

МИНИСТЕРСТВО
ОБРАЗОВАНИЯ И НАУКИ
УКРАИНЫ

Национальный технический университет
"Харьковский политехнический институт"

www.kpi.kharkov.ua

Кафедра прикладной математики

С.П. Иглин
iglin@kpi.kharkov.ua

Вариационное исчисление

с применением MATLAB

Учебное пособие и варианты
индивидуальных домашних заданий
для студентов всех форм обучения

Харьков 2000

УДК 517.2, 681.142.2

Иглин С.П. Вариационное исчисление с применением MATLAB. Учебное пособие и варианты индивидуальных домашних заданий для студентов всех форм обучения. – Харьков: НТУ ХПИ, 2000.

Рассмотрены основные разделы курса вариационного исчисления. По каждому из разделов приведены необходимые теоретические сведения. Подробно рассмотрено решение примеров, к которым составлены программы на языке **MATLAB**. Подобрано 30 вариантов примеров для самостоятельного решения. Электронная версия пособия оформлена в виде **M**-книги, что позволяет проводить вычисления непосредственно из **Word**-документа.

© С.П.Иглин, 2000 г.
e-mail: iglin@kpi.kharkov.ua

0. Введение

При изучении курса вариационного исчисления преподавателям часто приходится сталкиваться с проблемой подбора задач для индивидуальных домашних заданий. В имеющихся задачниках, в частности, в [3], мало однотипных вариантов. Кроме того, исходные данные этих задач подобраны так, что почти все они могут быть решены аналитически, без применения вычислительной техники. На практике такие задачи встречаются очень редко. В предлагаемом пособии все рассматриваемые задачи требуют для своего решения применения компьютеров и знания языков программирования. Поэтому настоящее учебное пособие можно рассматривать как комплексное пособие по вариационному исчислению, численным методам и программированию.

Мы выбрали для использования систему **MATLAB**, которая приобретает всё большее распространение [4]. **MATLAB** сочетает в себе гибкую систему символьного программирования, во многом совместимую с **Maple**, и мощные средства численного решения задач и отображения результатов. Решение каждого примера пособия включает в себя вывод дифференциального уравнения в символьском виде [9], аналитическое или численное его решение, и построение графика решения.

Начиная с версии 5, **MATLAB** позволяет создавать интерактивные документы, так называемые **М-книги**. **М-книгу** можно рассматривать как документ **MS Word** с встроенной в него работающей программой на **MATLAB**'е. Данное пособие представляет как раз такую **М-книгу**. Если у Вас есть электронный вариант этого пособия, Вы можете прямо из него запустить все программы и посмотреть полученные результаты.

Вы можете использовать и другие математические пакеты, например, **Maple**, **MathCad**, **Mathematica** и т.д. Во всех примерах подробно описано составление программ для решения заданий, и студенты смогут повторить её на других языках.

Каждая глава посвящена отдельной теме курса. Все главы построены одинаково. Вначале излагаются краткие теоретические сведения. Затем рассматривается решение примеров и составление программ. Решение везде доводится до численного результата с построением графика. В конце приведены варианты ИДЗ.

Особенно подробно рассмотрен пример 1а. Причины этого следующие.

1. Это первый пример, который решают студенты.
2. В этом примере есть все необходимые элементы, которые используются в других примерах: вывод уравнения, его решение, построение графика.
3. При составлении программ для других примеров используются фрагменты программы и исходные данные примера 1а. Многие из примеров последующих глав сформулированы в виде: «решить пример 1а при данном дополнительном условии и сравнить решение с решением примера 1а».

В пособии используются такие шрифты. Названия выделяются шрифтом **Bold**. В формулах используется *Italic*, при этом матрицы и векторы пишем шрифтом **Bold Italic**. При написании программ и результатов их работы используются соглашения ППП **Notebook** [4]. Ячейки ввода выделяются шрифтом **Curier New Bold зелёного цвета**. Ячейки вывода – **Curier New** и синим цветом. Ошибки – шрифтом **Curier New Bold красного цвета**. В печатном варианте пособия цвет шрифта не виден, но начертания шрифтов **Bold**, *Italic*, **Bold Italic**, **Curier New** и **Curier New Bold** различаются хорошо.

Данное пособие не заменяет учебники по вариационному исчислению. Теоретические основы излагаются в пособии в сжатом виде. Даются только те сведения, которые необходимы непосредственно для решения задач. В качестве учебников можно использовать, например, [1,2,5,6].

1. Элементарная задача вариационного исчисления

1.1. Краткие теоретические сведения

В курсе вариационного исчисления рассматриваются задачи исследования на экстремум функционалов. Функционалом называется правило, по которому каждой функции из некоторого их класса ставится в соответствие число. Рассмотрим функционал, зависящий от функции одной переменной и её производной:

$$J(y) = \int_{x_1}^{x_2} F(x, y, y') dx \rightarrow \text{extr} \quad (1.1)$$

с заданными граничными условиями:

$$\begin{cases} y(x_1) = y_1, \\ y(x_2) = y_2, \end{cases} \quad (1.2)$$

где $F(x, y, y')$ – непрерывная функция трёх переменных и дифференцируемая функция двух своих последних аргументов.

Необходимым условием экстремума функционала является равенство нулю его вариации, вычисленной на экстремальной функции $y_0(x)$:

$$\delta J(y) = 0. \quad (1.3)$$

Вариация функционала δJ – это главная, линейная относительно вариации функции δy , часть его приращения ΔJ . В нашем случае $\delta J(y)$ вызывается вариацией независимой переменной – функции $y(x)$ и её производной $y'(x)$: $y(x) = y_0(x) + \delta y(x)$; $y'(x) = y'_0(x) + \delta y'(x)$; причём в силу граничных условий на концах интервала $\delta y(x_1) = \delta y(x_2) = 0$. Вычислим вариацию функционала как линейную часть его приращения. Для этого разложим $F(x, y_0 + \delta y, y'_0 + \delta y')$ в ряд Тейлора в окрестности экстремальной функции с удержанием только линейных членов, а затем проинтегрируем по частям:

$$\begin{aligned} \delta J(y_0) &= \int_{x_1}^{x_2} (F(x, y_0 + \delta y, y'_0 + \delta y') - F(x, y_0, y'_0)) dx = \int_{x_1}^{x_2} (F_y \delta y + F_{y'} \delta y') dx = \\ &= \int_{x_1}^{x_2} F_y \delta y dx + \left(F_{y'} \delta y \right) \Big|_{x_1}^{x_2} - \int_{x_1}^{x_2} \frac{dF_{y'}}{dx} \delta y dx = \int_{x_1}^{x_2} \left(F_y - \frac{dF_{y'}}{dx} \right) \delta y dx = 0. \end{aligned} \quad (1.4)$$

Так как вариация функции $\delta y(x)$ – произвольная, то в силу основной леммы вариационного исчисления первый сомножитель под интегралом должен равняться нулю. Таким образом, функция, на которой достигается экстремум, должна удовлетворять дифференциальному уравнению

$$F_y - \frac{dF_{y'}}{dx} = 0. \quad (1.5)$$

Это уравнение называется дифференциальным уравнением Эйлера. Оно является в общем случае уравнением второго порядка и дополняется двумя граничными условиями (1.2). Любое его решение называется экстремалью. Это кривая, на которой может достигаться экстремум.

Так как уравнение Эйлера дополняется не начальными, а граничными условиями, то теорема Коши о существовании и единственности решения дифференциального уравнения здесь неприменима. Иными словами, экстремаль не обязательно существует, а если существует, то не обязательно единственна. Всё зависит от вида уравнения Эйлера (1.5) и разрешимости системы уравнений для граничных условий (1.2).

Рассмотрим некоторые частные случаи уравнения Эйлера (1.5).

1. Подынтегральная функция F не зависит от производной y' или зависит от неё линейно. В этом случае уравнение Эйлера становится алгебраическим, в его решении нет произвольных постоянных, и оно не обязательно удовлетворяет условиям (1.2). Если граничные условия (1.2) удовлетворяются, то мы получили экстремаль, а если нет – то нет и решений у данной вариационной задачи.
2. Частный случай случая 1: $F = P(x, y) + y' Q(x, y)$, причём $\partial P / \partial y = \partial Q / \partial x$. В этом случае уравнение (1.5) обращается в тождество $0=0$, и экстремалью будет любая кривая, соединяющая точки $M_1(x_1, y_1)$ и $M_2(x_2, y_2)$. Криволинейный интеграл (1.1) в этом случае не зависит от линии интегрирования, и вариационная задача теряет смысл.
3. Подынтегральная функция F не зависит явно от y . Уравнение Эйлера имеет первый интеграл $F_{y'} = C_1$. Это уравнение первого порядка, и оно решается легче, чем исходное уравнение второго порядка.
4. Если подынтегральная функция F не зависит явно от x , то уравнение Эйлера также имеет первый интеграл вида $F - y' F_{y'} = C_1$. Действительно, уравнение Эйлера (1.5) можно записать в виде $F_y - F_{y'y'} y' - F_{y'y''} y'' = 0$. Из-за явной независимости от x это выражение имеет вид $F_y - F_{y'y'} y' - F_{y'y''} y'' = 0$. Но такой же вид имеет и полная производная по x от выражения $F - y' F_{y'} = C_1$: $d(F - y' F_{y'}) / dx = F_x - y' F_{xy} - F_{y'y'} y' - y'' F_{y'y''} + (F_{y'y'} y' + F_{y'y''} y'') = F_{y'y''} y'' - F_{y'y''} y'' = 0$, что после сокращения на y' совпадает с уравнением Эйлера.

Чтобы проверить, действительно ли достигается экстремум на найденной экстремали, нужно воспользоваться достаточными условиями экстремума. Простейшее из них – это условие Лежандра. Для его

применения нужно вычислить F_{yy} и проверить знак этого выражения на кривых, близких к экстремали. Если $F_{yy} > 0$ для всех $y(x)$, близких к экстремали, и для любых $y'(x)$, то на данной экстремали достигается сильный минимум. Если же неравенство $F_{yy} > 0$ выполняется для всех $y(x)$, близких к экстремали, но только для $y'(x)$, близких к экстремали, то достигается слабый минимум. При $F_{yy} < 0$ достигается максимум (соответственно сильный или слабый).

1.2. Примеры выполнения заданий

1.2.а. Пример 1а

Найти экстремаль функционала

$$J(y) = \int_{-1}^1 (x^2 + y^2 + y'^2) dx; \quad \begin{cases} y(-1) = 1; \\ y(1) = 2. \end{cases} \quad (1.6)$$

Исследовать полученную экстремаль на достаточные условия экстремума. Построить график решения.

В этом примере подынтегральная функция $F(x, y, y')$ является функцией общего вида, поэтому составим уравнение Эйлера в виде (1.5) и решим его. Затем построим график решения. Попутно исследуем на выполнение достаточных условия экстремума.

Применим для решения задачи систему программирования **MATLAB**. Рекомендуется после написания каждой группы команд запустить её и посмотреть результаты.

Вначале очистим память от решавшихся ранее задач. Зададим формат отображения действительных чисел с 14 знаками. Напечатаем заголовок решаемой задачи. Зададим число точек для построения графика. Опишем символические переменные.

```
clear all % очищаем память
format long % формат отображения чисел с 14 знаками
disp('Решаем пример 1а') % выводим заголовок задачи
syms x y Dy D2y % описали символические переменные
Решаем пример 1а
```

Уравнение Эйлера является дифференциальным уравнением второго порядка. Для его решения используем принятые в **MATLAB** обозначения производных: Dy для y' и $D2y$ для y'' . Вводим подынтегральную функцию и граничные условия. Здесь Вы должны поставить свои исходные данные.

```
F=x^2+y^2+Dy^2; % вводим подынтегральную функцию
x1=-1; % вводим граничные условия
y1=1;
x2=1;
y2=2;
fprintf('Подынтегральная функция: F=%s\n',char(F))
fprintf('Граничные условия: y(%d)=%d; y(%d)=%d\n',x1,y1,x2,y2)
Подынтегральная функция: F=x^2+y^2+Dy^2
Граничные условия: y(-1)=1; y(1)=2
```

Подынтегральная функция F – символическое выражение, граничные условия $x1$, $y1$, $x2$ и $y2$ – действительные переменные. Найдём частные производные F_y и $F_{y'}$.

```
dFdY=diff(F,y) % вычисляем F_y
dFdY1=diff(F,Dy) % вычисляем F_{y'}
dFdY =
2*y
dFdY1 =
2*Dy
```

В уравнение Эйлера входит полная производная dF_y/dx . Вычисляем её по формуле

$$\frac{dF_y}{dx} = \frac{\partial F_y}{\partial x} + \frac{\partial F_y}{\partial y'} y' + \frac{\partial F_y}{\partial y''} y'' \quad (1.7)$$

Вычислим частные производные $\partial F_y / \partial x$, $\partial F_y / \partial y'$ и $\partial F_y / \partial y''$, а затем найдём полную производную dF_y/dx по (1.7).

```
d_dFdY1_dx = diff(dFdY1,x) % ∂(F_{y'})/∂x
d_dFdY1_dy=diff(dFdY1,y) % ∂(F_{y'})/∂y
```

```
d_dFdy1_dy1 = diff(dFdy1,Dy) %  $\partial(F_y)/\partial y = F_{y'y}$  - условие Лежандра
dFy1dx = d_dFdy1_dx + d_dFdy1_dy*Dy + d_dFdy1_dy1*D2y %  $d(F_y)/dx$ 
d_dFdy1_dx =
0
d_dFdy1_dy =
0
d_dFdy1_dy1 =
2
dFy1dx =
2*D2y
```

Составим левую часть дифференциального уравнения Эйлера и упростим её. Для решения дифференциального уравнения нужно представить его в виде строковой переменной. Преобразуем символическую переменную **Euler** в строку и добавим в конце равенство нулю.

```
Euler = simple(dFdy-dFy1dx) % левая часть уравнения Эйлера
deqEuler = [ char(Euler) ' = 0 ' ]; % составили уравнение
fprintf('Уравнение Эйлера: %s\n',deqEuler)
Euler =
2*y-2*D2y
У р а в н е н и е  Э й л е р а :  2*y-2*D2y=0
```

Команда **dsolve** позволяет находить как общее решение дифференциального уравнения, так и частное его решение, удовлетворяющее заданным начальным или граничным условиям. В следующих главах при решении других примеров нам нужно будет иметь общее решение уравнения Эйлера. Поэтому найдём вначале общее решение, а потом подставим в него граничные условия. Решаем уравнение Эйлера и проверяем, является ли решение единственным. Если нет, прекращаем работу программы.

```
Sol = dsolve(deqEuler,'x') % решаем уравнение Эйлера
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
Sol =
(C1*exp(x)^2+C2)/exp(x)
```

Программа будет работать дальше, если решение существует и единственное. Сформируем теперь уравнения для граничных условий. Подставим в найденное аналитическое решение **Sol** граничные точки **x1** и **x2**, преобразованные в символические переменные. Затем преобразуем эти символические выражения в строки, и приравняем их соответственно **y1** и **y2**.

```
SolLeft = subs(Sol,x,sym(x1)); % подставляем x1
SolRight = subs(Sol,x,sym(x2)); % подставляем x2
EqLeft = [char(SolLeft) '=' char(sym(y1))] % приравняли y1
EqRight = [char(SolRight) '=' char(sym(y2))] % приравняли y2
EqLeft =
(C1*exp(-1)^2+C2)/exp(-1)=1
EqRight =
(C1*exp(1)^2+C2)/exp(1)=2
```

Решаем полученную систему уравнений – находим значения произвольных постоянных **C1** и **C2**. Подставим найденные константы в аналитическое решение **Sol**. Ограничим полученное решение 14 знаками – этого вполне достаточно для построения графика решения.

```
Con = solve(EqLeft,EqRight); % решаем систему уравнений
C1=Con.C1 % присваиваем полученные решения
C2=Con.C2 % символическим константам C1 и C2
Solla = vpa(eval(Sol),14); % подставляем C1, C2, вычисляем с 14 знаками
fprintf('Уравнение экстремали:\n%s\n',char(Solla))
C1 =
(-2*exp(1)+exp(-1))/(exp(-1)^2-exp(1)^2)
C2 =
exp(1)*exp(-1)*(-exp(1)+2*exp(-1))/(exp(-1)^2-exp(1)^2)
```

Уравнение экстремали:

$$(.69877023730776 * \exp(x)^2 + .27331117318808) / \exp(x)$$

Зададим массив аргументов для рисования графика функции и вычислим значения функции. Нарисуем график, подпишем заголовок и координатные оси.

```
xpl = linspace(x1,x2); % задаём массив абсцисс
yla = subs(Solla,x,xpl); % вычислили ординаты
plot ( xpl, yla, '-r' ) % рисуем график
title ( '\bfExample 1a' ) % заголовок
xlabel('x') % метка оси OX
ylabel('y(x)') % метка оси OY
```

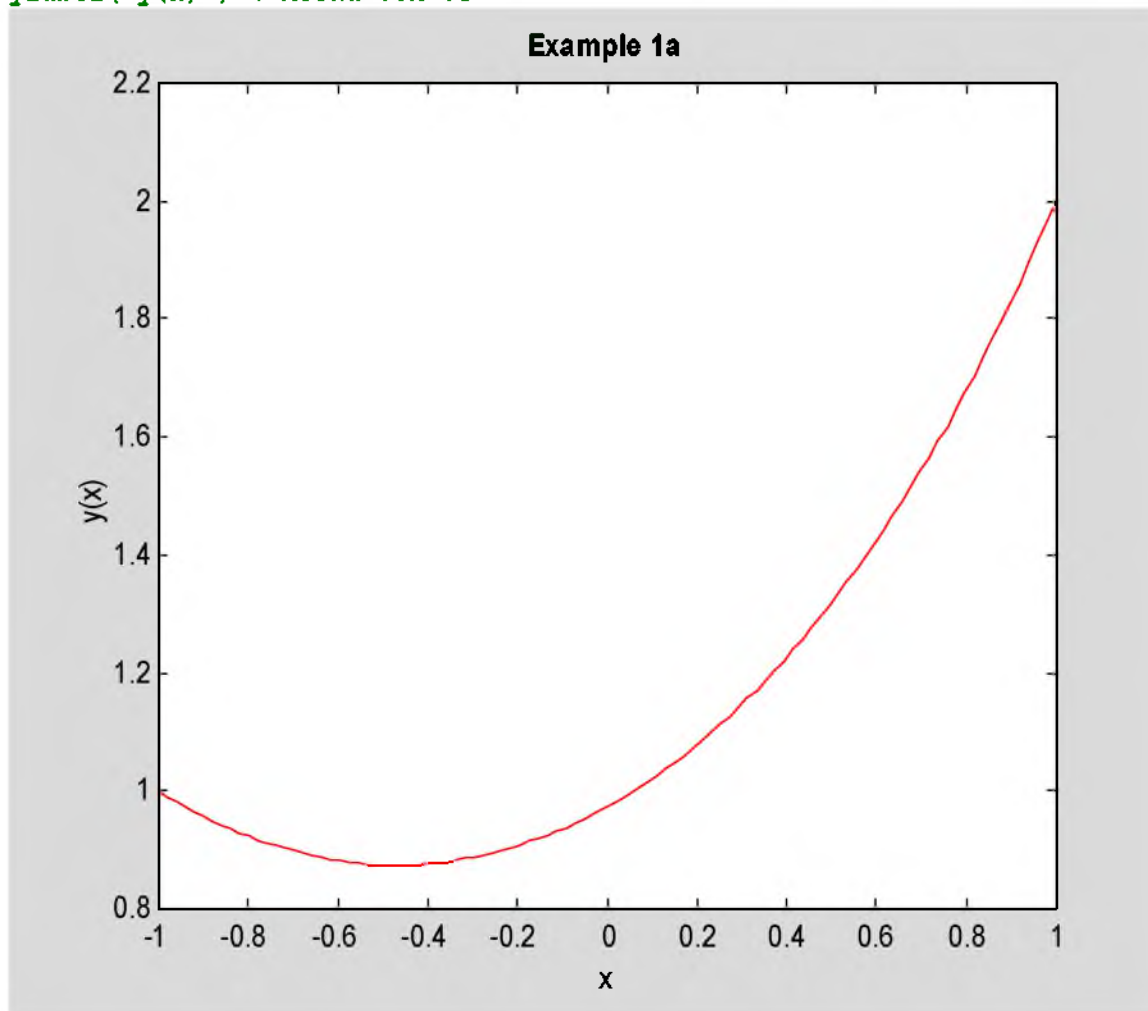


Рис. 1.1. Решение примера 1а

Ответ. Уравнение Эйлера после сокращения на -2 будет:

$$y'' - y = 0. \quad (1.8)$$

Его аналитическое решение

$$y(x) = C_1 e^x + C_2 e^{-x}. \quad (1.9)$$

После подстановки значений произвольных постоянных экстремаль функционала (1.6) имеет вид

$$y(x) = 0.69877 e^x + 0.27331 e^{-x}. \quad (1.10)$$

График её показан на рис.1.1. Так как $F_{yy'} = 2 > 0$, и это условие выполняется $\forall y(x)$ и $\forall y'(x)$, то в соответствии с достаточным условием Лежандра найденная экстремаль доставляет сильный минимум функционалу (1.6).

