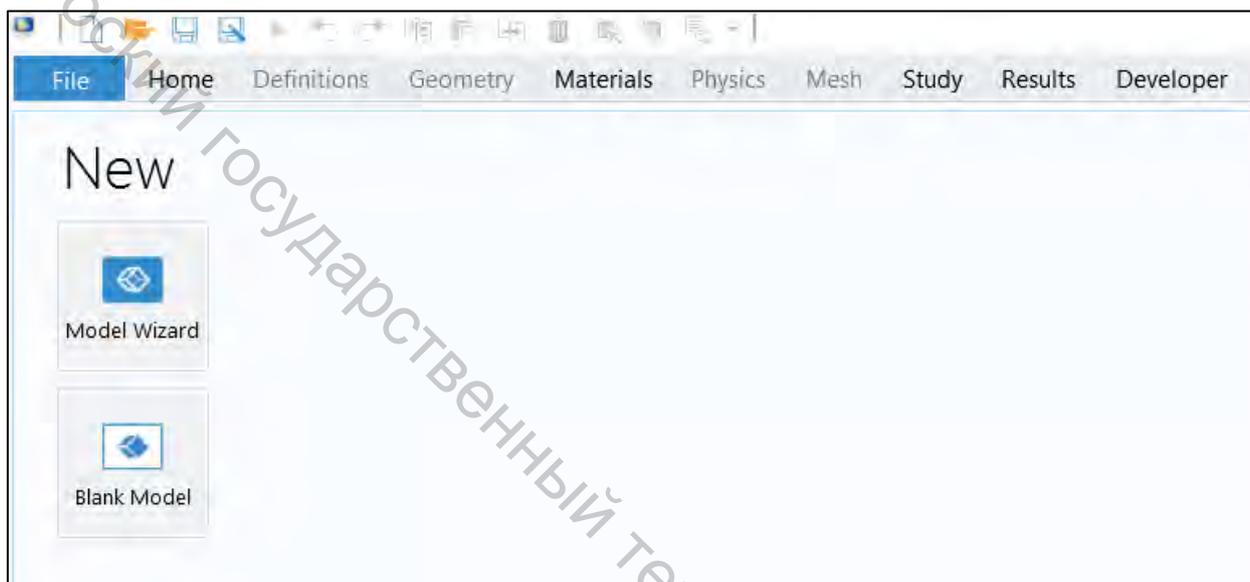


Рисунок 1.3 – Начало работы в COMSOL Multiphysics

**Шаг 1.** При создании модели с помощью мастера создания модели (Model Wizard) сначала необходимо выбрать размерность пространства. Возможные размерности: 3D, 2D, 1D, 0D. Выбираем 3D и нажимаем внизу Done (рис. 1.4).

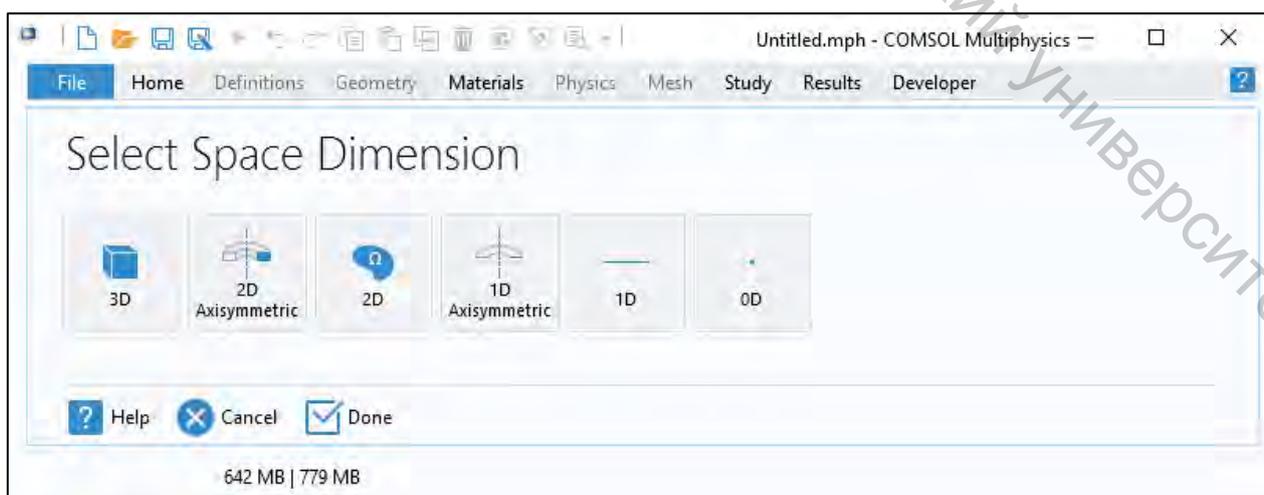


Рисунок 1.4 – Фрагмент окна программы на этапе выбора размерности

**Шаг 2.** Далее выбираем один или несколько физических интерфейсов. Поскольку мы будем решать механическую задачу, то открываем Structural Mechanics и выбираем подраздел Thermal-Structure Interaction. Находим подраздел Joule Heating and Thermal Expansion (рис. 1.5). Для добавления физического интерфейса следует нажать кнопку Add.

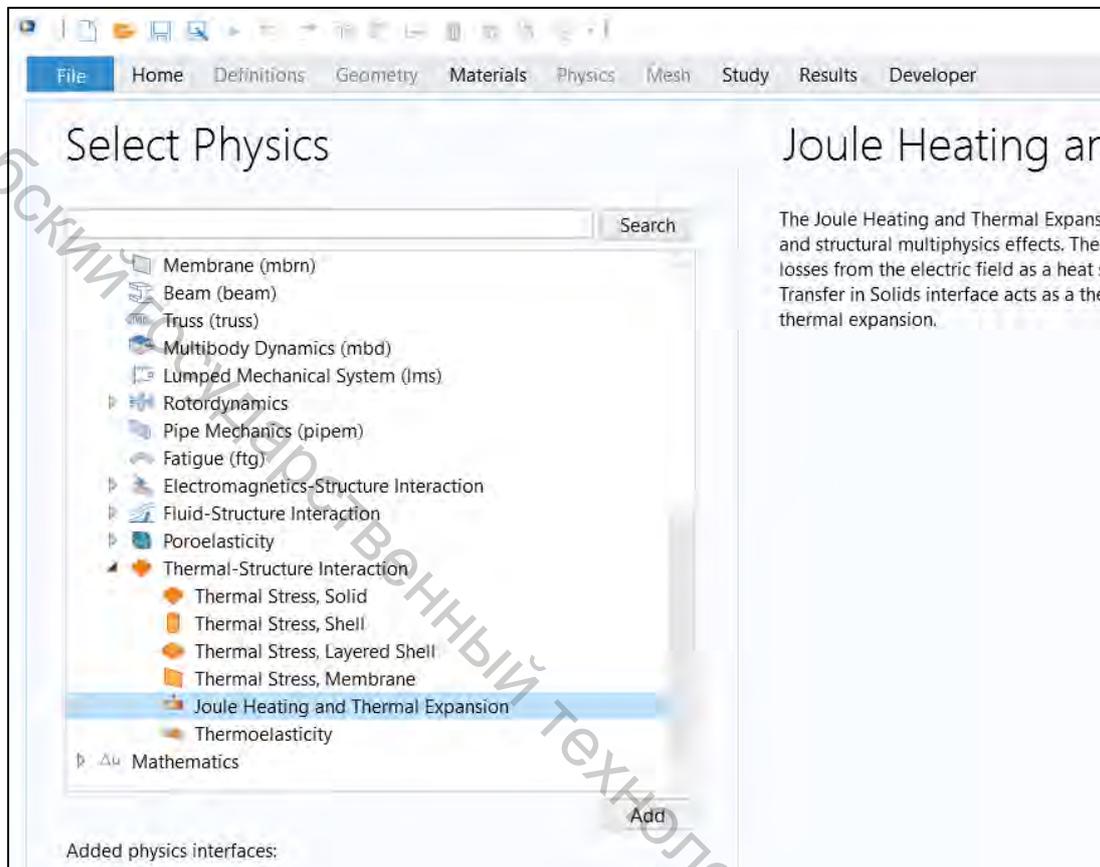


Рисунок 1.5 – Фрагмент окна программы на этапе Select Physics

В блоке Added physics interfaces появятся интерфейсы, которые будут использованы для решения этой задачи (рис. 1.6). Интерфейс в COMSOL – это некоторое графическое представление той математической модели, которая описывает нужный нам физический процесс.

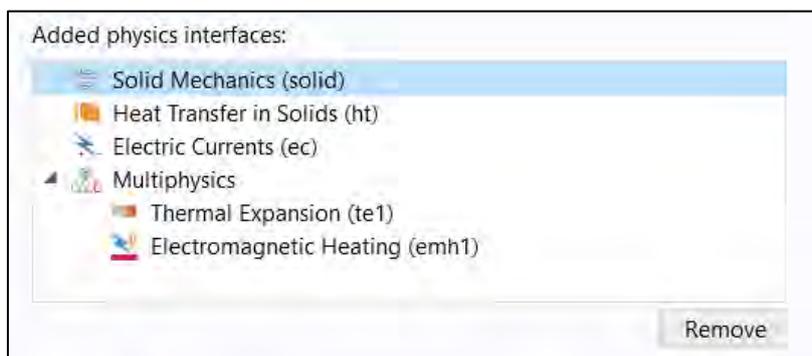


Рисунок 1.6 – Фрагмент окна программы на этапе Added physics interfaces

Имеются следующие интерфейсы: Electric Currents – служит для расчетов электрических токов, Heat Transfer in solids – для расчета поля температуры, Solid Mechanics – для расчета перемещений, механических деформаций. Эти процессы связаны между собой: протекание электрического тока приводит к нагреву. Нагрев приводит к механическим деформациям. Изменение температуры может так же влиять и на распределение токов (коэффициент электропроводности может зависеть от температуры). Связь между соответствующими уравнениями устанавливается в узле Multiphysics.

Нажимаем внизу кнопку Study.

**Шаг 3.** Выбираем тип исследования (рис. 1.7). Далее будем решать стационарную задачу, поэтому выбираем решатель Stationary.

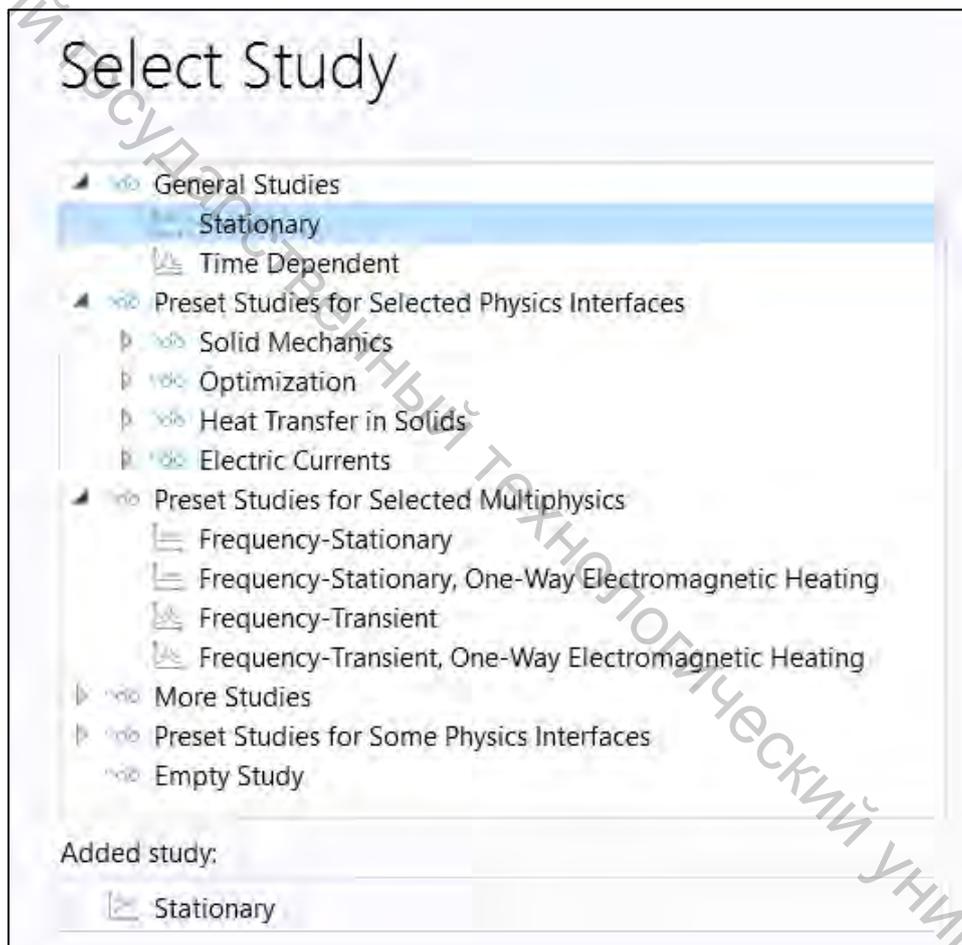


Рисунок 1.7 – Выбор типа решателя

Нажимаем кнопку Done.

Появляется основное окно для работы с COMSOL Multiphysics (рис. 1.8). Оно состоит из нескольких окон, которые можно изменять в размерах, двигать, закреплять и разъединять. По мере построения модели будут добавляться новые окна.

