УДК 004.9+537.212+001.891.57(075)
ББК В185.505я73-1
В 545

Витюгова Н. А., Задорожный А. М., Черкасский В. С. Моделирование электростатических полей в среде MATLAB: Метод. пособие / Новосиб. гос. ун-т. Новосибирск, 2009. Вып. 1. 52 с.

ISBN 978-5-94356-864-0

Компьютерный практикум по электродинамике предназначен для студентов 2-го курса физического факультета НГУ. Цель практикума – дать студенту возможность, используя компьютерную графику и численное моделирование, познакомиться с физическими закономерностями пространственной структуры электромагнитных полей и законами распространения электромагнитных волн. В пособии описывается раздел практикума «Моделирование электростатических полей», реализованный в среде MATLAB фирмы MathWorks.

Издание подготовлено в рамках реализации *Программы развития государственного образовательного учреждения высшего профессионального образования «Новосибирский государственный университет»* на 2009–2018.

ISBN 978-5-94356-864-0

Введение

Компьютерный практикум по электродинамике был разработан по инициативе заведующего кафедрой общей физики (КОФ) физического факультета Новосибирского государственного университета член-корреспондента РАН И. Н. Мешкова и впервые реализован в 1986 г. на ЭВМ типа «Электроника-60». Первая версия практикума была создана сотрудниками КОФ Н. А. Башлыковой (Витюговой), Н. А. Винокуровым, А. М. Задорожным, В. Г. Казаковым, Е. А. Переведенцевым и Ю. И. Эйдельманом [1, 2]. В последующем развитии практикума, переносе его на более современные персональные компьютеры приняли участие также В. Т. Астрелин, С. Г. Воропаев, Г. Л. Коткин, Б. А. Князев, Г. В. Меледин и В. С. Черкасский [3].

На первых этапах развития компьютерный практикум по электродинамике был основан на прикладном программном обеспечении ELDIN, специально написанном для этой цели и позволяющем:

- моделировать электростатические поля системы точечных зарядов;
- моделировать электростатические поля в области свободной от зарядов;
- моделировать аксиально-симметричные магнитные поля системы круговых витков с током;
- проводить Фурье-анализ сигналов;
- моделировать распространение электромагнитной волны в плоскослоистой среде.

Пакет программ ELDIN использовался не только на физическом факультете Новосибирского государственного университета, но и в ряде других российских вузов, а также в Гаванском высшем институте ядерных технологий. В 1994 г. «Компьютерный практикум по электродинамике» стал победителем Первого европейского конкурса учебного программного обеспечения. На финальной церемонии в Гейдельберге (Германия) 25–29 ноября 1994 г. коллектив авторов получил Certificate for the excellent software contribution to the European Academic Software Award 1994 and the participation in the final.

В последние годы появился ряд профессиональных программных продуктов, предназначенных для моделирования различных физических процессов и явлений. Обладающие мощными средствами визуализации, данные продукты позволяют существенно расширить существующий практикум по электродинамике. Одним из таких программных продуктов является MATLAB фирмы MathWorks, который используется на физическом факультете НГУ в курсе «Компьютерное моделирование физических процессов с использованием MATLAB» [4]. В связи с этим к настоящему времени часть задач компьютерного практикума по электродинамике переработана для решения в среде MATLAB.

В настоящем пособии описывается переработанный раздел компьютерного практикума по электродинамике «Моделирование электростатических полей». В новой версии практикума студентам предлагается исследовать электрические поля не только в областях, свободных от электрических зарядов (решение уравнения Лапласа), но и в областях, содержащих объемные заряды (решение уравнения Пуассона). Добавлены также задачи о формировании электрических полей в неоднородных средах. В процессе работы студенты узнают, как выглядят при графическом отображении поля различных конфигураций, осваивают методику вычисления ёмкости системы электродов, знакомятся с ролью краевых эффектов при оценке ёмкости таких систем, получают представление о точности численных методов при решении уравнений Лапласа и Пуассона.

Пособие предназначено для студентов второго курса физического факультета НГУ.

Авторы выражают благодарность за плодотворное обсуждение содержания учебного пособия В. А. Володину, И. А. Котельникову, А. Н. Матвеенко, А. Г. Погосову, С. Л. Синицкому.

1. Основные уравнения электростатики

В вакууме, в области, содержащей объемные заряды с плотностью $\rho(x, y, z)$, распределение электростатического потенциала $U(x, y, z)$ описывается **уравнением Пуассона** [5, 6]

$$div\, grad\, U = -4\pi\rho\ . \tag{1}$$

Используя оператор Лапласа $\Delta = div\, grad$, уравнение (1) обычно записывают в виде

$$\Delta U = -4\pi\rho\ . \tag{2}$$

В декартовых координатах уравнение Пуассона имеет вид

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} + \frac{\partial^2 U}{\partial z^2} = -4\pi\rho\ . \tag{3}$$

Уравнение Пуассона дает возможность определить потенциал поля объемных зарядов, если известно расположение этих зарядов. В областях, где нет электрических зарядов, уравнение Пуассона обращается в **уравнение Лапласа**

$$\Delta U = 0 \tag{4}$$

или в декартовых координатах

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} + \frac{\partial^2 U}{\partial z^2} = 0\ . \tag{5}$$

Напряженность электростатического поля $\vec{E}$ связана с потенциалом U соотношением

$$\vec{E} = -\, grad\, U \tag{6}$$

или

$$E_x = -\frac{\partial U}{\partial x},\ E_y = -\frac{\partial U}{\partial y},\ E_z = -\frac{\partial U}{\partial z}\ . \tag{7}$$

В диэлектрике уравнение для электростатического потенциала имеет вид

$$div(\, \varepsilon\, grad\, U\,) = -4\pi\rho\ , \tag{8}$$

где $\varepsilon(x, y, z)$ – диэлектрическая проницаемость среды. В случае однородного диэлектрика уравнение (7) сводится к обычному уравнению Пуассона:

$$\Delta U = -\frac{4\pi\rho}{\varepsilon}. \tag{9}$$

В проводнике $\vec{E} \equiv 0$. К поверхности проводника поле подходит по нормали, и его величина на поверхности определяется выражением

$$\vec{E}_n = 4\pi\sigma, \tag{10}$$

где σ – поверхностная плотность зарядов.

Для решения уравнения Пуассона или Лапласа в заданной области необходимо задать дополнительные **граничные условия** на ее границе. Граничные условия для задач электростатики могут быть двух типов – Дирихле или Неймана. Условия Дирихле подразумевают задание на границе электростатического потенциала U, а Неймана – поверхностного заряда, величина которого пропорциональна нормальной составляющей градиента потенциала на границе $\sigma \sim grad_n U$. Все задачи из настоящего пособия предполагают задание граничных условий Дирихле.

2. Моделирование электростатических полей с помощью пакета MATLAB PDE Toolbox

Пакет программ PDE Toolbox (Partial Differential Equation Toolbox) является пакетом расширения системы автоматизации математических расчетов MATLAB фирмы MathWorks, который предназначен для численного решения двухмерных дифференциальных уравнений в частных производных второго порядка. Для численного решения дифференциальных уравнений в пакете PDE Toolbox используется метод конечных элементов (см. Приложение 3). Пакет PDE Toolbox разработан как для начинающих, так и для опытных пользователей. Минимальное требование к пользователю – он должен уметь формулировать задачу на бумаге, т. е. задавать необходимое уравнение в частных производных, область его определения и граничные условия. В компьютерном практикуме по электродинамике используется только небольшая часть возможностей пакета PDE Toolbox, а именно раздел, специально посвященный решению уравнений электростатики, т. е. уравнений Пуассона или Лапласа.

Пакет PDE Toolbox содержит программные инструменты, которые позволяют:

- сформулировать задачу на основе двухмерного уравнения в частных производных, т. е. определить 2-D область, в которой ищется решение, задать граничные условия и коэффициенты дифференциального уравнения;

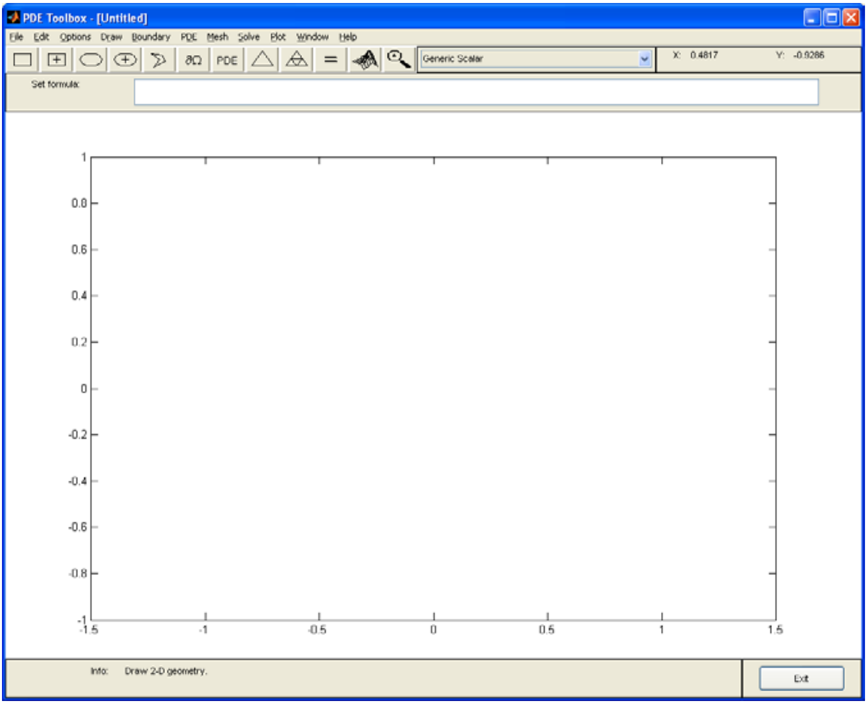
- численно решить поставленную задачу, т. е. сгенерировать неструктурированную сетку, на которой ищется решение, дискретизировать исходное дифференциальное уравнение, найти приближенное решение;

- визуализировать результат.

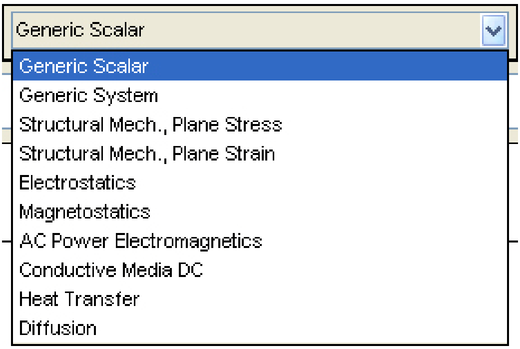
Пакет PDE Toolbox включает в себя также графический интерфейс пользователя (graphical user interface, GUI) pdetool, являющийся замкнутой графической оболочкой для решения дифференциальных уравнений в частных производных второго порядка. GUI-приложение pdetool значительно облегчает работу пользователя, обеспечивает интерактивный режим работы на всех этапах формулировки задачи, решения уравнений и визуализации результатов. Использование pdetool не требует знаний математики за пределами основ теории дифференциальных уравнений в частных производных, а также знаний численных методов их решения и языка программирования MATLAB. GUI-приложение pdetool предоставляет возможность пользоваться обычными физическими терминами, а не абстрактными коэффициентами.

Рассмотрим кратко основные этапы решения задачи в PDE Toolbox на примере задачи формирования однородного электрического поля в двухмерной области произвольной конфигурации, в которой отсутствуют электрические заряды. Подобные задачи формирования электростатических полей с помощью электродов определенной формы, находящихся под заданными потенциалами, являются типичными задачами многих экспериментальных исследований и разработок. Формирование электрических полей определенной конфигурации требуется, например, при создании электронных пушек, фотоэлектронных умножителей, электронно-оптических преобразователей и других приборов электронной оптики, экспериментальной физики и техники. Решение данной задачи сводится к решению уравнения Лапласа с определенно заданными граничными условиями.

Запустим графический интерфейс пакета PDE Toolbox. Для этого необходимо в командной строке MATLAB набрать «pdetool». После выполнения команды на экране монитора открывается окно PDE Toolbox:



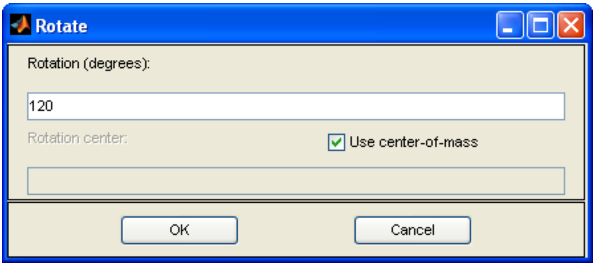
Выбираем в Ниспадающем меню краевую задачу **Electrostatics**, или в меню **Options** ⇨ **Application** ⇨ **Electrostatics**.



<u>В *режиме Draw*</u> моделируем форму 2-D области, которая может быть создана с помощью различных геометрических объектов: прямоугольника **(Rectangle)**, эллипса **(Ellipse)**, многоугольника **(Polygon)** и их комбинаций (первые 5 кнопок слева направо на панели инструментов ) или меню **Draw**.

Можно нарисовать объекты, которые могут перекрываться.