1.2.b. Пример 1b

Найти экстремаль функционала

$$J(y) = \int_{-1}^1 (y'^2 + 2y' \operatorname{sh} x - 5x^2) dx, \quad \begin{cases} y(-1) = 2; \\ y(1) = 3. \end{cases} \quad (1.11)$$

Исследовать на выполнение достаточных условий экстремума. Построить график решения.

В этом примере подынтегральная функция $F(x,y,y')$ не зависит явно от y . Первый интеграл уравнения Эйлера имеет вид $F_y = C_1$. Составим программу для решения этой вариационной задачи. Вначале введём исходные данные.

```
clear all % очищаем память
format long % формат отображения чисел с 14 знаками
disp('Решаем пример 1b') % выводим заголовок задачи
syms x Dy % описали символические переменные
F=Dy^2+2*Dy*sinh(x)-5*x^2; % вводим подынтегральную функцию
x1=-1; % вводим граничные условия
y1=2;
x2=1;
y2=3;
fprintf('Подынтегральная функция: F=%s\n',char(F))
fprintf('Граничные условия: y(%d)=%d; y(%d)=%d\n',x1,y1,x2,y2)
Решаем пример 1b
Подынтегральная функция: F=Dy^2+2*Dy*sinh(x)-5*x^2
Граничные условия: y(-1)=2; y(1)=3
```

У нас будет первый интеграл уравнения Эйлера, поэтому ни сама функция y , ни её вторая производная y'' нам не нужны, и мы их не описываем. Строим первый интеграл и решаем полученное дифференциальное уравнение. Названия констант C_1 и C_2 используются в команде `dsolve`, поэтому обозначим константу C .

```
dFdyl = simple(diff(F,Dy)) % F_y
deqEuler = [char(dFdyl) '=C'] % составили уравнение
Sol = dsolve(deqEuler,'x') % решаем уравнение Эйлера
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
dFdyl =
2*Dy+2*sinh(x)
deqEuler =
2*Dy+2*sinh(x)=C
Sol =
-cosh(x)+1/2*C*x+C1
```

В переменной `Sol` получено общее решение, произвольные постоянные обозначены C и C_1 . Найдём их. Для этого подставим в `Sol` граничные точки.

```
SolLeft = subs(Sol,x,sym(x1)) % подставляем x1
SolRight = subs(Sol,x,sym(x2)) % подставляем x2
SolLeft =
-cosh(1)-1/2*C+C1
SolRight =
-cosh(1)+1/2*C+C1
```

Приравняем полученные выражения соответственно y_1 и y_2 . Тем самым мы сформируем систему алгебраических уравнений. Решим её. Подставим найденные константы C и C_1 в решение `Sol`.

```
EqLeft = [char(SolLeft) '=' char(sym(y1))] % приравняли y1
EqRight = [char(SolRight) '=' char(sym(y2))] % приравняли y2
Con = solve(EqLeft,EqRight); % решаем систему уравнений
C=Con.C % присвоили полученные решения
C1=Con.C1 % символическим переменным C и C1
Sol1b = vpa(eval(Sol),14); % подставляем C1, C в решение с точностью 14 знаков
fprintf('Уравнение экстремали:\n%s\n',char(Sol1b))
EqLeft =
-cosh(1)-1/2*C+C1=2
EqRight =
-cosh(1)+1/2*C+C1=3
C =
```



```
1
C1 =
cosh(1)+5/2
Уравнение экстремали:
-1.*cosh(x)+.500000000000000*x+4.0430806348152
```

Дальнейшие действия не отличаются от описанных в примере 1а. Рисуем график и проверяем выполнение достаточных условий экстремума.

```
xp1 = linspace(x1,x2); % массив абсцисс
y1b = subs(Sol1b,x,xp1); % вычисляем ординаты
plot ( xp1, y1b, '-r' ) % рисуем график
title ( '\bfExample 1b' ) % заголовок
xlabel('x') % метка оси OX
ylabel('y(x)') % метка оси OY
Leg = diff(dFdy1,Dy) %  $F_{y'y'}$  - достаточное условие Лежандра
Leg =
2
```

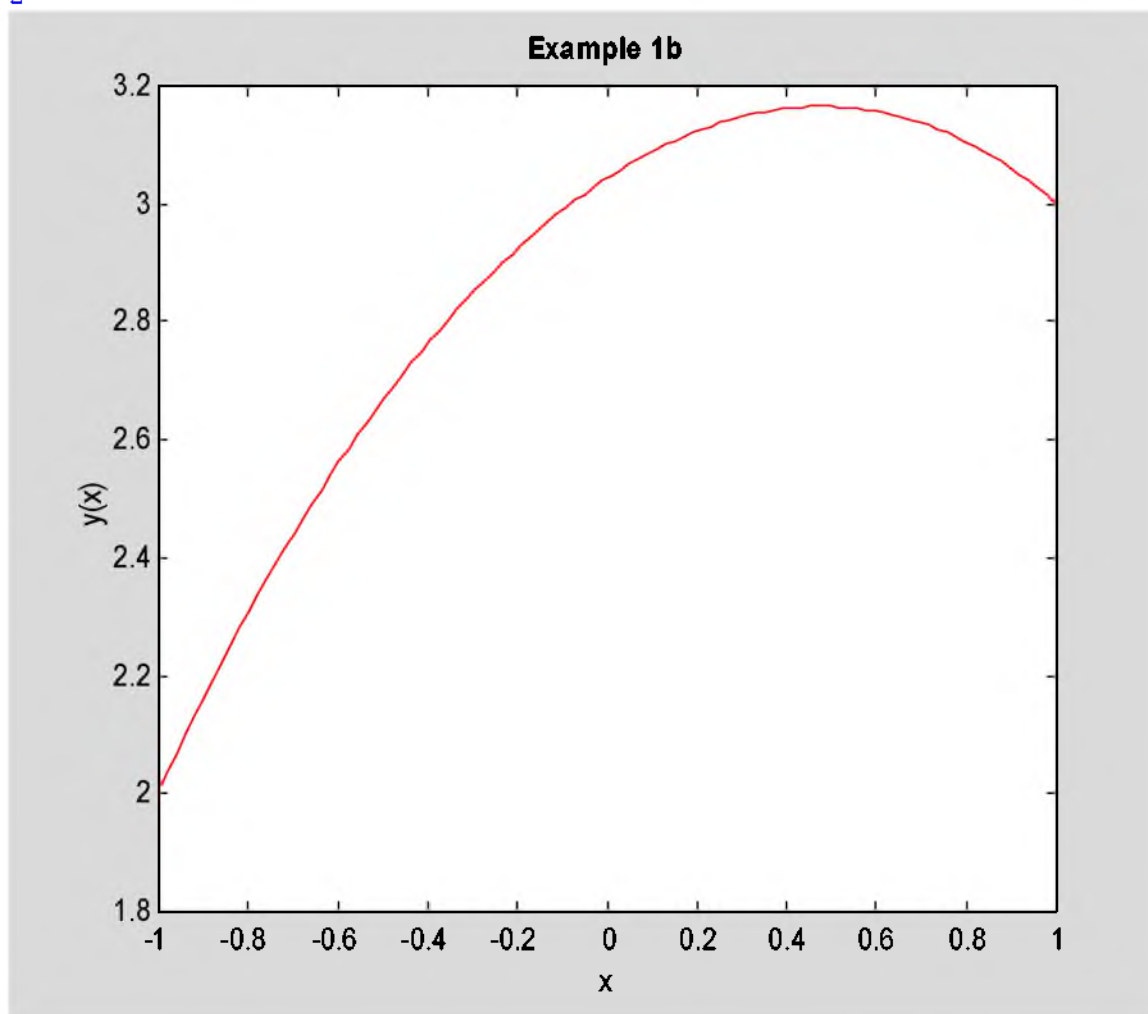


Рис. 1.2. Решение примера 1b

Ответ. Первый интеграл дифференциального уравнения Эйлера

$$2y'+2\operatorname{sh}x=C. \quad (1.12)$$

Экстремаль функционала (1.11) имеет вид

$$y(x)=-\operatorname{ch}x+0.5Cx+C_1 \approx -\operatorname{ch}x+0.5x+4.04308. \quad (1.13)$$

График её показан на рис.1.2. Так как $F_{y'y'}=2>0$, и это выполняется $\forall y(x)$ и $\forall y'(x)$, то по достаточному условию Лежандра имеем сильный минимум.

1.2.с. Пример 1с

Найти экстремаль функционала

$$J(y) = \int_1^3 y^2 y'^2 dx, \quad \begin{cases} y(1) = 4; \\ y(3) = 3. \end{cases} \quad (1.14)$$

Исследовать на достаточные условия. Построить график решения.

Здесь подынтегральная функция $F(x, y, y')$ не зависит явно от x . Первый интеграл уравнения Эйлера имеет вид $F - y'F_{y'} = C_1$. Составим программу для решения этого примера. Вначале введём исходные данные задачи.

```
clear all % очищаем память
format long % формат отображения чисел с 14 знаками
disp('Решаем пример 1с') % печатаем заголовок
syms x y Dy % описали символические переменные
F=y^2*Dy^2; % вводим подынтегральную функцию
x1=1;
y1=4;
x2=3;
y2=3;
fprintf('Подынтегральная функция: F=%s\n',char(F))
fprintf('Граничные условия: y(%d)=%d; y(%d)=%d\n',x1,y1,x2,y2)
Решаем пример 1с
Подынтегральная функция: F=y^2*Dy^2
Граничные условия: y(1)=4; y(3)=3
```

Выводим первый интеграл уравнения Эйлера. Решаем полученное уравнение.

```

dFdy1 = diff(F,Dy)      % Fy
Euler=simple(F-Dy*dFdy1) % левая часть уравнения Эйлера
deqEuler = [char(Euler) '=C'] % составили уравнение
Sol = dsolve(deqEuler,'x') % решаем уравнение Эйлера
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
dFdy1 =
2*y^2*Dy
Euler =
-y^2*Dy^2
deqEuler =
-y^2*Dy^2=C
Sol =
[ (-2*(-C)^(1/2)*x+C1)^(1/2) ]
[ -(-2*(-C)^(1/2)*x+C1)^(1/2) ]
[ (2*(-C)^(1/2)*x+C1)^(1/2) ]
[ -(2*(-C)^(1/2)*x+C1)^(1/2) ]
??? [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] []

```

Первая проблема! Решение данной задачи – не единственное. **MATLAB** не смог упростить первый интеграл уравнения Эйлера, и оставил его в виде: $-v^2 u'^2 = C$. Нам придётся упростить его вручную.

```
Euler = y*Dy % упрощаем вручную
deqEuler = [char(Euler) '=C'] % составили уравнение
Sol = dsolve(deqEuler,'x') % решаем уравнение Эйлера
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
Euler =
y*Dy
deqEuler =
y*Dy=C
Sol =
[ (2*C*x+C1)^(1/2)]
[ -(2*C*x+C1)^(1/2)]
??? [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] []
```

Уравнение упростилось, но оно по-прежнему имеет более одного решения. Для удовлетворения заданным граничным условиям мы должны выбрать первое решение. Выбираем его, и повторяем далее все операторы программы примера 1а.

```
SolLeft = subs(Sol(1),x,sym(x1)) % подставляем x1 в первое решение
SolRight = subs(Sol(1),x,sym(x2)) % подставляем x2 в первое решение
EqLeft = [char(SolLeft) '=' char(sym(y1))] % приравниваем y1
EqRight = [char(SolRight) '=' char(sym(y2))] % приравниваем y1
Con=solve(EqLeft,EqRight); % решаем систему уравнений
C=Con.C % присвоили полученные решения
C1=Con.C1 % символическим переменным C и C1
Sol1c = vpa(eval(Sol(1)),14); % подставляем C1, C2 в решение с 14 знаками
fprintf('Уравнение экстремали:\n%s\n',char(Sol1c))
xpl=linspace(x1,x2); % задаём массив абсцисс
y1c=subs(Sol1c,x,xpl); % вычисляем ординаты
plot ( xpl, y1c, '-r' ) % рисуем график функции
title ( '\bfExample 1c' ) % заголовок
xlabel('x') % метка оси OX
ylabel('y(x)') % метка оси OY
Leg = diff(dFdy1,Dy) % Fy'y' - достаточное условие Лежандра
SolLeft =
(2*C+C1)^(1/2)
SolRight =
(6*C+C1)^(1/2)
EqLeft =
(2*C+C1)^(1/2)=4
EqRight =
(6*C+C1)^(1/2)=3
C =
-7/4
C1 =
39/2
Уравнение экстремали:
(-3.5000000000000*x+19.500000000000)^(1/2)
Leg =
2*y^2
```

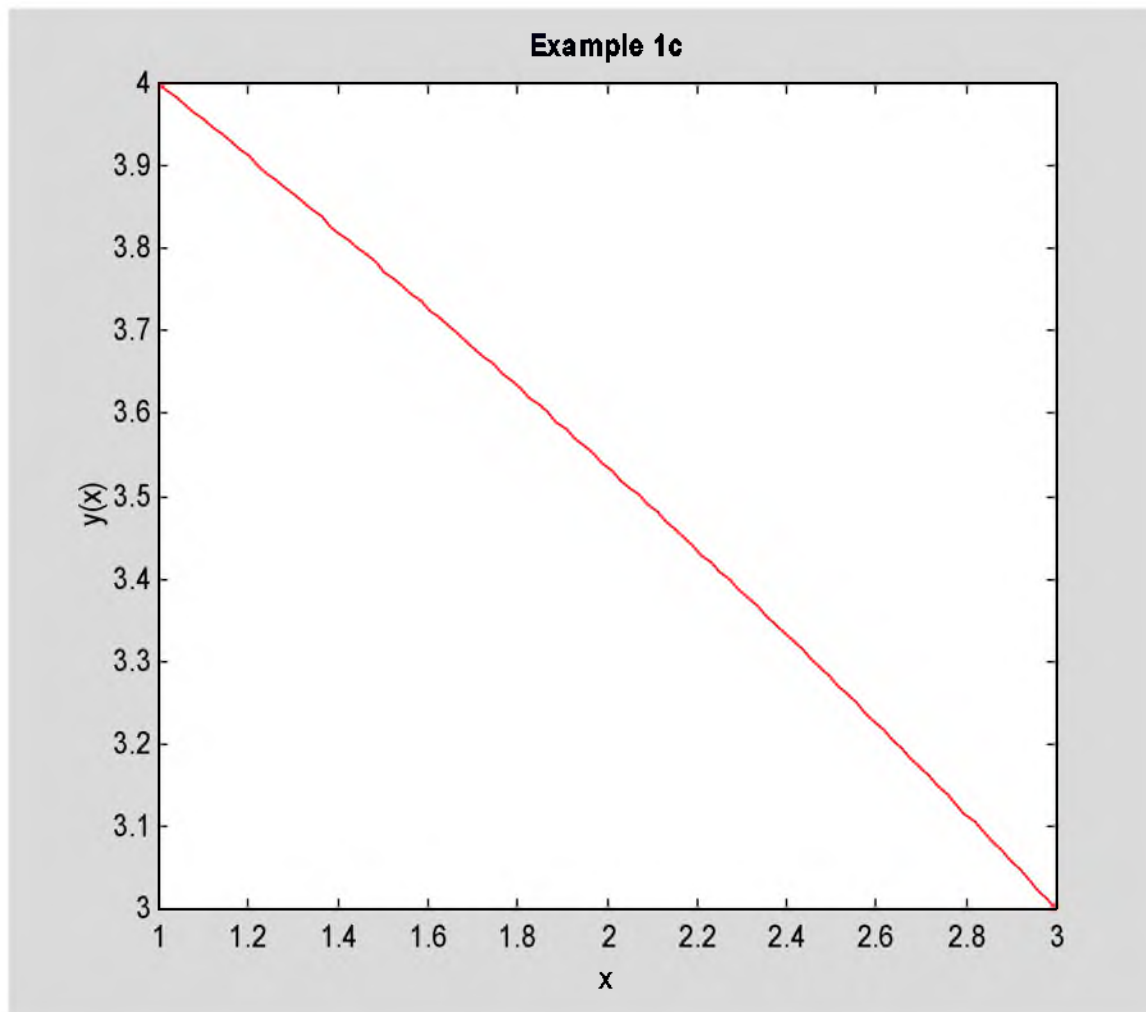


Рис. 1.3. Решение примера 1с

Ответ. Первый интеграл дифференциального уравнения Эйлера

$$yy' = C.$$

(1.15)

Экстремаль функционала (1.13) имеет вид

$$y(x) = \pm \sqrt{2Cx + C_1} = \sqrt{19.5 - 3.5x}.$$

(1.16)

График её показан на рис.1.3. Так как $F_{yy} = 2y^2 > 0$, имеем сильный минимум.