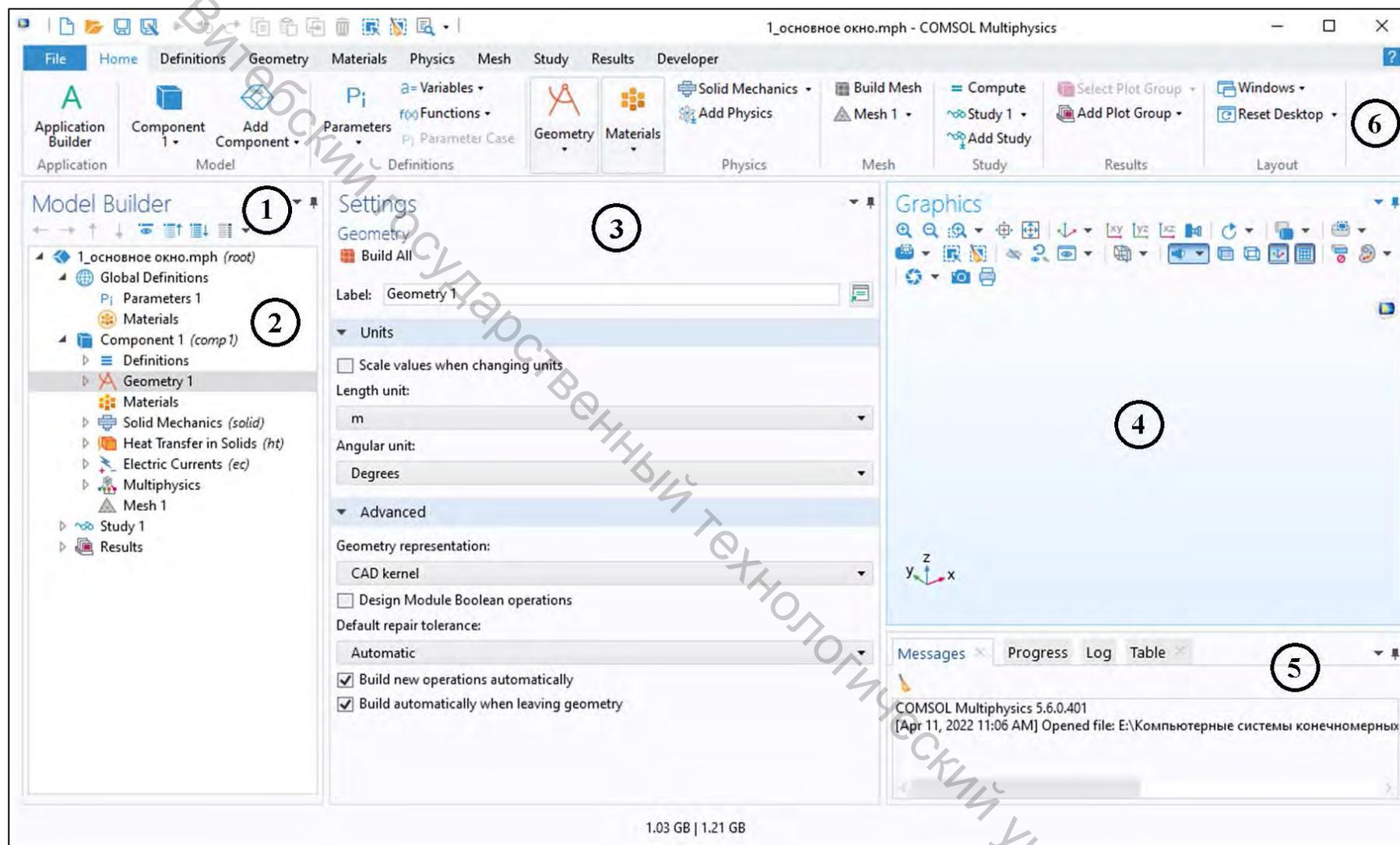


Рисунок 1.8 – Основное окно для работы в COMSOL Multiphysics

В левой части основного окна располагается панель построителя моделей Model Builder, которая включает в себя панель инструментов 1 и дерево модели 2, состоящее из перечня узлов.

Если пройтись по узлам сверху вниз, то можно увидеть все те шаги, которые необходимо осуществить для создания и решения модели, а именно необходимо задать геометрию модели, выбрать материалы, осуществить постановку задачи посредством выбора физических интерфейсов и граничных условий, осуществить дискретизацию задачи (построить сетку), выполнить решение, получить результаты и произвести их обработку. Настройка дерева моделей обычно производится сверху вниз.

В центральной части основного окна расположена панель настроек Settings 3. Она является характерной для тех узлов, которые выбираются в дереве модели. На этой панели задаются основные характеристики модели, включая размерность геометрии, свойства материалов, граничные и начальные условия, а также другая информация, которая может понадобиться для расчета модели.

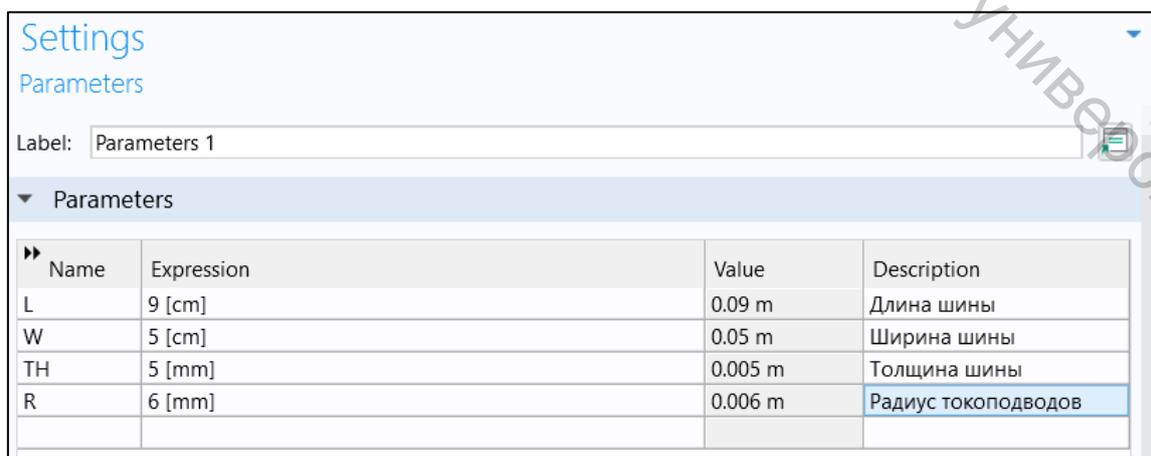
Правее располагается панель для вывода графических данных Graphics 4. На ней доступны операции поворота, масштабирования и выделения.

В нижней части основного окна располагаются информационные панели 5. В них отражаются сведения о работе, о длительности и ходе решения, статистике сеток, журнал решателя и таблицы результатов.

Лента Ribbon 6 расположена в верхней части основного окна. Ее вкладки содержат процессы моделирования и соответствующие функциональные элементы на каждом этапе. Большинство возможностей ленты также доступны из контекстных меню по правому щелчку в высших узлах дерева модели 2.

### 1.3 Геометрия модели

Узел Global Definitions позволяет задать глобальные параметры, переменные, функции, которые можно использовать в любых местах для расчета модели. Зададим геометрические размеры в виде параметров (рис. 1.9).



Name	Expression	Value	Description
L	9 [cm]	0.09 m	Длина шины
W	5 [cm]	0.05 m	Ширина шины
TH	5 [mm]	0.005 m	Толщина шины
R	6 [mm]	0.006 m	Радиус токопроводов

Рисунок 1.9 – Задание глобальных параметров

Данные именованные параметры можно использовать при построении геометрии. Вся работа с геометрическими моделями осуществляется в узле Geometry. Выбираем вкладку Geometry и в разделе настройки самого узла Settings выбираем базовую единицу измерения линейных размеров модели (рис. 1.10).

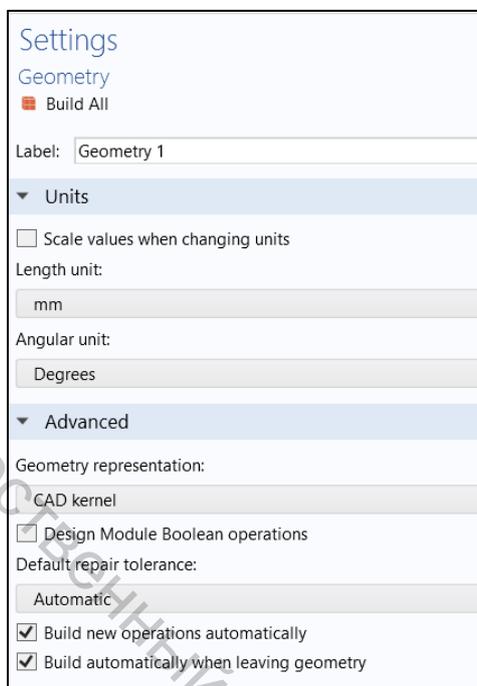


Рисунок 1.10 – Настройки узла Geometry

Построение геометрической фигуры объекта осуществляется с помощью выполнения операций, которые расположены на панели инструментов Geometry (рис. 1.11).

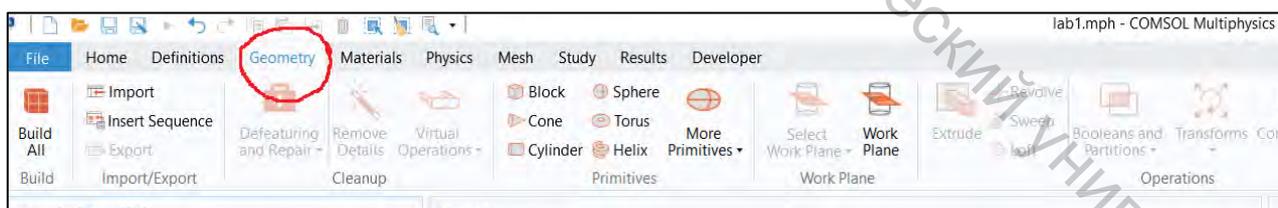


Рисунок 1.11 – Геометрические операции узла Geometry

Далее строим модель шины. Существует бесконечное число способов построения. Рассмотрим один из них.

1. В узле Geometry добавляем инструмент Work Plane (рабочая поверхность). Это инструмент, позволяющий рисовать двухмерные фигуры в трехмерном пространстве. В Settings для Plane выберем xz-plane.

2. На этой плоскости добавим квадрат со стороной (Settings → Side length) L и нажмем Build Selected.

3. Добавим еще квадрат со стороной (Side length)  $L = TH$ , который будет сдвинут по высоте на  $Position_{uw} = TH$ , и нажмем Build Selected.

4. Оставим разность между этими квадратами. Для этого в разделе Booleans and Partitions находим Difference (рис. 1.12).

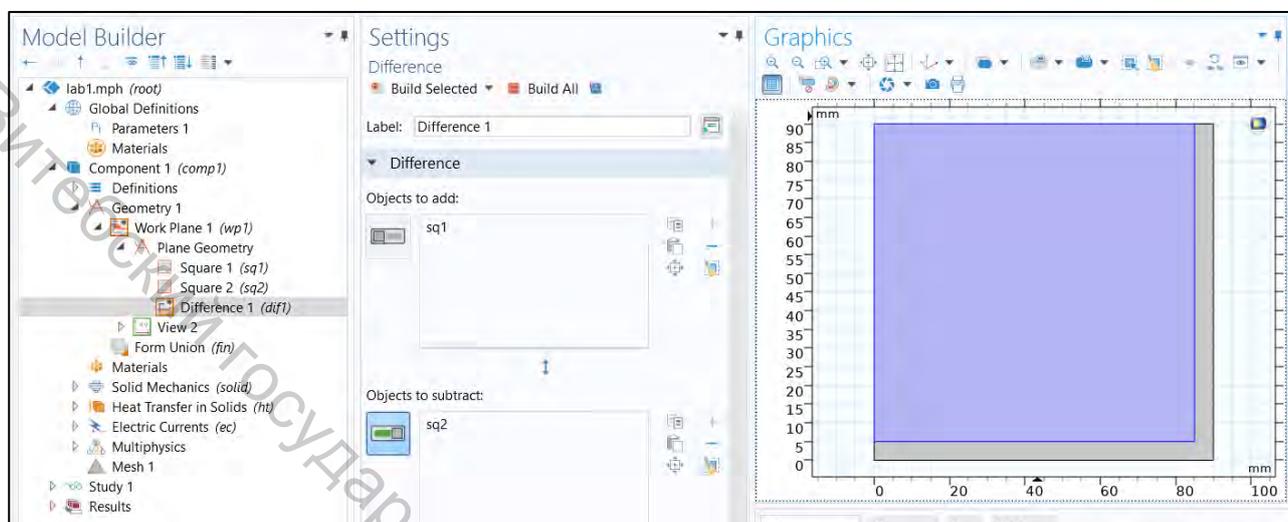


Рисунок 1.12 – Выполнение операции Difference

После нажатия Build Selected остается результат выполнения этой операции.

5. Чтобы скруглить углы на стыке двух прямоугольников, выбираем команду Fillet. Для этого выделяем внутренний узел и на панели Settings выполняем действия: Radius → записываем  $TH$  → нажимаем Build Selected. Для внешнего узла: Fillet → выбираем узел → Radius =  $2*TH$  → нажимаем Build Selected. Получаем поперечное сечение шины (рис.1.13).

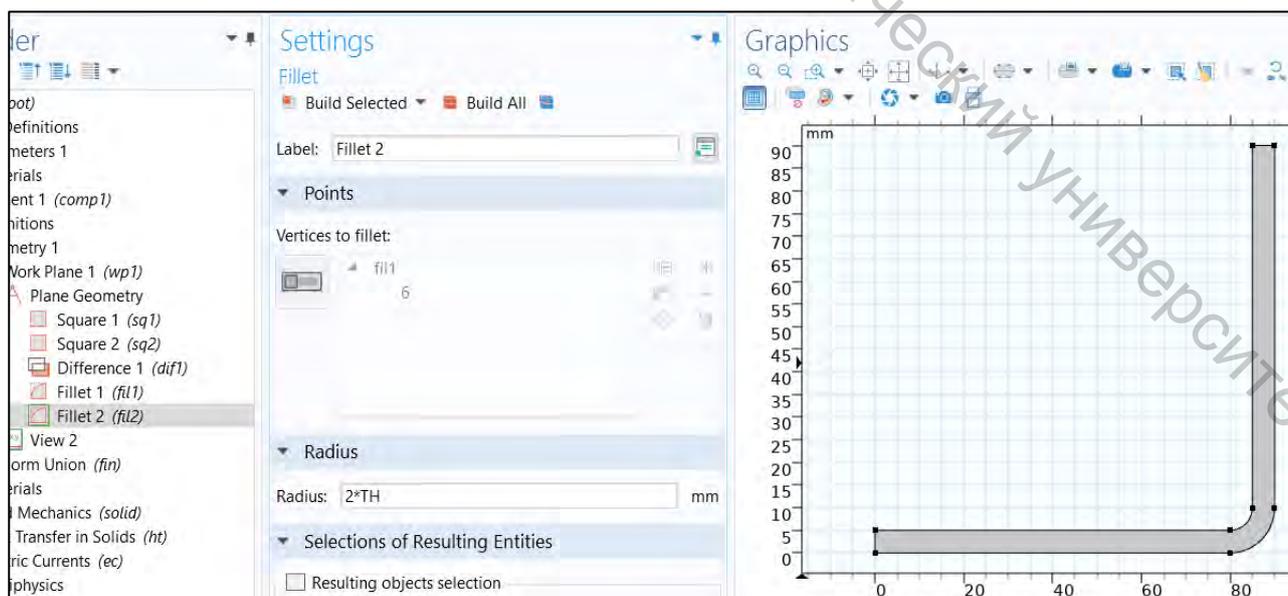


Рисунок 1.13 – Поперечное сечение шины

6. Добавим еще один квадрат. На панели Settings прописываем: Side length =  $2 * TH$ ; Position  $xw = L - 2 * TH$ . Нажимаем Build Selected. Этот квадрат содержит изогнутую часть пластины. Далее воспользуемся командой Booleans and Partitions → Partition Objects для того, чтобы разбить пластину на три части. Нажимаем Build Selected (рис. 1.14).