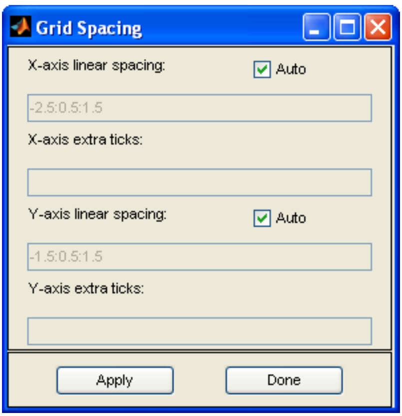
Объекты можно перемещать (щелкнув объект, нажмите левую кнопку мыши и удерживайте ее до появления голубой рамки, не отпуская кнопку мыши, переместите объект) и вращать. Чтобы объект повернуть, выберите в меню: **Draw ⇨ Rotate…**



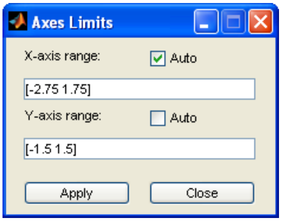
Перемещать и вращать можно и группы объектов, предварительно выделив эти объекты. Выделить все объекты можно, выбрав в меню **Edit ⇨ Select All** или нажав клавиши **Ctrl/A**. Объекты можно копировать, вырезать и вставлять среди выбранной области.

Форма 2-D области может быть сохранена в m-файл, который содержит Matlab команды, необходимые для создания модели, так что вам не придется повторять все шаги каждый раз.

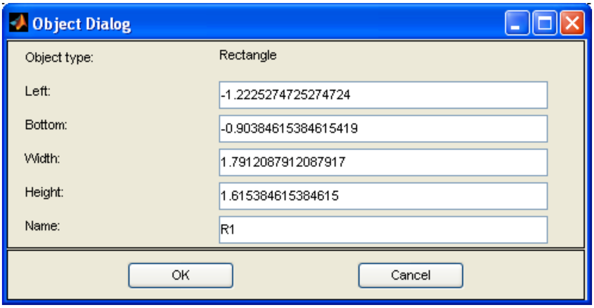
При создании формы 2-D области доступна функция **snap-to-grid**. Это означает, что объекты будут выравниваться по сетке. **Сетку (grid)** можно включать **Options ⇨ Grid** и отключать **Options⇨ Grid**, а также изменять масштаб и параметры сетки **Options ⇨ Grid Spacing…**



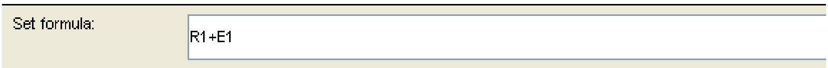
Для изменения характеристик осей координат выберите **Options⇨ Axes Limits…**



Дважды щелкнув мышкой по объекту, можно вызвать диалоговое окно, которое позволит изменить некоторые параметры объекта: например, координаты левого нижнего угла, ширину и высоту для прямоугольника.

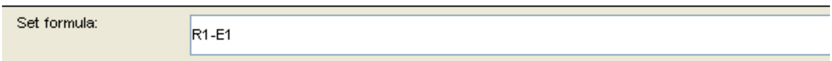


Для более сложной формы, такой, как квадрат с отверстием, можно нарисовать квадрат и круг, позиционируя их в зависимости от постановки задачи.

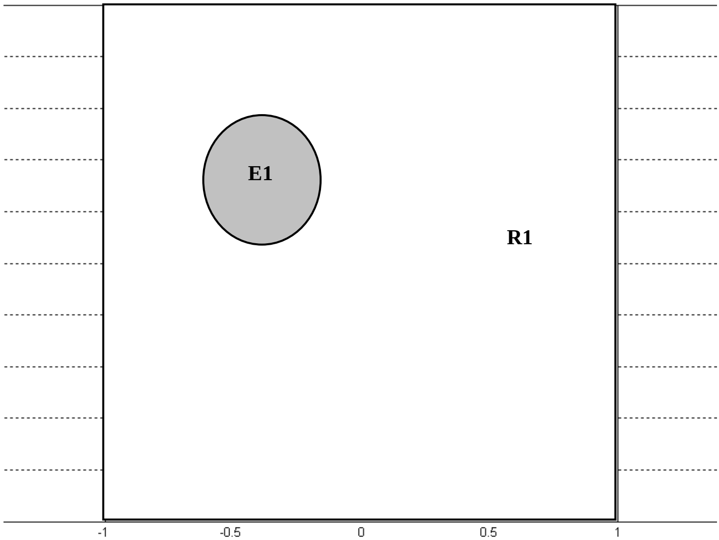


В окне формулы видно, что Matlab определяет область как «прямоугольник + эллипс», и можно просто изменить формулу на «прямоугольник – эллипс», для того, чтобы создать отверстие.

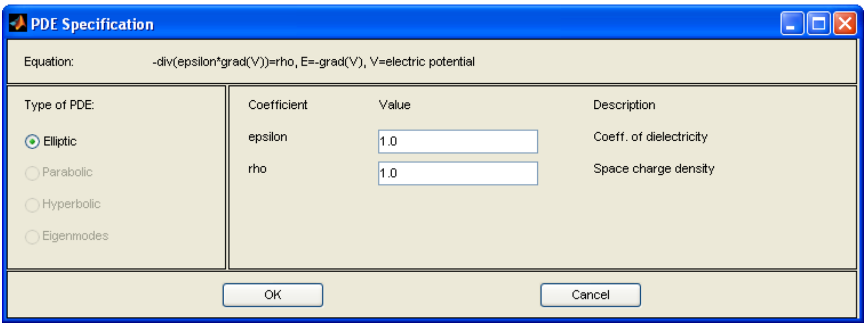
Отредактируйте формулу, которая используется при создании формы 2-D области.



При создании формы 2-D области объекты могут быть объединены с помощью строки формул. Каждому объекту автоматически присваивается уникальное имя, которое отображается в графическом режиме на самом объекте. При наборе формул с помощью имен ссылаются на объекты, т. е. в заданной формуле название указывает на множество точек внутри объекта. В результате формой 2-D области является множество точек, удовлетворяющих формуле. По умолчанию формой 2-D области является объединение всех объектов.



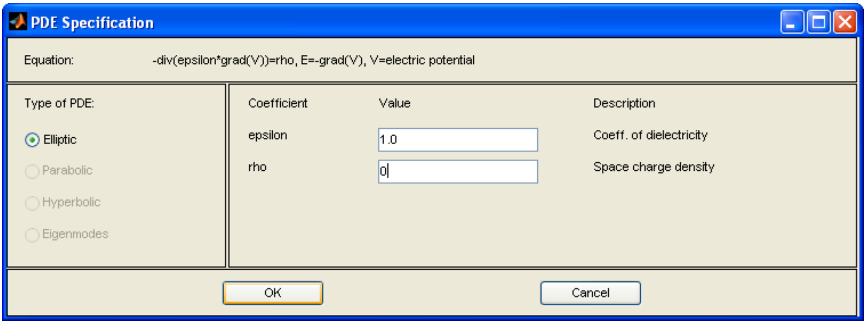
<u>В *режиме PDE*</u> указываем параметры уравнения в частных производных: **Coeff. of dielectricity** (Коэффициент диэлектрической проницаемости**)** и **Space charge density** (Объемная плотность заряда)[1], нажав кнопку PDE на панели инструментов или выбрав в меню **PDE** ⇨ **PDE Specification…**. Появится следующее диалоговое окно:



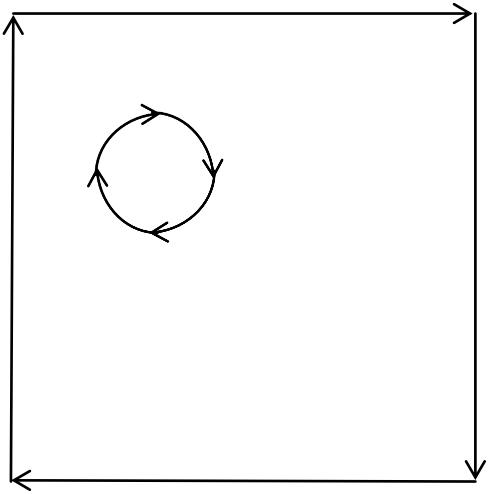
В области, свободной от зарядов, потенциал электростатического поля удовлетворяет уравнению Лапласа, поэтому при моделировании подобных полей объемная плотность заряда (Space charge density) rho равна нулю.

---

[1] В отличие от привычного для студентов выражения $\Delta U = -4\pi\rho$ в программе используется система единиц без множителя 4π перед объемной плотностью заряда.



<u>В режиме *Boundary*</u> задаем граничные условия. Выберите режим границ, нажав кнопку ∂Ω на панели инструментов или выбрав в меню **Boundary ⇨ Boundary Mode** (переключение в режим ввода граничных условий, все границы будут окрашены красным цветом и возле каждой границы появится стрелка).



На разных границах можно задать различные граничные условия, дважды щелкнув левой кнопкой мыши на каждой из границ[2].

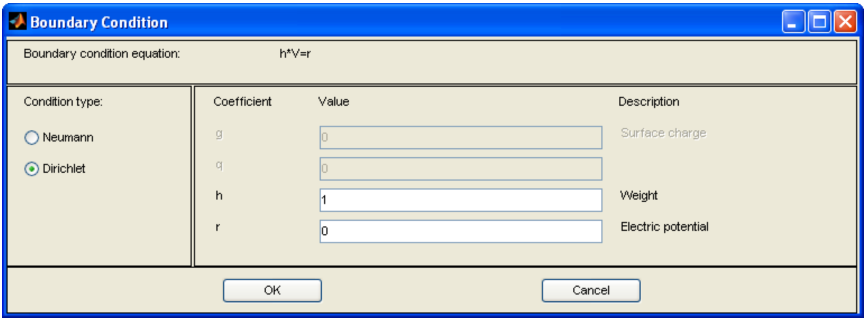
Выберите границу для задания граничных условий (граница чернеет, когда выбрана), а затем меню **Boundary ⇨ Specify Boundary Conditions…** или двойной щелчок по границе (ввод параметров граничных условий).

---

[2] Можно выделить несколько границ (если они имеют одинаковые граничные условия), удерживая клавишу **Shift**, или все границы одновременно, нажав клавиши **Ctrl/A**.

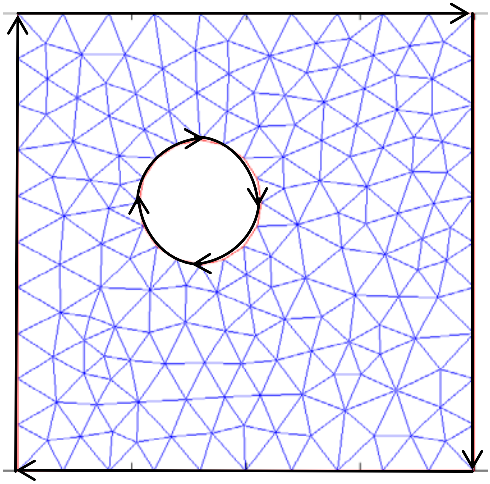
По данной команде раскрывается диалоговое окно, предлагающее ввести нужные значения параметров.

Существуют два основных типа граничных условий: условие Дирихле $h \cdot U = r$ на границе $\partial\Omega$ и условие Неймана $\vec{n} \cdot (c\nabla U) + qU = g$ на границе $\partial\Omega$. Выбрав условие Дирихле, вы должны установить коэффициент h равным единице **(Weight)**, а r – значение потенциала **(Electric potential)** на границе области (граничные условия).



Задайте такие граничные условия, чтобы в двухмерной области было однородное электрическое поле.

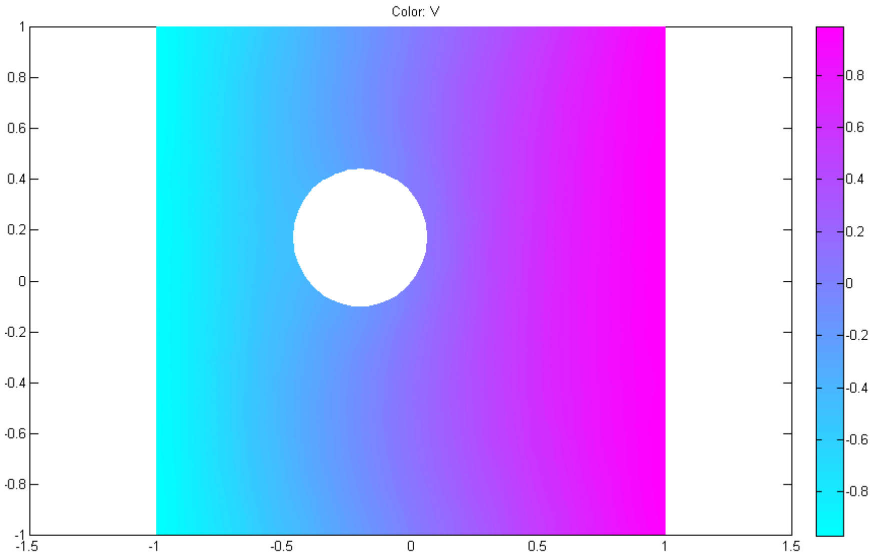
В *режиме Mesh* можно управлять построением сетки конечных элементов.



Первоначальную сетку можно сгенерировать, нажав кнопку  на панели инструментов или выбрав меню **Mesh ⇨ Initialize Mesh**, после чего область разбивается на треугольники. Поле считается в узлах сетки и в дальнейшем между узлами сетки интерполируется.