1.3. Задание

Для своего варианта функционалов а), б), в) найти экстремали, построить их графики и исследовать на выполнение достаточных условий экстремума.

1.4. Варианты заданий

Вариант 1.

а). $J(y) = \int_{-1}^1 (y'^2 + 4y^2 - 8xy + 2x^2) dx$; $y(-1) = 3$; $y(1) = 1$;

б). $J(y) = \int_0^2 (y'^2 - 4y'e^{2x} + \sin^2 x) dx$; $y(0) = 1$; $y(2) = -2$;

в). $J(y) = \int_0^1 y\sqrt{1+y'^2} dx$; $y(0) = 2$; $y(1) = 3$;

Вариант 2.

а). $J(y) = \int_{-1}^1 (y'^2 - 4y^2 + 2xy - x^2) dx$; $y(-1) = 2$; $y(1) = 4$;

b). $J(y) = \int_0^2 (y'^2 - 4y' \sin 2x - x^2) dx; \quad y(0)=1; \quad y(2)=-1;$

c). $J(y) = \int_0^1 \frac{\sqrt{1+y'^2}}{y} dx; \quad y(0)=2; \quad y(1)=1;$

Вариант 3.

a). $J(y) = \int_{-1}^1 (y'^2 + 4y^2 + 4x^2 y + x \cos x) dx; \quad y(-1)=2; \quad y(1)=0.5;$

b). $J(y) = \int_0^2 (y'^2 - 4y' \cos 2x + 5 \sin 3x) dx; \quad y(0)=2; \quad y(2)=-3;$

c). $J(y) = \int_0^1 y y'^2 dx; \quad y(0)=2; \quad y(1)=1;$

Вариант 4.

a). $J(y) = \int_0^2 (y'^2 + 9y^2 + 2xy - x \sin x) dx; \quad y(0)=1; \quad y(2)=2;$

b). $J(y) = \int_1^3 \left(y'^2 - \frac{4y'}{x} + x \sin x \right) dx; \quad y(1)=1; \quad y(3)=-2;$

c). $J(y) = \int_0^1 \sqrt{y(1+y'^2)} dx; \quad y(0)=1; \quad y(1)=3;$

Вариант 5.

a). $J(y) = \int_{-2}^0 (y'^2 - 4y^2 + 2y + x e^{2x}) dx; \quad y(-2)=0; \quad y(0)=1;$

b). $J(y) = \int_{-1}^1 (y'^2 - 2y' e^x + \cos x) dx; \quad y(-1)=2; \quad y(1)=3;$

c). $J(y) = \int_1^3 y \sqrt{y'} dx; \quad y(1)=2; \quad y(3)=8;$

Вариант 6.

a). $J(y) = \int_0^1 (y'^2 - 9y^2 + 2y \sin x - x^2 e^x) dx; \quad y(0)=1; \quad y(1)=-1;$

b). $J(y) = \int_{-1}^1 \left(y'^2 - \frac{2y'}{1+x^2} + e^{2x} \right) dx; \quad y(-1)=0; \quad y(1)=3;$

c). $J(y) = \int_0^2 y \sqrt{1+y'^2} dx; \quad y(0)=-1; \quad y(2)=-3;$

Вариант 7.

a). $J(y) = \int_{-1}^1 (y'^2 + 4y^2 + 6y e^x + 2x \cos x) dx; \quad y(-1)=1; \quad y(1)=3;$

b). $J(y) = \int_{-1}^1 (y'^2 + 4y' e^x \cos x - \sin x) dx; \quad y(-1)=1; \quad y(1)=2;$

c). $J(y) = \int_0^2 y y'^2 dx; \quad y(0)=1; \quad y(2)=3;$

Вариант 8.

a). $J(y) = \int_{-1}^1 (y'^2 + y^2 + 4y e^x - x \sin x) dx; \quad y(-1)=1; \quad y(1)=3;$

b). $J(y) = \int_1^3 (y'^2 - y' \ln x + 2x) dx; \quad y(1)=2; \quad y(3)=-1;$

c). $J(y) = \int_0^2 \frac{\sqrt{1+y'^2}}{y} dx; \quad y(0)=4; \quad y(2)=2;$

Вариант 9.

a). $J(y) = \int_{-1}^1 (y'^2 + 4y^2 + 8ye^{2x} + 3x^2) dx; \quad y(-1)=1; \quad y(1)=3;$

b). $J(y) = \int_{-1}^1 (y' + y'^2 \cos^2 x - \sin^2 x) dx; \quad y(-1)=1; \quad y(1)=-2;$

c). $J(y) = \int_0^2 y \sqrt{y'} dx; \quad y(0)=2; \quad y(2)=4;$

Вариант 10.

a). $J(y) = \int_0^2 (2y'^2 + 2y^2 + y \cos x - 5x) dx; \quad y(0)=2; \quad y(2)=2;$

b). $J(y) = \int_1^3 (y' + y'^2 \sin^2 x + e^{2x}) dx; \quad y(1)=-1; \quad y(3)=4;$

c). $J(y) = \int_0^2 \sqrt{y(1+y'^2)} dx; \quad y(0)=2; \quad y(2)=1;$

Вариант 11.

a). $J(y) = \int_0^2 (2y'^2 + 2y^2 + xy \sin x + 6xe^x) dx; \quad y(0)=1; \quad y(2)=2;$

b). $J(y) = \int_0^2 (y' + y'^2 e^x - \sin x) dx; \quad y(0)=2; \quad y(2)=-1;$

c). $J(y) = \int_1^3 y \sqrt{1+y'^2} dx; \quad y(1)=2; \quad y(3)=3;$

Вариант 12.

a). $J(y) = \int_0^2 (2y'^2 - 2y^2 + y \sin 2x - x^2 \sin x) dx; \quad y(0)=-1; \quad y(2)=4;$

b). $J(y) = \int_{0.5}^{1.5} (y' + y'^2 \sin 2x - \cos 2x) dx; \quad y(0.5)=1; \quad y(1.5)=2;$

c). $J(y) = \int_1^3 y y'^2 dx; \quad y(1)=2; \quad y(3)=5;$

Вариант 13.

a). $J(y) = \int_0^2 (2y'^2 - 2y^2 + y \cos x + xe^{2x}) dx; \quad y(0)=1; \quad y(2)=2;$

b). $J(y) = \int_1^2 (y' + xy'^2 - x^2 y') dx; \quad y(1)=2; \quad y(2)=-1;$

c). $J(y) = \int_{0.5}^{2.5} \frac{\sqrt{1+y'^2}}{y} dx; \quad y(0.5)=2; \quad y(2.5)=1;$

Вариант 14.

a). $J(y) = \int_1^2 (2y'^2 - 2y^2 + ye^{2x} \sin 3x - x \sin x) dx; \quad y(1)=2; \quad y(2)=3;$

b). $J(y) = \int_{-1}^1 (y' + e^x y'^2 - xy') dx; \quad y(-1)=0; \quad y(3)=2;$

c). $J(y) = \int_1^3 y \sqrt{y'} dx; \quad y(1)=2; \quad y(3)=6;$

Вариант 15.

a). $J(y) = \int_{-1}^1 (2y'^2 + 2y^2 + ye^x + 4xe^{2x}) dx; \quad y(-1)=1; \quad y(1)=2;$

b). $J(y) = \int_{-1}^1 (y' + y'^2 \sec^2 x + xy') dx; \quad y(-1)=-1; \quad y(1)=0;$

c). $J(y) = \int_1^3 \sqrt{y(1+y'^2)} dx; \quad y(1)=1; \quad y(3)=4;$

Вариант 16.

a). $J(y) = \int_{-1}^1 (2y'^2 + 2y^2 + 3ye^x \cos x - 5x^2 e^{2x}) dx; \quad y(-1)=2; \quad y(1)=1;$

b). $J(y) = \int_0^2 (y' + y'^2 + x^2 y'^2) dx; \quad y(0)=1; \quad y(2)=-2;$

c). $J(y) = \int_1^4 y \sqrt{1+y'^2} dx; \quad y(1)=3; \quad y(4)=8;$

Вариант 17.

a). $J(y) = \int_{-1}^1 (2y'^2 + 2y^2 + ye^{2x} + 4 \sin x) dx; \quad y(-1)=4; \quad y(1)=3;$

b). $J(y) = \int_{-0.5}^{0.5} (y' + y'^2 \cos 2x - \sin 2x) dx; \quad y(-0.5)=1; \quad y(0.5)=0.5;$

c). $J(y) = \int_0^4 xy'^2 dx; \quad y(0)=2; \quad y(4)=4;$

Вариант 18.

a). $J(y) = \int_{-1}^1 (2y'^2 + 2y^2 + 3ye^{2x} \sin x - 5 \cos x) dx; \quad y(-1)=3; \quad y(1)=4;$

b). $J(y) = \int_{-1}^1 (y' + y'^2 e^{2x} - 2xy') dx; \quad y(-1)=2; \quad y(1)=1;$

c). $J(y) = \int_1^4 \frac{\sqrt{1+y'^2}}{y} dx; \quad y(1)=4; \quad y(4)=2;$

Вариант 19.

a). $J(y) = \int_0^2 (y'^2 + 4y^2 + 6ye^{2x} \cos x - x^2) dx; \quad y(0)=1; \quad y(2)=3;$

b). $J(y) = \int_{0.5}^{1.5} ((2-6x)y' + y'^2 \cos^2 x) dx; \quad y(0.5)=-1; \quad y(1.5)=-2;$

c). $J(y) = \int_0^4 y \sqrt{y'} dx; \quad y(0)=2; \quad y(4)=5;$

Вариант 20.

a). $J(y) = \int_0^2 (y'^2 + 4y^2 + 4ye^x \sin x + x^2 \sin x) dx; \quad y(0)=2; \quad y(2)=3;$

b). $J(y) = \int_0^2 \left(y'^2 - \frac{2y'}{\sqrt{1+x^2}} + \sin 3x \right) dx; \quad y(0) = -1; \quad y(2) = 3;$

c). $J(y) = \int_0^4 \sqrt{y(1+y'^2)} dx; \quad y(0) = 4; \quad y(4) = 1;$

Вариант 21.

a). $J(y) = \int_0^2 (y'^2 + 4y^2 - 8y \cos x + 4x^2) dx; \quad y(0) = 1; \quad y(2) = 3;$

b). $J(y) = \int_2^4 \left(y'^2 + \frac{2y'}{1-x^2} + e^{3x} \right) dx; \quad y(2) = -1; \quad y(4) = 2;$

c). $J(y) = \int_1^5 y \sqrt{1+y'^2} dx; \quad y(1) = 3; \quad y(5) = 6;$

Вариант 22.

a). $J(y) = \int_2^4 (y'^2 - 4y^2 + 4y \cos 2x - 3x^2) dx; \quad y(2) = 1; \quad y(4) = 4;$

b). $J(y) = \int_{2.5}^4 \left(y'^2 + \frac{4y'}{4-x^2} + \sin 2x \right) dx; \quad y(2.5) = 2; \quad y(4) = 4;$

c). $J(y) = \int_1^5 y y'^2 dx; \quad y(1) = 2; \quad y(5) = 6;$

Вариант 23.

a). $J(y) = \int_0^2 (y'^2 - 4y^2 + 4ye^x \sin 2x + x^2) dx; \quad y(0) = 2; \quad y(2) = 3;$

b). $J(y) = \int_2^4 (\sqrt{x^2 - 1} y'^2 + y^4 - e^x) dx; \quad y(2) = 1; \quad y(4) = -2;$

c). $J(y) = \int_1^5 \frac{\sqrt{1+y'^2}}{y} dx; \quad y(1) = 4; \quad y(5) = 2;$

Вариант 24.

a). $J(y) = \int_{-1}^0 (y'^2 - 9y^2 + 4y \sin 3x + 5x^2) dx; \quad y(-1) = 2; \quad y(0) = 0;$

b). $J(y) = \int_1^3 (y'^2 + 2xy' \ln x - \ln x) dx; \quad y(1) = -1; \quad y(3) = 2;$

c). $J(y) = \int_1^5 y \sqrt{y'} dx; \quad y(1) = 1; \quad y(5) = 7;$

Вариант 25.

a). $J(y) = \int_0^1 (y'^2 - 9y^2 + 4ye^{2x} \cos 3x) dx; \quad y(0) = 3; \quad y(1) = 2;$

b). $J(y) = \int_{-1}^1 (y'^2 + 2y'e^x \sin x - e^x \cos x) dx; \quad y(-1) = 2; \quad y(1) = 3;$

c). $J(y) = \int_1^5 \sqrt{y(1+y'^2)} dx; \quad y(1) = 6; \quad y(5) = 1;$

Вариант 26.

a). $J(y) = \int_0^2 (y'^2 + 4y^2 - 8y \cos x + 4x^2) dx; \quad y(0) = 1; \quad y(2) = 3;$

b). $J(y) = \int_2^4 \left(y'^2 + \frac{2y'}{1-x^2} + e^{3x} \right) dx; \quad y(2) = -2; \quad y(4) = 2;$

c). $J(y) = \int_2^5 y \sqrt{1+y'^2} dx; \quad y(2) = 4; \quad y(5) = 6;$

Вариант 27.

a). $J(y) = \int_2^4 (y'^2 - 4y^2 + 4y \cos 2x - 3x^2) dx; \quad y(2) = 2; \quad y(4) = 3;$

b). $J(y) = \int_3^5 \left(y'^2 + \frac{4y'}{4-x^2} + \sin 3x \right) dx; \quad y(3) = 5; \quad y(5) = 1;$

c). $J(y) = \int_2^5 y y'^2 dx; \quad y(2) = 3; \quad y(5) = 6;$

Вариант 28.

a). $J(y) = \int_0^2 \left(y'^2 - 4y^2 + 4y e^x \sin \frac{x}{2} + 2x^2 \right) dx; \quad y(0) = 1; \quad y(2) = 3;$

b). $J(y) = \int_2^4 (\sqrt{x^2 - 1} y'^2 + 2y' - e^x) dx; \quad y(2) = 1; \quad y(4) = -2;$

c). $J(y) = \int_2^5 \frac{\sqrt{1+y'^2}}{y} dx; \quad y(2) = 3; \quad y(5) = 2;$

Вариант 29.

a). $J(y) = \int_0^1 (y'^2 - 9y^2 + 6y \sin 3x + 5x^2) dx; \quad y(0) = 3; \quad y(1) = 1;$

b). $J(y) = \int_1^4 (y'^2 + 2xy' \ln x - \ln x) dx; \quad y(1) = -1; \quad y(4) = 3;$

c). $J(y) = \int_2^5 y \sqrt{y'} dx; \quad y(2) = 4; \quad y(5) = 7;$

Вариант 30.

a). $J(y) = \int_0^3 (y'^2 + 4y^2 + 4y e^x \sin x + x^2 \sin x) dx; \quad y(0) = 2; \quad y(3) = 4;$

b). $J(y) = \int_0^3 \left(y'^2 - \frac{2y'}{\sqrt{1+x^2}} + \sin 3x \right) dx; \quad y(0) = -1; \quad y(3) = 2;$

c). $J(y) = \int_0^3 y \sqrt{1+y'^2} dx; \quad y(0) = 4; \quad y(3) = 2;$

2. Экстремаль функционала, зависящего от нескольких функций

2.1. Краткие теоретические сведения

Рассмотрим задачу отыскания экстремума функционала, зависящего от нескольких функций одной переменной и их первых производных:

$$J(y_1, y_2, \dots, y_n) = \int_{x_1}^{x_2} F(x, y_1, y_2, \dots, y_n, y'_1, y'_2, \dots, y'_n) dx \rightarrow \text{extr}. \quad (2.1)$$

Экстремаль функционала должна удовлетворять в общем случае $2n$ граничным условиям:

$$\begin{cases} y_1(x_1) = y_{11}; & y_2(x_1) = y_{21}; & \dots & y_n(x_1) = y_{n1}; \\ y_1(x_2) = y_{12}; & y_2(x_2) = y_{22}; & \dots & y_n(x_2) = y_{n2}. \end{cases} \quad (2.2)$$

Как и ранее (см. глава 1), необходимым условием экстремума такого функционала – равенство нулю его вариации, вычисленной на экстремали $\{y_{10}(x), y_{20}(x), \dots, y_{n0}(x)\}$. Так как функционал достигает экстремума на множестве варьируемых функций с закреплёнными концами, то он тем более будет достигать экстремума на более узком классе функций: когда варьируется только одна из них, а остальные зафиксированы. Значит, каждая из функций $y_i(x)$ должна удовлетворять уравнению Эйлера (1.5). Таким образом, для нахождения экстремалей мы получаем систему n дифференциальных уравнений Эйлера

$$\begin{cases} F_{y_i} - \frac{dF_{y'_i}}{dx} = 0; \\ i = \overline{1, n} \end{cases} \quad (2.3)$$

дополненную $2n$ граничными условиями (2.2).

Данную вариационную задачу при $n=2$ можно интерпретировать геометрически. Обозначим функции через $y(x)$ и $z(x)$. Тогда задача сводится к отысканию линии, проходящей через заданные две точки $M_1(x_1, y_1, z_1)$ и $M_2(x_2, y_2, z_2)$, и доставляющей экстремум функционалу (2.1).

2.2. Пример выполнения задания

Найти экстремаль функционала, зависящего от двух функций, при заданных граничных условиях, и построить график экстремали в виде двух функций $y(x)$, $z(x)$ и в виде пространственной кривой.

$$J(y, z) = \int_{-2}^2 (y'^2 + z'^2 + 2yz) dx; \quad \begin{cases} y(-2) = 1; \\ y(2) = 0; \end{cases} \quad \begin{cases} z(-2) = 0; \\ z(2) = 2. \end{cases} \quad (2.4)$$

Применим систему **MATLAB** для решения данной задачи. При написании будем использовать фрагменты программы решения примера 1а.