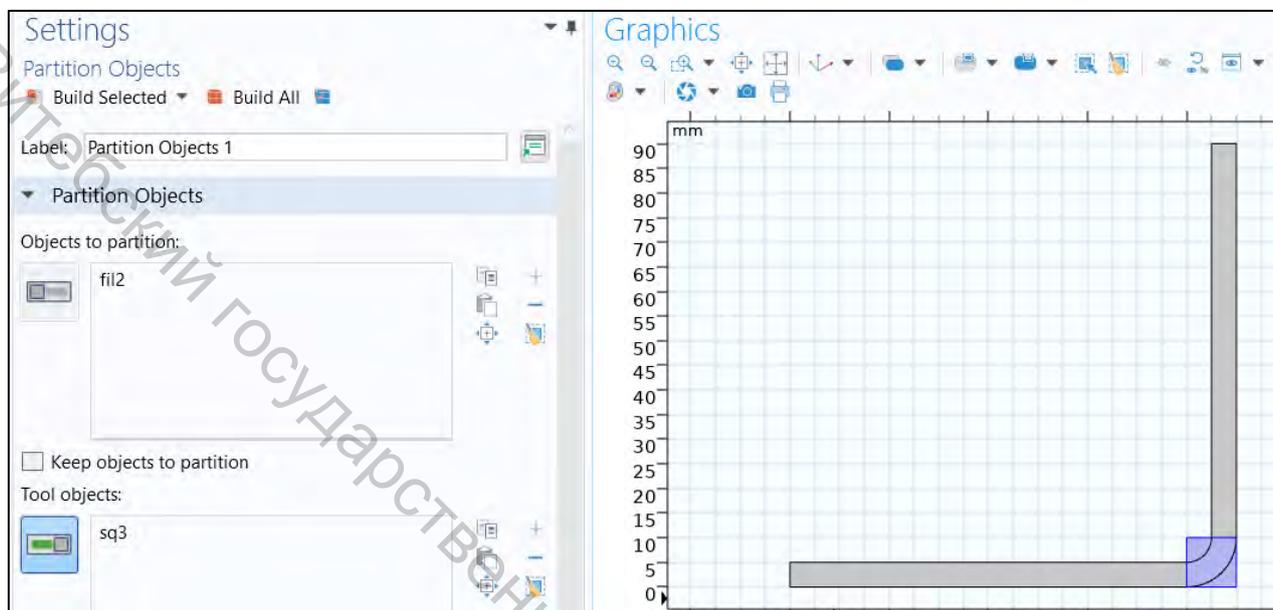


Рисунок 1.14 – Разбивка сечения на три части

7. Для превращения двухмерного объекта в трехмерный, воспользуемся вкладкой Extrude (правой мышью на Work Plane). Вытягивать будем на ширину пластины. Для этого на панели Settings записываем Distances(mm) =  $W$ , а также выделяем Reverse direction и нажимаем Build Selected. Получаем трехмерный объект.

8. Далее изображаем токоподводы. Для этого в дереве модели выбираем Geometry (правой мышью) и добавляем Work Plane. На панели Settings выполняем: Plane type → Face parallel (выберем вертикальную пластину). Затем переходим в Model Builder → Plane Geometry → рисуем круг радиуса  $R$  и нажимаем Build Selected. Круг будет построен в центре соответствующей грани. Для Work Plane используем команду Extrude (Distances (mm) =  $2 * TH$ , выбираем Reverse direction). Получаем цилиндр (рис. 1.15).

Токоподводы также можно построить при помощи команды Cylinder. Устанавливаем: Radius =  $R$ , Height =  $2 * TH$ , Position  $x = L/4$ ,  $y = W/4$ ,  $z = -TH$ . После ввода данных параметров появляется один из цилиндров.

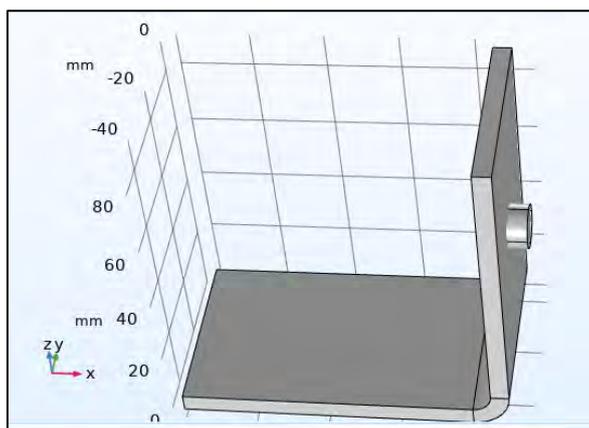


Рисунок 1.15 – Построение токоподвода

Кроме этого, мы можем скопировать объект и сдвинуть его на некоторое расстояние. Для этого на вкладке Geometry переходим в Transforms и выбираем Copy. Выбираем объект, который нужно скопировать (нижний цилиндр), и задаем смещение по оси  $ou$  на величину  $W/2$  (рис. 1.16).

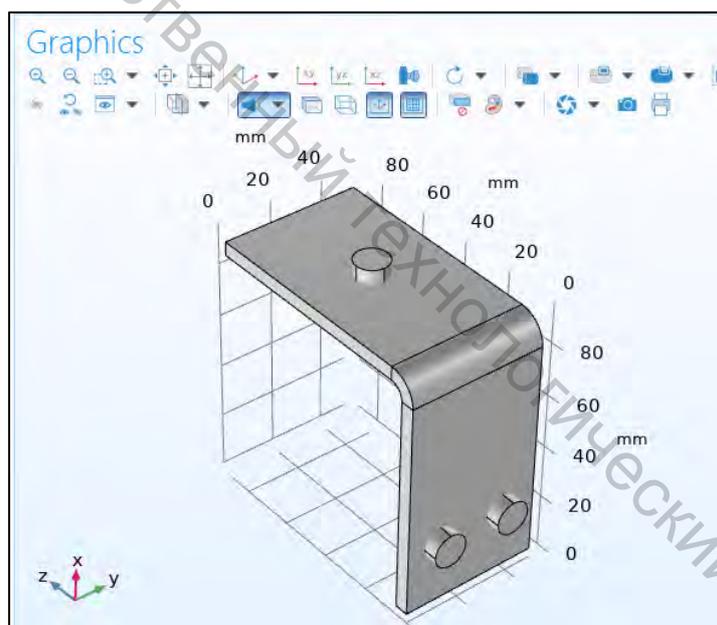


Рисунок 1.16 – Геометрическая модель электрической шины

## 1.4 Свойства материалов

Зададим свойства материалов нашего объекта. В узле Materials (материалы) хранятся свойства материалов для всех физических интерфейсов и геометрических областей узла Component (компонент). Электрическая шина изготовлена из меди, а токоподводы – из титанового сплава, окружающая среда – воздух. Оба материала доступны во встроенной библиотеке материалов. Будем учитывать зависимость коэффициента электропроводности металлов от температуры.

В построителе моделей (Model Builder) нажимаем правой кнопкой мыши Materials и выбираем Add Material from Library. По умолчанию окно открывается в правой части рабочего стола. Будем использовать встроенную библиотеку Built-in, которая относится к базовой платформе. Находим Titanium beta-215 и дважды щелкаем по материалу. Узел Titanium beta-215 добавится в построитель моделей. При этом первый материал связывается со всеми элементами расчетной области. Находим медь (Copper), нажимаем Add to Component. Для второго материала необходимо указать, к каким геометрическим объектам этот материал относится. Для этого на панели Graphics выделяем нужные нам объекты левой мышью. Они отображаются на панели Settings.

В узле Materials (материалы) щелкаем по Copper (медь). На панели Settings узла Material изучаем раздел Material Contents (содержимое материала). Просмотрим раздел Material Contents для титана. Раздел Material Contents включает в себя полезную информацию о свойствах материалов, используемых в модели.

Теперь учтем зависимость коэффициента электропроводности от температуры. Для этого воспользуемся справочными данными. Раскрываем узел Titanium beta-215 и переходим в узел Basic, в котором сгруппированы основные свойства этого материала. Переходим в Settings для этого узла в раздел Model Inputs, позволяющий указать, от каких физических параметров мы хотим описать зависимость свойств. Нажимаем «+» и переходим в Physical Quantity. Выбираем General → Temperature. Далее в Model Builder → Titanium → нажимаем правой кнопкой мыши Basic → выбираем Function-Interpolation, поскольку данные представлены в виде таблицы. Таблицу можно заполнять вручную или загрузить из текстового файла (приложение А) (рис. 1.17).

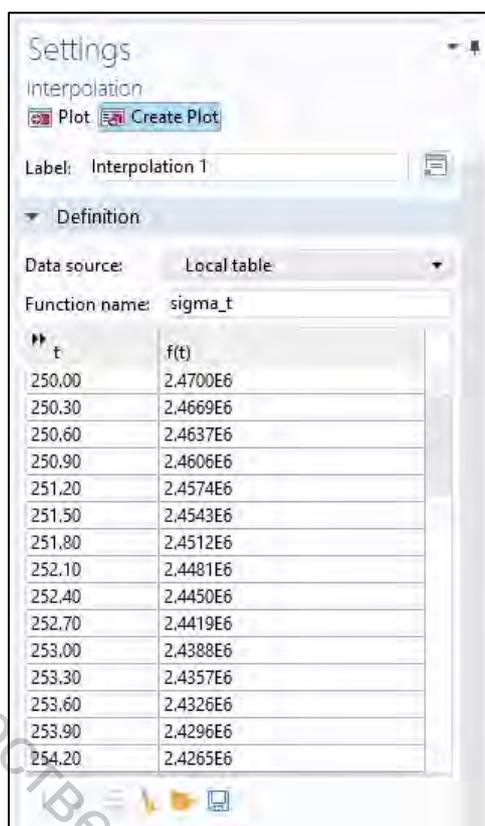


Рисунок 1.17 – Свойства титана

В первом столбце приводится значение температуры (К), во втором – коэффициент электропроводности (См/м). Следует отметить, что Сименс (русское обозначение – См; международное обозначение – S) – единица измерения электрической проводимости в Международной системе единиц (СИ), величина, обратная Ому.

Зададим имя функции Function name: sigma\_t. Укажем единицы измерения в разделе Units: Arguments: K; Functoin: S/m.

В Model Builder выбираем Titanium → Basic. Переходим на панель Settings и заменяем число, стоящее в строке electrical conductivity, на sigma\_t(T) (рис. 1.18).

Output Properties					
Property	Variable	Expression	Unit	Size	Info
Relative permeability	mur_iso ;...	1	1	3x3	
Electrical conductivity	sigma_iso...	sigma_t(T)	S/m	3x3	
Coefficient of thermal...	alpha_iso...	7.06e-6[1/K]	1/K	3x3	
Heat capacity at const...	Cp	710[J/(kg*K)]	J/(kg*K)	1x1	
Relative permittivity	epsilon_r_i...	1	1	3x3	
Density	rho	4940[kg/m^3]	kg/m^3	1x1	
Thermal conductivity	k_iso ; kii...	7.5[W/(m*K)]	W/(m*K)	3x3	

Рисунок 1.18 – Задание зависимости электропроводности от температуры

Аналогично добавляем зависимость электропроводности меди от температуры (приложение Б).

Таким образом, не только загружены основные свойства из библиотеки, но и добавлена зависимость коэффициента электропроводности от температуры в виде интерполяционной функции.

## 1.5 Физический интерфейс и граничные условия

В данной задаче решается три уравнения. При этом нужно настраивать мультифизические связи. Граничные условия:

- напряжение на шине – 50 мВ;
- температура воздуха – 20 °С;
- коэффициент теплоотдачи – 20 Вт/м<sup>2</sup>К;
- токоподводы жестко заземлены.

В Model Builder выбираем Parameters → Settings и добавляем параметры, которые будем использовать для задания граничных условий (рис. 1.19).

Name	Expression	Value	Description
L	9 [cm]	0.09 m	Длина шины
W	5 [cm]	0.05 m	Ширина шины
TH	5 [mm]	0.005 m	Толщина шины
R	6 [mm]	0.006 m	Радиус токоподводов
T0	20[degC]	293.15 K	Температура воздуха
htc	20[W/m^2/K]	20 W/(m <sup>2</sup> ·K)	Теплоотдача
V0	50[mV]	0.05 V	Напряжение

Рисунок 1.19 – Параметры модели

Переходим к физическим интерфейсам и настраиваем все необходимые параметры. В мультифизический интерфейс для данной задачи входят узлы: Solid Mechanics, Heat Transfer in Solids, Electric Currents. Решаемые в интерфейсе уравнения отображаются в разделе Equation панели Settings для соответствующих узлов физических интерфейсов.

Для каждого интерфейса есть набор узлов, добавленных по умолчанию.

Для Solid Mechanics: Linear Elastic Material – узел, отвечающий за материальную модель (по умолчанию используется модель упругого линейного материала); Free – граничные условия по умолчанию (отсутствие нагрузок); Initial value – начальное значение (для нестационарных задач задается значение переменной в начальный момент времени, для стационарных – величина Initial value играет роль начального приближения к искомому решению). Заметим, что узел Initial value имеется во всех интерфейсах.

В Heat Transfer in Solids: Solid – также задает материальную модель (коэффициент теплопроводности, плотность и др.); Thermal Insulation – теплоизоляция, отсутствие теплового потока на внешних границах.

В Electric Currents: Current Conservation – задается коэффициент электропроводности и диэлектрическая проницаемость; Electric Insulation – отсутствие электрических токов на внешних поверхностях, направленных по нормали к этой поверхности.

Если решать задачу с заданными начальными условиями, то получим тривиальное решение (получим нулевые перемещения, нулевые градиенты температур). Поэтому переопределим некоторые условия. Для этого можно использовать вкладку Physics на панели инструментов.

Для узла Solid Mechanics по условию задачи токоподводы жестко заземлены. В Model Builder выбираем Solid Mechanics и правой клавишей мыши переходим на граничные условия, которые называются Fixed Constraint. На панели Settings выбираем сечения токоподводов.

В интерфейсе Heat Transfer зададим теплообмен с окружающей средой. В COMSOL есть специальный узел Component → Definitions. Правой кнопкой мыши выбираем Shared Properties → Ambient Properties. Для  $T_{amb}$  задаем температуру  $T_0$ . Возвращаемся в интерфейс Heat Transfer in Solids и описываем теплообмен с окружающей средой. Для этого выбираем правой кнопкой мыши Heat Flux. Переключаемся на Convective heat flux и задаем коэффициент теплоотдачи Heat transfer coefficient  $h = h_{tc}$ ,  $T_{ext} = \text{Ambient temperature (ampr1)}$ . Выбираем границы, на которые это условие действует. Это все поверхности за исключением сечений токоподводов.

В интерфейсе Electric Currents граничные условия задаются в виде напряжения, приложенного к шине. Для этого правой кнопкой мыши выбираем узел Ground, задающий нулевой потенциал в двух сечениях (рис. 1.20).

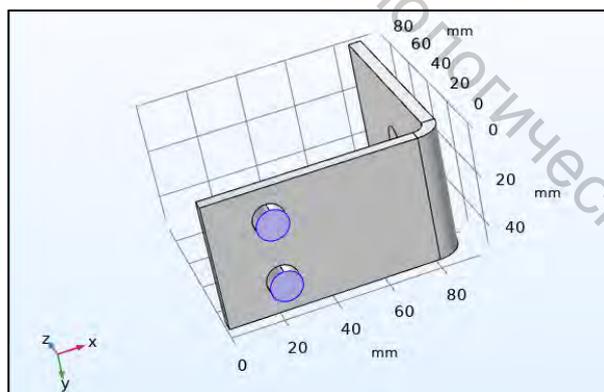


Рисунок 1.20 – Выбор нулевого потенциала

Также добавляем еще одно граничное условие в Electric Potential. Выбираем оставшееся сечение и задаем  $V_0 = V_0$ .