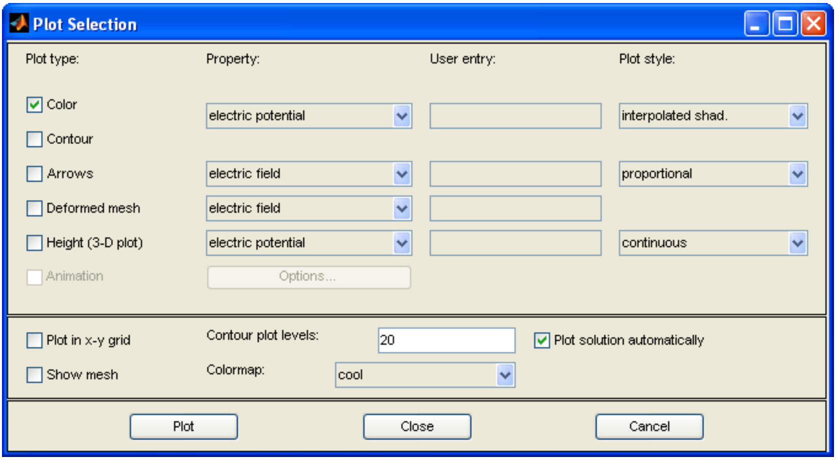
Первоначальную сетку можно уточнить (детализировать), выбрав меню **Mesh ⇨ Refine Mesh** или нажав кнопку  на панели инструментов. Уточняем сетку до тех пор, пока не получим более точное решение задачи. Сетка должна быть детализирована до такой степени, чтобы регулярный массив сетки элементами охватывал нарушения в форме 2-D области («резолюция»), но не настолько, чтобы множество новых элементов ненамного улучшало решение при значительном увеличении времени, которое требуется для расчетов (точка «убывающего плодородия почвы»). Вы также можете «встряхнуть» сетку **Mesh ⇨ Jiggle Mesh**, чтобы получить оптимальную композицию треугольных элементов. Если при уточнении (детализации) сетки ошибочно была сгенерирована слишком мелкая сетка, вернитесь к первоначальному варианту, нажав кнопку  на панели инструментов или выбрав меню **Mesh ⇨ Initialize Mesh**.

<u>В режиме Solve</u> решается уравнение в частных производных. Решите уравнение, нажав кнопку  на панели инструментов, или выбрав меню **Solve ⇨ Solve PDE**. По умолчанию решение строится по осям PDETool.

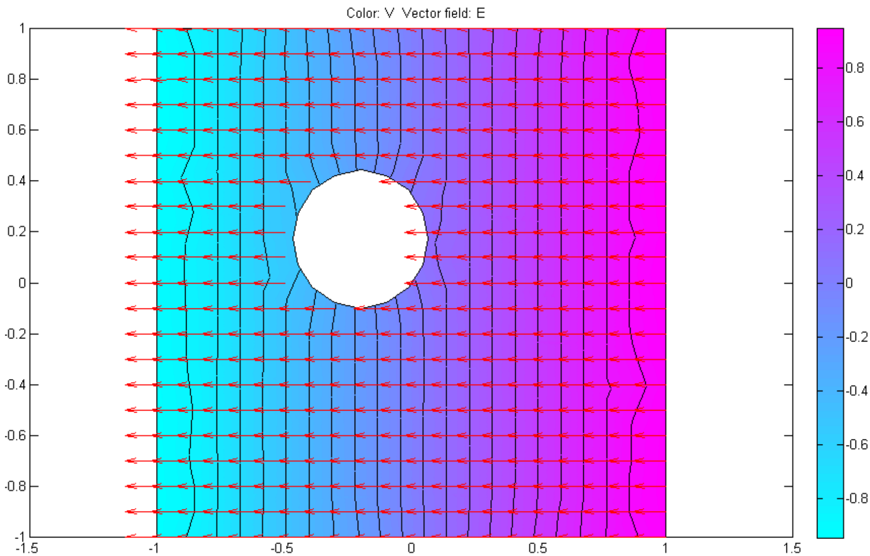


<u>В режиме Plot</u> можно выбрать различные методы визуализации: поверхность, сетка, контур, стрелки (векторное поле). Для графика поверхности можно выбирать между интерполяцией и плоскими схемами визуализации. Сетка может быть спрятана. Для параболических и гиперболических уравнений можно анимировать процесс изменения решения с течением времени. Можно показать решение как 2-D в окне

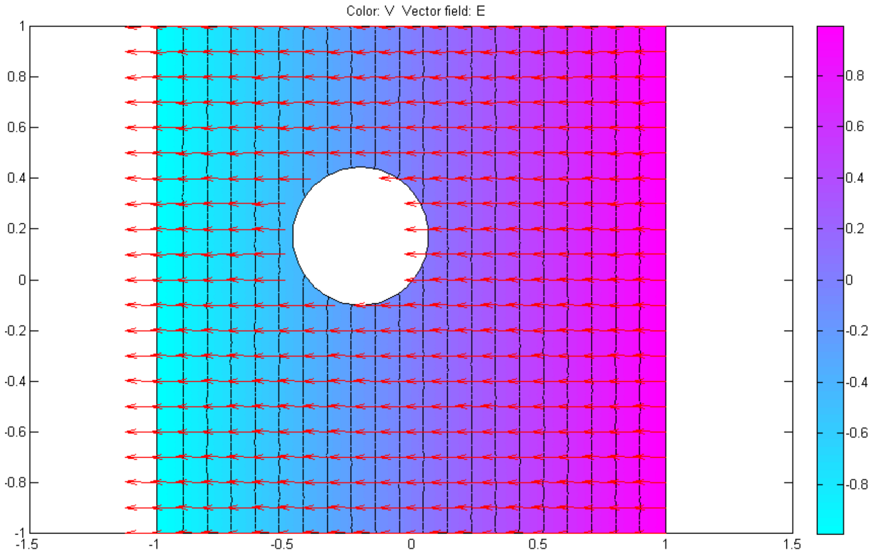
PDETool, так и 3-D в отдельном окне. Различные типы решений можно выбрать, нажав на кнопку  на панели инструментов или путем выбора меню **Plot ⇨ Parameters…** Откроется диалоговое окно **Plot Selection**, в котором можно выбрать различные графические представления решения и их свойства.



Например, поставьте флажки в полях: Contour (контур (эквипотенциали)) и Arrows (стрелки (векторное поле)).



Видно, что необходимо уточнить сетку, поскольку эквипотенциали и векторное поле, отображающие однородное поле, непараллельны. После трехкратного уточнения сетки поле выглядит так:



Вы можете вернуться, изменить форму 2-D области, уравнение или граничные условия.

Теперь у вас есть почти все, что необходимо, чтобы приступить к решению задач формирования электростатических полей в области, свободной от зарядов с помощью проводящих электродов, находящихся под заданными потенциалами, на Matlab.

После ознакомления с приемами работы пакета PDE можно начать решать задачи. В нашем задачнике приведено 15 задач.

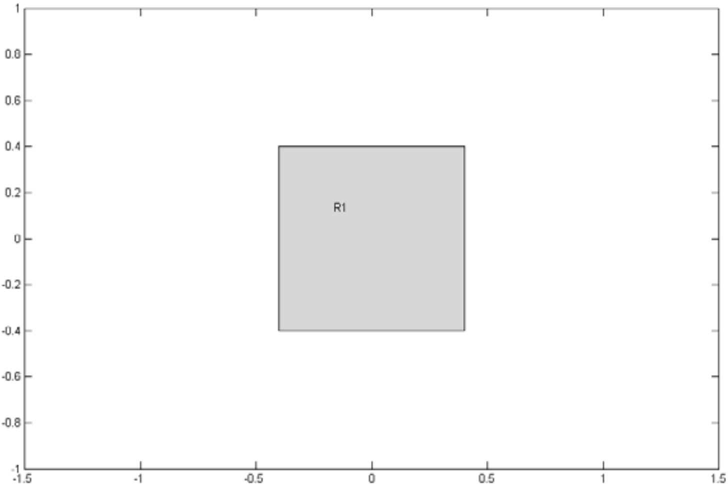
3. Примеры задач

Задача 1. Задайте граничные условия для вертикального и «наклонного» однородного поля и убедитесь в однородности.

Задача 2. Внутри квадрата сформируйте поле электростатической квадрупольной линзы: $E_x = ky$, $E_y = kx$ путем надлежащего расчетного выбора граничных условий для U $(U_{гр} = ?)$ (Приложение 1, задачи 7.18, 7.19).

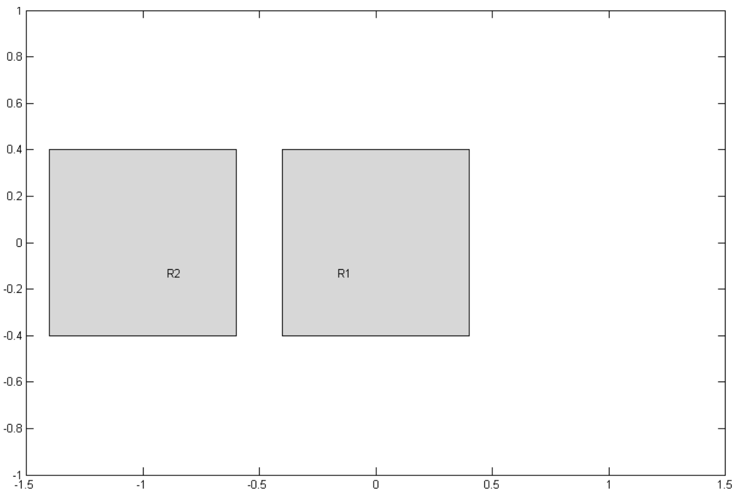
> ***Указание.*** 2-D область разместите так, чтобы ее центр совпадал с началом координат (для удобства вычислений).

При поэлементном делении, умножении и возведении в степень матрицы на матрицу необходимо перед знаком /,* и ^ ставить точку «**.**» (правило MATLAB для выполнения поэлементных операций) [4].



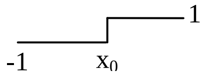
Задача 3. Задайте на сторонах квадрата граничные условия $U_л = U_п = +1$, $U_н = U_в = -1$, сравните поле с решением п. 2. Объясните поведение U и $\vec{E}$ в окрестности центра квадрата и вблизи его вершин.

Указание. 2-D область разместите левее или правее квадрупольной линзы, сравнивать будет удобнее.



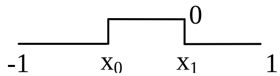
Задача 4. а) Задайте на сторонах квадрата ненулевые граничные условия ($U_л = U_п = +1$, $U_н = U_в = -1$) не на всей длине сторон, а лишь в средних их частях и исследуйте вид поля в зависимости от ширины этих «электродов».

Указание. Воспользуйтесь функцией sign.
Значение *sign(x)* равно *1,* если $x > 0$; *0,* если $x = 0$ и *-1,* если $x < 0$, т. е. задав на границе функцию $sign(x - x_0)$, получим ступеньку



,

если зададим функцию $sign\,(x - x_0) - sign\,(x - x_1)$, получим ступеньку



.

б) Исследуйте поведение U и $\vec{E}$ в окрестности точки, где граничные значения потенциала испытывают скачок.

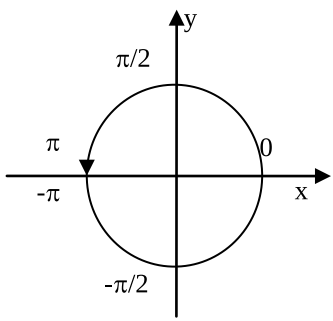
Указание. Воспользуйтесь меню **Plot ⇨ Parameters…** Откроется диалоговое окно **Plot Selection**, в котором можно выбрать 3-D представление решения: **Electric field** или **Electric potential**.

Задача 5. Задайте такое чередование значений U на границах, чтобы в окрестности центра квадрата возникло поле $E_x \propto x^2$ (и вообще $E_x \propto x^n$).

Указание. В двумерном случае решение уравнения Лапласа можно записать в виде:

$$U(r,\alpha) = \sum_{m=1}^{\infty} a_m \cdot r^m \cdot cos(m\alpha),\text{ где } r \text{ и } \alpha \text{ – полярные координаты.}$$

2-D область задайте окружностью единичного радиуса.
Угол определяется с помощью функции
$\alpha = a\,tan(y/x)$
(угол α ограничен интервалом $[-\pi/2, \pi/2]$)
или функции $\alpha = a\,tan2(y,x)$
(угол α ограничен интервалом $[-\pi, \pi]$).



Убедитесь, что $E_x \propto x^n$, проанализировав 3-D решение для потенциала и поля.

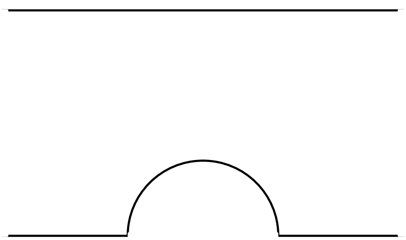
Задача 6. Найдите потенциал и электрическое поле тонкой равномерно заряженной по объему нити. Сравните полученное численное решение с известным аналитическим решением внутри и снаружи нити (Приложение 1, задача 1.47б).

Указание. Решайте уравнение Пуассона в 2-D области единичного радиуса. Расположенную в центре области равномерно заряженную нить радиусом ~0,1 задайте с помощью функции sign при определении коэффициентов уравнения Пуассона.