Вначале опишем необходимые данные и введём их.

```
clear all
format long
disp('Решаем пример 2')
syms x y z Dy D2y Dz D2z % описали переменные
F=Dy^2+Dz^2+2*y*z; % подынтегральная функция
x1=-2;
y1=1;
z1=0;
x2=2;
y2=0;
z2=2;
fprintf('Подынтегральная функция: F=%s\n', char(F))
fprintf('Граничные условия слева: y(%d)=%d; z(%d)=%d\n', x1, y1, x1, z1)
fprintf('Граничные условия справа: y(%d)=%d; z(%d)=%d\n', x2, y2, x2, z2)
Решаем пример 2
Подынтегральная функция: F=Dy^2+Dz^2+2*y*z
Граничные условия слева: y(-2)=1; z(-2)=0
Граничные условия справа: y(2)=0; z(2)=2
```

Находим частные производные F_y , F_z , $F_{y'}$ и $F_{z'}$, и вычисляем полные производные dF_y/dx и dF_z/dx по правилу (1.7).

```
dFdy = diff(F, y)
dFdy1 = diff(F, Dy);
```

```
d_dFdy1_dx = diff(dFdy1,x);
d_dFdy1_dy = diff(dFdy1,y);
d_dFdy1_dy1 = diff(dFdy1,Dy);
d_dFdy1_dz = diff(dFdy1,z);
d_dFdy1_dz1 = diff(dFdy1,Dz);
dFy1dx = d_dFdy1_dx + d_dFdy1_dy*Dy + d_dFdy1_dy1*D2y + d_dFdy1_dz*Dz +
d_dFdy1_dz1*D2z
dFdZ=diff(F,z)
dFdZ1 = diff(F,Dz);
d_dFdZ1_dx = diff(dFdZ1,x);
d_dFdZ1_dy = diff(dFdZ1,y);
d_dFdZ1_dy1 = diff(dFdZ1,Dy);
d_dFdZ1_dz = diff(dFdZ1,z);
d_dFdZ1_dz1 = diff(dFdZ1,Dz);
dFz1dx = d_dFdZ1_dx + d_dFdZ1_dy*Dy + d_dFdZ1_dy1*D2y + d_dFdZ1_dz*Dz +
d_dFdZ1_dz1*D2z
dFdy =
2*z
dFy1dx =
2*D2y
dFdZ =
2*y
dFz1dx =
2*D2z
```

Из этих величин формируем систему уравнений Эйлера

```
EulerY = simple(dFdy-dFy1dx)
EulerZ = simple(dFdZ-dFz1dx)
deqEulerY = [char(EulerY) '=0']; % уравнение Y
deqEulerZ = [char(EulerZ) '=0']; % уравнение Z
fprintf('Система уравнений Эйлера:\n%s\n%s\n',deqEulerY,deqEulerZ)
EulerY =
2*z-2*D2y
EulerZ =
2*y-2*D2z
Система уравнений Эйлера:
2*z-2*D2y=0
2*y-2*D2z=0
```

Решаем систему, проверяем существование и единственность решения, печатаем аналитические решения.

```
Sol = dsolve(deqEulerY,deqEulerZ,'x'); % решаем
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
SolY = Sol.y
SolZ = Sol.z
SolY =
1/4*C1*exp(-x)+1/4*C1*exp(x)+1/2*C1*cos(x)-1/4*C2*exp(-
x)+1/4*C2*exp(x)+1/2*C2*sin(x)+1/4*C3*exp(-x)+1/4*C3*exp(x)-1/2*C3*cos(x)-
1/2*C4*sin(x)+1/4*C4*exp(x)-1/4*C4*exp(-x)
SolZ =
1/4*C1*exp(-x)+1/4*C1*exp(x)-1/2*C1*cos(x)-1/2*C2*sin(x)+1/4*C2*exp(x)-
1/4*C2*exp(-x)+1/4*C3*exp(-x)+1/4*C3*exp(x)+1/2*C3*cos(x)-1/4*C4*exp(-
x)+1/4*C4*exp(x)+1/2*C4*sin(x)
```

Подставляем в полученное решение граничные условия, формируем систему уравнений и находим из её решения произвольные постоянные. Найденные константы подставляем в решение. Для ускорения решения системы уравнений все константы вычисляем перед решением системы с точностью до 14 знаков.

```
SolLeftY = subs(SolY,x,sym(x1)); % x1 в y
SolLeftZ = subs(SolZ,x,sym(x1)); % x1 в z
```

```

SolRightY = subs(SolY,x,sym(x2)); % x2 в y
SolRightZ = subs(SolZ,x,sym(x2)); % x2 в z
EqLeftY = [char(vpa(SolLeftY,14)) '=' char(sym(y1))];
EqLeftZ = [char(vpa(SolLeftZ,14)) '=' char(sym(z1))];
EqRightY = [char(vpa(SolRightY,14)) '=' char(sym(y2))];
EqRightZ = [char(vpa(SolRightZ,14)) '=' char(sym(z2))];
Con = solve(EqLeftY,EqLeftZ,EqRightY,EqRightZ);
C1 = Con.C1
C2 = Con.C2
C3 = Con.C3
C4 = Con.C4
Sol2Y=vpa(eval(Sol.y),14);
Sol2Z=vpa(eval(Sol.z),14);
fprintf('Уравнения экстремали: \ny(x)=%s\nz(x)=%s\n',char(Sol2Y),char(Sol2Z))
EqLeftY =
1.6730244272683*C1-2.2680789173363*C2+2.0891712638155*C3-1.3587814905107*C4=1
EqLeftZ =
2.0891712638155*C1-1.3587814905107*C2+1.6730244272683*C3-2.2680789173363*C4=0
EqRightY =
1.6730244272683*C1+2.2680789173364*C2+2.0891712638154*C3+1.3587814905107*C4=0
EqRightZ =
2.0891712638154*C1+1.3587814905107*C2+1.6730244272683*C3+2.2680789173364*C4=2
C1 =
.80010116205603772792330153473626
C2 =
-.75588248652807617767056096145498
C3 =
-.40139781880493258439008582742141
C4 =
.89374276891397209091577934075248
Уравнения экстремали:
y(x)=.65210765216302e-1*exp(-
1.*x)+.13414090640925*exp(x)+.60074949043049*cos(x)-.82481262772102*sin(x)
z(x)=.65210765216302e-1*exp(-1.*x)+.13414090640925*exp(x)-
.60074949043049*cos(x)+.82481262772102*sin(x)

```

Вычисляем значения функций. Рисуем графики функций: $y(x)$ сплошной красной линией и $z(x)$ пунктирной синей линией на одном рисунке.

```

xpl = linspace(x1,x2); % массив абсцисс
y2a = subs(Sol2Y,x,xpl); % вычисляем ординаты
z2a = subs(Sol2Z,x,xpl); % вычисляем ординаты
plot ( xpl,y2a,'-r', xpl,z2a,'--b' ) % рисуем график
title('\bfExample 2') % заголовок
xlabel('x') % метка оси ОХ
ylabel('y(x), z(x)') % метка оси ОУ

```

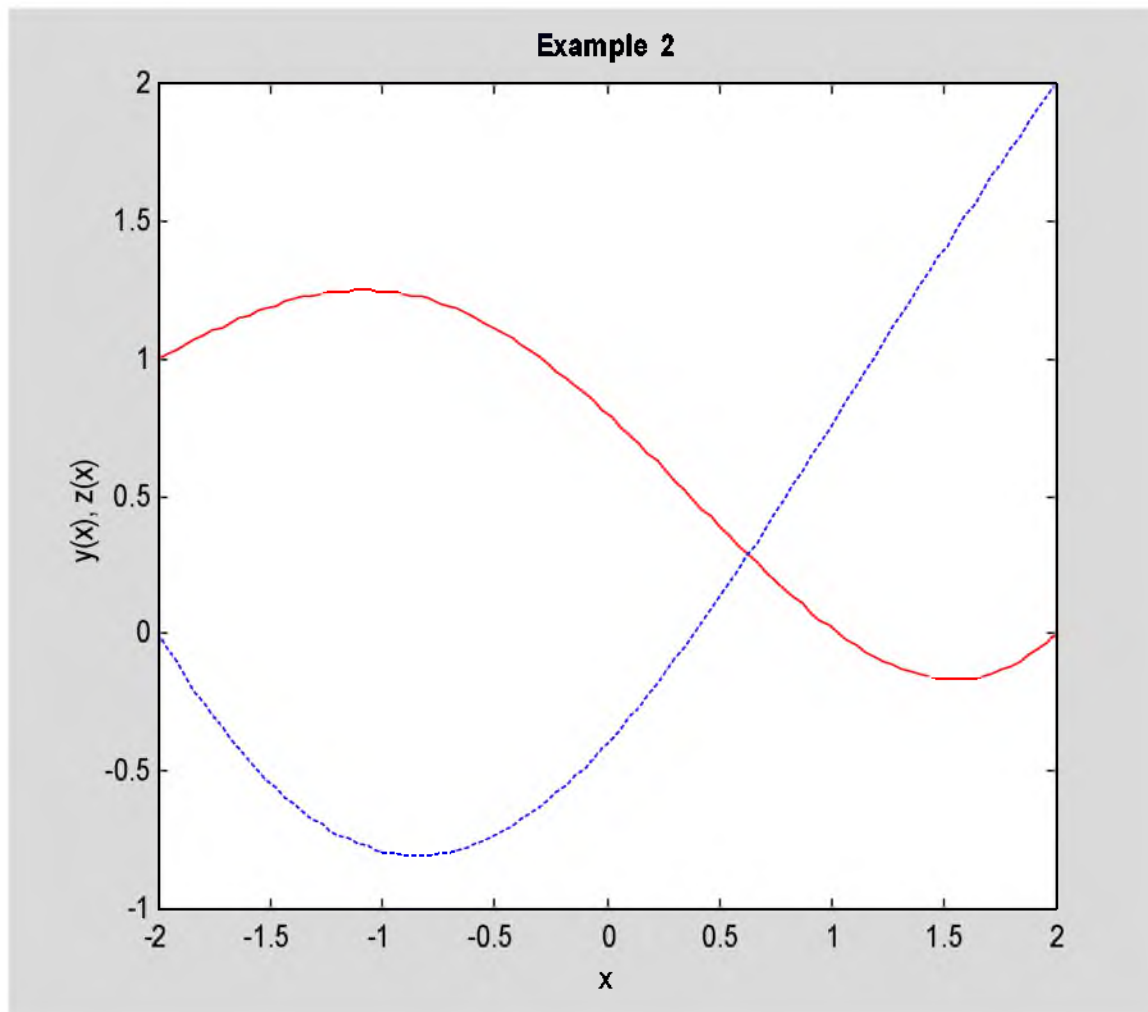


Рис. 2.1. Решение примера 2 в виде двух линий

На другом рисунке рисуем трёхмерный график полученной линии. Выбираем точку просмотра. Показываем сетку и контур.

```
plot3(xp1,y2a,z2a,'-r') % Рисуем 3D график
title('\bfExample 2') % заголовок
xlabel('x') % метка оси OX
ylabel('y(x)') % метка оси OY
zlabel('z(x)') % метка оси OZ
view(205,30) % точка просмотра
grid on % показали сетку
box on % показали внешний контур
```

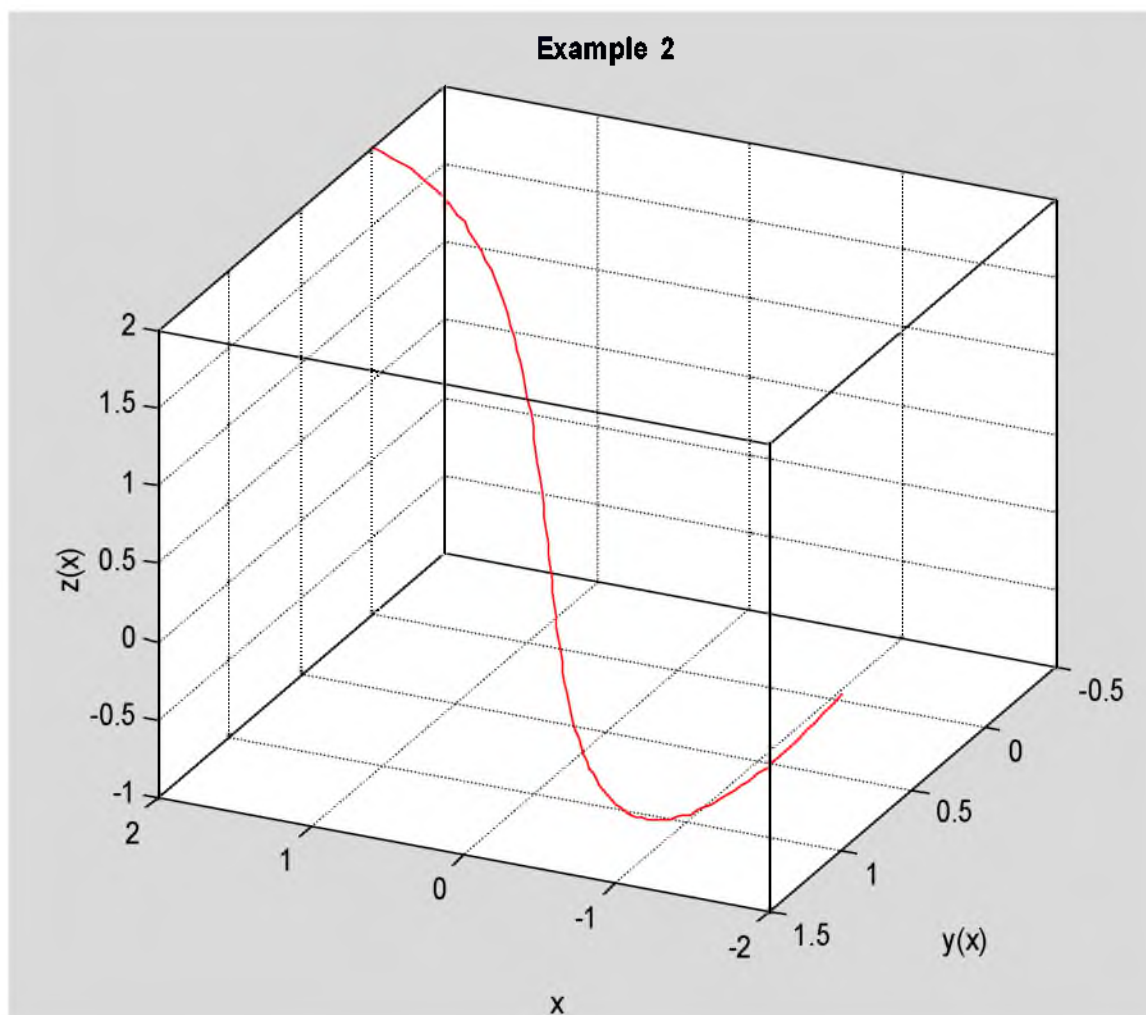


Рис. 2.2. Решение примера 2 в виде пространственной линии

Ответ. Система уравнений Эйлера после сокращения на -2 имеет вид

$$\begin{cases} y'' - z = 0; \\ z'' - y = 0. \end{cases} \quad (2.5)$$

После подстановки произвольных постоянных уравнения экстремали

$$\begin{cases} y = 0.06521e^{-x} + 0.13414e^x + 0.60075\cos x - 0.82481\sin x; \\ z = 0.06521e^{-x} + 0.13414e^x - 0.60075\cos x + 0.82481\sin x. \end{cases} \quad (2.6)$$

На рис.2.1 показаны двумерные графики линий $y(x)$ (сплошная красная линия) и $z(x)$ (пунктирная синяя), а на рис.2.2 – трёхмерный график пространственной кривой.

2.3. Задание

Для своего варианта функционала найти экстремаль и построить её график.

2.4. Варианты заданий

Вариант 1. $J(y, z) = \int_0^1 (2y'z' - y^2 + z^2 - 2ye^x) dx;$ $y(0)=0; \quad z(0)=1;$
 $y(1)=1; \quad z(1)=0;$

Вариант 2. $J(y, z) = \int_0^1 (2y'z' + y^2 + z^2 - z\sin x) dx;$ $y(0)=0; \quad z(0)=1;$
 $y(1)=1; \quad z(1)=0;$

Вариант 3. $J(y, z) = \int_0^1 (2y'z' + y^2 - z^2 + 2z\cos x) dx;$ $y(0)=0; \quad z(0)=1;$
 $y(1)=1; \quad z(1)=0;$

Вариант 4. $J(y, z) = \int_0^1 (y^2 + z^2 + 2y'z' + ye^x) dx;$ $y(0)=0; \quad z(0)=1;$
 $y(1)=1; \quad z(1)=0;$

- Вариант 5. $J(y, z) = \int_0^1 (y^2 + 4yz + z^2 + y'^2 + z'^2 + 2ze^x) dx$; $y(0)=0$; $z(0)=1$;
 $y(1)=1$; $z(1)=0$;
- Вариант 6. $J(y, z) = \int_0^1 ((y+z)^2 + y'^2 + z'^2 + 2y \sin x) dx$; $y(0)=0$; $z(0)=1$;
 $y(1)=1$; $z(1)=0$;
- Вариант 7. $J(y, z) = \int_0^1 ((y-z)^2 + y'^2 - z'^2 + 2z \cos x) dx$; $y(0)=0$; $z(0)=1$;
 $y(1)=1$; $z(1)=0$;
- Вариант 8. $J(y, z) = \int_{-1}^1 (2y'z' - y^3 + z^3 - 2y \cos x) dx$; $y(-1)=2$; $z(-1)=1$;
 $y(1)=0$; $z(1)=2$;
- Вариант 9. $J(y, z) = \int_{-1}^1 (2y'z' + y^2 + z^2 + 2ze^x) dx$; $y(-1)=3$; $z(-1)=0$;
 $y(1)=1$; $z(1)=2$;
- Вариант 10. $J(y, z) = \int_{-1}^1 (2y'z' + y^2 - z^2 - 2z \sin x) dx$; $y(-1)=2$; $z(-1)=0$;
 $y(1)=0$; $z(1)=2$;
- Вариант 11. $J(y, z) = \int_{-1}^1 (y^2 + z^2 - 2y'z' - y \cos x) dx$; $y(-1)=2$; $z(-1)=0$;
 $y(1)=0$; $z(1)=2$;
- Вариант 12. $J(y, z) = \int_{-1}^1 (y^2 + 4yz + z^2 - y'^2 - z'^2 + 2ye^{-x}) dx$; $y(-1)=2$; $z(-1)=0$;
 $y(1)=0$; $z(1)=2$;
- Вариант 13. $J(y, z) = \int_{-1}^1 ((y+z)^2 - y'^2 - z'^2 + 2xz) dx$; $y(-1)=2$; $z(-1)=0$;
 $y(1)=0$; $z(1)=2$;
- Вариант 14. $J(y, z) = \int_{-1}^1 ((y-z)^2 + y'^2 - z'^2 + 2xy) dx$; $y(-1)=1$; $z(-1)=0$;
 $y(1)=0$; $z(1)=2$;
- Вариант 15. $J(y, z) = \int_0^2 (2y'z' - y^3 + z^3 + 2y \sin x) dx$; $y(0)=1$; $z(0)=-2$;
 $y(2)=-1$; $z(2)=1$;
- Вариант 16. $J(y, z) = \int_0^2 (2y'z' + y^2 + z^2 + 2z \cos x) dx$; $y(0)=1$; $z(0)=-2$;
 $y(2)=-1$; $z(2)=1$;
- Вариант 17. $J(y, z) = \int_0^2 (2y'z' + y^2 - z^2 + 2ze^x) dx$; $y(0)=1$; $z(0)=-2$;
 $y(2)=-1$; $z(2)=1$;
- Вариант 18. $J(y, z) = \int_0^2 (y^2 + z^2 + 2y'z' + z \sin x) dx$; $y(0)=1$; $z(0)=-2$;
 $y(2)=-1$; $z(2)=1$;
- Вариант 19. $J(y, z) = \int_0^2 (y^2 + 4yz + z^2 - y'^2 - z'^2 + ze^{3x}) dx$; $y(0)=1$; $z(0)=-2$;
 $y(2)=-1$; $z(2)=1$;
- Вариант 20. $J(y, z) = \int_0^2 ((y+z)^2 - y'^2 - z'^2 + 2ye^{2x}) dx$; $y(0)=1$; $z(0)=-2$;
 $y(2)=-1$; $z(2)=1$;
- Вариант 21. $J(y, z) = \int_0^2 ((y-z)^2 + y'^2 - z'^2 + x^2 z) dx$; $y(0)=1$; $z(0)=-2$;
 $y(2)=-1$; $z(2)=1$;
- Вариант 22. $J(y, z) = \int_{-2}^2 (2y'z' - y^2 + z^2 + ze^{2x}) dx$; $y(-2)=0$; $z(-2)=2$;
 $y(2)=3$; $z(2)=1$;
- Вариант 23. $J(y, z) = \int_{-2}^2 (2y'z' + y^2 + z^2 + 2ye^{-x}) dx$; $y(-2)=0$; $z(-2)=2$;
 $y(2)=3$; $z(2)=1$;
- Вариант 24. $J(y, z) = \int_{-2}^2 (2y'z' + y^2 - z^2 + 2ye^x) dx$; $y(-2)=0$; $z(-2)=2$;
 $y(2)=3$; $z(2)=1$;
- Вариант 25. $J(y, z) = \int_{-2}^2 (y^2 + z^2 - 2y'z' + 2ze^{-x}) dx$; $y(-2)=0$; $z(-2)=2$;
 $y(2)=3$; $z(2)=1$;
- Вариант 26. $J(y, z) = \int_{-1}^2 (2y'z' + y^2 - z^2 + 2z \sin x) dx$; $y(-1)=2$; $z(-1)=0$;
 $y(2)=0$; $z(2)=2$;
- Вариант 27. $J(y, z) = \int_{-2}^2 (y^2 + z^2 + 2y'z' - y \cos x) dx$; $y(-2)=2$; $z(-2)=0$;
 $y(2)=0$; $z(2)=2$;