Перейдем к узлу мультифизических связей Multiphysics. Они предназначены для решения системы связанных уравнений. Узел Thermal Expansion связывает интерфейсы Heat Transfer in Solid и Solid Mechanics. Это позволяет рассчитать термическое расширение, обусловленное нагревом. Можно убрать галочку с Thermoelastic damping. Узел Electromagnetic Heating

связывает интерфейсы Electric Currents и Heat Transfer in Solid. Например, когда электропроводность зависит от температуры.

## 1.6 Mesh (Сетка)

Для решения данной задачи используется метод конечных элементов. Поэтому важным этапом при решении является дискретизация пространства. Это разбиение нашей непрерывной геометрической фигуры объекта на конечные элементы. COMSOL позволяет строить сетку как в автоматическом, так и в ручном режимах. Для работы с сетками предназначен узел Mesh. Воспользуемся автоматическим сеточным генератором. Кликая по узлу Mesh, автоматически выбирается Physics → controlled mesh, т. е. строится сетка на основе физических интерфейсов. Параметр, который можно менять, – размер сетки (Element size). Нажимаем Build All и сетка сгенерирована в виде тетраэдров.

## 1.7 Исследование, визуализация и обработка

Следующий этап – решение задачи, причем:

- в одной модели можно использовать несколько типов решателей;
- каждый решатель можно настроить независимо;
- для большинства задач можно использовать автоматические настройки решателей.

В COMSOL доступен параметрический анализ: Parametric Sweep – перебор параметров; Material sweep – перебор материалов; Function sweep – перебор функций.

Проведем следующие типы исследования:

- стационарное исследование (искомые характеристики процесса не зависят от времени);
- параметрическое исследование – решение задачи для нескольких значений напряжения на шине (от 0,01 до 0,05 В с шагом 0,005 В);
- переходной процесс – охлаждение шины при отключении тока (температура шины изменяется во времени).

После создания сетки автоматически подключается стационарный решатель (Stationary) и на вкладке Study панели инструментов становится доступна кнопка Compute. Для нашей задачи мы получим графики, которые построены по умолчанию. Переходим в Model Builder → Results → Stress → Surface и получаем график эквивалентных напряжений по Мизесу (теория Mises-Hencky). Теория утверждает, что пластичный материал начинает повреждаться в местах, где напряжение по Мизесу становится равным предельному напряжению (рис. 1.21).

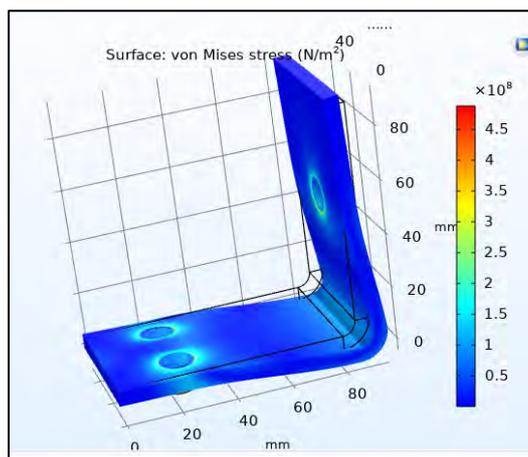


Рисунок 1.21 – График напряжений по Мизесу

На графике справа находится шкала, позволяющая сопоставить цвет и численное значение. Видна форма деформации. Для наглядности COMSOL автоматически увеличивает деформации. Причем величину увеличения на рисунке можно посмотреть в Model Builder → Results → Surface → Deformation. Значения выводятся в Н/м<sup>2</sup>, в Settings → Unit можно переключиться на МПа (МПа). В Stress (solid) → Color Legend поставим галочки в Show maximum and minimum values и Show units. Появится минимальное и максимальное значение и единицы измерений.

Также по умолчанию строится график температуры Temperature → Surface. По умолчанию график строится в Кельвинах (K), в Unit переключаемся на градусы Цельсия (degC). В Color Legend включаем Show maximum and minimum values и Show units. В силу достаточно низкой теплопроводности и электропроводности титана токоотводы будут перегреваться.

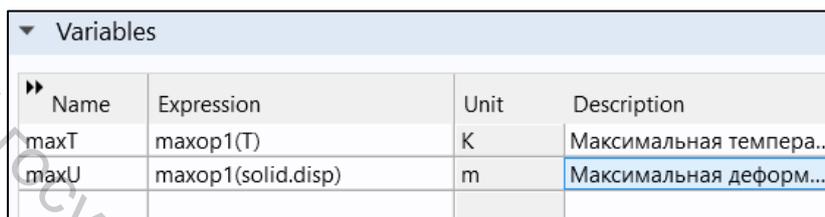
Мы можем построить свой график. Воспользуемся на панели инструментов вкладкой Results → 3d Plot Group. Назовем график: Settings → Label → Heat Source (мощность теплоты). На панели выбираем Definitions → Surface (распределение по поверхности). Переходим в Settings → Expression → нажимаем кнопку Replace Expression. Откроется список, сгруппированный по разделам. В поисковой строке набираем heat source и выбираем Total heat source. Из рисунка видно, что максимум тепловыделения приходится на титановый токоподвод. То есть график позволяет проанализировать, где больше всего выделяется теплоты.

Далее можно построить график электрического потенциала Electric Potential. Для этого на панели инструментов выбираем Results → 3d Plot Group. Назовем график: Settings → Label → Current Density. На панели выбираем Definitions → Surface. Переходим в Settings → Expression → нажимаем кнопку Replace Expression → Electric Current → Currents and charge → Current density norm (абсолютное значение плотности тока). Нажимаем Settings Plot. На панели Settings выбираем Range, для лучшей визуализации можно менять цветовой диапазон.

## Порядок выполнения работы

**Задание 1.** Выполните пункты 1.1–1.7.

**Задание 2.** Установить, как будет меняться максимальная температура и максимальная деформация при изменении напряжения от 20 до 60 мВ. Для этого в Model Builder в узле Definitions добавить Nonlocal Couplings → Maximum. Это позволит находить максимальное значение любой переменной в выделенной области. В Graphics выбрать всю область. Добавить узел Variables (рис. 1.22).



Name	Expression	Unit	Description
maxT	maxop1(T)	K	Максимальная темпера...
maxU	maxop1(solid.disp)	m	Максимальная деформ...

Рисунок 1.22 – Вид узла Variables

В настройках решателя открыть Study → Stationary и на панели Settings → Study Extensions → поставить галочку для Auxiliary sweep (вспомогательная развертка). Эта опция позволит настроить параметрическое исследование. Нажимаем кнопку Range и указываем диапазон: Start 20, Step 5, Stop 60, replace. Единицу измерения выставляем – mV. Запускаем решение.

После завершения расчетов посмотреть графики при разных напряжениях, выбирая значения в Settings → Parameter value. Для наглядности следует зафиксировать шкалу цветов. В Model Builder → Result → Stress → Surface выделить Manual color range. Выставить минимальное значение 0, а максимальное – 420. Перейдя на Deformation, выставить коэффициент Scalar factor – 50. Сделать вывод, как зависит деформация от напряжения.

**Задание 3.** Построить график зависимости максимальной температуры от напряжения.

В разделе Result выбрать Derived Values → Global Evaluation и вывести переменные, рассчитанные ранее maxT(degC). Нажав Evaluate, посмотреть рассчитанную таблицу. Нажать кнопку Table Graph.

Аналогично построить график зависимости максимальной деформации maxU (mm) от напряжения.

**Задание 4.** Найти напряжение, при котором максимальная деформация будет равна 0,1 мм.

Для этого добавить в Component → Definitions → Variables переменную goal, Expression =  $\text{abs}(\text{maxU} - 0.1[\text{mm}])$ , Description = Целевая функция.

В настройках решателя Study1 добавить Optimization. В Settings: Method = COBYLA; Objective Function (+) = comp1.goal; Control Variables and Parameters = V0, Initial value = 35[mV], Scale = 0.01, Lower bound = 20[mV], Upper bound = 60[mV]; Objective scaling = Manual; Scale =  $1e-5$ .

В Study → Step1: Stationary убрать галочку в Settings → Study Extensions → Auxiliary sweep. И запустить расчет. В таблицу будут выводиться значения параметра и целевой функции.

Для того, чтобы вывести искомую функцию, перейти в Derived Value → Global Evaluation и в Settings → Expression задать:  $V_0$  и  $\max U$ . Нажать Evaluate.

## ЛАБОРАТОРНАЯ РАБОТА 2. СОЗДАНИЕ ПРИЛОЖЕНИЯ В APPLICATION BUILDER

### Порядок выполнения работы

Создать приложение для расчета максимальных значений температуры и деформаций. Приложение должно иметь интерфейс, предназначенный для взаимодействия с моделью электрической шины и содержащий поля ввода для значений толщины шины, приложенного напряжения и температуры воздуха.

Результатами будут значение максимальной температуры, максимальной деформации и график механических напряжений распределения температуры.

1. В файле лабораторной работы 1 удалить: в Model Builder → Study1 → Optimization; в Solver Configurations → Parametric Solutions1 –  $V_0 = 0.0423$ ; в Job Configurations → Optimization 1 и Parametric Sweep1. Во вкладке Study1 → Solver Configuration правой кнопкой мыши выбрать Reset Solver to Default.

2. Перейти на панели инструментов в среду Application Builder, то есть в среду разработки приложений. Далее воспользуйтесь мастером New Form. Предусмотрено несколько шаблонов интерфейса. Выбрать, например, Toolbar, sections, and graphics (рис. 1.23). И нажать Content.

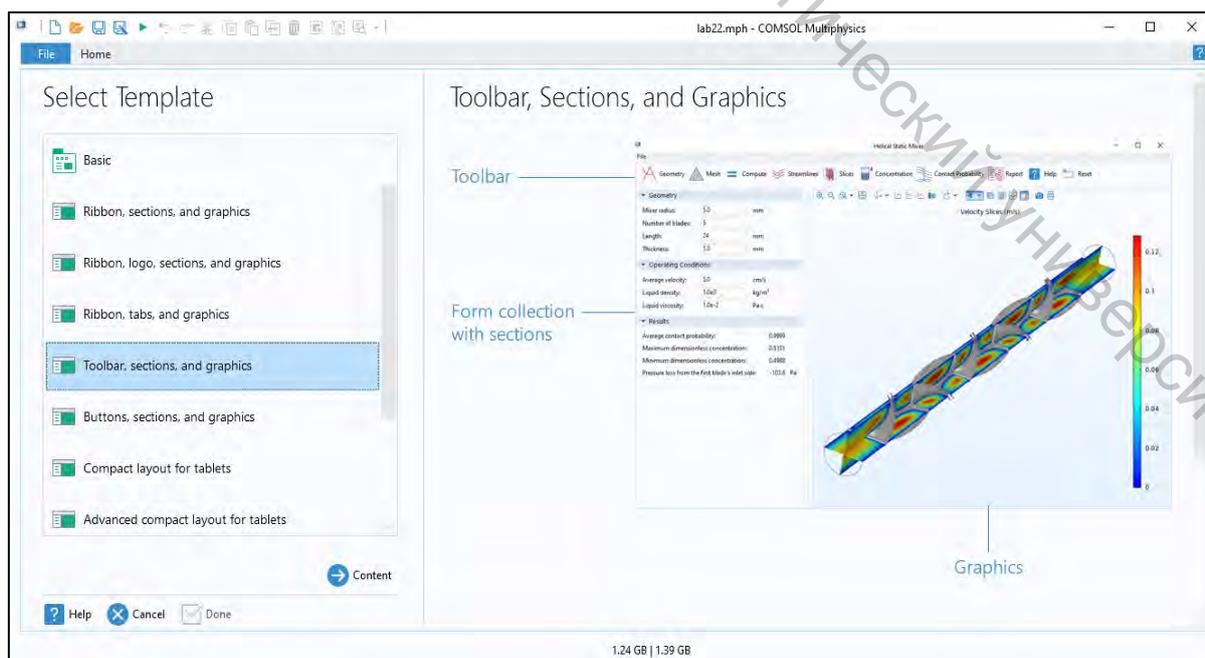


Рисунок 1.23 – Выбор шаблона интерфейса

3. В Selected добавить команды построения – Plot Geometry1, сетки – Plot Mesh и запуска расчета – Compute Study1.

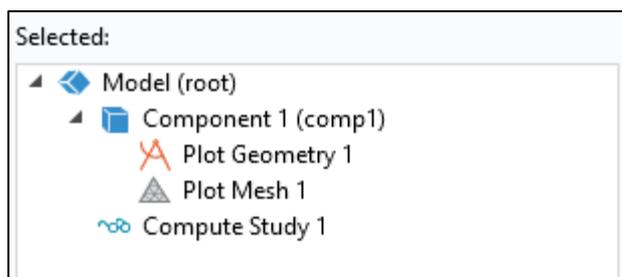


Рисунок 1.24 – Выбор команд построения

4. Перейти на вкладку Input/Output и выбрать Model (root) → Global Definitions → Parameters: задать толщину шины (ТН) и нажать на «←→». При этом ввести имя формы Title = Входные данные. Добавить также во входные данные напряжение (V0), температуру воздуха (T0).

5. Создать форму с именем Title = результаты. В нее вывести максимальную температуру Result → Derived Value → Global Evaluation1 и нажать на «←→», а также максимальную деформацию Global Evaluation2 и нажать на «←→». Нажать кнопку Done.

6. Настроить графическое окно. Кликнуть на него. Для вывода графика напряжений выделить на панели Setting в Model → Results → Stress (solid). Появится окно, представленное на рисунке 1.25.

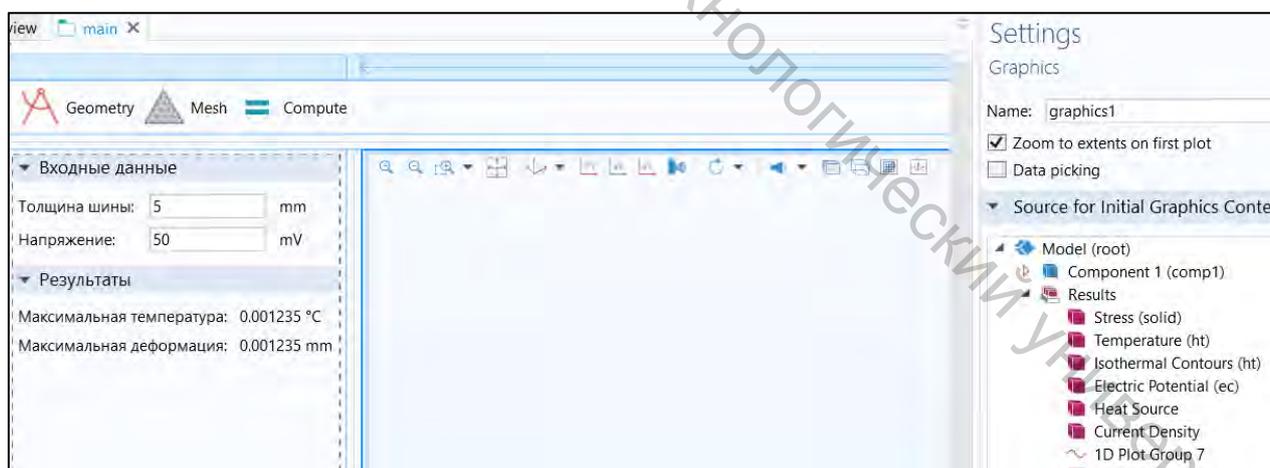


Рисунок 1.25 – Настройка графического окна

Чтобы график после завершения расчетов обновлялся, нажать в верхней части формы на Compute, в Toolbar Items выделить Compute и нажать Edit. Выбрать команды, которые должны выполняться при нажатии на кнопку. Для этого перейти во вкладку Choose commands to run и добавить команду Result → Stress, нажимая кнопку Plot (рис. 1.26–1.28).