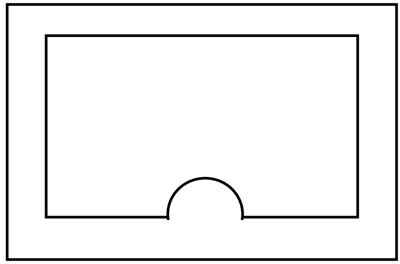
Задача 7. Найдите потенциал электрического поля, создаваемый металлическим цилиндром (ось которого перпендикулярна направлению поля), помещенным во внешнее однородное поле. Сравните с аналогичной задачей для шара (Приложение 1, задача 1.51).

Задача 8. Найдите поле диэлектрического цилиндра (ось которого перпендикулярна направлению поля), помещенного во внешнее однородное поле (Приложение 1, задача 2.8а).

Задача 9. Найдите поле внутри конденсатора, если на одной из пластин образовалась проводящая цилиндрическая вмятина.

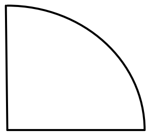
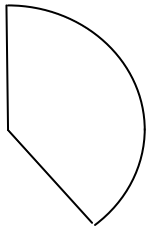


Задача 10. Найдите потенциал и поле в конденсаторе, состоящем из двух коробов, во внутреннем коробе вмятина цилиндрической формы.

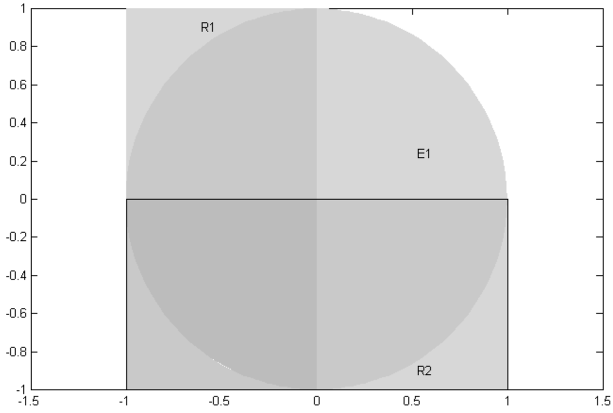


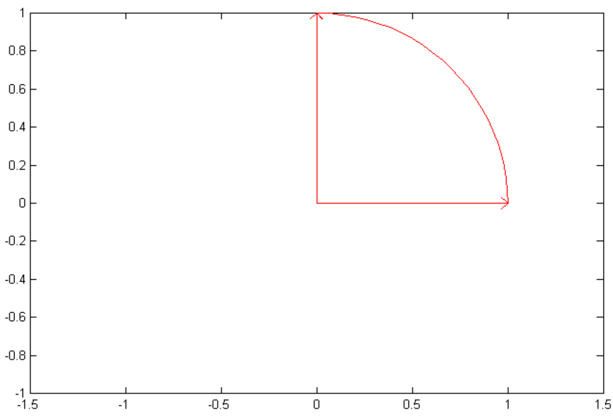
Задача 11. Задайте потенциал на нижней границе $U = U_0 \cdot sin\alpha x$, а на других $U = 0$ (Приложение 1, задача 1.49а).

Задача 12. Найдите потенциал в двугранном угле (90^0 и 135^0) между двумя эквипотенциальными металлическими полуплоскостями с потенциалами U_1 и U_2 .

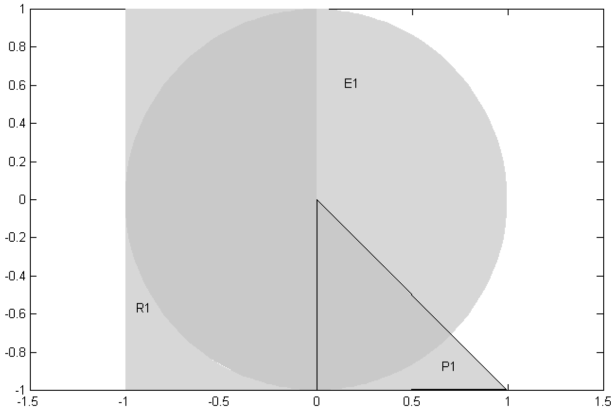
а)  б) 

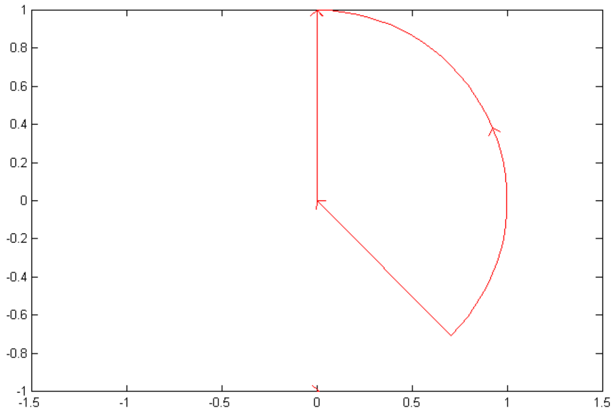
Указание. а) 2-D область: из круга вычитаются два прямоугольника.



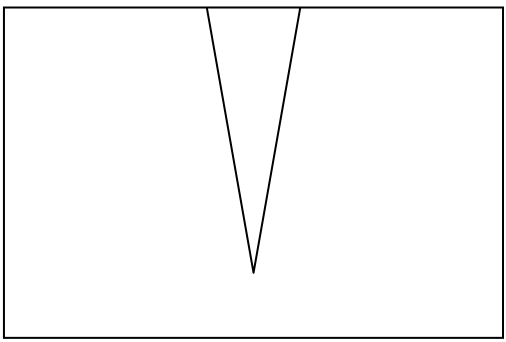


б) 2-D область: из круга вычитаются прямоугольник и треугольник.



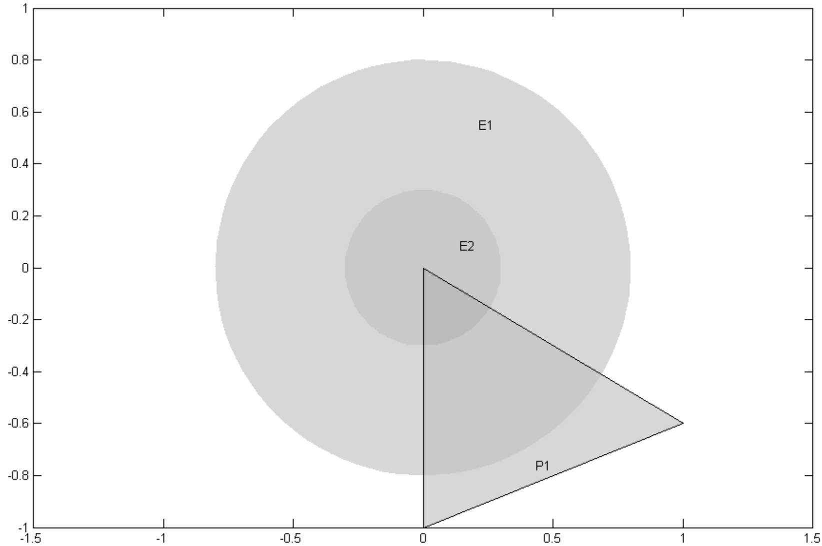


Задача 13. Найдите потенциал тонкого клина («игла в коробке»).

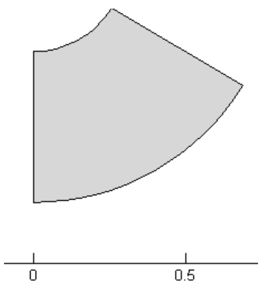


Задача 14. Найдите потенциал и поле в цилиндрическом конденсаторе, с внутренним радиусом a и внешним b, у которого часть пространства между электродами (сектор с углом α) заполнена диэлектриком с диэлектрической проницаемостью ε.

Указание. 2-D область – из круга вычитается круг и умножается на треугольник (операция пересечения).

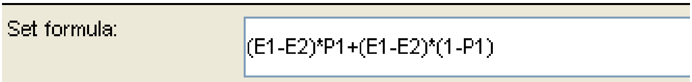


В результате получим сектор кольца.

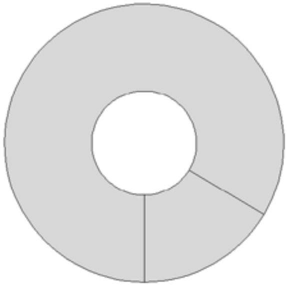


Эта область заполнена диэлектриком с диэлектрической проницаемостью ε.

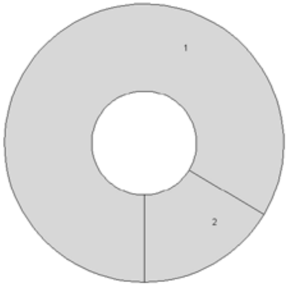
Для получения оставшейся части кольца умножаем вышеупомянутую разность на дополнение к треугольнику (1 – P1) и полученные две части формы складываем.



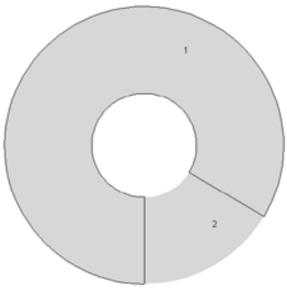
В результате получаем кольцо, разбитое на две зоны.



Можно показать номера зон **PDE ⇨ Show Subdomain Labels**.



Переключитесь в режим ввода параметров (коэффициентов) уравнения в частных производных **PDE ⇨ PDE Mode** и выделите 1-ю зону.

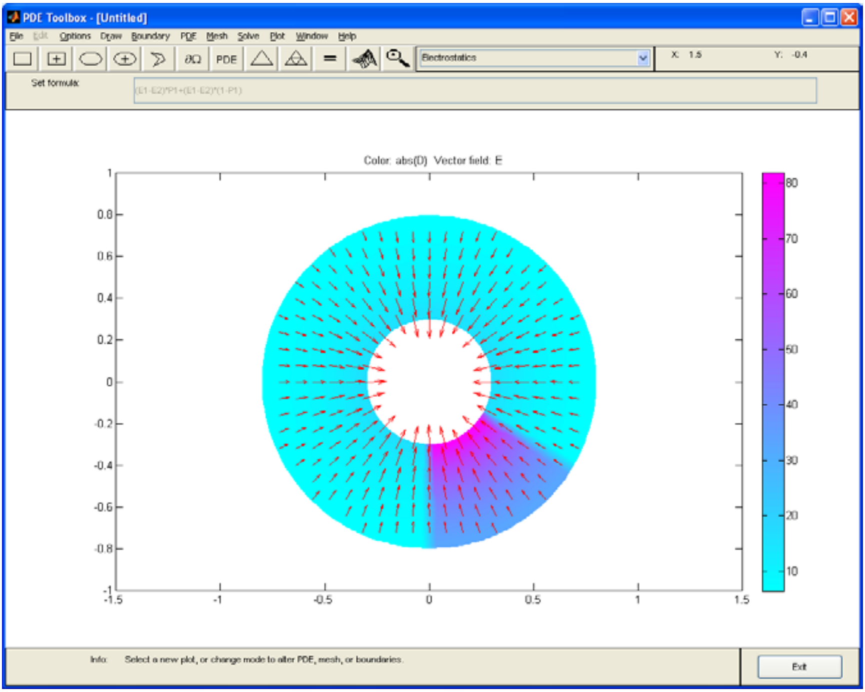


Введите параметры **PDE ⇨ PDE Specification…**

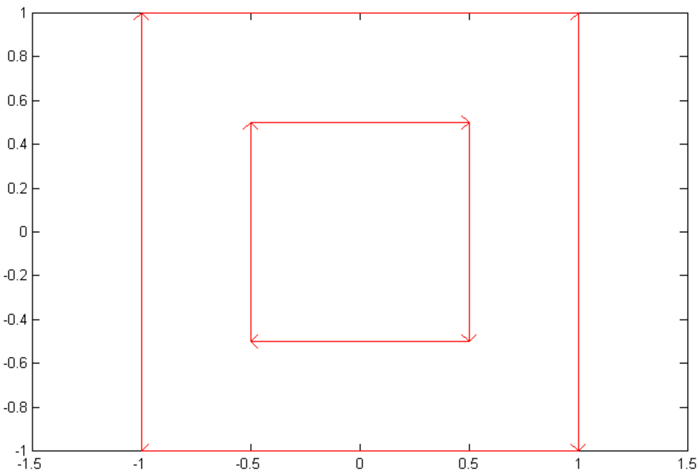
Аналогично выделите и введите PDE параметры для зоны 2.

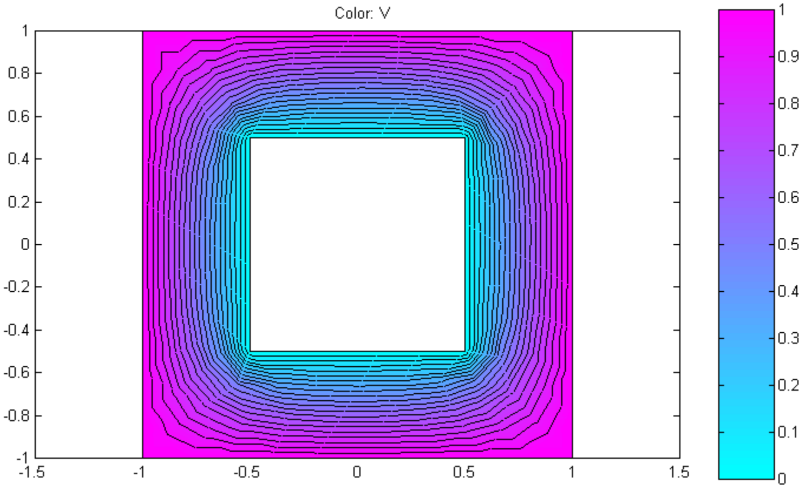
Переключитесь в режим ввода граничных условий **Boundary ⇨ Boundary Mode**, выделите границы и задайте граничные значения **Boundary ⇨ Specify Boundary Conditions…**

Решите уравнение и настройте графику.