Вариант 28. $J(y, z) = \int_{-2}^2 (y^2 + 4yz + z^2 - y'^2 - z'^2 + 2ye^{-2x}) dx$, $y(-2) = 3$, $z(-2) = 0$,
 $y(2) = 1$, $z(2) = 2$;

Вариант 29. $J(y, z) = \int_{-2}^2 ((y+z)^2 - y'^2 - z'^2 + 4xz) dx$, $y(-2) = 2$, $z(-2) = 0$,
 $y(2) = 0$, $z(2) = 2$;

Вариант 30. $J(y, z) = \int_{-2}^2 ((y-z)^2 + y'^2 - z'^2 + 4xy) dx$, $y(-2) = 1$, $z(-2) = 0$,
 $y(2) = 0$, $z(2) = 2$;

3. Экстремаль функционала, зависящего от производных высших порядков

3.1. Краткие теоретические сведения

Исследуем на экстремум функционал, зависящий от функции одной переменной и её производных 1-го и 2-го порядка

$$J(y) = \int_{x_1}^{x_2} F(x, y, y', y'') dx \rightarrow \text{extr} \quad (3.1)$$

с заданными граничными условиями

$$\begin{cases} y(x_1) = y_1; & y'(x_1) = y'_1; \\ y(x_2) = y_2; & y'(x_2) = y'_2. \end{cases} \quad (3.2)$$

Как и для других задач, необходимым условием экстремума функционала (3.1) является равенство нулю его вариации, вычисленной на экстремали $y_0(x)$: $\delta J(y_0) = 0$. В силу граничных условий на концах интервала $\delta y(x_1) = \delta y(x_2) = \delta y'(x_1) = \delta y'(x_2) = 0$. Вычислим δJ как линейную часть приращения. Разложим первое слагаемое в ряд Тейлора в окрестности экстремали, и удержим только линейные члены. Слагаемое, содержащее 1-ю производную, проинтегрируем по частям один раз, а слагаемое, содержащее 2-ю производную – 2 раза.

$$\begin{aligned} \delta J(y_0) &= \int_{x_1}^{x_2} (F(x, y_0 + \delta y, y'_0 + \delta y', y''_0 + \delta y'') - F(x, y_0, y'_0, y''_0)) dx = \int_{x_1}^{x_2} (F_y \delta y + F_{y'} \delta y' + F_{y''} \delta y'') dx = \\ &= \int_{x_1}^{x_2} F_y \delta y dx + \left(F_{y'} \delta y \right) \Big|_{x_1}^{x_2} - \int_{x_1}^{x_2} \frac{dF_{y'}}{dx} \delta y dx + \left(F_{y''} \delta y' \right) \Big|_{x_1}^{x_2} - \int_{x_1}^{x_2} \frac{dF_{y''}}{dx} \delta y' dx = \\ &= \int_{x_1}^{x_2} \left(F_y - \frac{dF_{y'}}{dx} \right) \delta y dx - \left(\frac{dF_{y''}}{dx} \delta y' \right) \Big|_{x_1}^{x_2} + \int_{x_1}^{x_2} \frac{d^2 F_{y''}}{dx^2} \delta y dx = \int_{x_1}^{x_2} \left(F_y - \frac{dF_{y'}}{dx} + \frac{d^2 F_{y''}}{dx^2} \right) \delta y dx = 0. \end{aligned} \quad (3.3)$$

В силу произвольности вариации функции $\delta y(x)$ по основной лемме вариационного исчисления первый сомножитель под интегралом должен равняться нулю. Таким образом, экстремаль должна удовлетворять уравнению

$$F_y - \frac{dF_{y'}}{dx} + \frac{d^2 F_{y''}}{dx^2} = 0, \quad (3.4)$$

которое называется уравнением Эйлера-Пуассона. Оно является в общем случае уравнением 4-го порядка и дополняется граничными условиями (3.2).

Если функционал зависит от производных более высоких порядков, то уравнение Эйлера-Пуассона выводится аналогично. Оно будет иметь вид

$$F_y - \frac{dF_{y'}}{dx} + \frac{d^2 F_{y''}}{dx^2} - \dots + (-1)^n \frac{d^n F_{y^{(n)}}}{dx^n} = 0 \quad (3.5)$$

и дополняться $2n$ граничными условиями: значения искомой функции и её производных до $n-1$ -го порядка включительно на концах интервала x_1 и x_2 должны равняться заданным величинам.

3.2. Пример выполнения задания

Найти экстремаль функционала при заданных граничных условиях:

$$J(y) = \int_{-1}^1 (y'''^2 - 2y'^2 + 4yy' + y^2 - 2y \sin x) dx, \quad \begin{cases} y(-1) = 1; & y(1) = 2; \\ y'(-1) = -1; & y'(1) = -1. \end{cases} \quad (3.6)$$

Применим для решения задачи **MATLAB**. Вначале вводим исходные данные.

```
clear all
format long
disp('Решаем пример 3')
syms x y Dy D2y D3y D4y % описали переменные
F=D2y^2-2*Dy^2+4*y*Dy+y^2-2*y*sin(x); % функция
x1=-1;
y1=1;
Dy1=-1;
x2=1;
y2=2;
Dy2=-1;
```

```
fprintf('Подынтегральная функция: F=%s\n',char(F))
fprintf('Граничные условия слева: y(%d)=%d;   y' '(%d)=%d\n',x1,y1,x1,Dy1)
fprintf('Граничные условия справа: y(%d)=%d;   y' '(%d)=%d\n',x2,y2,x2,Dy2)
Решаем пример 3
Подынтегральная функция: F=D2y^2-2*Dy^2+4*y*Dy+y^2-2*y*sin(x)
Граничные условия слева: y(-1)=1;   y'(-1)=-1
Граничные условия справа: y(1)=2;   y'(1)=-1
```

Находим частные производные F_y , F_y' и F_{yy} .

```
dFdy = diff(F,y)
dFdy1 = diff(F,Dy)
dFdy2 = diff(F,D2y)
dFdy =
4*Dy+2*y-2*sin(x)
dFdy1 =
-4*Dy+4*y
dFdy2 =
2*D2y
```

Нам надо сформировать полные производные dF_y/dx и d^2F_y/dx^2 . Для их формирования используем формулу (1.7). Находим dF_y/dx .

```
d_dFdy1_dx = diff(dFdy1,x);
d_dFdy1_dy = diff(dFdy1,y);
d_dFdy1_dy1 = diff(dFdy1,Dy);
d_dFdy1_dy2 = diff(dFdy1,D2y);
dFy1dx = d_dFdy1_dx + d_dFdy1_dy*Dy + d_dFdy1_dy1*D2y + d_dFdy1_dy2*D3y
dFy1dx =
4*Dy-4*D2y
```

Дважды дифференцируем по x F_y' :

```
d_dFdy2_dx = diff(dFdy2,x);
d_dFdy2_dy = diff(dFdy2,y);
d_dFdy2_dy1 = diff(dFdy2,Dy);
d_dFdy2_dy2 = diff(dFdy2,D2y);
dFy2dx = d_dFdy2_dx + d_dFdy2_dy*Dy + d_dFdy2_dy1*D2y + d_dFdy2_dy2*D3y;
d_dFdy2dx_dx=diff(dFy2dx,x);
d_dFdy2dx_dy=diff(dFy2dx,y);
d_dFdy2dx_dy1 = diff(dFy2dx,Dy); % d((dFy')/dx)/dy'
d_dFdy2dx_dy2 = diff(dFy2dx,D2y); % d((dFy')/dx)/dy''
d_dFdy2dx_dy3 = diff(dFy2dx,D3y); % d((dFy')/dx)/dy'''
d2Fy2dx2 = d_dFdy2dx_dx + d_dFdy2dx_dy*Dy + d_dFdy2dx_dy1*D2y +
d_dFdy2dx_dy2*D3y + d_dFdy2dx_dy3*D4y
d2Fy2dx2 =
2*D4y
```

Формируем уравнение Эйлера-Пуассона.

```
Euler = simple(dFdy-dFy1dx+d2Fy2dx2)
deqEuler = [char(Euler) '=0']; % составили уравнение
fprintf('Уравнение Эйлера-Пуассона:\n%s\n',deqEuler)
Euler =
2*y-2*sin(x)+4*D2y+2*D4y
Уравнение Эйлера-Пуассона:
2*y-2*sin(x)+4*D2y+2*D4y=0
```

Находим общее решение, проверяем существование и единственность.

```
Sol = dsolve ( deqEuler, 'x' ) % решаем уравнение
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
```

```
end
Sol =
-1/8*sin(x)*x^2-
1/4*cos(x)*x+1/4*sin(x)+C1*sin(x)+C2*cos(x)+C3*sin(x)*x+C4*cos(x)*x
```

Вычисляем производную от полученного решения – она необходима для формирования граничных условий. Находим частное решение.

```
dydx = diff(Sol,x) % нашли производную
slY=subs(Sol,x,sym(x1)); % подставляем x1 в y(x)
slDY=subs(dydx,x,sym(x1)); % подставляем x1 в y'(x)
srY=subs(Sol,x,sym(x2)); % подставляем x2 в y(x)
srDY=subs(dydx,x,sym(x2)); % подставляем x2 в y'(x)
elY = [char(vpa(slY,14)) '=' char(sym(y1))]; % приравниваем y1
elDY = [char(vpa(slDY,14)) '=' char(sym(Dy1))]; % приравниваем Dy1
erY = [char(vpa(srY,14)) '=' char(sym(y2))]; % приравниваем y2
erDY = [char(vpa(srDY,14)) '=' char(sym(Dy2))]; % приравниваем Dy2
Con = solve(elY,elDY,erY,erDY); % решаем систему уравнений
C1 = Con.C1
C2 = Con.C2
C3 = Con.C3
C4 = Con.C4
Sol3 = vpa(eval(Sol),14); % подставляем C1-C4
fprintf('Уравнение экстремали:\ny(x)=%s\n',char(Sol3))
dydx =
-1/8*cos(x)*x^2+C1*cos(x)-C2*sin(x)+C3*cos(x)*x+C3*sin(x)-C4*sin(x)*x+C4*cos(x)
elY =
.2989170336605e-1-.84147098480790*C1+.54030230586814*C2+.84147098480790*C3-
.54030230586814*C4=1
elDY =
-.67537788233518e-1+.54030230586814*C1+.84147098480790*C2-1.3817732906760*C3-
.30116867893976*C4=-1
erY =
-.2989170336605e-
1+.84147098480790*C1+.54030230586814*C2+.84147098480790*C3+.54030230586814*C4=2
erDY =
-.67537788233518e-1+.54030230586814*C1-.84147098480790*C2+1.3817732906760*C3-
.30116867893976*C4=-1
C1 =
-.63119810539481762303733212277005
C2 =
1.4248525550551816404175707298948
C3 =
.86770535427108120537106182262121
C4 =
1.9637646983880351164662332538540
Уравнение экстремали:
y(x)=-.12500000000000*sin(x)*x^2+1.7137646983880*cos(x)*x-
.38119810539482*sin(x)+1.4248525550552*cos(x)+.86770535427108*sin(x)*x
```

Задаём массив аргументов, вычисляем функцию и печатаем её.

```
xpl = linspace(x1,x2); % задаём массив аргументов
y3=subs(Sol3,x,xpl); % вычисляем значения функции
plot ( xpl, y3, '-r' ) % рисуем
title ( '\bfExample 3' ) % заголовок
xlabel('x') % метка оси OX
ylabel('y(x)') % метка оси OY
```

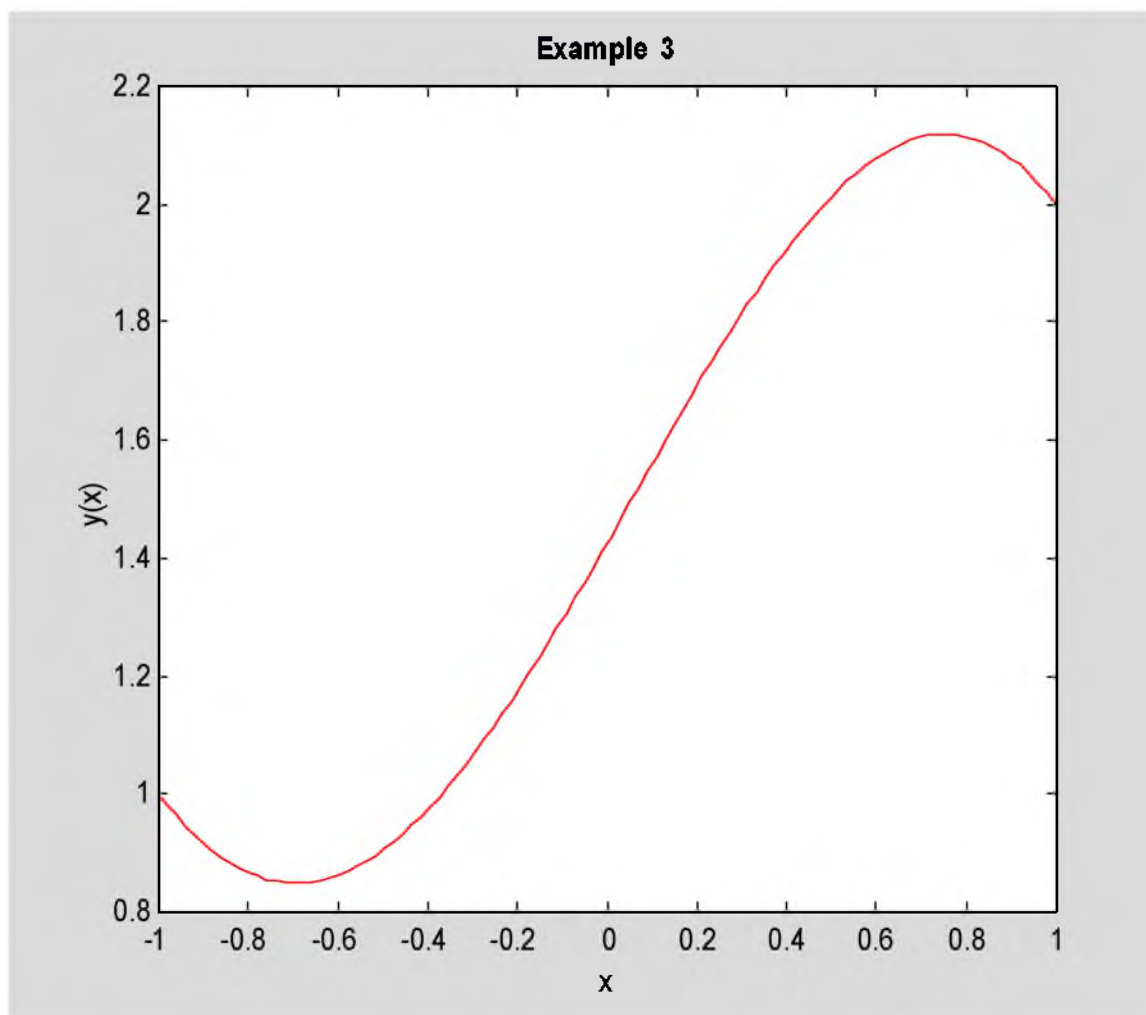


Рис. 3.1. Решение примера 3

Ответ. Уравнение Эйлера-Пуассона после сокращения на 2 имеет вид

$$y'' + 2y'' + y = \sin x. \quad (3.7)$$

Его частное решение, удовлетворяющее граничным условиям (3.2), будет

$$y = -0.125x^2 \sin x + 0.867705x \sin x + 1.71376x \cos x - 0.381198 \sin x + 1.42485 \cos x. \quad (3.8)$$

График решения показан на рис.3.1.

3.3. Задание

Для своего варианта функционала найти экстремаль и построить её график.