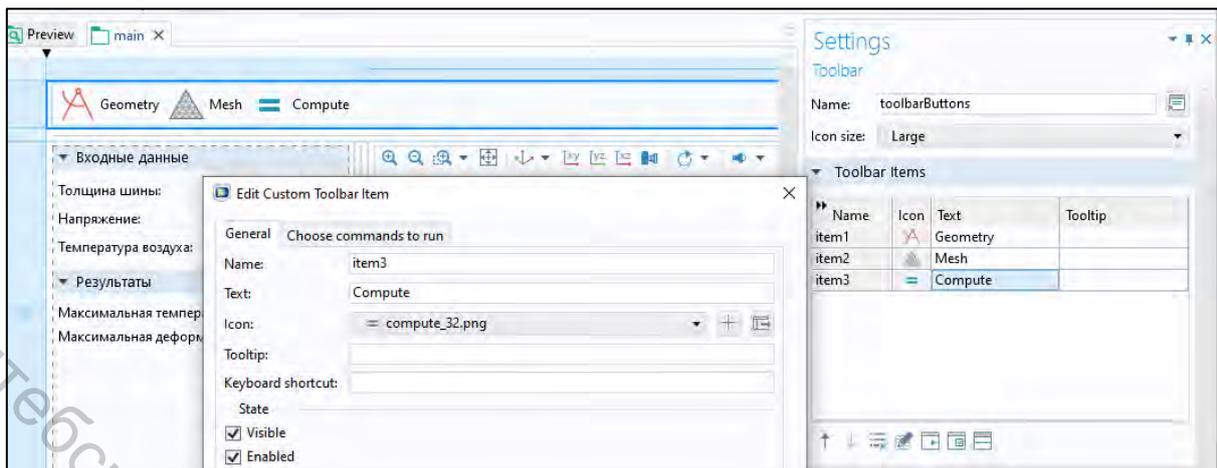


Рисунок 1.26 – Настройка графического окна для обновления (шаг 1)

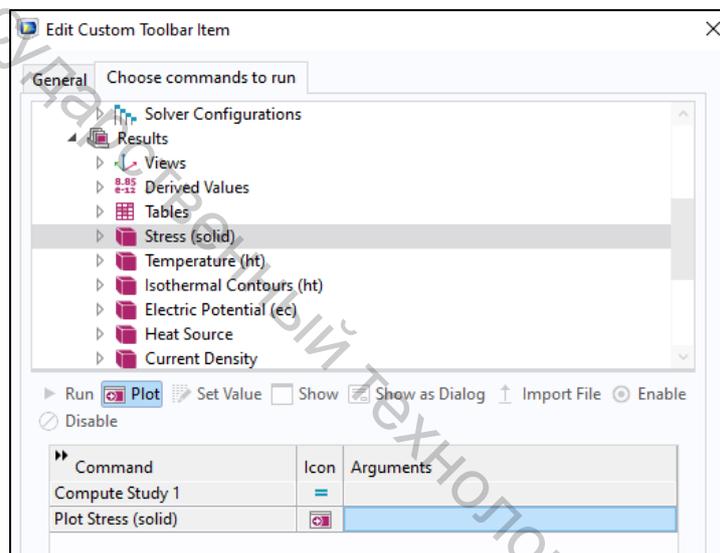


Рисунок 1.27 – Настройка графического окна для обновления (шаг 2)

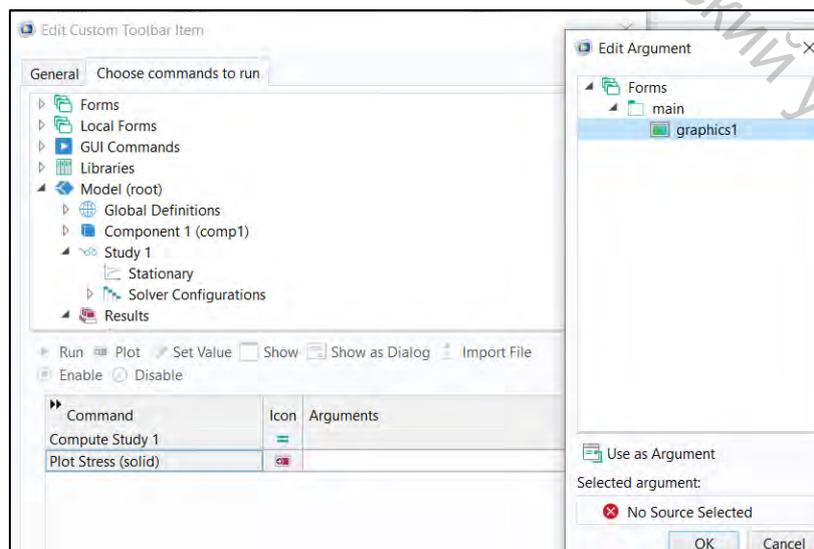


Рисунок 1.28 – Настройка графического окна для обновления (шаг 3)

Чтобы добавить Argument, перейти в Edit Argument → Form → main → graphics1 и нажать ОК.

7. Протестировать полученное приложение для разной толщины пластины и разных напряжений.

8. Создать исполняемый файл приложения. Для этого нажать кнопку Compiler, выбрать место сохранения и под какую платформу компилируется приложение.

## 2 РЕШЕНИЕ СИСТЕМ ЛИНЕЙНЫХ УРАВНЕНИЙ

### 2.1 Основные понятия

Определителем или детерминантом второго порядка называется разность:

$$\Delta = \begin{vmatrix} a_{11} & b_{12} \\ a_{21} & b_{22} \end{vmatrix} = a_{11}b_{22} - a_{21}b_{12}, \quad (2.1)$$

где  $a_{11}, b_{12}, a_{21}, b_{22}$  – элементы определителя.

Элементы  $a_{11}, b_{22}$  образуют главную диагональ определителя, а элементы  $a_{21}, b_{12}$  – побочную диагональ.

**Минором** определителя некоторого элемента

$$\Delta = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}$$

называется такой новый определитель, который получается из определителя  $\Delta$  вычёркиванием строки и столбца, проходящих через данный элемент. Например, минором определителя, соответствующим элементу  $a_{22}$ , будет

$$M_{22} = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = \begin{vmatrix} a_{11} & a_{13} \\ a_{31} & a_{33} \end{vmatrix}.$$

**Алгебраическим дополнением** некоторого элемента называется соответствующий его минор, взятый со знаком «+» или «-» в зависимости от того, будет ли сумма номеров строки и столбца, которым принадлежит элемент, чётным или нечётным числом.

Алгебраическое дополнение элемента  $a_{ij}$  обозначают  $A_{ij}$ . Таким образом,

$$A_{ij} = (-1)^{i+j} M_{ij}. \quad (2.2)$$

Определитель равен сумме произведений элементов какой-либо строки или столбца на их алгебраическое дополнение.

**Матрица** – прямоугольная таблица, образованная из элементов некоторого множества и состоящая из  $m$  строк и  $n$  столбцов. Матрицу  $A$  называют матрицей размера  $m \times n$  и пишут  $A_{m \times n}$ . Матрица, определитель которой равен нулю, называется вырожденной (или особенной), а матрица, определитель которой отличен от нуля, – невырожденной (или неособенной).

Для каждой обратимой матрицы существует только одна обратная матрица, которая определяется формулой

$$A^{-1} = \frac{1}{\Delta} B = \frac{1}{\Delta} \begin{pmatrix} A_{11} & A_{21} & \dots & A_{n1} \\ A_{12} & A_{22} & \dots & A_{n2} \\ \dots & \dots & \dots & \dots \\ A_{1n} & A_{2n} & \dots & A_{nn} \end{pmatrix}. \quad (2.3)$$

Матрица  $B$  называется присоединенной.

Системой  $m$  линейных уравнений с  $n$  неизвестными называют систему уравнений вида:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1; \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2; \\ \dots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = b_m. \end{cases}$$

## 2.2 Формулы Крамера

Для простоты будем рассматривать систему из трёх уравнений с тремя неизвестными:

$$\begin{cases} a_{11}x + a_{12}y + a_{13}z = b_1 \\ a_{21}x + a_{22}y + a_{23}z = b_2 \\ a_{31}x + a_{32}y + a_{33}z = b_3 \end{cases}. \quad (2.4)$$

Предположим, что  $\Delta \neq 0$ . Решение системы запишется в виде

$$x = \frac{\Delta_x}{\Delta}, \quad y = \frac{\Delta_y}{\Delta}, \quad z = \frac{\Delta_z}{\Delta},$$

где

$$\Delta_x = \begin{vmatrix} b_1 & a_{12} & a_{13} \\ b_2 & a_{22} & a_{23} \\ b_3 & a_{32} & a_{33} \end{vmatrix}, \Delta_y = \begin{vmatrix} a_{11} & b_1 & a_{13} \\ a_{21} & b_2 & a_{23} \\ a_{31} & b_3 & a_{33} \end{vmatrix}, \Delta_z = \begin{vmatrix} a_{11} & a_{12} & b_1 \\ a_{21} & a_{22} & b_2 \\ a_{31} & a_{32} & b_3 \end{vmatrix}.$$

Определители  $\Delta_x, \Delta_y, \Delta_z$  получаются из определителя  $\Delta$  при помощи замены соответственно его первого, второго и третьего столбца столбцом свободных членов системы.

### 2.3 Матричный метод решения системы линейных уравнений

Пусть дана система (2.4). Введем обозначения

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}, X = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}, \tilde{B} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}.$$

Тогда систему (2.4) в матричном виде можно записать:  $A \cdot X = \tilde{B}$ . Матрица-решение  $X$  находится как произведение  $A^{-1}$  и  $\tilde{B}$

$$X = A^{-1} \tilde{B}. \quad (2.5)$$

## ЛАБОРАТОРНАЯ РАБОТА 3. СОЗДАНИЕ ПРИЛОЖЕНИЯ РЕШЕНИЯ СИСТЕМ ЛИНЕЙНЫХ УРАВНЕНИЙ

### *Краткие теоретические сведения*

Рассмотрим пример создания приложения решения системы уравнений матричным методом на языке программирования C#. Основные компоненты программы:

comboBox1 – элемент управления позволяет пользователю выбирать порядок системы уравнений;

dataGridView1 – элемент управления для отображения данных в табличном формате матрицы коэффициентов;

dataGridView2 – для отображения обратной матрицы;

dataGridViewB – для отображения правой части системы уравнений;

dataGridViewResult – для отображения матрицы решения системы уравнений матричным методом;

label1 – 4 – отображение текста для пояснения;

button1 – кнопка для вычисления обратной матрицы;

button2 – кнопка для вычисления определителя;

button3 – кнопка для очистки формы;

result – кнопка для отображения результата.

Порядок определителя 3

groupBox2 Введите матрицу

	1	2	3
	3	3	1
▶	1	6	7
*			

Вычислить определитель 20

groupBox3 Введите правую часть

	3
	4
▶	5
*	

groupBox1 обратная матрица

▶	0,75	0,2	-0,35
	-1	0,2	0,4
*	0,75	-0,2	-0,15

groupBox3 Решить СУ

▶	1,3
	-0,2
*	0,7

ОЧИСТИТЬ

Рисунок 2.1 – Настройка графического окна для обновления

Исходный код программы:

```
public partial class Form1 : Form
{
    public double[,] matrix;
    public double[,] inversematrix;
    public double[,] InverseMatrix;
    public double[,] submatrix;
    private int razmer;
    private double determinant;
    public Form1()
    {
        InitializeComponent();
    }
    private void comboBox1_SelectedIndexChanged(object sender,
EventArgs e)
    {
        razmer =
Convert.ToInt32((comboBox1.SelectedItem.ToString()));
        dataGridView1.RowCount = razmer;
        dataGridView1.ColumnCount = razmer;
        dataGridView2.RowCount = razmer;
        dataGridView2.ColumnCount = razmer;
        dataGridViewResult.RowCount = razmer;
    }
}
```

```

dataGridViewB.RowCount = razmer;
for (int k = 0; k < razmer; k++)
{
    dataGridView1.Columns[k].Width = 50;
    dataGridView2.Columns[k].Width = 50;
}
}
private void Form1_Load(object sender, EventArgs e)
{
    comboBox1.SelectedIndex = 0;
    dataGridView1.RowCount = 2;
    dataGridView1.ColumnCount = 2;
    dataGridView2.RowCount = 2;
    dataGridView2.ColumnCount = 2;
    dataGridViewB.RowCount = 2;
    dataGridViewB.ColumnCount = 1;
    dataGridViewResult.RowCount = 2;
    dataGridViewResult.ColumnCount = 1;
    for (int k = 0; k < razmer; k++)
        dataGridView1.Columns[k].Width = 50;
}
private static double DetRec(double[,] matrix)
{
    if (matrix.Length == 4)
    {
        return matrix[0, 0] * matrix[1, 1] - matrix[0, 1] *
matrix[1, 0];
    }
    double sign = 1, result = 0;
    for (int i = 0; i < matrix.GetLength(1); i++)
    {
        double[,] minor = GetMinor(matrix, i);
        result += sign * matrix[0, i] * DetRec(minor);
        sign = -sign;
    }
    return result;
}
private static double[,] GetMinor(double[,] matrix, int n)
{
    double[,] result = new double[matrix.GetLength(0) - 1,
matrix.GetLength(0) - 1];
    for (int i = 1; i < matrix.GetLength(0); i++)
    {
        for (int j = 0, col = 0; j < matrix.GetLength(1); j++)
        {
            if (j == n)
                continue;
            result[i - 1, col] = matrix[i, j];
            col++;
        }
    }
}
}