Задача 15. Вычислите ёмкость конденсатора, образованного внутренним прямоугольным электродом и окружающим его прямоугольным экраном. Сравните вычисленные значения с оценкой, сделанной «вручную» с пренебрежением краевыми эффектами.





Решение, полученное в результате численного расчета, может быть использовано для выполнения дополнительных расчетов. Конечно, можно написать программу, которая, используя функции пакета PDE, найдет требуемое решение и вычислит на его основании все требуемые параметры. В настоящем практикуме это не предполагается. Наша задача – используя готовый графический интерфейс, решить поставленную задачу и определить требуемые параметры. Не всегда достаточно вычислить и нарисовать вид силовых линий или линий постоянного потенциала. В некоторых случаях требуется найти определенные характеристики исследуемого поля или объекта. Одним из таких примеров является вычисление ёмкости системы «труба в трубе».

Как известно, ёмкость системы из двух проводников определяется отношением заряда на одном из объектов (например, на наружной трубе), к разности потенциалов между этими цилиндрами. Что касается разности потенциалов, то проблем не возникает: задание потенциалов на граничных элементах входит в постановку нашей задачи. Определение заряда, который в полученном случае образуется на поверхности, – это нетривиальная задача. Поскольку заряд на эквипотенциальной поверхности пропорционален нормальной производной к соответствующей поверхности, то после получения численного решения (потенциала во всех точках заданной сетки) необходимо взять численную производную. Поскольку решение известно только с одной стороны от границы да еще в точках, отстоящих от границы на разных расстояниях, то вычисление нормальной производной выполняется с большой погрешностью. Желающие могут это проверить самостоятельно. Найдя однородное поле в прямоугольнике, попробуйте найти нормальную производную к границе и определить распределение заряда на границе.

К счастью, для вычисления ёмкости системы есть более надежный и точный метод. Вспомним, как определяется энергия системы. С одной стороны, она равна

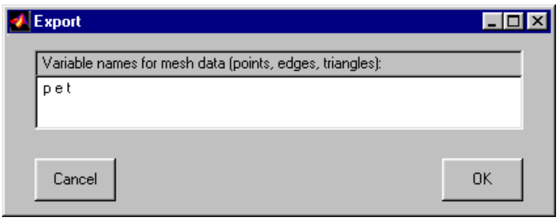
$$W = \frac{CU^2}{2},$$

а с другой стороны, энергия такой системы равна интегралу по пространству от плотности энергии, т. е.

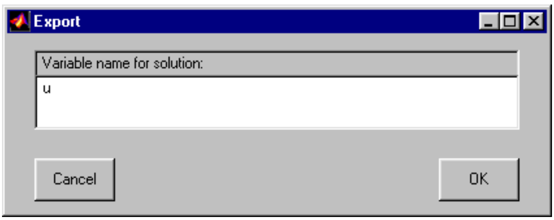
$$\frac{CU^2}{2} = \int \frac{E^2}{8\pi} dV .$$

Из этого равенства следует такой алгоритм: по полученному сеточному решению находим полную энергию системы, заменяя интегрирование на суммирование по всем узлам сетки с учетом размеров каждой ячейки сетки, после чего по формуле энергии определяем ёмкость. Поскольку такое действие в графическом интерфейсе не заложено, то его надо сделать самостоятельно. Для этого необходимо получить из решения задачи собственно значение потенциала во всех узлах сетки и описание самой сетки.

Для получения информации о сетке используется пункт меню **Mesh ⇨ Export mesh**. После вызова этого пункта в рабочей области Matlab появятся три массива: **p, e, t** (массивы характеризующие конечноэлемент-ную сетку).



После этого необходимо сохранить в рабочей области само решение. Для этого используется пункт меню **Solve ⇨ Export solution**. В рабочем пространстве появится массив значений потенциала **u**.



Теперь можно воспользоваться функцией, которая вычисляет электрическое поле и интеграл от плотности энергии, после чего вычисляется ёмкость. Эта функция называется **capas(p, t, u, V)**. Параметры **p, t, u** берутся из рабочей области, а величина разности потенциалов между обкладками конденсатора задается значением параметра **V**.

В окне Matlab вызовите программу capas с параметрами

```
>> capas(p, t, u, 1).
```

Результатом вычисления является ёмкость единицы длины трубы

```
ans =
   0.8337,
```

Текст программы:

```
function c=capas(p,t,u,V);

%Вычисление ёмкости по решению уравнения Лапласа
%Вычисление поля E

[Ex,Ey]=pdegrad(p,t,u);
Ex2=Ex.^2;
Ey2=Ey.^2;
w=(Ex2+Ey2)/(8*pi);
[ar,a1,a2,a3]=pdetrg(p,t);
W=w*ar';
c=2*W/V^2;
```

Установите знак и порядок величины поправок, вносимых краевыми эффектами при оценке ёмкости, изучая случай расположения внутреннего электрода вблизи стенки экрана.

Приложение 1. Задачи

Все задачи в приложении взяты из [7] с сохранением нумерации задачника.

Задача 1.47б. Используя уравнение Пуассона, симметрию задачи, конечность и непрерывность потенциала и его производной, найти потенциал цилиндра радиуса R равномерно заряженного по объему с линейной плотностью κ.

Ответ: $U = \kappa \cdot \left(1 - \dfrac{r^2}{R^2}\right)$ при $r < R$ и

$$U = -2 \cdot \kappa \cdot ln\frac{r}{R} \text{ при } r > R.$$

Задача 1.49. Плоскость $z = 0$ заряжена с плотностью $\sigma(x,y) = \sigma_0 \cdot sin(\alpha x) \cdot sin(\beta y)$, где σ_0, α, β – постоянные. Найти потенциал этой системы зарядов.

Задача 1.51. Незаряженный металлический шар радиуса R вносится в электрическое поле, которое при отсутствии шара было однородным и равным E_0. Определить результирующее поле и плотность поверхностных зарядов на шаре. Что изменится, если заменить шар цилиндром, ось которого перпендикулярна полю?

Задача 2.8а. Однородный шар радиуса a с диэлектрической проницаемостью ε_1 погружен в однородный неограниченный диэлектрик ε_2. На большом расстоянии от шара в диэлектрике имеется однородное электрическое поле E_0. Найти потенциал и напряженность электрического поля во всем пространстве, а также распределение связанных зарядов на шаре и его поляризованность.

Задача 7.18. Показать, что фокусное расстояние F для пучка заряженных частиц в параксиальном приближении для тонкой электростатической квадрупольной линзы с потенциалом

$$U = U_0 (x^2 - y^2) / h^2$$

удовлетворяет соотношениям

$$F_x^{-1} = p \cdot sin(pl) \approx p^2 l ,$$

$$F_y^{-1} = -p \cdot sh(pl) \approx -p^2 l ,$$

где $p = \frac{1}{h} \cdot \sqrt{\frac{qU}{mv_0^2}}$.

Задача 7.19. Показать, что для системы двух последовательных квадрупольных линз, расположенных на расстоянии d, имеющих фокусные расстояния $F_x = F$, $F_y = -F$ и повернутых в плоскостях X, Y друг относительно друга на 90^0, результирующие фокусные расстояния по осям X и Y одинаковы и равны $\sim F^2 / d$ при $d << F$.

Приложение 2. Интерфейс пакета PDETool

Меню

File (Файл)

New (Ctrl + N) – создать новую PDE-модель. Область координатных осей очищается, переменные, описывающие PDE-модель, освобождаются. Если в это время была открыта несохранённая PDE-модель, то раскроется диалоговое окно, предлагающее сохранить её.

Open… (Ctrl + O) – открыть ранее сохранённую в m-файле PDE-модель. Перед открытием файла будет развёрнута стандартная диалоговая панель открытия файла.

Save (Ctrl + S) – сохранение в виде m-файла открытой PDE-модели. Открытая PDE-модель сохраняется под текущим именем. Существующий m-файл заменяется.

Save As… (сохранить как) – сохранение в виде m-файла открытой PDE-модели. Перед сохранением раскроется стандартная диалоговая панель записи файла.

Print… – печать фигуры PDETool.

Exit (Ctrl + W) – закрытие приложения PDETool. Если в это время была открыта несохранённая PDE-модель, то раскроется диалоговое окно, предлагающее сохранить её.

Edit (Редактирование)

Undo (Ctrl + Z) – отмена последнего действия.

Cut (Ctrl + X) – вырезать выделенный геометрический объект и поместить его в буфер.

Copy (Ctrl + C) – копировать выделенный геометрический объект в буфер.

Paste (Ctrl + V) – вставить геометрический объект из буфера.

Clear (Ctrl + R) – удалить выделенный геометрический объект.

Select All (Ctrl + A) – выделить все геометрические объекты.

Options (Возможности)

Grid – показать / скрыть координатную сетку в объекте Axes.

Grid Spacing – установить пределы и шаг координатной сетки. Возможен произвольный неравномерный шаг.

Snap – округлять координаты указателя мыши при показе их значений пользователю. В правой части панели инструментов имеется область отображения значений координат *x* и *y* точки, в которой находится указатель мыши. Если режим Snap включён, то отображаемые координаты округляются до ближайших значений координат линий сетки. Иначе выдаётся по четыре значащие цифры каждой координаты.

Axes Limits – установить значения пределов координатных осей. По данной команде разворачивается диалоговое окно, предлагающее ввести нужные значения.

Axes Equal – установить на экране одинаковый масштаб по осям *x* и *y*.

Turn Off Toolbar Help – отключить выдачу подсказок по кнопкам инструментальной панели PDETool. В нижней левой части фигуры PDETool имеется область подсказок. Если указатель мыши подвести к кнопке инструментальной панели, то в названной области будет выдана подсказка о назначении этой кнопки (при выключенном режиме Turn Off Toolbar Help).

Zoom – включить режим показа в увеличенном масштабе выделяемой прямоугольной области в PDE-модели. Если при включенном режиме Zoom выделить мышью некоторую прямоугольную область, то сразу при отпускании левой кнопки мыши эта область будет показана в увеличенном масштабе (Axes Limits будут соответствовать границам этой области).

Application – переключение типа PDE-задачи (список краевых задач, поддерживаемых приложением PDETool). По данной команде список отображается в виде подменю.

Refresh – обновить изображение PDE-модели в поле axes. Если изображение PDE-модели в поле axes по каким-либо причинам было испорчено, то по данной команде оно отображается снова.

Draw (Прорисовка)

Draw Mode – переключение в режим ввода формы 2-D области (прорисовки).

Rectangle/square – ввод прямоугольника или квадрата с помощью мыши. Исходной точкой прямоугольника является его верхняя левая вершина. Если при вводе будет удерживаться клавиша Ctrl, то будет прорисовываться квадрат. Стороны прямоугольника всегда параллельны осям координат.

Rectangle/square (centered) – то же. Исходной точкой прямоугольника является его центр.

Ellipse/circle – ввод эллипса или круга с помощью мыши. Исходной точкой эллипса является его верхняя левая точка. Если при вводе будет удерживаться клавиша Ctrl, то будет прорисовываться круг. Главные оси эллипса всегда параллельны осям координат.

Ellipse/circle (centered) – то же. Исходной точкой эллипса является его центр.

Polygon – прорисовка многоугольника. Ввод производится отрезками ломаной линии, до ее замыкания.

Rotate – поворот выделенных геометрических объектов на некоторый угол вокруг некоторой точки. По данной команде раскрывается диалоговое окно, предлагающее ввести угол в градусах и координаты центра поворота. Есть возможность осуществлять поворот относительно центра масс. Положительное направление поворота – против часовой стрелки.

Export Geometry Description, Set Formula, Labels… – экспорт в базовую рабочую область переменных описания формы 2-D области.

Boundary (Граница)

Boundary Mode (Ctrl + B) – переключение в режим ввода граничных условий.

Specify Boundary Conditions… – ввод параметров граничных условий. По данной команде раскрывается диалоговое окно, предлагающее ввести нужные значения параметров.

Show Edge Labels – показать номера граничных сегментов.

Show Subdomain Labels – показать номера зон.

Remove Subdomain Border – удалить границу зон.

Remove All Subdomain Borders – удаление всех границ зон.

Export Decomposed Geometry, Boundary Cond's… – экспорт в базовую рабочую область переменных описания граничных условий.

PDE (Partial Differential Equation – уравнение в частных производных)

PDE Mode – переключение в режим ввода параметров PDE.

Show Subdomain Labels – показать номера зон.

PDE Specification… – ввод параметров PDE. По данной команде раскрывается диалоговое окно, предлагающее ввести нужные значения параметров.

Export PDE Coefficients… – экспорт в базовую рабочую область переменных, описывающих распределение PDE-коэффициентов в расчётной области.