3.4. Варианты заданий

Вариант 1. $J(y) = \int_0^1 (y''^2 - 2y'^2 + y^2 - 2ye^x) dx$; $y(0)=2$; $y(1)=0$;
 $y'(0)=1$; $y'(1)=-1$;

Вариант 2. $J(y) = \int_0^1 (y''^2 - y'^2 + 2y \sin x) dx$; $y(0)=2$; $y(1)=0$;
 $y'(0)=1$; $y'(1)=-1$;

Вариант 3. $J(y) = \int_0^1 (y''^2 + 4y'y'' + y'^2 - 2ye^x) dx$; $y(0)=2$; $y(1)=0$;
 $y'(0)=1$; $y'(1)=-1$;

Вариант 4. $J(y) = \int_0^1 (y''^2 - y'y'' + y'^2 - 2ye^{-x}) dx$; $y(0)=2$; $y(1)=0$;
 $y'(0)=1$; $y'(1)=-1$;

Вариант 5. $J(y) = \int_0^1 (y''^2 - y'^2 - 4ye^{-x}) dx$; $y(0)=2$; $y(1)=0$;
 $y'(0)=1$; $y'(1)=-1$;

Вариант 6. $J(y) = \int_0^1 (y''^2 - y'^2 + yy' - yx) dx$; $y(0)=2$; $y(1)=0$;
 $y'(0)=1$; $y'(1)=-1$;

- Вариант 7. $J(y) = \int_0^1 (y'''^2 - 2y'^2 + y^2 + 2ye^{-x}) dx$; $y(0)=2$; $y(1)=0$;
 $y'(0)=1$; $y'(1)=-1$;
- Вариант 8. $J(y) = \int_0^1 (y'''^2 - y'^2 + ye^x) dx$; $y(0)=2$; $y(1)=0$;
 $y'(0)=1$; $y'(1)=-1$;
- Вариант 9. $J(y) = \int_0^1 (y'''^2 + 3y'y'' + y'^2 + 2xy) dx$; $y(0)=2$; $y(1)=0$;
 $y'(0)=1$; $y'(1)=-1$;
- Вариант 10. $J(y) = \int_0^1 (y'''^2 - 4y'y'' + y'^2 + 2y \sin x) dx$; $y(0)=2$; $y(1)=0$;
 $y'(0)=1$; $y'(1)=-1$;
- Вариант 11. $J(y) = \int_0^1 (y'''^2 - y'^2 + 2ye^x) dx$; $y(0)=2$; $y(1)=0$;
 $y'(0)=1$; $y'(1)=-1$;
- Вариант 12. $J(y) = \int_0^1 (y'''^2 - y'^2 + 4yy' + 2ye^{-x}) dx$; $y(0)=3$; $y(1)=1$;
 $y'(0)=0$; $y'(1)=1$;
- Вариант 13. $J(y) = \int_{-1}^1 (y'''^2 - 2y'^2 + y^2 + 2y \sin x) dx$; $y(-1)=1$; $y(1)=2$;
 $y'(-1)=-1$; $y'(1)=1$;
- Вариант 14. $J(y) = \int_{-1}^1 (y'''^2 - y'^2 + 2y \cos x) dx$; $y(-1)=1$; $y(1)=2$;
 $y'(-1)=-1$; $y'(1)=1$;
- Вариант 15. $J(y) = \int_{-1}^1 (y'''^2 + 4y'y'' + y'^2 + 2ye^{-x}) dx$; $y(-1)=1$; $y(1)=2$;
 $y'(-1)=-1$; $y'(1)=1$;
- Вариант 16. $J(y) = \int_{-1}^1 (y'''^2 - y'y'' + y'^2 + ye^x) dx$; $y(-1)=1$; $y(1)=2$;
 $y'(-1)=-1$; $y'(1)=1$;
- Вариант 17. $J(y) = \int_{-1}^1 (y'''^2 - y'^2 + 4xy) dx$; $y(-1)=1$; $y(1)=2$;
 $y'(-1)=-1$; $y'(1)=1$;
- Вариант 18. $J(y) = \int_{-1}^1 (y'''^2 - y'^2 + 2yy' + ye^x) dx$; $y(-1)=1$; $y(1)=2$;
 $y'(-1)=-1$; $y'(1)=1$;
- Вариант 19. $J(y) = \int_0^2 (y'''^2 - 2y'^2 + y^2 + 2ye^x) dx$; $y(0)=1$; $y(2)=4$;
 $y'(0)=0$; $y'(2)=-1$;
- Вариант 20. $J(y) = \int_0^2 (y'''^2 - y'^2 + 2ye^{-x}) dx$; $y(0)=1$; $y(2)=4$;
 $y'(0)=0$; $y'(2)=-1$;
- Вариант 21. $J(y) = \int_0^2 (y'''^2 + 3y'y'' + y'^2 + 4ye^x) dx$; $y(0)=1$; $y(2)=4$;
 $y'(0)=0$; $y'(2)=-1$;
- Вариант 22. $J(y) = \int_0^2 (y'''^2 - 4y'y'' + y'^2 + 4ye^{-x}) dx$; $y(0)=1$; $y(2)=4$;
 $y'(0)=0$; $y'(2)=-1$;
- Вариант 23. $J(y) = \int_0^2 (y'''^2 - y'^2 + 2ye^{-x}) dx$; $y(0)=1$; $y(2)=4$;
 $y'(0)=0$; $y'(2)=-1$;
- Вариант 24. $J(y) = \int_0^2 (y'''^2 - y'^2 + 2yy' + 2xy) dx$; $y(0)=1$; $y(2)=4$;
 $y'(0)=0$; $y'(2)=-1$;
- Вариант 25. $J(y) = \int_0^2 (y'''^2 - 2y'^2 + y^2 + 4xy) dx$; $y(0)=1$; $y(2)=4$;
 $y'(0)=0$; $y'(2)=1$;
- Вариант 26. $J(y) = \int_0^1 (y'''^2 - 4y'y'' + y'^2 + 2y \sin x) dx$; $y(0)=2$; $y(1)=0$;
 $y'(0)=1$; $y'(1)=-1$;
- Вариант 27. $J(y) = \int_0^1 (y'''^2 - y'^2 + 2ye^x) dx$; $y(0)=2$; $y(1)=0$;
 $y'(0)=1$; $y'(1)=-1$;
- Вариант 28. $J(y) = \int_0^1 (y'''^2 - y'^2 + 4yy' + 2ye^{-x}) dx$; $y(0)=3$; $y(1)=1$;
 $y'(0)=0$; $y'(1)=1$;
- Вариант 29. $J(y) = \int_{-1}^1 (y'''^2 - 2y'^2 + y^2 + 2y \sin x) dx$; $y(-1)=1$; $y(1)=2$;
 $y'(-1)=-1$; $y'(1)=1$;

Вариант 30. $J(y) = \int_{-1}^1 (y'''^2 - y^2 + 2y \cos x) dx;$ $y(-1)=1; \quad y(1)=2;$
 $y'(-1)=-1; \quad y'(1)=1;$

4. Экстремаль функционала, зависящего от функции нескольких переменных

4.1. Краткие теоретические сведения

Рассмотрим задачу исследования на экстремум функционала, зависящего от функции двух переменных и её частных производных 1-го порядка:

$$J(z(x,y)) = \iint_D F\left(x,y,z,\frac{\partial z}{\partial x},\frac{\partial z}{\partial y}\right) dS \rightarrow \text{extr} \quad (4.1)$$

с заданными условиями на контуре C – границе области D :

$$z(x,y)|_C = z_c(x,y). \quad (4.2)$$

Будем далее обозначать $p = \partial z / \partial x$, $q = \partial z / \partial y$. Необходимым условием экстремума функционала является равенство нулю его вариации, вычисленной на экстремали $z_0(x,y)$: $\delta J(z_0) = 0$. Найдём эту вариацию как линейную часть приращения функционала. Эта вариация вызывается вариациями функций z , p и q , причём на контуре C $\delta z = 0$. После разложения функции $F(x,y,z,p,q)$ в ряд Тейлора в окрестности экстремали и удержания линейных членов вариация функционала имеет вид

$$\delta J(z_0) = \iint_D (F_z \delta z + F_p \delta p + F_q \delta q) dS. \quad (4.3)$$

В главах 1 и 3 мы преобразовывали все слагаемые, кроме первого, с помощью интегрирования по частям. Здесь этот приём применить не удаётся, так как у нас двойной интеграл. Однако мы можем применить формулу Грина, дающую тот же результат. Заметим прежде всего, что

$$\frac{\partial}{\partial x}(F_p \delta z) = \frac{\partial F_p}{\partial x} \delta z + F_p \delta p; \quad \frac{\partial}{\partial y}(F_q \delta z) = \frac{\partial F_q}{\partial y} \delta z + F_q \delta q. \quad (4.4)$$

Здесь $\partial F_p / \partial x$ и $\partial F_q / \partial y$ – так называемые “полные частные производные”, то есть частные производные, вычисляемые при условии, что в дифференцируемой функции $z = z(x,y)$, $p = p(x,y)$, $q = q(x,y)$. Подставив (4.4) в (4.3), получим

$$\delta J(z_0) = \iint_D \left(\frac{\partial}{\partial x}(F_p \delta z) + \frac{\partial}{\partial y}(F_q \delta z) + F_z \delta z - \frac{\partial F_p}{\partial x} \delta z - \frac{\partial F_q}{\partial y} \delta z \right) dS. \quad (4.5)$$

Вычислим интеграл от первых двух слагаемых по формуле Грина, положив $Q(x,y) = F_p \delta z$, $P(x,y) = -F_q \delta z$.

$$\iint_D \left(\frac{\partial}{\partial x}(F_p \delta z) + \frac{\partial}{\partial y}(F_q \delta z) \right) dS = \iint_D \left(\frac{\partial Q}{\partial x} - \frac{\partial P}{\partial y} \right) dS = \oint_C P dx + Q dy = \oint_C F_p \delta z dy - F_q \delta z dx = 0, \quad (4.6)$$

так как на контуре C – границе области D : $\delta z = 0$. Таким образом, в интеграле (4.5) остаются только три последних слагаемых. В силу произвольности вариации функции $\delta z(x,y)$ по основной лемме вариационного исчисления должен быть равен нулю множитель при δz в подынтегральной функции:

$$F_z - \frac{\partial F_p}{\partial x} - \frac{\partial F_q}{\partial y} = 0. \quad (4.7)$$

Уравнение (4.7) называется уравнением Эйлера-Остроградского. Это дифференциальное уравнение в частных производных, оно дополняется граничным условием (4.2).

Если функционал зависит от функции n переменных $z(x_1, x_2, \dots, x_n)$, то уравнение Эйлера-Остроградского будет иметь вид

$$F_z - \sum_{k=1}^n \frac{\partial F_{p_k}}{\partial x_k} = 0, \quad (4.8)$$

где $p_k = \partial z / \partial x_k$.

4.2. Пример выполнения задания

Решить вариационную задачу

$$J(z) = \iint_D \left(z_x^2 - 2z_y^2 + 2yz \left(\sin \pi x + \frac{x}{5} \right) \right) dS \rightarrow \text{extr}, \quad D: \begin{cases} 0 \leq x \leq 1; \\ 0 \leq y \leq 2; \end{cases} \quad z|_C = \frac{x}{10} + \frac{y^2}{50}. \quad (4.9)$$

Составим программу для решения данной задачи на языке **MATLAB**.

```
clear all
format long
disp('Решаем пример 4')
syms x y z Dx Dz D2zx2 D2zxy D2zy2 % переменные
F = Dx^2 - 2*Dz^2 + 2*y*z*(sin(pi*x)+x/5); % подынтегральная функция
```

```

zc = x/10+y^2/50; % функция граничных условий
x1=0; % координаты ограничивающего прямоугольника
x2=1;
y1=0;
y2=2;
fprintf('Подынтегральная функция: F=%s\n',char(F))
fprintf('Граничное условие на контуре: z=%s\n',char(zc))
fprintf('Область: %d<=x<=%d; %d<=y<=%d\n',x1,x2,y1,y2)
Решаем пример 4
Подынтегральная функция: F=Dzx^2-2*Dzy^2+2*y*z*(sin(pi*x)+1/5*x)
Граничное условие на контуре: z=1/10*x+1/50*y^2
Область: 0<=x<=1; 0<=y<=2

```

Находим частные производные F_z , F_p и F_q .

```

dFdZ = diff(F,z)
dFdP = diff(F,Dzx)
dFdQ = diff(F,Dzy)
dFdZ =
2*y*(sin(pi*x)+1/5*x)
dFdP =
2*Dzx
dFdQ =
-4*Dzy

```

Сформируем полные частные производные $\partial F_p / \partial x$ и $\partial F_q / \partial y$. При их формировании учитываем, что $z=z(x,y)$, $p=p(x,y)$, $q=q(x,y)$. Используем формулу (1.7).

```

d_dFdP_dx = diff(dFdP,x);
d_dFdP_dz = diff(dFdP,z);
d_dFdP_dp = diff(dFdP,Dzx);
d_dFdP_dq = diff(dFdP,Dzy);
dFpdx = d_dFdP_dx + d_dFdP_dz*Dzx + d_dFdP_dp*D2zx2 + d_dFdP_dq*D2zxy
d_dFdQ_dy = diff(dFdQ,y);
d_dFdQ_dz = diff(dFdQ,z);
d_dFdQ_dp = diff(dFdQ,Dzx);
d_dFdQ_dq = diff(dFdQ,Dzy);
dFqdy = d_dFdQ_dy + d_dFdQ_dz*Dzy + d_dFdQ_dp*D2zxy + d_dFdQ_dq*D2zy2
dFpdx =
2*D2zx2
dFqdy =
-4*D2zy2

```

Формируем уравнение Эйлера-Остроградского.

```

Euler = simple(dFdZ-dFpdx-dFqdy)
EuR = -subs(Euler,{z,D2zx2,D2zy2,D2zxy},{0,0,0,0}); % правая часть
EuL = Euler+EuR; % левая часть
deqEuler = [ char(EuL) '=' char(EuR) ]; % уравнение
fprintf('Уравнение Эйлера-Остроградского:\n%s\n',deqEuler)
Euler =
2*y*sin(pi*x)+2/5*y*x-2*D2zx2+4*D2zy2
Уравнение Эйлера-Остроградского:
-2*D2zx2+4*D2zy2=-2*y*sin(pi*x)-2/5*y*x

```

Для решения дифференциальных уравнений в частных производных в **MATLAB**'е есть специальный инструментальный – Partial Differential Equation Toolbox (PDE), в котором используется метод конечных элементов (FEM). Для применения PDE нужно привести дифференциальное уравнение к виду

$$-\operatorname{div}(C \operatorname{grad} u) + au = f \quad (4.10)$$

и дополнить его граничными условиями типа Дирихле

$$hu = r \quad (4.11)$$

или Неймана

$$\operatorname{div}(C \operatorname{gradu}) + qu = g. \quad (4.12)$$

Здесь $u(x,y)$ – искомая функция, C – матрица 2×2 , элементы которой являются коэффициентами при $\partial^2 u / \partial x^2$, $\partial^2 u / \partial x \partial y$, $\partial^2 u / \partial y^2$; a – коэффициент при u ; f – правая часть. h, r, q, g – заданные функции x, y . Величины C, a, f, h, r, q, g могут быть как постоянными, так и переменными. В последнем случае они должны вычисляться в центрах тяжести конечных элементов и задаваться как массивы.

Процесс решения дифференциального уравнения в частных производных при помощи PDE состоит из следующих этапов.

- задание геометрии (области решения) и построение FEM-сетки;
- задание граничных условий (функций h, r, q, g);
- задание функций, входящих в дифференциальное уравнение (C, a, f);
- решение дифференциального уравнения;
- отображение результатов в виде графика.

Задать геометрию в виде прямоугольника проще всего с помощью команды **poimesh**. Она формирует сетку на квадрате $[-1,1] \times [-1,1]$. Эта команда возвращает 3 выходных параметра: **p**, **e** и **t**. В переменной **p** возвращаются координаты узлов сформированной сетки: 1-я строка – x -е. 2-я – y -е. Массив **p** имеет размеры $2 \times n_p$, где n_p – число узлов. В переменной **t** возвращаются данные по треугольникам. Это массив размером $4 \times n_{el}$, где n_{el} – число элементов. Первые 3 строки содержат номера узлов для каждого элемента в порядке обхода против часовой стрелки. 4-е число в нашем случае – всегда 1. В переменной **e** возвращаются данные по граничным точкам сетки. Размер массива **e** $7 \times n_b$, где n_b – число участков границы в FEM-сетке. В каждом столбце массива **e** содержатся данные по одной граничной линии. 1-е и 2-е числа – это номера узлов (в том порядке, в котором они перечислены в массиве **p**).

Сформируем треугольную FEM – сетку и нарисуем её. Напечатаем количество узлов и элементов.

```
del=min(x2-x1,y2-y1)/20 % размер элемента
nx=round((x2-x1)/del)
ny=round((y2-y1)/del)
[p,e,t]=poimesh('squareg',nx,ny); % формируем FEM-сетку
p(1,:)=(p(1,:)+1)/2*(x2-x1)+x1;
p(2,:)=(p(2,:)+1)/2*(y2-y1)+y1; % приводим к прямоугольной
np=size(p,2); % число узлов
nel=size(t,2); % число элементов
fprintf('Число узлов сетки разбиения np=%d\n',np)
fprintf('Число конечных элементов nel=%d\n',nel)
pdemesh(p,e,t) % рисуем сетку
da=daspect; % текущие масштабы осей
da(1:2)=min(da(1:2)); % выравниваем масштабы
daspect(da); % установили одинаковые масштабы
title('\bfFEM Grid') % заголовок
xlabel('x')
ylabel('y') % метки осей
del =
    0.0500000000000000
nx =
    20
ny =
    40
Число узлов сетки разбиения np=861
Число конечных элементов nel=1600
```

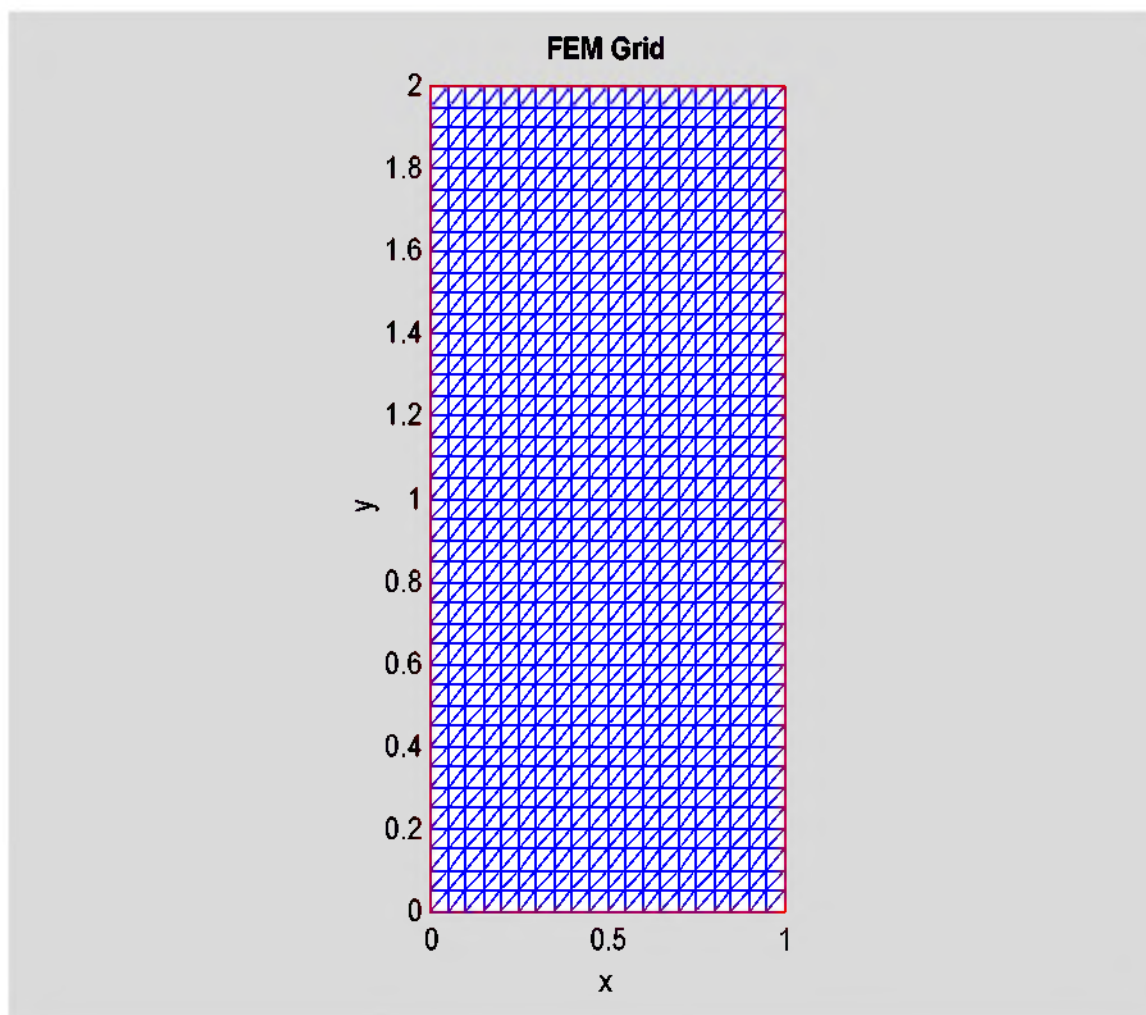


Рис. 4.1. Разбивка области на конечные элементы

Следующий этап – это задание **граничных условий**. Граничные условия можно задать или в виде матрицы граничных условий (boundary condition matrix), или в виде *.m-файла граничных условий (boundary M-file). Второй вариант проще. Файл граничных условий должен иметь такую структуру:

```
[q,g,h,r]=pdebound(p,e,u,time)
```

Входные параметры: **p,e** – данные по сетке разбиения, **u** – решение, **time** – время. Выходные параметры – матрицы граничных условий Дирихле (4.11) или Неймана (4.12). PDE позволяют решать параболические и гиперболические уравнения (зависящие от времени), а также нелинейные задачи, поэтому граничные условия и параметры дифференциального уравнения могут зависеть также от времени и решения.