```

```

        return result;
    }
    private void button2_Click(object sender, EventArgs e)
    {
        matrix = new double[razmer, razmer];
        for (int i = 0; i < razmer; i++)
            for (int j = 0; j < razmer; j++)
                matrix[i, j] =
                    Convert.ToDouble(dataGridView1.Rows[i].Cells[j].Value);
        determinant = DetRec(matrix);
        label4.Text = determinant.ToString();
    }
    private void button3_Click(object sender, EventArgs e)
    {
        //очистка
        for (int i = 0; i < razmer; i++)
            for (int j = 0; j < razmer; j++)
                {
                    dataGridView1.Rows[i].Cells[j].Value = string.Empty;
                    dataGridView2.Rows[i].Cells[j].Value = string.Empty;
                }
        for (int i = 0; i < razmer; i++)
        {
            dataGridViewB.Rows[i].Cells[0].Value = string.Empty;
            dataGridViewResult.Rows[i].Cells[0].Value = string.Empty;
        }
        //или
        //dataGridView1.Rows.Clear();
        //dataGridView1.Columns.Clear();
        //dataGridView2.Rows.Clear();
        //dataGridView2.Columns.Clear();
        label4.Text = " ";
    }
    public double[,] transpose(double[,] matrix, int N)
    {
        double t;
        for (int i = 0; i < N; i++)
        {
            for (int j = i + 1; j < N; j++)
            {
                t = matrix[i, j];
                matrix[i, j] = matrix[j, i];
                matrix[j, i] = t;
            }
        }
        return matrix;
    }
    public double[,] inverse(double[,] matr, int n)
    {
        if (n == 2)
        {

```

```

        double a = matrix[0, 0], b = matrix[0, 1];
        double c = matrix[1, 0], d = matrix[1, 1];
        double den = a * d - b * c;
        if (den == 0)
            throw new InvalidOperationException();
        double m = 1 / den;
        var inv = new double[2, 2];
        inv[0, 0] = +m * d;
        inv[0, 1] = -m * b;
        inv[1, 0] = -m * c;
        inv[1, 1] = +m * a;
        return inv;
    }
    double[,] inversematrix = new double[razmer, razmer];
    double[,] matrix_alg_dop = new double[razmer - 1, razmer -
1];
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < n; j++)
        {
            int row = 0;
            for (int i2 = 0; i2 < n; i2++)
            {
                int col = 0;
                if (i2 == i) continue;
                for (int j2 = 0; j2 < n; j2++)
                {
                    if (j2 != j)
                    {
                        matrix_alg_dop[row, col] = matrix[i2, j2];
                        col++;
                    }
                }
                row++;
            }
            inversematrix[i, j] = DetRec(matrix_alg_dop) /
determinant * ((i + j) % 2 == 0 ? 1 : -1);
        }
    }
    double[,] transposed = transpose(inversematrix, n);
    return transposed;
}

private void button1_Click(object sender, EventArgs e)
{
    matrix = new double[razmer, razmer];
    for (int i = 0; i < razmer; i++)
        for (int j = 0; j < razmer; j++)
            matrix[i, j] =
Convert.ToDouble(dataGridView1.Rows[i].Cells[j].Value);
    determinant = DetRec(matrix);
}

```



### Порядок выполнения работы

**Задание 1.** Написать приложение для нахождения определителя матрицы размера  $N \times N$ .

**Задание 2.** Написать приложение для нахождения обратной матрицы размером  $N \times N$ . Данные могут считываться из файла, и результат выводится на форму и в файл.

**Задание 3.** Написать приложение вычисления системы  $N$  линейных уравнений с  $n$ -неизвестными матричным методом. Предусмотреть считывание данных с формы и из файла и вывод результата на форму и в файл.

**Задание 4.** Написать приложение решения системы  $N$  линейных уравнений с  $n$ -неизвестными методом Крамера.

## 3 МЕТОД КОНЕЧНЫХ РАЗНОСТЕЙ (МКР)

Метод конечных разностей (МКР) – метод численного решения краевых задач для дифференциальных уравнений, называемый также методом сеток. Суть метода состоит в следующем. На рассчитываемую область наносится сетка с узлами. Все производные, входящие в дифференциальные уравнения и граничные условия, приближенно заменяются соответствующими разностными отношениями (по формулам численного дифференцирования) и, таким образом, выражаются через неизвестные узловые значения искомой функции. В результате приходят к системе линейных алгебраических уравнений относительно значений функций в узлах сетки. Решение этой системы с последующей интерполяцией в промежутках между узлами позволяет, в конечном счете, получить приближенное решение рассматриваемой задачи.

Большим преимуществом этого метода является слабая зависимость от граничных условий задачи, геометрии конструкций и характера исходного напряженного состояния. Недостатком является высокий порядок систем алгебраических уравнений.

Рассмотрим дифференциальное уравнение

$$y'' + p(x)y' + q(x)y - f(x) = 0; \quad y(x_0) = y_0, y(x_n) = y_n. \quad (3.1)$$

Разобьем отрезок  $[x_0, x_n]$  на  $n$  равных частей с шагом  $h = \frac{x_n - x_0}{n}$ .

Точки разбиения называются узлами, а их совокупность – сеткой на отрезке  $[x_0, x_n]$ . Значения в узлах искомой функции и ее производных обозначим  $y_i = y(x_i), y'_i = y'(x_i), y''_i = y''(x_i)$ . Наша задача определить  $y_1, y_2, \dots, y_{n-1}$ . Это и будет численным решением уравнения.

Для  $i$ -точки можно записать:

$$y'_i = \frac{y_{i+1} - y_i}{h} \text{ – правая производная,}$$

$$y'_i = \frac{y_i - y_{i-1}}{h} \text{ – левая производная,}$$

$$y'_{icp} = \frac{\left(\frac{y_{i+1} - y_i}{h}\right) + \left(\frac{y_i - y_{i-1}}{h}\right)}{2} = \frac{y_{i+1} - y_{i-1}}{2h} \text{ – средняя производная.}$$

Вторая производная – это изменение первых производных.

$$y''_i = \frac{\left(\frac{y_{i+1} - y_i}{h} - \frac{y_i - y_{i-1}}{h}\right)}{h} = \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2}.$$

Уравнение (3.1) для  $(x_i, y_i)$   $i=2, \dots, n-2$  можно записать в виде

$$\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + p(x_i) \frac{y_{i+1} - y_{i-1}}{2h} + q(x_i)y_i - f(x_i) = 0, \quad (3.2)$$

или

$$\begin{cases} \left(\frac{1}{h^2} - \frac{p(x_i)}{2h}\right) y_{i-1} + \left(-\frac{2}{h^2} + q(x_i)\right) y_i + \left(\frac{1}{h^2} + \frac{p(x_i)}{2h}\right) y_{i+1} = f(x_i) & (i = 2, \dots, n-2) \\ \left(-\frac{2}{h^2} + q(x_1)\right) y_1 + \left(\frac{1}{h^2} + \frac{p(x_1)}{2h}\right) y_2 = f(x_1) - \left(\frac{1}{h^2} - \frac{p(x_1)}{2h}\right) y_0 & (i = 1) \\ \left(\frac{1}{h^2} - \frac{p(x_{n-1})}{2h}\right) y_{n-2} + \left(-\frac{2}{h^2} + q(x_{n-1})\right) y_{n-1} = f(x_{n-1}) - \left(\frac{1}{h^2} + \frac{p(x_{n-1})}{2h}\right) y_n & (i = n-1). \end{cases} \quad (3.3)$$

## ЛАБОРАТОРНАЯ РАБОТА 4. СОЗДАНИЕ ПРИЛОЖЕНИЯ РЕШЕНИЯ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ ВТОРОГО ПОРЯДКА С ПОСТОЯННЫМИ КОЭФФИЦИЕНТАМИ МЕТОДОМ КОНЕЧНЫХ РАЗНОСТЕЙ

### *Краткие теоретические сведения*

В качестве примера рассмотрим решение уравнения

$$a \cdot \phi(x) + b \cdot \frac{d}{dx} \phi(x) + c \cdot \frac{d^2}{dx^2} \phi(x) = d; \quad \phi(x_0) = y_0, \phi(x_n) = y_n \quad (3.4)$$

методом конечных разностей в Maple.



$$\begin{pmatrix} a\Delta x^2 + b\Delta x - 2c & c & 0 & 0 \\ (c - b\Delta x) & a\Delta x^2 + b\Delta x - 2c & c & 0 \\ 0 & (c - b\Delta x) & a\Delta x^2 + b\Delta x - 2c & c \\ 0 & 0 & (c - b\Delta x) & a\Delta x^2 + b\Delta x - 2c \end{pmatrix} \cdot \begin{pmatrix} \varphi_1 \\ \varphi_2 \\ \varphi_3 \\ \varphi_4 \end{pmatrix} = \begin{pmatrix} d\Delta x^2 + \varphi_0(c - b\Delta x) \\ d\Delta x^2 \\ d\Delta x^2 \\ d\Delta x^2 - c\varphi_5 \end{pmatrix}$$

Функция для вычисления коэффициентов матрицы системы линейных уравнений размерности n имеет в Maple следующий вид:

```
restart with(LinearAlgebra): with(plots):
matr:=proc(n) local m,i:
m:=Matrix(n):
for i from 1 to n do
if (i-1>0) then
m[i,i-1]:=c(x0+i·Δx)-b(x0+i·Δx)·Δx:
end if:
m[i,i]:=a(x0+i·Δx)·Δx2+b(x0+i·Δx)·Δx-2·c(x0+i·Δx):
if (i+1≤n) then
m[i,i+1]:=c(x0+i·Δx):
end if:
end do:
m
end proc:
```

Функция для вычисления столбца свободных коэффициентов для систем линейных уравнений:

```
free:=proc(n) local m,i:
m:=Matrix(n,1):
for i from 1 to n do
if (i=1) then
m[i,1]:=d(x0+i·Δx)·Δx2-phi_0·(c(x0+i·Δx)-b(x0+i·Δx)·Δx):
elif (i=n) then
m[i,1]:=d(x0+i·Δx)·Δx2-c(x0+i·Δx) phi_L:
else
m[i,1]:=d(x0+i·Δx)·Δx2:
end if:
end do:
m
end proc:
```

Введем исходные данные:

```

x0:=-10:
xL:=10:
n:=10:
Δx=(xL-x0)/(n+1):
phi_0:=-10:
phi_L:=4:
a:=x→0:
b:=x→x:
c:=x→6:
d:=x→0:

```

Решение системы будем искать при помощи следующей команды:

```
rez:=evalf(LinearSolve(matr(n),free(n)));
```

Точное решение дифференциального уравнения можно получить с помощью команды **dsolve**:

```

rez2:=dsolve(
{
a(x)·φ(x)+b(x)· $\frac{d}{dx}$ φ(x)+
c(x)· $\frac{d^2}{dx^2}$ φ(x)=d(x),φ(x0)=phi_0,φ(xL)=phi_L},
numeric,output=procedurelist):
rr:=x→rhs(rez2(x)[2]):

```

Построим графики:

```

display(plot(rr,color="DarkGreen",legent="Точное"),
plot([seq(x0+i·Δx,i=1..n)],
rez+[1],color=red,legent="Метод конечных элементов"))

```

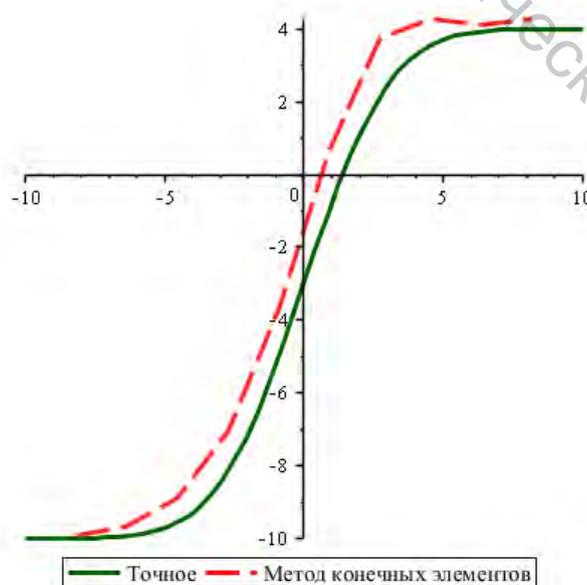


Рисунок 3.1 – Точное и приближенное решение МКР дифференциального уравнения второго порядка со специальной правой частью

## Порядок выполнения работы

**Задание 1.** Разработать в Maple приложение для решения неоднородного дифференциального уравнения второго порядка с постоянными коэффициентами методом конечных разностей:

$$a \cdot \phi(x) + b \cdot \frac{d}{dx} \phi(x) + c \cdot \frac{d^2}{dx^2} \phi(x) = d; \phi(x_0) = y_0, \phi(x_n) = y_n,$$

где  $a, b, c, d$  – постоянные действительные числа;  
 $x_0=0$  и  $x_n=1$ .

Таблица 3.1 – Данные к заданию 1 по вариантам

Вариант	$a$	$b$	$c$	$d$	$y_0$	$y_n$	Вариант	$a$	$b$	$c$	$d$	$y_0$	$y_n$
1	2	-3	2	1	0	1	9	-1	1	3	2	0	1
2	1	-3	3	3	0	1	10	2	2	2	-1	0	1
3	2	-1	-2	2	1	2	11	2	2	3	5	1	2
4	1	-3	-1	5	0	1	12	3	1	2	6	0	1
5	2	-3	4	-3	0	1	13	2	-2	4	1	1	2
6	1	-2	3	-1	1	2	14	1	3	5	6	2	3
7	3	-2	2	4	1	2	15	3	2	5	-7	1	2
8	-2	-4	1	4	0	1							

**Задание 2.** Найти точное решение дифференциального уравнения из задания 1 и сравнить результаты.

## ЛАБОРАТОРНАЯ РАБОТА 5. СОЗДАНИЕ ПРИЛОЖЕНИЯ РЕШЕНИЯ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ ВТОРОГО ПОРЯДКА С ПРОИЗВОЛЬНЫМИ КОЭФФИЦИЕНТАМИ МЕТОДОМ КОНЕЧНЫХ РАЗНОСТЕЙ

### Краткие теоретические сведения

При создании Windows-приложения на языке C# для анализа математических выражений можно воспользоваться библиотекой info.lundin.math. Описанный в ней класс ExpressionParser преобразует строковое выражение в математическую функцию и возвращает ее значение (типа double) для заданного параметра. Установочный пакет info.lundin.math.dll. Его можно скачать: <http://www.lundin.info/mathparser>.

Рассмотрим простой пример использования данной библиотеки.

Основные компоненты программы:

textBox1 – textBox для ввода данных;  
expression – textBox для ввода выражения;  
resultEnd – textBox для вывода результата расчета;  
Button1 – кнопка для вычисления.