Mesh (Сетка)

Mesh Mode – переключение в режим построения конечноэлементной сетки.

Initialize Mesh (Ctrl + I) – инициализация (генерация) конечноэлементной сетки.

Refine Mesh (Ctrl + M) – переопределение (сгущение) конечноэлементной сетки во всей расчётной области.

Jiggle Mesh – регуляризация конечноэлементной сетки. Конечноэлементная сетка перестраивается таким образом, чтобы показатель нерегулярности всех конечных элементов не превышал установленной в PDETool величины.

Undo Mesh Change – отменить последнее изменение конечноэлементной сетки.

Display Triangle Quality – отобразить в цвете значения показателя регулярности для всех конечных элементов (треугольников) сетки.

Show Node Labels – показать номера узлов конечноэлементной сетки.

Show Triangle Labels – показать номера конечных элементов (треугольников).

Parameters… – установить параметры генератора сетки. По данной команде раскрывается диалоговое окно, предлагающее ввести нужные значения параметров.

Export Mesh – экспорт конечноэлементной сетки в базовую рабочую область.

Solve (Решение)

Solve PDE (Ctrl + E) – решить краевую задачу (вычислить узловое распределение искомой величины).

Parameters… – установить параметры PDE. По данной команде раскрывается диалоговое окно, предлагающее ввести нужные значения параметров.

Export Solution… – экспорт решения (узлового распределения) в базовую рабочую область.

Plot (График)

Plot Solution (Ctrl + P) – отобразить решение PDE.

Parameters… – установить параметры отображения решения.

Export Movie… – экспорт в базовую рабочую область переменной, содержащей информацию, необходимую для анимации решения нестационарной PDE-задачи.

Window (Окно)

Переключение между окнами, фигурами и приложениями Matlab. Подменю содержит список соответствующих окон.

Help (Помощь)

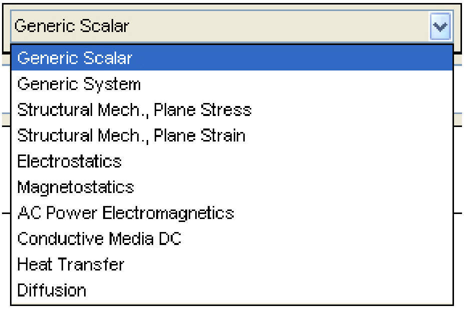
Help… (Ctrl + H) – открыть стандартное окно справочной системы Matlab.

About… – о приложении PDETool.

Панель инструментов

Непосредственно под главным меню находится Панель инструментов.

-  – см. пункт меню **Draw ⇨ Rectangle/square**.
-  – см. пункт меню **Draw ⇨ Rectangle/square (centered)**.
-  – см. пункт меню **Draw ⇨ Ellipse/circle**.
-  – см. пункт меню **Draw ⇨ Ellipse/circle (centered)**.
-  – см. пункт меню **Draw ⇨ Polygon**.
-  – переключение в режим ввода граничных условий, то же, что и **Boundary ⇨ Boundary Mode**. Если мышью сделать двойной щелчок на каком-либо граничном сегменте, то для него выполнится команда **Boundary ⇨ Specify Boundary Conditions…**
-  – ввод параметров (коэффициентов) PDE для всей расчётной области или для выделенных зон, то же, что и **PDE ⇨ PDE Specification…**
-  – инициализация (генерация) конечноэлементной сетки. Выполняется то же действие, что и по команде **Mesh ⇨ Initialize Mesh**.
-  – переопределение (сгущение) конечноэлементной сетки во всей расчётной области. Выполняется то же действие, что и по команде **Mesh ⇨ Refine Mesh**.
- – решить краевую задачу (вычислить узловое распределение искомой величины). Выполняется то же действие, что и по команде **Solve ⇨ Solve PDE**.
-  – установить параметры отображения решения. Выполняется то же действие, что и по команде **Plot ⇨ Parameters…** По данной команде будет развёрнуто диалоговое окно, предлагающее ввести нужные значения параметров. Если нажать кнопку **Plot** будет отображено решение PDE с учётом вновь указанных режимов, окно не закрывается. Нажатие кнопки **Cancel** отменяет изменения параметров и закрывает диалоговое окно.
-  – включить / отключить режим Zoom. Выполняется то же действие, что и по команде **Options ⇨ Zoom**.
- **Ниспадающее меню** – содержит список краевых задач, поддерживаемых GUI-приложением PDETool. Выполняется то же действие, что и по команде **Options ⇨ Application**.



- **Generic Scalar** – скалярные краевые задачи: эллиптические, параболические, гиперболические, задачи на собственные частоты.
- **Generic System** – векторные PDE с неоднородно распределёнными тензорными коэффициентами.
- **Structural Mesh., Plane Stress** – механические напряжения в упругой среде с ортотропными линейными свойствами.
- **Structural Mesh., Plane Strain** – перемещения в упругой напряжённо-деформированной среде с ортотропными линейными свойствами.
- **Electrostatics** – электростатическая задача.
- **Magnetostatics** – магнитостатическая задача.
- **AC Power Electromagnetics** – переменное гармоническое электромагнитное поле.
- **Conductive Media DC** – стационарное электрическое поле в проводящей среде.
- **Heat Transfer** – теплопередача.
- **Diffusion** – уравнение диффузии.

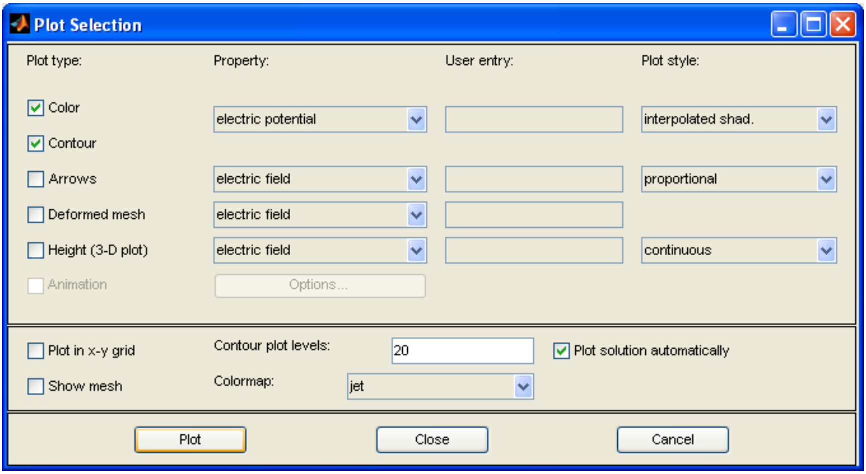
Строка ввода

Строка ввода **Set Formula** должна содержать формулу описания формы 2-D области расчётной области:

По синтаксису это должно быть выражение, операндами которого являются метки (идентификаторы) геометрических объектов. Операция объединения обозначается знаком «+» (плюс), операция исключения – знаком «–» (минус), операция пересечения – знаком умножения «*» (звёздочка).

Визуализация полученного решения

GUI-приложение PDETool переходит в режим визуализации решения по команде меню **Plot ⇨ Parameters…** Картина отображаемого в объекте axes поля зависит от параметров визуализации, которые можно настроить этой командой. Диалоговое окно настройки параметров визуализации имеет вид



В режим визуализации решения также можно перейти и при нажатии кнопки  (Specify PDE solution plot) на панели инструментов.

Диалоговое окно «Plot Selection» состоит из двух панелей. Верхняя панель состоит из четырёх групп:

1) «Plot type»;
2) «Property»;
3) «User entry»;
4) «Plot style».

1. Группа «Plot type» состоит из флажков (CheckBox – флажки используют, когда необходимо выбрать два или более варианта из предложенных), позволяющих указать, что требуется графически отображать:

   - Color – потенциал, модуль поля, модуль индукции как скалярные функции двух переменных (x, y) могут быть представлены в виде цветной карты;
   - Contour – изображение потенциала, модуля поля, модуля индукции изолиниями;

- Arrows – представление векторных полей ($\vec{E} = -grad\,U$, $\vec{D} = \varepsilon \cdot \vec{E}$) стрелками;
- Deformed mesh – отображение поля на деформированной сетке;
- Height (3-D plot) – отображение потенциала, модуля поля, модуля индукции в виде поверхности в отдельном окне;
- Animation – анимация решения нестационарной задачи.

2. Группа «Property» состоит из выпадающих списков, позволяющих выбрать визуализируемое поле:
   - electric potential (*U*) – распределение электрического потенциала;
   - electric field ($\vec{E} = -grad\,U$) – распределение модуля напряжённости электрического поля;
   - electric displacement ($\vec{D} = \varepsilon \cdot \vec{E}$) – распределение модуля электрической индукции;
   - user entry – распределение скалярной величины, интересующей пользователя.
3. Если в соответствующем списке нет требуемого поля, то можно воспользоваться строками ввода в колонке «User entry».
4. Группа «Plot style» содержит выпадающие списки, позволяющие выбрать стили визуализации соответствующих полей:
   - interpolated shad – использование кусочно-линейной интерполяции при отображении;
   - flat shading – отображение в кусочно-постоянном виде;
   - proportional – каждая стрелочка имеет длину, пропорциональную модулю вектора в соответствующей точке наблюдения;
   - normalized – все стрелочки имеют одинаковую длину;
   - continuous – непрерывная интерполяция при построении поверхностного графика;
   - discontinuous – кусочно-постоянный поверхностный график.

Нижняя панель состоит из трёх из флажков (CheckBox), строки ввода и выпадающего списка:

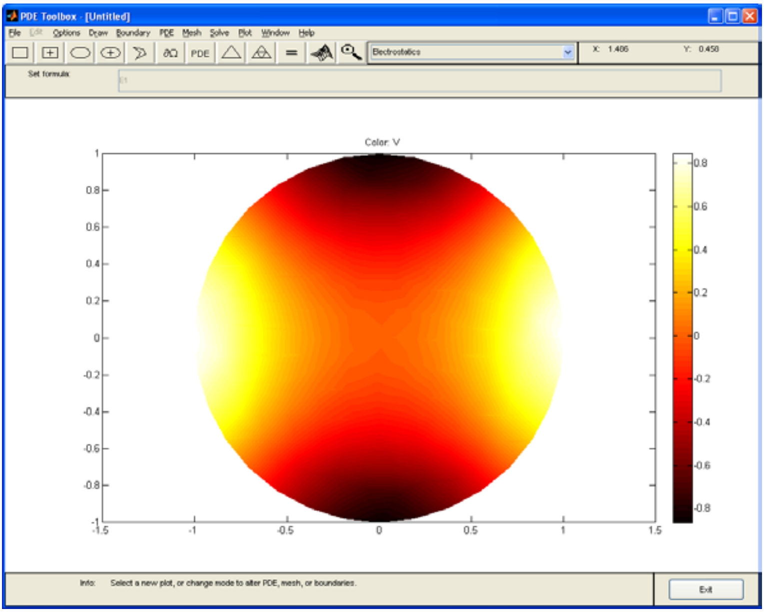
- Plot in x-y grid – построение на x-y сетке;
- Show mesh – показать конечноэлементную сетку;

- Plot solution automatically – автоматическая визуализация решения, получаемого в результате выполнения команды меню **Solve ⇨ Solve PDE** (или кнопка **=** (Solve PDE) на панели инструментов);
- Contour plot levels – число линий равного уровня отображаемого потенциала (модуля поля, модуля индукции);
- Colormap – выбор палитры цветной карты потенциала (модуля поля, модуля индукции).

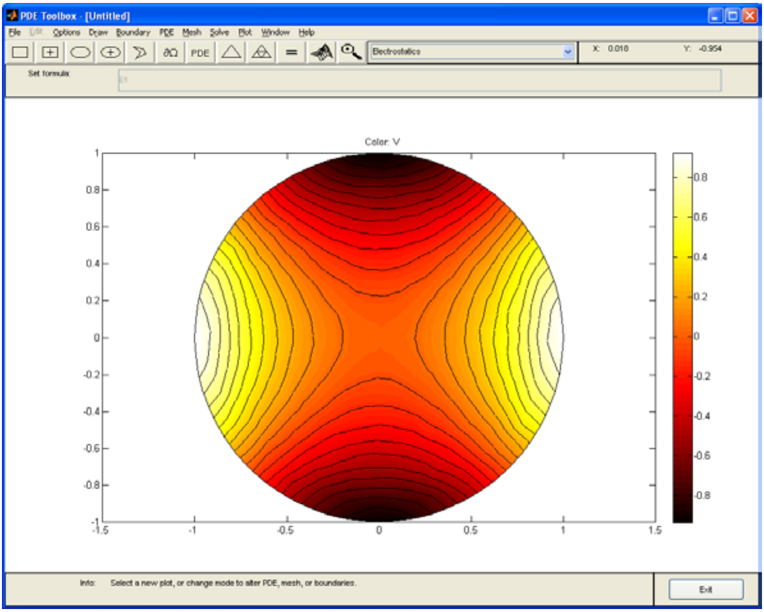
Внизу диалогового окна имеются три кнопки: Plot, Close, Cancel.

- Plot – отобразить решение PDE с учётом вновь указанных параметров, окно не закрывается;
- Close – закрыть диалоговое окно;
- Cancel – отменить изменения параметров и закрыть диалоговое окно.