Для граничных условий Неймана матрицы **q** и **g** должны содержать значения параметров q и g в средних точках границ. Размер матрицы **q**: $N^2 \times n_b$, где N – число уравнений системы, а n_b – число сторон конечных элементов, выходящих на границу. Размер матрицы **g**: $N \times n_b$. В каждом из столбцов этих матриц должны возвращаться коэффициенты q и g в средних точках соответствующей границы. Для нескольких уравнений элементы q должны следовать по столбцам. Так, например, для 2-х уравнений в каждом столбце матрицы **q** нужно вернуть q_{11} , q_{21} , q_{12} , q_{22} . В случае граничных условий Дирихле в этих матрицах должны возвращаться нулевые значения.

Для граничных условий Дирихле должны формироваться матрицы **h** и **r**. Матрица **h** имеет размеры $N^2 \times (2n_b)$, и содержит значения h сперва во всех начальных точках каждой границы, и сразу за ними – в конечных точках границ. Размер матрицы **r**: $N \times (2n_b)$, этот массив заполняется аналогично.

Сформируем файл для вычисления граничных условий Дирихле. Число уравнений у нас $N=1$, число граничных точек n_b находим из массива **e**. Формируем строки для записи в файл и записываем их. Файл должен быть размещён в каком-либо каталоге, доступном системе **MATLAB**.

```
s{1}='function [q,g,h,r]=boundmem(p,e,u,time)'; % заголовок  
s{2}='nb=size(e,2)'; % определили размерности  
s{3}='q=zeros(1,nb); g=zeros(1,nb)'; % задали массивы условий Неймана  
s{4}='h=ones(1,2*nb); r=zeros(1,2*nb)'; % задали массивы условий Дирихле  
s{5}='xb=[p(1,e(1,:)),p(1,e(2,:))];' % столбец координат x
```

```

s{6}='yb=[p(2,e(1,:)),p(2,e(2,:))];'; % столбец координат y
zcf=subs(zc,{x,y},{sym('xb'),sym('yb')}); % формула для подстановки
s{7}=['r=' vectorize(zcf) ''];
disp('Текст файла граничных условий boundmem.m:')
fprintf('%s\n',s{:})
fid = fopen('C:\Iglin\Matlab\boundmem.m','w');
fprintf(fid,'%s\n',s{:});
fclose(fid); % закрываем файл
Текст файла граничных условий boundmem.m:
function [q,g,h,r]=boundmem(p,e,u,time)
nb=size(e,2);
q=zeros(1,nb); g=zeros(1,nb);
h=ones(1,2*nb); r=zeros(1,2*nb);
xb=[p(1,e(1,:)),p(1,e(2,:))];
yb=[p(2,e(1,:)),p(2,e(2,:))];
r=1./10.*xb+1./50.*yb.^2;

```

Следующий этап – задание функций, входящих в дифференциальное уравнение. Каждая из функций C , a , f может быть задана в следующих видах:

- В виде константы.
- В виде вектор-строки значений функции в центрах масс треугольников. Если для матрицы C задаются 2 строки, то подразумевается, что C – диагональная, и задаются c_{11} и c_{22} . Если для матрицы C задаются 4 строки, то подразумевается, что это элементы матрицы c в порядке c_{11} , c_{21} , c_{12} , c_{22} .
- В виде текстового выражения, составленного по правилам MATLAB, по которому можно вычислить соответствующую функцию в центрах масс треугольников. Эта функция может зависеть от переменных x , y , sd , u , ux , uy , t . Смысл этих переменных: t – время (скаляр), остальные переменные – векторы-строки, представляющие значения в центрах масс треугольников: x , y – координаты центров масс, sd – номер подобласти, u , ux , uy – решение и его частные производные.
- В виде последовательности выражений, рассмотренных в предыдущем пункте. Эти выражения должны быть отделены друг от друга знаками `!`. Они воспринимаются как различные задания функции в различных подобластях. Этим выражений должно быть столько, сколько есть различных подобластей, т.е. `max(t(4,:))`, где t – массив, возвращаемый командой `initmesh`.
- В виде имени определённой пользователем функции MATLAB (то есть *.m-файла), который представляет функцию, зависящую от аргументов $(p,t,u,time)$, где p,t – данные по сетке разбиения, u – решение, $time$ – время.

Найдём C , a . Учтём, что коэффициенты при $\partial^2 u / \partial x \partial y$ и $\partial^2 u / \partial y \partial x$ одинаковые. Вычислим правую часть уравнения Эйлера (4.7) сначала аналитически, а потом в центрах тяжести конечных элементов.

```

a=eval(subs(EuL,{z,D2zx2,D2zxy,D2zy2},{1,0,0,0}))
c11=-eval(subs(EuL,{z,D2zx2,D2zxy,D2zy2},{0,1,0,0}));
c12=-eval(subs(EuL,{z,D2zx2,D2zxy,D2zy2},{0,0,1,0}))/2;
c22=-eval(subs(EuL,{z,D2zx2,D2zxy,D2zy2},{0,0,0,1}));
c=[c11;c12;c12;c22]
fp = subs(EuR,{x,y},{p(1,:),p(2,:)}); % f в узлах
f = (fp(t(1,:))+fp(t(2,:))+fp(t(3,:)))/3; % f в ц.т. конечных элементов
a =
0
c =
2
0
0
-4

```

Решаем уравнение. Рисуем график решения. Показываем конечноэлементную сетку. Надписываем оси. Выравниваем масштабы по осям Ox и Oy . Выбираем палитру. Показываем сетку и контур.

```

u = assempde('boundmem',p,e,t,c,a,f); % решили
pdeplot(p,e,t,'xydata',u,'zdata',u,'mesh','on','colorbar','off'); % рисуем
title('\bfExample 4') % заголовок
xlabel('x')
ylabel('y')
zlabel('z(x,y)')

```



```
v = axis; % границы осей
da = daspect; % масштаб осей
da(1:2) = min(da(1:2)); % min из двух
daspect(da) % выравнили масштаб осей
axis(v); % оставили границы
colormap(gray) % выбрали палитру
grid on % показали сетку
box on % показали внешний контур
```

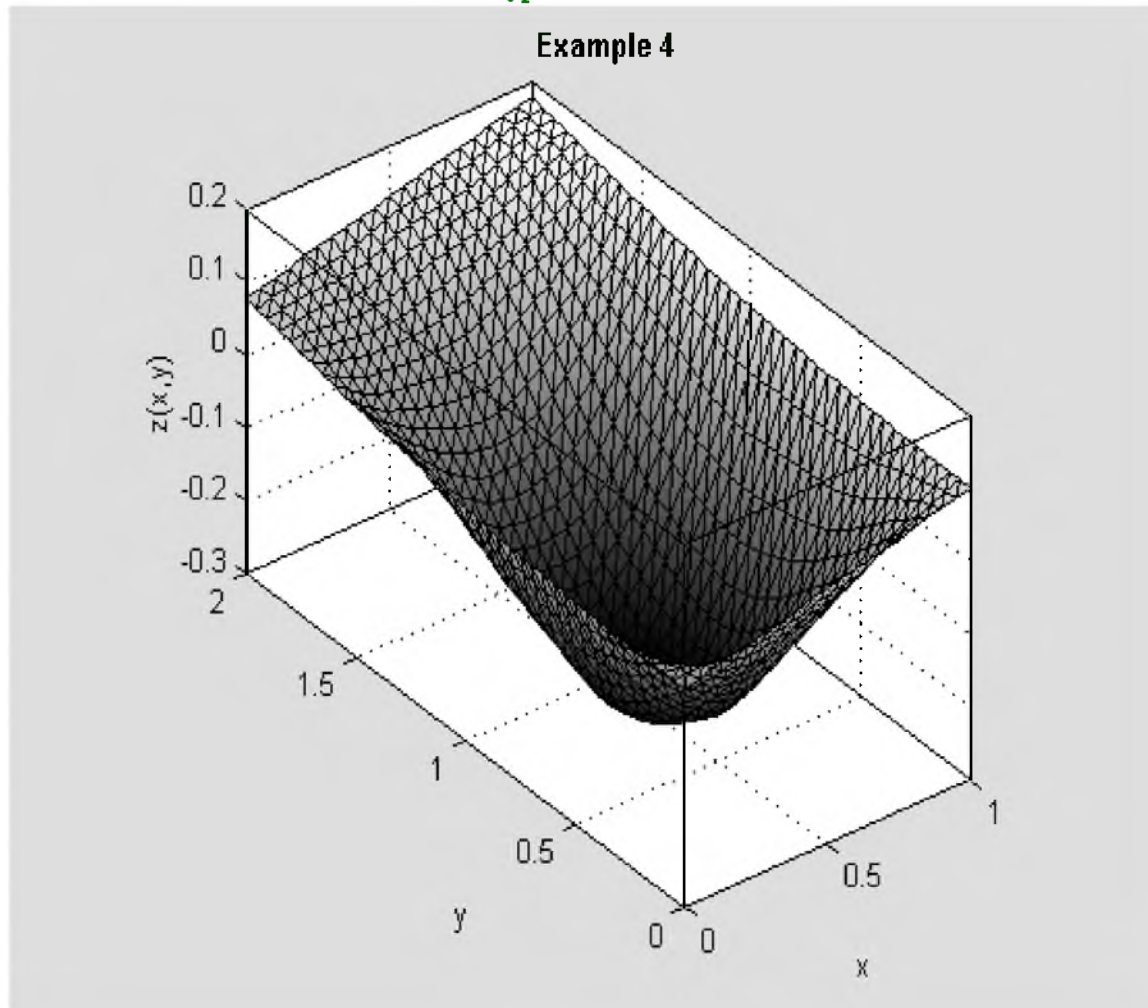


Рис. 4.2. Решение примера 4

Ответ. Дифференциальное уравнение Эйлера-Остроградского после сокращения на -2 имеет вид

$$\frac{\partial^2 z}{\partial x^2} - 2 \frac{\partial^2 z}{\partial y^2} = y \sin \pi x + \frac{xy}{5}. \quad (4.13)$$

Его решение методом конечных элементов приведено на рис.4.3.

4.3. Задание

Для своего варианта функционала вывести дифференциальное уравнение Эйлера-Остроградского и решить его МКЭ.

4.4. Варианты заданий

Вариант 1. $J(z) = \iint_D (z_x^2 + z_y^2 + 2xyz) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ 0 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2}{50} + \frac{y^2}{100}.$

Вариант 2. $J(z) = \iint_D \left(z_x^2 + 2z_y^2 + 2z \sin \pi x \sin \frac{\pi y}{2} \right) dS$; $D: \begin{cases} 0 \leq x \leq 1; \\ 0 \leq y \leq 2; \end{cases} \quad z|_C = \frac{x^2}{50} - \frac{y}{100}.$

Вариант 3. $J(z) = \iint_D (z_x^2 + 3z_y^2 + 2yz) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ 0 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2}{100} + \frac{y}{50}.$

Вариант 4. $J(z) = \iint_D (z_x^2 + 4z_y^2 + 2yz \sin x) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ 0 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2}{100} + \frac{y}{50}.$

Вариант 5. $J(z) = \iint_D (z_x^2 + 5z_y^2 + 2yz \sin x) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ 0 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2}{200} - \frac{y}{100}.$

Вариант 6. $J(z) = \iint_D (2z_x^2 + z_y^2 + 2yz \cos x) dS$; $D: \begin{cases} -1 \leq x \leq 1; \\ 0 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2}{100} + \frac{y}{200}.$

Вариант 7. $J(z) = \iint_D (3z_x^2 + z_y^2 + 2xz \cos y) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ -1 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2}{200} - \frac{y}{100}.$

Вариант 8. $J(z) = \iint_D (4z_x^2 + z_y^2 + 2xyz) dS$; $D: \begin{cases} -1 \leq x \leq 1; \\ 0 \leq y \leq 2; \end{cases} \quad z|_C = \frac{x^2 + y^2}{100}.$

Вариант 9. $J(z) = \iint_D (5z_x^2 + z_y^2 + 2xyz) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ -1 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x + y^2}{200}.$

Вариант 10. $J(z) = \iint_D (2z_x^2 + 3z_y^2 + 2x^2 yz) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ 0 \leq y \leq 2; \end{cases} \quad z|_C = \frac{x}{100} + \frac{y^2}{200}.$

Вариант 11. $J(z) = \iint_D (2z_x^2 - z_y^2 + 2xyz) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ 0 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2 + y^2}{100}.$

Вариант 12. $J(z) = \iint_D (3z_x^2 - z_y^2 + 2z \sin \pi x \sin \frac{\pi y}{2}) dS$; $D: \begin{cases} 0 \leq x \leq 1; \\ 0 \leq y \leq 2; \end{cases} \quad z|_C = \frac{x^2 - 2y}{200}.$

Вариант 13. $J(z) = \iint_D (2z_x^2 - z_y^2 + yz) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ 0 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2 + y}{100}.$

Вариант 14. $J(z) = \iint_D (5z_x^2 - z_y^2 + 2yz \sin x) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ 0 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2}{200} + \frac{y}{50}.$

Вариант 15. $J(z) = \iint_D (z_x^2 - 2z_y^2 + 2xz \sin \pi y) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ 0 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2}{200} - \frac{y}{50}.$

Вариант 16. $J(z) = \iint_D (z_x^2 - 3z_y^2 + 2yz \cos x) dS$; $D: \begin{cases} -1 \leq x \leq 1; \\ 0 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2}{100} + \frac{y}{50}.$

Вариант 17. $J(z) = \iint_D (3z_x^2 - 4z_y^2 + 2xz \cos y) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ -1 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2}{200} - \frac{y}{100}.$

Вариант 18. $J(z) = \iint_D (5z_x^2 - 4z_y^2 + 2xyz) dS$; $D: \begin{cases} -1 \leq x \leq 1; \\ 0 \leq y \leq 2; \end{cases} \quad z|_C = \frac{x^2}{100} + \frac{y^2}{200}.$

Вариант 19. $J(z) = \iint_D (4z_x^2 - 5z_y^2 + 2xyz) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ -1 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x}{100} + \frac{y^2}{50}.$

Вариант 20. $J(z) = \iint_D (2z_x^2 - 3z_y^2 + 2x^2 yz) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ 0 \leq y \leq 2; \end{cases} \quad z|_C = \frac{x}{50} + \frac{y^2}{80}.$

Вариант 21. $J(z) = \iint_D (2z_x^2 - z_y^2 + z^2 + 2xyz) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ 0 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2}{100} + \frac{y^2}{160}.$

Вариант 22. $J(z) = \iint_D (3z_x^2 - z_y^2 + z^2 + 2z \sin \pi x \sin \frac{\pi y}{2}) dS$; $D: \begin{cases} 0 \leq x \leq 1; \\ 0 \leq y \leq 2; \end{cases} \quad z|_C = \frac{x^2}{200} - \frac{y}{120}.$

Вариант 23. $J(z) = \iint_D (4z_x^2 - z_y^2 + z^2 + 2yz) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ 0 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2 + y}{100}.$

Вариант 24. $J(z) = \iint_D (5z_x^2 - z_y^2 + z^2 + 2yz \sin x) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ 0 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2}{200} + \frac{y}{100}.$

Вариант 25. $J(z) = \iint_D (z_x^2 - 2z_y^2 + z^2 + 2xz \sin \pi y) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ 0 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2}{200} - \frac{y}{80}.$

Вариант 26. $J(z) = \iint_D (2z_x^2 + z_y^2 + z^2 + 2yz \cos x) dS$; $D: \begin{cases} -1 \leq x \leq 1; \\ 0 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2}{100} + \frac{y}{60}.$

Вариант 27. $J(z) = \iint_D (z_x^2 + 2z_y^2 + z^2 + 2xz \cos y) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ -1 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x^2}{100} - \frac{y}{60}.$

Вариант 28. $J(z) = \iint_D (z_x^2 + 3z_y^2 + z^2 + 2xyz) dS$; $D: \begin{cases} -1 \leq x \leq 1; \\ 0 \leq y \leq 2; \end{cases} \quad z|_C = \frac{x^2}{200} + \frac{y^2}{100}.$

Вариант 29. $J(z) = \iint_D (z_x^2 + 4z_y^2 + z^2 + 2xyz) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ -1 \leq y \leq 1; \end{cases} \quad z|_C = \frac{x+y^2}{100}.$

Вариант 30. $J(z) = \iint_D (z_x^2 + 5z_y^2 + z^2 + 2x^2 yz) dS$; $D: \begin{cases} 0 \leq x \leq 2; \\ 0 \leq y \leq 2; \end{cases} \quad z|_C = \frac{x}{100} + \frac{y^2}{160}.$

5. Естественные граничные условия

5.1. Краткие теоретические сведения

Рассмотрим задачу вариационного исчисления для функционала (1.1), зависящего от функции одной переменной и её производной. Но, в отличие от (1.2), пусть задано граничное условие только на левом конце: $y(x_1)=y_1$. На правом конце x_2 значение функции может быть произвольным, то есть экстремум функционала (1.1) ищется на классе функций с закреплённым левым концом и свободным правым. Этот класс является более широким, чем рассмотренный в главе 1 класс функций с двумя закреплёнными концами.