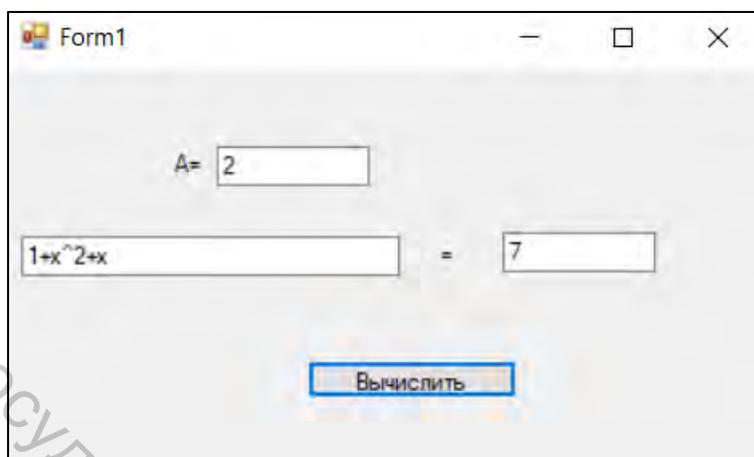


Рисунок 3.2 – Ввод и расчет формулы с формы

Исходный код программы:

```
using info.lundin.math;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
namespace Expression
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        private void button1_Click(object sender, EventArgs e)
        {
            double A = Convert.ToDouble(textBox1.Text);
            string expr = expression.Text;
            double result= Parse(A, expr);
            resultEnd.Text = Convert.ToString(result);
        }
        // Метод, преобразующий введённую формулу к исполняемому виду и
        // подстановки значения переменной
    }
}
```

```

private double Parse(double x, string expression)
{
    // Импортируем парсер
    ExpressionParser parser = new ExpressionParser();
    // Добавляем переменную
    parser.Values.Add("x", x);
    // Преобразуем выражение и подставляем значение переменной
    double result = parser.Parse(expression);
    //parser.Values.Clear();
    // Возвращаем результат вычисления
    return result;
}
}
}

```

### Порядок выполнения работы

Разработать приложение для решения дифференциального уравнения методом конечных разностей согласно варианту.

Таблица 3.2 – Данные по вариантам

1	$a \frac{d\varphi}{dx} + b \frac{d^2\varphi}{dx^2} = cx; \quad x \in [x_0; x_L];$ $\varphi(x = x_0) = \varphi_0; \quad \varphi(x = x_L) = \varphi_L$	2	$b \frac{d^2\varphi}{dx^2} = c \frac{1}{x}; \quad x \in [x_0; x_L];$ $\varphi(x = x_0) = \varphi_0; \quad \varphi(x = x_L) = \varphi_L$
3	$b \frac{d^2\varphi}{dx^2} = cx^2; \quad x \in [x_0; x_L];$ $\varphi(x = x_0) = \varphi_0; \quad \varphi(x = x_L) = \varphi_L$	4	$b \cdot x^2 \frac{d^2\varphi}{dx^2} = c; \quad x \in [x_0; x_L];$ $\varphi(x = x_0) = \varphi_0; \quad \varphi(x = x_L) = \varphi_L$
5	$\sin(a \cdot x) \frac{d\varphi}{dx} = bx^2 + cx;$ $x \in [x_0; x_L];$ $\varphi(x = x_0) = \varphi_0; \quad \varphi(x = x_L) = \varphi_L$	6	$a\sqrt{x} \frac{d\varphi}{dx} + b \cdot x \frac{d^2\varphi}{dx^2} = 0;$ $x \in [x_0; x_L];$ $\varphi(x = x_0) = \varphi_0; \quad \varphi(x = x_L) = \varphi_L$
7	$a \frac{d\varphi}{dx} + b \frac{1}{x} \frac{d^2\varphi}{dx^2} = cx^2;$ $x \in [x_0; x_L];$ $\varphi(x = x_0) = \varphi_0; \quad \varphi(x = x_L) = \varphi_L$	8	$a \cos(x) \frac{d\varphi}{dx} + b \frac{d^2\varphi}{dx^2} = c \frac{1}{x};$ $x \in [x_0; x_L];$ $\varphi(x = x_0) = \varphi_0; \quad \varphi(x = x_L) = \varphi_L$
9	$a \frac{d\varphi}{dx} + bx \frac{d^2\varphi}{dx^2} = 0;$ $x \in [x_0; x_L];$ $\varphi(x = x_0) = \varphi_0; \quad \varphi(x = x_L) = \varphi_L$	10	$a \cdot x \frac{d\varphi}{dx} = c \frac{1}{x};$ $x \in [x_0; x_L];$ $\varphi(x = x_0) = \varphi_0; \quad \varphi(x = x_L) = \varphi_L$

11	$a \frac{d\varphi}{dx} + b \frac{d^2\varphi}{dx^2} = \sin(cx);$ $x \in [x_0; x_L];$ $\varphi(x = x_0) = \varphi_0; \quad \varphi(x = x_L) = \varphi_L$	12	$ax^2 \frac{d\varphi}{dx} + b \cos(x) \frac{d^2\varphi}{dx^2} = c;$ $x \in [x_0; x_L];$ $\varphi(x = x_0) = \varphi_0; \quad \varphi(x = x_L) = \varphi_L$
13	$a \sin(x) \frac{d\varphi}{dx} + b \cos(x) \frac{d^2\varphi}{dx^2} = 0;$ $x \in [x_0; x_L];$ $\varphi(x = x_0) = \varphi_0; \quad \varphi(x = x_L) = \varphi_L$	14	$a \cdot x^3 \frac{d\varphi}{dx} = c \frac{1}{x};$ $x \in [x_0; x_L];$ $\varphi(x = x_0) = \varphi_0; \quad \varphi(x = x_L) = \varphi_L$
15	$\frac{a}{x} \cdot \frac{d^2\varphi}{dx^2} = bx^2; \quad x \in [x_0; x_L]$ $\varphi(x = x_0) = \varphi_0; \quad \varphi(x = x_L) = \varphi_L$		

Примечание:  $a, b, c, d, x_0, x_L, \varphi_0, \varphi_L$  – задавать самостоятельно.

## 4 МЕТОД СЕТОК ДЛЯ РЕШЕНИЯ УРАВНЕНИЙ С ЧАСТНЫМИ ПРОИЗВОДНЫМИ

В случае двух независимых переменных эти уравнения можно записать в виде

$$a \frac{\partial^2 u}{\partial x^2} + 2b \frac{\partial^2 u}{\partial x \partial y} + c \frac{\partial^2 u}{\partial y^2} + d \frac{\partial u}{\partial x} + e \frac{\partial u}{\partial y} + fu = g. \quad (4.1)$$

Здесь  $u = u(x, y)$  – искомая функция.

В зависимости от соотношения между коэффициентами существуют различные виды уравнений. При  $a=b=c=f=0, d \neq 0, e \neq 0$  получается уравнение первого порядка вида

$$\frac{\partial u}{\partial x} + p \frac{\partial u}{\partial y} = q, \quad (4.2)$$

называемое уравнением переноса.

Если хотя бы один из коэффициентов  $a, b, c$  в (4.1) отличен от нуля, то получим уравнения второго порядка:

1.  $\frac{\partial^2 u}{\partial t^2} = a^2 \frac{\partial^2 u}{\partial x^2}$  – волновое уравнение, описывающее процесс поперечных колебаний стержня, электрические колебания в проводе и т. д.

2.  $\frac{\partial u}{\partial t} = a \frac{\partial^2 u}{\partial x^2}, a > 0$  – уравнение теплопроводности или диффузии.

Описывает распределение температуры в заданной области пространства и ее

изменение во времени. Искомая функция  $u=u(x,t)$  – задает температуру в точке с координатами  $x$  в момент времени  $t$ .

3.  $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0$  – уравнение Лапласа. К данному уравнению приводят

задачи об электрических и магнитных полях, задачи гидродинамики, диффузии и т. д.

## ЛАБОРАТОРНАЯ РАБОТА 6. СОЗДАНИЕ ПРИЛОЖЕНИЯ РЕШЕНИЯ НЕСТАЦИОНАРНОГО УРАВНЕНИЯ ТЕПЛОПРОВОДНОСТИ

### *Краткие теоретические сведения*

Рассмотрим задачу для уравнения теплопроводности: найти функцию  $u(x,t)$ , удовлетворяющую уравнению

$$\frac{\partial u}{\partial t} = a^2 \frac{\partial^2 u}{\partial x^2}, \quad (4.3)$$

начальному условию

$$u(x, 0) = f(x), \quad 0 \leq x \leq s$$

и краевым условиям

$$u(0, t) = \varphi(t), \quad u(s, t) = \psi(t) \quad (t > 0).$$

Введем прямоугольную сетку узлов в полосе  $t \geq 0, 0 \leq x \leq s$ , образуемую точками пересечения двух семейств параллельных прямых:

$$x = ih \quad (i = 0, 1, \dots), \quad t = jl \quad (j = 0, 1, \dots).$$

Заменим в каждом узле  $(x_i, t_j)$  вторую производную разностным отношением

$$\left( \frac{\partial^2 u}{\partial x^2} \right)_{ij} \approx \frac{u_{i+1,j} - 2u_{ij} + u_{i-1,j}}{h^2},$$

а первые производные одним из двух разностных отношений:

$$\left(\frac{\partial u}{\partial t}\right)_{ij} \approx \frac{u_{i,j+1} - u_{ij}}{\ell}, \quad \left(\frac{\partial u}{\partial t}\right)_{ij} \approx \frac{u_{i,j} - u_{ij-1}}{\ell}.$$

Получим два типа конечно-разностных уравнений:

$$\frac{u_{i,j+1} - u_{ij}}{\ell} = a^2 \frac{u_{i+1,j} - 2u_{ij} + u_{i-1,j}}{h^2}, \quad (4.4)$$

$$\frac{u_{i,j} - u_{ij-1}}{\ell} = a^2 \frac{u_{i+1,j} - 2u_{ij} + u_{i-1,j}}{h^2}. \quad (4.5)$$

Первая разностная схема является явной схемой, вторая – неявная. Пусть

$$\sigma = \frac{a^2 \ell}{h^2}. \quad (4.6)$$

С учетом (4.6) уравнения (4.4) и (4.5) примут вид

$$u_{i,j+1} = (1 - 2\sigma)u_{ij} + \sigma(u_{i+1,j} + u_{i-1,j}), \quad (4.7)$$

$$(1 - 2\sigma)u_{i,j} - \sigma(u_{i+1,j} + u_{i-1,j}) - u_{i,j-1} = 0. \quad (4.8)$$

Доказано, что уравнение (4.7) будет устойчиво при  $0 < \sigma \leq \frac{1}{2}$ , а (4.8) – при любом значении  $\sigma$ . Чтобы получить наиболее простой вид уравнения (4.7), положим  $\sigma = 1/2$ , тогда

$$u_{i,j+1} = \frac{u_{i+1,j} + u_{i-1,j}}{2}.$$

Рассмотрим решение двумерной нестационарной задачи теплопроводности методом конечных разностей в Maple

$$\frac{\partial \phi}{\partial t} - k \left( \frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} \right) = 0.$$

Заменяя в каждом узле первую и вторую производные разностными отношениями, получим

$$\frac{\phi_{i,j}^{n+1} - \phi_{i,j}^n}{\Delta t} - k \cdot \left( \frac{\phi_{i,j-1}^n - 2 \cdot \phi_{i,j}^n + \phi_{i,j+1}^n}{\Delta x^2} - \frac{\phi_{i-1,j}^n - 2 \cdot \phi_{i,j}^n + \phi_{i+1,j}^n}{\Delta y^2} \right) = 0.$$

ИЛИ

$$\phi_{i,j}^{n+1} = \phi_{i,j}^n + \Delta t \cdot k \cdot \left( \frac{\phi_{i,j-1}^n - 2 \cdot \phi_{i,j}^n + \phi_{i,j+1}^n}{\Delta x^2} - \frac{\phi_{i-1,j}^n - 2 \cdot \phi_{i,j}^n + \phi_{i+1,j}^n}{\Delta y^2} \right) = 0.$$

Напишем приложение в Maple:

```
restart  
interface(rtablesize=25);
```

Количество интервалов:

```
N:=10:  
M:=10:
```

Начальные и граничные условия:

```
Phi_0:=50:  
Phi_l:=0:  
Phi_r:=100:  
Phi_u:=200:  
Phi_d:=-10:
```

Коэффициент теплопроводности:

```
k:=1.172*10^-5:
```

Размер области:

```
L_x:=0.3:  
L_y:=0.2:
```

Размеры интервалов:

```
dx:=L_x/N  
dy:=L_y/M
```

Шаг по времени:

```
dt:=1/(2*(k/dx^2+k/dy^2))
```

Общее время:

```
T:=200*dt
```

Формируем матрицу начальных и граничных значений в узлах:

```

for i from 1 by 1 to M+1 do
  for j from 1 by 1 to N+1 do
    if i=1 then  $\Phi[i,j] := \Phi_u$ 
    else if i=M+1 then  $\Phi[i,j] := \Phi_d$ 
    else if j=1 then  $\Phi[i,j] := \Phi_l$ 
    else if i=N+1 then  $\Phi[i,j] := \Phi_r$ 
    else  $\Phi[i,j] := \Phi_0$ 
    end if
  end if
end if
end if
end do
end do
F[0] := Matrix( $\Phi$ , M+1, N+1)

```

Вычисляем по итерационной формуле:

```

for t from 1 by 1 to T/dt do
  F[t] := Matrix(F[t-1]):
  for i from 2 by 1 to M do
    for j from 2 by 1 to N do
      F[t][i,j] := F[t-1][i,j] + dt * k * ((F[t-1][i,j-1] -
      2 * F[t-1][i,j] + F[t-1][i,j+1]) / dx2 + (F[t-1][i-1,j] -
      2 * F[t-1][i,j] + F[t-1][i+1,j]) / dy2)
    end do:
  end do:
end do:

```

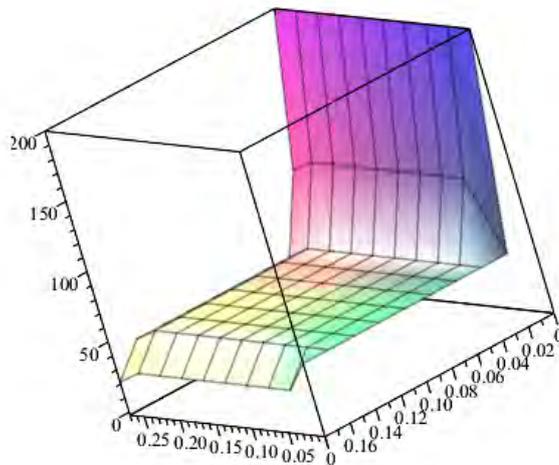
Отображаем результаты расчетов:

```

with(plots):
data := [seq([seq([dx * (j-1), dy * (i-1), F[A][i,j], j=1..M)], i=1..N)]:
animate(surfd, [data, axes=boxed], A=[seq(i, i=1..100)])

```

A = 1.



### **Порядок выполнения работы**

Найти решение  $u(x, t)$  для нестационарного уравнения теплопроводности с постоянными коэффициентами:

$$\frac{\partial u}{\partial t} = \alpha^2 \frac{\partial^2 u}{\partial x^2}, \quad \alpha = 1,$$

с начальными условиями

$$u(x, 0) = f(x), \quad 0 \leq x \leq 1,$$

и граничными условиями

$$u(0, t) = a, \quad u(1, t) = b.$$

Для решения задачи построить сетку из 11 узлов по  $x$  ( $i = 0, 1, \dots, 10$ ) и провести вычисления для 12 слоев по  $t$  ( $j = 0, 1, \dots, 11$ ). Вычисления выполнить с шагом  $h$  по  $x$ , равным 0,1, и шагом по  $t$ , равным 0,005. Отобразить графически решение задачи.