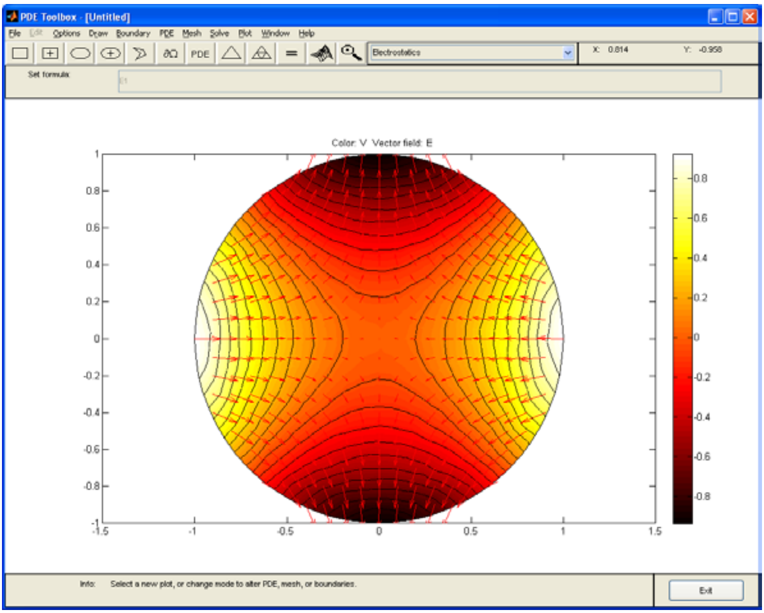
Выполнив команду меню **Solve ⇨ Solve PDE**, получим цветную карту потенциала (в окне «Plot Selection» по умолчанию стоит флажок Plot solution automatically (автоматическая визуализация решения) и в группе «Property» – electric potential).



Если в колонке «Plot type» установить еще флажок «Contour», остальные параметры визуализации оставить без изменения, то карта потенциала приобретет вид



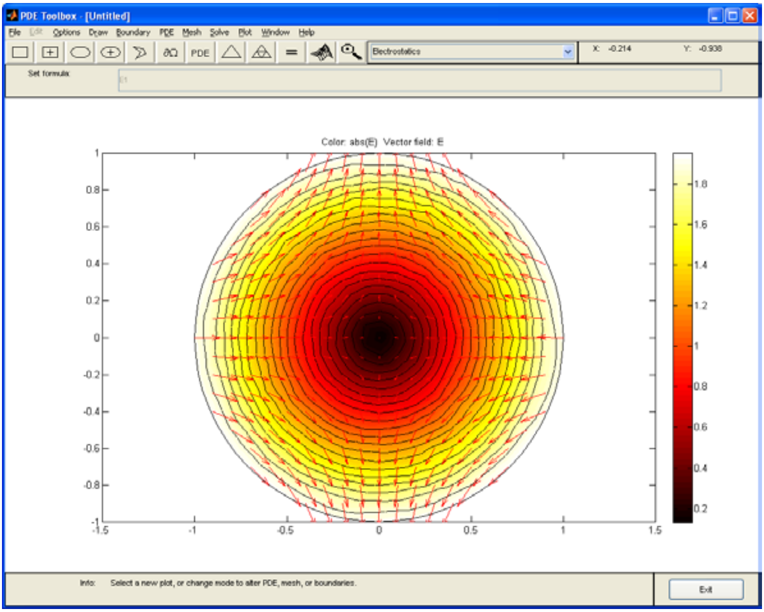
Чтобы показать векторное электрическое поле, нужно установить флажок «Arrows».



Если в группе «Property» выбрать «electric field» или «electric displacement», будет выведен модуль соответствующего векторного поля.

Тип графика «Color» и «Contour» позволяет вывести потенциал, модуль поля, модуль индукции как скалярные функции двух переменных (x, y). «Arrows» дополняет плоскую картину, показывая векторное поле $\vec{E} = -grad\,U$ или $\vec{D} = \varepsilon \cdot \vec{E}$ стрелками.

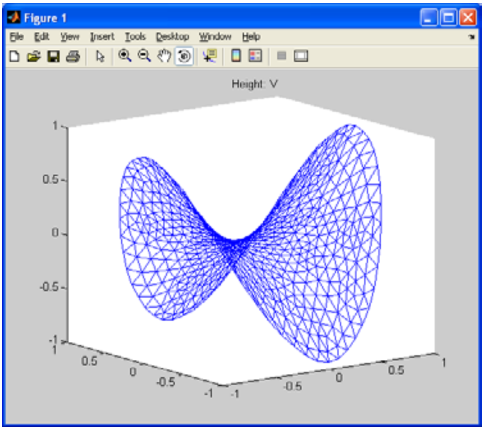
Ниже показана цветная карта модуля напряжённости электрического поля с нанесенными на нее изолиниями и стрелками $\vec{E} = -grad\,U$.



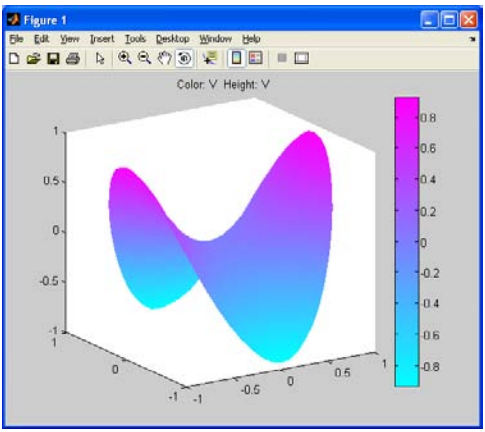
Если в группе «Property» выбран пункт «User entry», то в соответствующую строку ввода в группе «User entry» нужно вписать матричное выражение, описывающее скалярное или векторное поле, которое надо отобразить в поле «axes» фигуры PDETool. Значением этого выражения для **векторного поля** является либо матрица-столбец, либо прямоугольная матрица, состоящая из двух строк. В первом случае столбец должен состоять из $2 \cdot NP$ членов, где NP – число узлов конечноэлементной сетки. Первая половина столбца – узловое распределение ***x***-составляющей векторного поля, вторая половина – узловое распределение ***y***-составляющей. Во втором случае первая строка – узловое распределение ***x***-составляющей векторного поля, вторая строка – узловое распределение ***y***-составляющей.

Для **скалярного поля** это либо матрица-столбец размера (NP , 1), либо матрица-строка размера (1, NE), где NP – число узлов конечноэлементной сетки, NE – число конечных элементов. В случае скалярной краевой задачи любого типа в выражении можно применять следующие переменные: ***u*** – рассчитанный в ходе решения PDE потенциал в виде столбцовой матрицы узлового распределения; ***ux***, ***uy*** – матрицы-строки узлового распределения частной производной по ***x*** и по ***y*** названного выше потенциала ***u***.

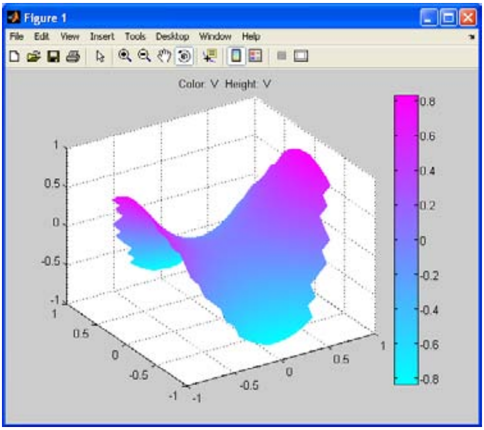
Отображение скалярной функции (потенциала, модуля поля, модуля индукции) в виде трёхмерного графика происходит при установленном флажке «Height (3-D plot)». Если все остальные флажки сброшены, то потенциал имеет вид (в программе электрический потенциал обозначен V)



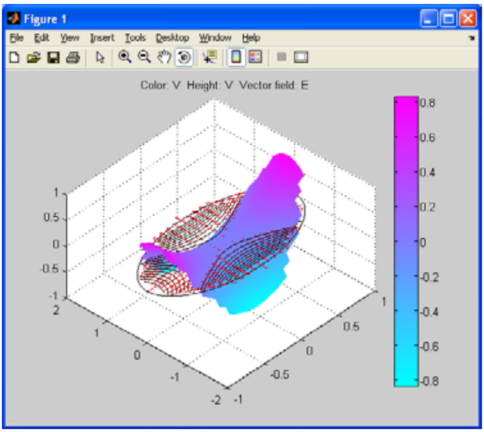
Если при том же решении PDE установить флажок «Color», то потенциал примет вид



Трёхмерный график приобретает наиболее наглядный вид при установленных флажках «Color», «Plot in x-y grid» (действует только при установленном флажке «Height (3D plot)»), «Show mesh» (действует только при установленном флажке «Color»).



При отображении скалярной функции в виде поверхности (Height (3-D plot)), флажки «Contour» и «Arrows» дают возможность изобразить скалярную функцию (потенциал, модуль поля или модуль индукции) изолиниями и векторное поле ($\vec{E} = -grad\,U$ или $\vec{D} = \varepsilon \cdot \vec{E}$) стрелками на плоскости *XY*.

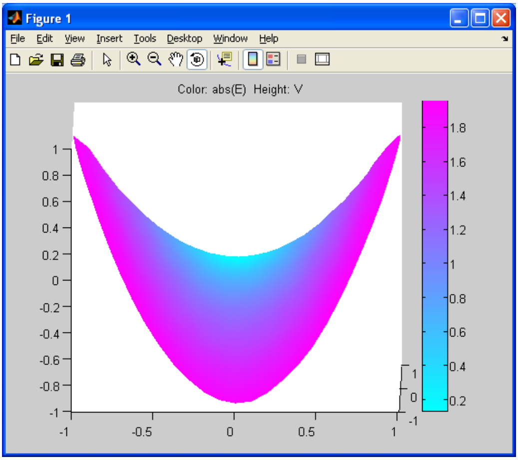


Графическое окно MATLAB – это обычное масштабируемое и перемещаемое окно Windows-приложений.

Меню этого окна похоже на меню окна командного режима работы системы MATLAB. Однако при внимательном просмотре заметен ряд отличий.

Несмотря на множество кнопок, пользование панелью инструментов 3D-графики достаточно просто, если представить себе, что вы смотрите на предмет через объектив фотокамеры. Наглядные рисунки на кнопках поясняют смысл их действия – это перемещение и вращение 3D-рисунков относительно тех или иных координатных осей, вывод легенды и шкалы цветов и др.

Для управления положением и вращением трехмерного графика можно использовать клавиши перемещения курсора. Эффект вращения графика иллюстрирует рисунок, где показан предыдущий график после его поворота при нажатой клавише →.



В отличие от поворота мышью (также возможного) перемещение и повороты с помощью клавиш управления курсором ←,(↑, →, ↓) при выбранном типе перемещения дают плавное перемещение или вращение фигуры. Таким образом, осуществляется анимация (оживление) трехмерной графики.

М-файл

Работа в PDETool может быть сохранена в любой момент (создание формы 2-D области, задание параметров PDE, задание граничных условий, решение уравнения), а затем, при открытии в PDETool соответствующего файла, продолжена, начиная с любого момента или для дальнейшего анализа.

M-файл Matlab содержит команды, необходимые для создания формы 2-D области. Он также может содержать дополнительные команды: задать граничные условия, определить PDE, создать сетку, решить PDE, построить решение. Этот m-файл можно сохранить и открыть из меню File.

M-файл – это функция Matlab, а не сценарий. Во избежание совпадения имени файла с именами переменных, используемых в функции и в основном рабочем пространстве, называйте файл так, чтобы имя файла совпадало с названием модели. Файл похож на представленный во фрагменте ниже:

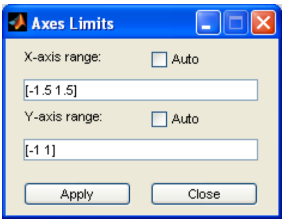
function pdemodel

`[pde_fig,ax]=pdeinit;` {команда запускает PDETool. Если PDETool уже запущен, текущая модель будет заменена новой.}

`pdetool('appl_cb',5);` {выбираем краевую задачу Electrostatics}

{Следующие команды создали масштаб, деления осей PDETool и другие параметры пользователя.}

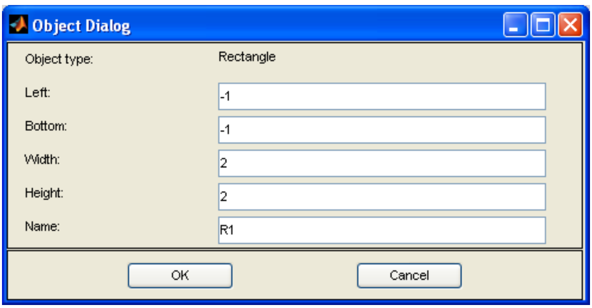
set(ax,'DataAspectRatio',[1 1 1]);
set(ax,'PlotBoxAspectRatio',[1.5 1 1]);
set(ax,'XLim',[-1.5 1.5]);
set(ax,'YLim',[-1 1]);



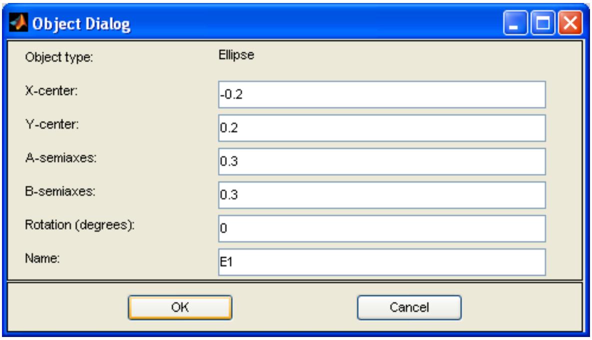
set(ax,'XTickMode','auto');
set(ax,'YTickMode','auto');