Таблица 4.1 – Данные по вариантам

№	$f(x)$	$a$	$b$	№	$f(x)$	$a$	$b$
1	$x \cdot (x - 1)$	0	0	9	$(x^2 + 0,5) \cdot \cos(2\pi x)$	0	0
2	$x^3 + x^2 - x$	0	1	10	$\sin(\pi x) \cdot \cos(x)$	0	0
3	$x^2 \cdot (1 - x)$	0	0	11	$x \cdot \sin(2 \cdot (x - 1))$	0	0
4	$1 - x^4$	1	0	12	$\ln(0,5 + x) \cdot (x - 1)$	0,7	0
5	$x \cdot \sin(2\pi x)$	0	-0,3	13	$x \cdot \sin(4(x - 1)) - x$	0	-1
6	$(x - 1) \cdot \sin^2 x$	0	0	14	$x \cdot \cos(2\pi x)$	0	1
7	$4x^2(x - 1)$	0	0,5	15	$x \cdot e^{-x} \cdot (x^4 - 2)$	0	-0,4
8	$10x^3 \cdot (x - 1)$	0,5	0				

## **ЛАБОРАТОРНАЯ РАБОТА 7. РАСЧЕТ КОЛЕБАНИЯ СТРУНЫ**

### **Краткие теоретические сведения**

Рассмотрим задачу для уравнения свободных колебаний однородной струны. Необходимо найти функцию  $u(x, t)$ , удовлетворяющую уравнению

$$\frac{\partial^2 u}{\partial t^2} = a^2 \frac{\partial^2 u}{\partial x^2}, \quad (4.9)$$

начальным условиям

$$u(x, 0) = f(x), \quad \frac{\partial u}{\partial t}(x, 0) = \Phi(x), \quad 0 \leq x \leq s \quad (4.10)$$

и граничным условиям

$$u(0, t) = \varphi(t); \quad u(s, t) = \psi(t), \quad t > 0. \quad (4.11)$$

Построим в полосе  $t \geq 0; 0 \leq x \leq s$  прямоугольную сетку

$$x_i = ih \quad (i = 0, 1, \dots), \quad t_j = jl \quad (j = 0, 1, \dots).$$

Заменяем производные в уравнении (4.9) разностными отношениями:

$$\left( \frac{\partial^2 u}{\partial x^2} \right)_{ij} \approx \frac{u_{i+1,j} - 2u_{ij} + u_{i-1,j}}{h^2}, \quad \left( \frac{\partial^2 u}{\partial t^2} \right)_{ij} \approx \frac{u_{i,j+1} - 2u_{ij} + u_{i,j-1}}{\ell^2}.$$

Получим

$$\frac{u_{i,j+1} - 2u_{ij} + u_{i,j-1}}{\ell^2} = a^2 \frac{u_{i+1,j} - 2u_{ij} + u_{i-1,j}}{h^2}. \quad (4.12)$$

Введя обозначение  $\alpha = \frac{a\ell}{h}$ , получим разностное уравнение

$$u_{i,j+1} = 2u_{ij} - u_{i,j-1} + \alpha^2 (u_{i+1,j} - 2u_{ij} + u_{i-1,j}), \quad (4.13)$$

которое является устойчивым при  $\alpha \leq 1$ .

Уравнение (4.13) принимает наиболее простой вид при  $\alpha = 1$

$$u_{i,j+1} = u_{i+1,j} + u_{i-1,j} - u_{i,j-1}. \quad (4.14)$$

Из уравнения (4.14) видно, что для получения значений  $u(x, t)$  на  $(j+1)$  слое используются значения  $u(x, t)$  на слоях  $j$  и  $(j-1)$ . Для начала вычисления необходимо знать значения  $u(x, t)$  на первых двух слоях:  $j=0, j=1$ .

Чтобы их определить, заменим в начальном условии (4.10) производную

$$\frac{\partial u}{\partial t}(x, 0) = \Phi(x),$$

разностным отношением

$$\frac{u_{i,1} - u_{i,0}}{\ell} = \Phi(x_i) = \Phi_i.$$

Для определения  $u(x, t)$  на слоях  $j=0$ ,  $j=1$  получим  $u_{i,0} = f(x_i) = f_i$  и  $u_{i,1} = u_{i,0} + \ell\Phi_i = f_i + \ell\Phi_i$ .

В качестве примера рассмотрим решение в Maple уравнения

$$\frac{\partial^2 \phi}{\partial t^2} = a^2 \frac{\partial^2 \phi}{\partial x^2},$$

Заменяя производные разностными отношениями, получим

$$\frac{\phi_i^{n+1} - 2\phi_i^n + \phi_i^{n-1}}{\Delta t^2} = a^2 \frac{\phi_{i-1}^n - 2\phi_i^n + \phi_{i+1}^n}{\Delta x^2}$$

$$\phi_i^{n+1} = 2\phi_i^n - \phi_i^{n-1} + \Delta t^2 \cdot a^2 \cdot \frac{\phi_{i-1}^n - 2\phi_i^n + \phi_{i+1}^n}{\Delta x^2}.$$

Напишем приложение:

```
restart
interface(rtablesize=25);
N:=30:
M:=300:
 $\Phi_0 := x \rightarrow \text{piecewise}(x \leq 0.6, 0.3/0.6 \cdot x, -0.3/0.4 \cdot (x - 0.6) + 0.3):$ 
 $\Phi_1 := 0:$ 
 $\Phi_r := 0:$ 
a:=1:
Lx:=1:
dx:=Lx/N
dt:=0.02
T:=M*dt
for i from 1 by 1 to N+1 do
  if i=1 then  $\Phi[i] := \Phi_1$ 
  else if i=N+1 then  $\Phi[i] := \Phi_r$ 
  else  $\Phi[i] := \Phi_0(dx \cdot (i-1))$ 
  end if
end if
end do
F[0]:=Vector( $\Phi$ , N+1)
F[1]:=Vector(F[0]):
for t from 2 by 1 to T/dt do
```

```

F[t]:= Vector(F[t-1]):
for i from 2 by 1 to N do
  F[t][i]:=2·F[t-1][i]-F[t-2][i]+dt2·a2·
  (F[t-1][i-1]-2·F[t-1][i]+F[t-1][i+1])/dx2
end do:
end do:

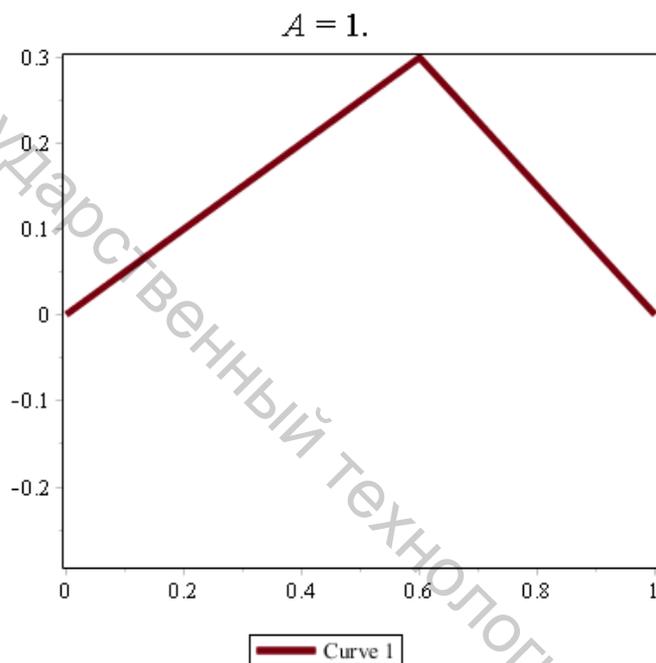
```

Построим график:

```

with(plots):
data:=A→[seq([dx·(i-1),F[A][i]],i=1..N+1)]:
animate(plot,[data(A),axes=boxed],A=[seq(i,i=1..M)])

```



### Порядок выполнения работы

Решить задачу о колебании струны единичной длины с закрепленными концами:

$$\frac{\partial^2 u}{\partial t^2} = a^2 \frac{\partial^2 u}{\partial x^2}, \quad a = 1$$

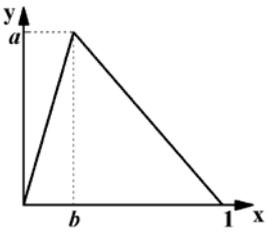
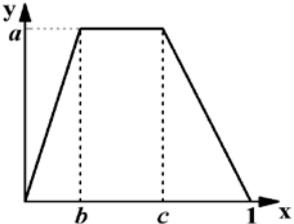
с начальными условиями

$$u(x,0) = f(x), \quad \frac{\partial u}{\partial t}(x,0) = 0, \quad 0 \leq x \leq 1$$

и нулевыми граничными условиями

$$u(0,t) = u(1,t) = 0.$$

Таблица 4.2 – Данные по вариантам

Вариант	$f(x)$	$a$	$b$	Вариант	$f(x)$	$a$	$b$
1		1	0,1	9	$x \cdot \sin(2(x-1))$		
2		2	0,1	10	$4x^3 \cdot (x-1)$		
3		4	0,2	11		1	0,1
4		6	0,3	12		3	0,2
5		8	0,4	13		5	0,4
6	$x \cdot (x^2 - 1)$			14		7	0,6
7	$\sin(\pi x^2)$			15	9	0,8	
8	$\sin(\pi x) \cdot \cos(x)$						

## ЛИТЕРАТУРА

1. Буханько, А. А. Приближенные методы решения краевых задач для обыкновенных дифференциальных уравнений, уравнений с частными производными и интегральных уравнений : учеб.-метод. пособие / А. А. Буханько, О. П. Чостковская. – Самара : Изд-во Самар. гос. аэрокосм. ун-та, 2011. – 68 с.

2. Введение в Comsol Multiphysics [Электронный ресурс]. – Режим доступа: <https://bylib.org/book/3106980/029e7c>. – Дата доступа: 28.04.2022.

3. Дифференциальное и интегральное исчисление функций одной и многих переменных. Дифференциальные уравнения : учеб.-метод. пособие / В. В. Цегельник [и др.]. – Минск : БГУИР, 2018. – 188 с.

4. Математика. Дифференциальные уравнения и системы дифференциальных уравнений. Операционное исчисление : практикум для студентов первого курса специальностей 1-40 05 01-01 «Информационные системы и технологии (в проектировании и производстве)», 1-36 01 01 «Технология машиностроения», 1-53 01 01-01 «Автоматизация технологических процессов и производств (машиностроение и приборостроение)», 1-43 01 07 «Техническая эксплуатация энергооборудования организаций» / УО «ВГТУ» ; сост. А. В. Коваленко [и др.]. – Витебск, 2019. – 100 с.

5. Рабочий процесс создания модели и запуска расчёта в COMSOL Multiphysics [Электронный ресурс]. – Режим доступа: <https://www.comsol.ru/video/setting-up-and-running-a-simulation-with-comsol-multiphysics>. – Дата доступа: 28.04.2022.

6. Сьярле, Ф. Метод конечных элементов для эллиптических задач / Ф. Сьярле ; пер. с англ. Ю. И. Квасова ; под ред. Н. Н. Яненко. – Москва : Мир, 1980. – 512 с.

## ПРИЛОЖЕНИЕ А

### ТЕМПЕРАТУРНАЯ ЗАВИСИМОСТЬ КОЭФФИЦИЕНТА ЭЛЕКТРОПРОВОДНОСТИ (В СИМЕНС/МЕТР) ДЛЯ ТИТАНА

250.00	2.4700E6	259.91	2.3697E6
250.30	2.4669E6	260.21	2.3668E6
250.60	2.4637E6	260.51	2.3639E6
250.90	2.4606E6	260.81	2.3610E6
251.20	2.4574E6	261.11	2.3581E6
251.50	2.4543E6	261.41	2.3552E6
251.80	2.4512E6	261.71	2.3523E6
252.10	2.4481E6	262.01	2.3494E6
252.40	2.4450E6	262.31	2.3465E6
252.70	2.4419E6	262.61	2.3437E6
253.00	2.4388E6	262.91	2.3408E6
253.30	2.4357E6	263.21	2.3380E6
253.60	2.4326E6	263.51	2.3351E6
253.90	2.4296E6	263.81	2.3323E6
254.20	2.4265E6	264.11	2.3295E6
254.50	2.4234E6	264.41	2.3265E6
254.80	2.4204E6		
255.11	2.4174E6		
255.41	2.4143E6		
255.71	2.4113E6		
256.01	2.4083E6		
257.21	2.3963E6		
257.51	2.3933E6		
257.81	2.3903E6		
258.11	2.3874E6		
258.41	2.3844E6		
258.71	2.3814E6		
259.01	2.3785E6		
259.31	2.3756E6		
259.61	2.3726E6		

## ПРИЛОЖЕНИЕ Б

### ТЕМПЕРАТУРНАЯ ЗАВИСИМОСТЬ КОЭФФИЦИЕНТА ЭЛЕКТРОПРОВОДНОСТИ ДЛЯ МЕДИ

250.00	7.2726E7	311.22	5.5809E7
253.06	7.1640E7	314.29	5.5167E7
256.12	7.0585E7	317.35	5.4540E7
259.18	6.9562E7	320.41	5.3927E7
262.24	6.8568E7	323.47	5.3328E7
265.31	6.7602E7	326.53	5.2741E7
268.37	6.6662E7	329.59	5.2168E7
271.43	6.5749E7	332.65	5.1606E7
274.49	6.4860E7	335.71	5.1057E7
277.55	6.3995E7	338.78	5.0519E7
280.61	6.3153E7	341.84	4.9992E7
283.67	6.2333E7	344.90	4.9476E7
286.73	6.1534E7	347.96	4.8970E7
289.80	6.0755E7	351.02	4.8475E7
292.86	5.9995E7	354.08	4.7989E7
295.92	5.9255E7	357.14	4.7513E7
298.98	5.8532E7	360.20	4.7047E7
302.04	5.7827E7	363.27	4.6589E7
305.10	5.7138E7	366.33	4.61405E7
308.16	5.6466E7		

Учебное издание

## **КОМПЬЮТЕРНЫЕ СИСТЕМЫ КОНЕЧНОМЕРНЫХ РАСЧЕТОВ**

Методические указания по выполнению лабораторных работ

Составители:

Дунина Елена Брониславовна

Соколова Анна Сергеевна

Бизюк Андрей Николаевич

Редактор *Т.А. Осипова*

Корректор *Т.А. Осипова*

Компьютерная верстка *А.С. Соколова*

---

Подписано к печати 11.05.2022. Формат 60x90<sup>1/16</sup>. Усл. печ. листов 3,4.  
Уч.-изд. листов 4,3. Тираж 30 экз. Заказ № 134.

Учреждение образования

«Витебский государственный технологический университет»  
210038, г. Витебск, Московский пр., 72.

Отпечатано на ризографе учреждения образования

«Витебский государственный технологический университет».

Свидетельство о государственной регистрации издателя, изготовителя,  
распространителя печатных изданий № 1/172 от 12 февраля 2014 г.

Свидетельство о государственной регистрации издателя, изготовителя,  
распространителя печатных изданий № 3/1497 от 30 мая 2017 г.