`% Geometry description:` {последовательность команд рисования}

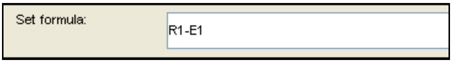
pderect([-1 1 1 -1],'R1');



pdeellip(-0.2,0.2,0.3,0.3,0,'E1');



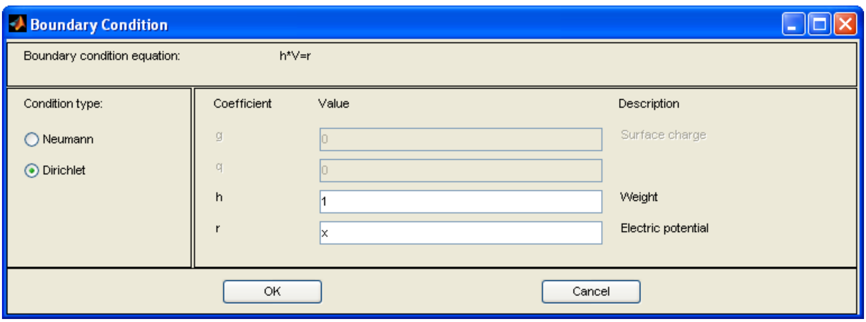
set(findobj(get(pde_fig,'Children'),'Tag','PDEEval'),'String','R1-E1')



{квадрат, с отверстием}

% Boundary conditions: {граничные условия}

pdetool('changemode',0)
pdesetbd(8,'dir',1,'1','x')
pdesetbd(7,'dir',1,'1','x')
pdesetbd(6,'dir',1,'1','x')
pdesetbd(5,'dir',1,'1','x')
pdesetbd(4,'dir',1,'1','x')
pdesetbd(3,'dir',1,'1','x')
pdesetbd(2,'dir',1,'1','x')
pdesetbd(1,'dir',1,'1','x')



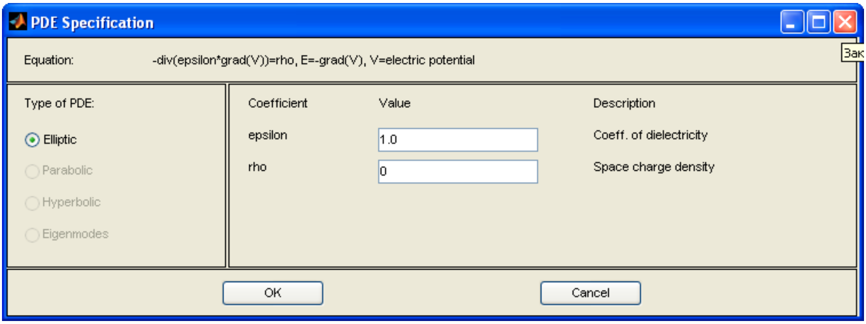
% Mesh generation: {генерация сетки}

setappdata(pde_fig,'Hgrad',1.3);

```
setappdata(pde_fig,'refinemethod','regular');
setappdata(pde_fig,'jiggle',char('on','mean',''));
pdetool('initmesh')
pdetool('refine')
pdetool('refine')
pdetool('refine')
```

% PDE coefficients: {параметры (коэффициенты) PDE}

```
pdeseteq(1,'1.0','0.0','0','1.0','0:10','0.0','0.0','[0 100]')
setappdata(pde_fig,'currparam',['1.0';'0  '])
```



% Solve parameters: {параметры решения PDE}

```
setappdata(pde_fig,'solveparam',...
str2mat('0','30144','10','pdeadworst',...
'0.5','longest','0','1E-4','','fixed','Inf'))
```

% Plotflags and user data strings: {параметры графика}

```
setappdata(pde_fig,'plotflags',[1 1 1 1 1 1 1 1 0 0 0 1 1 1 0 1 0 1]);
setappdata(pde_fig,'colstring','');
setappdata(pde_fig,'arrowstring','');
setappdata(pde_fig,'deformstring','');
setappdata(pde_fig,'heightstring','');
```

% Solve PDE: {решение PDE}

```
pdetool('solve')
```

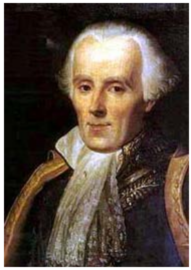
Приложение 3. Обо всем

Симеон Дени Пуассон
(*Poisson, Simeon-Denis*)
(21.06.1781, Питивье – 25.04.1840, Париж)

французский математик, механик и физик

Петер Густав Лежен Дирихле
(Johann Peter Gustav Lejeune Dirichlet)
(13.2.1805, Дюрен – 5.5. 1859, Геттинген)

немецкий математик



Пьер Симон Лаплас
(*Pierre-Simon Laplace*)
(23.3.1749, Бомон-ан-Ож, Нормандия – 5.3.1827, Париж)
французский астроном, математик и физик

Уравнение Пуассона – эллиптическое дифференциальное уравнение в частных производных, которое, среди прочего, описывает

- электростатическое поле;
- стационарное поле температуры;
- поле давления;
- поле потенциала скорости в гидродинамике.

Оно названо в честь знаменитого французского физика и математика Симеона Дени Пуассона.

Это уравнение имеет вид $\Delta\varphi = f$,

где Δ – оператор Лапласа или лапласиан, а f – действительная или комплексная функция на некотором многообразии. Когда в качестве многообразия выступает Евклидово пространство, оператор Лапласа часто обозначается как ∇^2 и уравнение Пуассона принимает вид

$$\nabla^2\varphi = f \ .$$

В трёхмерной декартовой системе координат уравнение принимает вид:

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}\right)\varphi(x,y,z) = f(x,y,z).$$

Если f стремится к нулю, то уравнение Пуассона превращается в уравнение Лапласа (уравнение Лапласа – частный случай уравнения Пуассона):

$$\Delta\varphi = 0 .$$

Эллиптические уравнения – класс дифференциальных уравнений в частных производных.

В общем виде могут быть записаны как $Au = f$, где u – неизвестная функция, A – эллиптический оператор, а f – известная функция пространственных координат.

Пример эллиптического уравнения – уравнение Лапласа.

Эллиптические уравнения противопоставляются параболическим и гиперболическим, хотя данная классификация не является исчерпывающей.

Задача Дирихле – задача отыскания в области D евклидова пространства гармонической функции u, которая на границе ∂D области D совпадает с наперёд заданной непрерывной функцией $\phi : \partial D \to R$. Задачу отыскания регулярного в области решения эллиптического уравнения 2-го порядка, принимающего наперёд заданные значения на границе области, также называют задачей Дирихле, или первой краевой задачей.

Краевая задача – дифференциальное уравнение (система дифференциальных уравнений) с заданными линейными соотношениями между значениями искомых функций на начале и конце интервала интегрирования.

Метод конечных элементов (*МКЭ*) – численный метод решения задач, широко используемый для решения задач механики, теплообмена, гидродинамики и электромагнитных полей [8]. 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.

Возникновение метода конечных элементов связано с решением задач космических исследований в 1950-х годах (идея МКЭ была разработана советскими учёными ещё в 1936 году, но из-за неразвитости вычислительной техники метод не получил развития). Этот метод возник из строительной механики и теории упругости, а уже затем было получено его математическое обоснование. Существенный толчок в своём развитии МКЭ получил в 1963 году после того, как было доказано то, что его можно рассматривать как один из вариантов распространённого в строительной механике метода Рэлея – Ритца, который путём минимизации потенциальной энергии сводит задачу к системе линейных уравнений равновесия. После того, как была установлена связь МКЭ с процедурой минимизации, он стал применяться к задачам, описываемым уравнениями Лапласа или Пуассона. Область применения МКЭ значительно расширилась, когда было установлено (в 1968 году), что уравнения, определяющие элементы в задачах, могут быть легко получены с помощью вариантов метода взвешенных невязок, таких как метод Галёркина или метод наименьших квадратов. Это сыграло важную роль в теоретическом обосновании МКЭ, так как позволило применять его при решении многих типов дифференциальных уравнений. Таким образом, метод конечных элементов превратился в общий метод численного решения дифференциальных уравнений или систем дифференциальных уравнений.

С развитием вычислительных средств возможности метода постоянно расширяются, также расширяется и класс решаемых задач.

Метод конечных элементов (МКЭ) является сеточным методом, предназначенным для решения задач, для которого модель объекта задаётся системой дифференциальных уравнений в частных производных с заданными краевыми условиями.

Идея метода заключается в следующем: в методе взвешенных невязок воспользоваться простыми пробными и весовыми функциями, но не во всей области S, а в её отдельных подобластях (конечных элементах), а точность решения задачи обеспечить использованием большого числа конечных элементов (КЭ), при этом КЭ могут быть простой формы и вычисление интегралов по ним не должно вызывать особых затруднений. Математически переход от метода взвешенных невязок к МКЭ осуществляется с использованием специальных пробных функций, которые также называются глобальными базисными функциями, обладающих следующими свойствами:

- в узле аппроксимации функции имеют значение равное единице;
- отличны от нуля только в КЭ, содержащих этот узел аппроксимации, во всей остальной области равны нулю.

Этапы решения задач с общих позиций

Выбор конечного элемента

В одномерной задаче это просто отрезок прямой. В двумерной задаче – треугольный (симплекс) элемент, четырёхугольный элемент, в общем случае любая фигура, с помощью которой можно разбить исследуемую область на непересекающиеся подобласти, но при этом следует учитывать, что чем сложнее элемент, тем с большими сложностями придётся столкнуться при вычислении интегралов, поэтому наиболее распространенными являются треугольный элемент и четырёхугольный со сторонами параллельными осям координат. Для трёхмерной области – тетраэдр и параллелепипед.

Разбиение области на КЭ

Существуют различные способы автоматического разбиения области на конечные элементы, исполнение этого этапа вручную утомительно и часто приводит к ошибкам.

Получение функции формы

В зависимости от соотношения требований точности задачи и возможностей вычислительной системы выбирают линейную, квадратичную или более высокую степень функции формы, при этом число узлов аппроксимации должно быть минимум на единицу больше порядка аппроксимирующей функции.

Получение матрицы жёсткости и вектора нагрузок

Этот этап получил своё название из строительной механики, поскольку именно там впервые был применён МКЭ. На этом этапе используется метод взвешенных невязок в пределах одного КЭ.

Ансамблирование или получение глобальных матрицы жёсткости и вектора нагрузок

Учёт граничных условий

Решение системы алгебраических уравнений

При решении может быть учтена особенность матрицы коэффициентов, поскольку она, как правило, имеет ленточную форму.

Литература

1. *Башлыкова Н. А., Винокуров Н. А., Казаков В. Г., Задорожный А. М., Переведенцев Е. А., Эйдельман Ю. И.* Лабораторные работы в терминальном классе по электродинамике. Новосибирск: НГУ, 1987. Вып. 1. Описание терминальных задач. Инструкции по работе с программами.
2. *Башлыкова Н. А., Задорожный А. М., Казаков В. Г., Переведенцев Е. А., Эйдельман Ю. И.* Лабораторные работы в терминальном классе по электродинамике. Новосибирск: НГУ, 1988. Вып. 2. Описание терминальных задач. Инструкции по работе с программами.
3. *Астрелин В. Т., Башлыкова Н. А., Воропаев С. Г., Задорожный А. М., Казаков В. Г., Коткин Г. Л., Меледин Г. В., Переведенцев Е. А., Черкасский В. С., Эйдельман Ю. И.* Практикум по электродинамике в терминальном классе / Под ред. Б. А. Князева. Новосибирск: Новосиб. гос. ун-т, 1992.
4. *Коткин Г. Л., Черкасский В. С.* Компьютерное моделирование физических процессов с использованием MATLAB. Новосибирск: Новосиб. гос. ун-т, 2001.
5. *Мешков И. Н., Чириков Б. В.* Электромагнитное поле. Новосибирск: Наука, 1987. Ч. 1.
6. *Яковлев В. И.* Классическая электродинамика. Электричество и магнетизм. Новосибирск: Новосиб. гос. ун-т, 2003. Ч. 1.
7. *Меледин Г. В., Черкасский В. С.* Электродинамика в задачах. Новосибирск: НГУ, 2009. Ч. 1.
8. *Галлагер Р.* Метод конечных элементов. Основы: Пер. с англ. М.: Мир, 1984.
9. *Деклу Ж.* Метод конечных элементов: Пер. с фр. М.: Мир, 1976.

Оглавление

Учебное издание

Витюгова Нина Алексеевна,
Задорожный Александр Максимович,
Черкасский Валерий Семенович

Моделирование электростатических полей в среде MATLAB

Методическое пособие
Выпуск 1

Редактор *К. В. Шмугурова*

Компьютерная верстка *Н. А. Витюговой*

Подписано в печать 25.12.2009. Формат 60х84 /16.
Офсетная печать.
Уч.-изд. л. 3,25. Усл. печ. л. 3. Тираж 100 экз.
Заказ №
Редакционно-издательский центр НГУ.
630090, Новосибирск 90, ул. Пирогова, 2.