Необходимым условием экстремума функционала является равенство нулю его вариации: $\delta J=0$. Так как функционал $J(y)$ достигает экстремума на классе функций с произвольным значением $y(x_2)$, то он тем более будет достигать экстремума на более узком классе функций: с неподвижной правой точкой. Следовательно, экстремаль должна удовлетворять дифференциальному уравнению Эйлера (1.5). Это уравнение дополняется граничным условием на левом конце: $y(x_1)=y_1$. Недостающее второе граничное условие находим из необходимого условия экстремума: $\delta J=0$. Вычислим вариацию функционала и приравняем её нулю. Разложим подынтегральную функцию F в ряд Тейлора в окрестности экстремали, и удержим в этом разложении только линейные члены. Проинтегрируем полученное выражение по частям и воспользуемся тем, что экстремаль удовлетворяет уравнению Эйлера.

$$\begin{aligned}\delta J(y_0) &= \int_{x_1}^{x_2} (F(x, y_0 + \delta y, y'_0 + \delta y') - F(x, y_0, y'_0)) dx = \int_{x_1}^{x_2} (F_y \delta y + F_{y'} \delta y') dx = \\ &= \int_{x_1}^{x_2} F_y \delta y dx + \left(F_{y'} \delta y \right) \Big|_{x_1}^{x_2} - \int_{x_1}^{x_2} \frac{dF_{y'}}{dx} \delta y dx = \left(F_{y'} \delta y \right) \Big|_{x_2} + \int_{x_1}^{x_2} \left(F_y - \frac{dF_{y'}}{dx} \right) \delta y dx = \left(F_{y'} \delta y \right) \Big|_{x_2} = 0.\end{aligned}\quad (5.1)$$

В силу произвольности $\delta y(x_2)$ получаем второе граничное условие:

$$F_{y'}(x_2) = 0. \quad (5.2)$$

Это граничное условие называется естественным. Его смысл следующий: если из всех экстремалей, удовлетворяющих заданному граничному условию слева и различным граничным условиям справа, выбрать функцию, доставляющую экстремум функционалу, то эта функция будет удовлетворять естественному граничному условию на правом конце (5.2).

Аналогично решается задача для функционала вида (2.1), зависящего от нескольких функций, если граничные условия заданы только на одном из концов (например, на левом). В этом случае недостающие граничные условия на правом конце имеют вид

$$\begin{cases} F_{y_i}(x_2) = 0; \\ i = [1, \dots, n]. \end{cases} \quad (5.3)$$

Задача для самостоятельного решения. Выведите естественное граничное условие для функционала (3.1), зависящего от функции и её производных до 2-го порядка включительно, если граничные условия (3.2) заданы только на левом конце интервала x_1 , а на правом конце x_2 заданы или только значение функции (вариант 1), или только значение производной (вариант 2), или ничего не задано (вариант 3).

5.2. Примеры выполнения заданий

5.2.а. Пример 5а

Найти экстремаль функционала (1.6), рассмотренного ранее в примере 1а, при том же самом граничном условии на левом конце, и при незаданном граничном условии на правом конце. Сравнить решение с решением задачи 1а.

Составим программу для решения этой задачи. Так как нам нужно сравнить решение с 1а, поэтому воспользуемся программой для примера 1а, как заготовкой. Оставим полностью решение примера 1а, вплоть до заполнения таблицы, но уберём печать результатов и рисование графика. Оставим только печать левой части естественного граничного условия (5.1) $F_{y'}$.

```
clear all
format long
disp('Решаем пример 5а')
syms x y Dy D2y % описали символические переменные
F=x^2+y^2+Dy^2; % вводим подынтегральную функцию
x1=-1; % вводим граничные условия
y1=1;
x2=1;
y2=2;
fprintf('Подынтегральная функция: F=%s\n',char(F))
```

```
fprintf('Граничные условия: y(%d)=%d;   y(%d)=%d\n',x1,y1,x2,y2)
dFdy = diff(F,y);
dFdy1 = diff(F,Dy); % левая часть естественного граничного условия
fprintf('Естественное граничное условие:   %s=0\n',char(dFdy1))
d_dFdy1_dx=diff(dFdy1,x);
d_dFdy1_dy=diff(dFdy1,y);
d_dFdy1_dy1 = diff(dFdy1,Dy);
dFyldx = d_dFdy1_dx + d_dFdy1_dy*Dy + d_dFdy1_dy1*D2y;
Euler = simple(dFdy-dFyldx);
deqEuler = [ char(Euler) '=0' ]; % составили уравнение
Sol = dsolve(deqEuler,'x'); % решаем уравнение Эйлера
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
SolLeft = subs(Sol,x,sym(x1)); % подставляем x1
SolRight = subs(Sol,x,sym(x2)); % подставляем x2
EqLeft = [char(SolLeft) '=' char(sym(y1))];
EqRight = [char(SolRight) '=' char(sym(y2))];
Con=solve(EqLeft,EqRight);
C1=Con.C1;
C2=Con.C2;
Sol1a = vpa(eval(Sol),14); % подставляем C1, C2
xpl = linspace(x1,x2);
yla = subs(Sol1a,x,xpl);
Решаем пример 5а
Подынтегральная функция: F=x^2+y^2+Dy^2
Граничные условия: y(-1)=1;   y(1)=2
Естественное граничное условие:   2*Dy=0
```

Формируем естественное граничное условие на правом конце. Для этого в формулу (5.1) подставляем сначала y и y' , а затем x_2 во все выражения вместо x .

```
dydx = diff(Sol,x); % производная dy/dx
RightNat = subs(dFdy1,{y,Dy},{Sol,dydx}); % подставили y, y'
SolRightNat = subs(RightNat,x,sym(x2)); % подставили x2
EqRightNat = [char(vpa(SolRightNat,14)) '=0'];
fprintf('Производная от общего решения уравнения Эйлера:\n%s\n',char(dydx))
fprintf('Естественное граничное условие:\n%s\n',EqRightNat)
Производная от общего решения уравнения Эйлера:
2*C1*exp(x)-(C1*exp(x)^2+C2)/exp(x)
Естественное граничное условие:
5.4365636569180*C1-.73575888234290*C2=0
```

Решаем систему уравнений **EqLeft** и **EqRightNat** – находим произвольные постоянные. Подставляем их в решение.

```
Con=solve(EqLeft,EqRightNat);
C1=Con.C1
C2=Con.C2
Sol5a = vpa(eval(Sol),14); % подставляем C1, C2
fprintf('Уравнение экстремали:\ny=%s\n',char(Sol5a))
C1 =
.48891587702804336134757322619033e-1
C2 =
.36126268430179560669376716483447
Уравнение экстремали:
y=(.48891587702804e-1*exp(x)^2+.36126268430180)/exp(x)
```

Далее, как и в примере 1а, заполняем таблицу и строим графики решений примеров 1а и 5а на одном рисунке: красная сплошная линия – решение примера 5а, а синяя пунктирная линия – решение примера 1а.

```
y5a = subs(Sol5a,x,xpl);
```

```
plot ( xpl, yla, '--b', xpl, y5a, '-r' )
title ( '\bfExample 5a' ) % заголовок
xlabel('x')
ylabel('y(x)') % метки осей
```

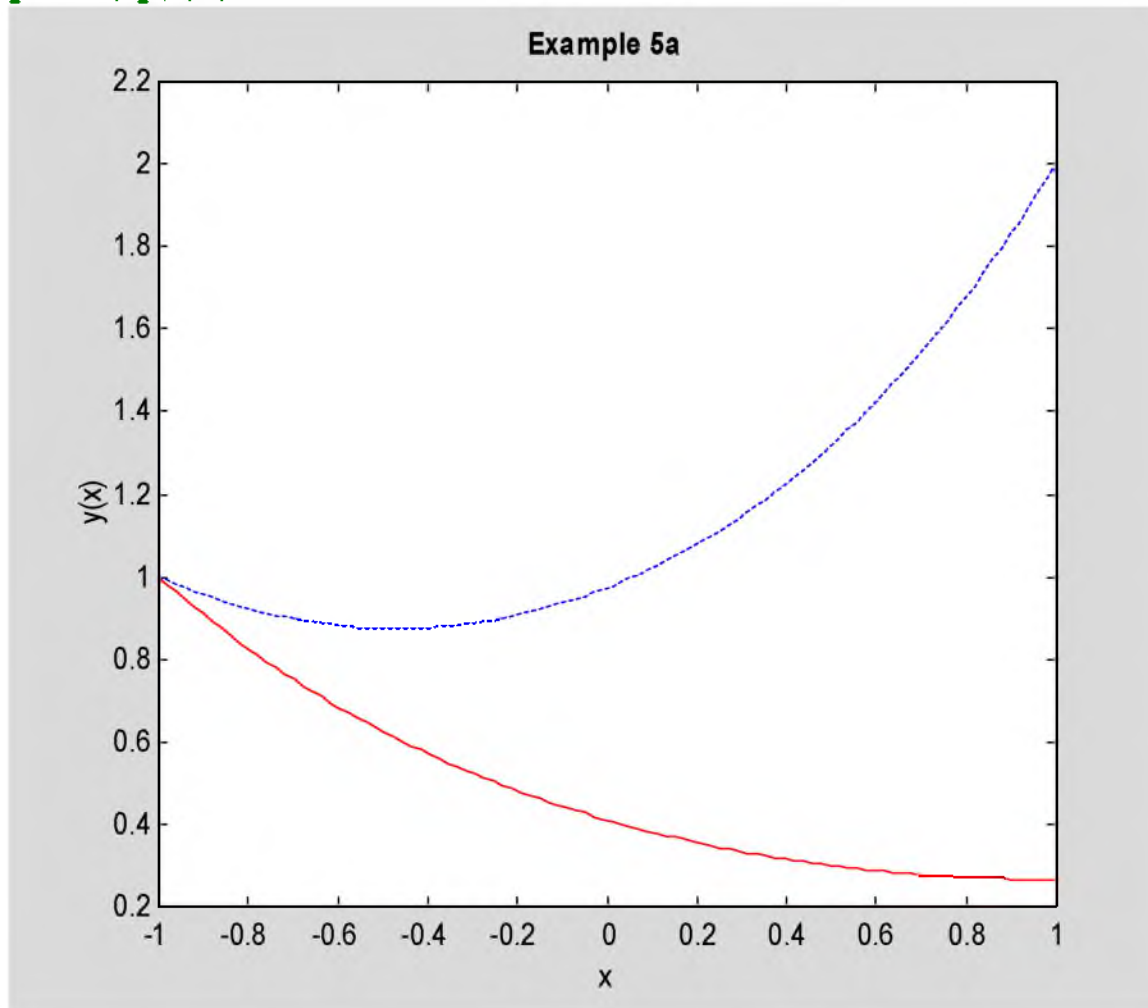


Рис. 5.1. Решение примера 5а

Ответ. Естественное граничное условие на правом конце имеет вид

$$F_{y'}(x_2) = 2y'(x_2) = 0. \quad (5.4)$$

После подстановки произвольных постоянных экстремаль

$$y(x) = 0.04889e^x + 0.36126e^{-x}. \quad (5.5)$$

График её приведен на рис.5.1 сплошной красной линией. Пунктирная синяя линия – решение примера 1а.

5.2.b. Пример 5b

Найти экстремаль функционала (2.4), рассмотренного ранее в примере 2, при тех же самых граничных условиях на левом конце, и при незадаанных граничных условиях на правом конце. Сравнить решение с решением примера 2.

Программу для решения этого примера будем составлять на основе программы для примера 2. Оставим в ней решение примера 2. Уберём только все печати, кроме естественных граничных условий $F_{y'}$ и $F_{z'}$. Графики также пока не строим.

```
clear all
format long
disp('Решаем пример 5b')
syms x y z Dy D2y Dz D2z % описали переменные
F=Dy^2+Dz^2+2*y*z; % подынтегральная функция
x1=-2;
y1=1;
z1=0;
```

```

x2=2;
y2=0;
z2=2;
fprintf('Подынтегральная функция: F=%s\n',char(F))
fprintf('Граничные условия слева: y(%d)=%d;    z(%d)=%d\n',x1,y1,x1,z1)
fprintf('Граничные условия справа: y(%d)=%d;    z(%d)=%d\n',x2,y2,x2,z2)
dFdy = diff(F,y);
dFdy1 = diff(F,Dy) % dF/dy';
d_dFdy1_dx = diff(dFdy1,x); % d(dF/dy')/dx
d_dFdy1_dy = diff(dFdy1,y);
d_dFdy1_dy1 = diff(dFdy1,Dy);
d_dFdy1_dz = diff(dFdy1,z);
d_dFdy1_dz1 = diff(dFdy1,Dz);
dFy1dx = d_dFdy1_dx + d_dFdy1_dy*Dy + d_dFdy1_dy1*D2y + d_dFdy1_dz*Dz +
d_dFdy1_dz1*D2z;
dFdZ = diff(F,z);
dFdZ1 = diff(F,Dz); % dF/dz'
d_dFdZ1_dx = diff(dFdZ1,x); % d(dF/dz')/dx
d_dFdZ1_dy = diff(dFdZ1,y);
d_dFdZ1_dy1 = diff(dFdZ1,Dy);
d_dFdZ1_dz = diff(dFdZ1,z);
d_dFdZ1_dz1 = diff(dFdZ1,Dz);
dFz1dx = d_dFdZ1_dx + d_dFdZ1_dy*Dy + d_dFdZ1_dy1*D2y + d_dFdZ1_dz*Dz +
d_dFdZ1_dz1*D2z;
fprintf('Естественные граничные условия на правом
конце:\n%s=0\n%s=0\n',char(dFdy1),char(dFdZ1))
EulerY = simple(dFdy-dFy1dx);
EulerZ = simple(dFdZ-dFz1dx);
deqEulerY = [char(EulerY) '=0']; % уравнение Y
deqEulerZ = [char(EulerZ) '=0']; % уравнение Z
Sol = dsolve(deqEulerY,deqEulerZ,'x'); % решаем
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
SolLeftY = subs(Sol.y,x,sym(x1)); % x1 в y
SolLeftZ = subs(Sol.z,x,sym(x1)); % x1 в z
SolRightY = subs(Sol.y,x,sym(x2)); % x2 в y
SolRightZ = subs(Sol.z,x,sym(x2)); % x2 в z
EqLeftY = [char(vpa(SolLeftY,14)) '=' char(sym(y1))];
EqLeftZ = [char(vpa(SolLeftZ,14)) '=' char(sym(z1))];
EqRightY = [char(vpa(SolRightY,14)) '=' char(sym(y2))];
EqRightZ = [char(vpa(SolRightZ,14)) '=' char(sym(z2))];
Con = solve(EqLeftY,EqLeftZ,EqRightY,EqRightZ);
C1 = Con.C1;
C2 = Con.C2;
C3 = Con.C3;
C4 = Con.C4;
Sol2Y = vpa(eval(Sol.y),14);
Sol2Z = vpa(eval(Sol.z),14);
xpl = linspace(x1,x2); % массив абсцисс
y2a = subs(Sol2Y,x,xpl);
z2a = subs(Sol2Z,x,xpl);
Решаем пример 5b
Подынтегральная функция: F=Dy^2+Dz^2+2*y*z
Граничные условия слева: y(-2)=1;    z(-2)=0
Граничные условия справа: y(2)=0;    z(2)=2
dFdy1 =
2*Dy
Естественные граничные условия на правом конце:
2*Dy=0
2*Dz=0

```

Формируем естественные граничные условия (5.3): подставляем в F_y и F_z вычисленные y, z, y', z' , а затем $x=x_2$, и приравниваем полученные выражения нулю.

```
dydx = diff(Sol.y,x);
dzdx = diff(Sol.z,x);
rnY = subs(dFdy1,{y,z,Dy,Dz},{Sol.y,Sol.z,dydx,dzdx});
rnZ = subs(dFdZ1,{y,z,Dy,Dz},{Sol.y,Sol.z,dydx,dzdx});
SolRightNatY = subs(rnY,x,sym(x2)); % x2
SolRightNatZ = subs(rnZ,x,sym(x2));
EqNatY = [char(vpa(SolRightNatY,14)) ' = 0 '];
EqNatZ = [char(vpa(SolRightNatZ,14)) ' = 0 '];
fprintf('Производные от общего решения уравнения
Эйлера:\n%s\n%s\n',char(dydx),char(dzdx))
fprintf('Естественные граничные условия:\n%s\n%s\n',EqNatY,EqNatZ)
Производные от общего решения уравнения Эйлера:
-1/4*C1*exp(-x)+1/4*C1*exp(x)-1/2*C1*sin(x)+1/4*C2*exp(-
x)+1/4*C2*exp(x)+1/2*C2*cos(x)-1/4*C3*exp(-x)+1/4*C3*exp(x)+1/2*C3*sin(x)-
1/2*C4*cos(x)+1/4*C4*exp(x)+1/4*C4*exp(-x)
-1/4*C1*exp(-x)+1/4*C1*exp(x)+1/2*C1*sin(x)-
1/2*C2*cos(x)+1/4*C2*exp(x)+1/4*C2*exp(-x)-1/4*C3*exp(-x)+1/4*C3*exp(x)-
1/2*C3*sin(x)+1/4*C4*exp(-x)+1/4*C4*exp(x)+1/2*C4*cos(x)
Естественные граничные условия:
2.7175629810214*C1+3.3460488545366*C2+4.5361578346728*C3+4.1783425276309*C4=0
4.5361578346728*C1+4.1783425276309*C2+2.7175629810214*C3+3.3460488545366*C4=0
```

Решаем полученную систему уравнений, и подставляем значения найденных констант в $y(x)$ и $z(x)$.

```
Con = solve(EqLeftY,EqLeftZ,EqNatY,EqNatZ);
C1 = Con.C1
C2 = Con.C2
C3 = Con.C3
C4 = Con.C4
Sol5bY = vpa(eval(Sol.y),14);
Sol5bZ = vpa(eval(Sol.z),14);
fprintf('Уравнения экстремали:\nyu=%s\nz=%s\n',char(Sol5bY),char(Sol5bZ))
C1 =
.38721244783797732687245309443672
C2 =
-.76196653418377063849264065529893
C3 =
-.24944462837945648240354713354661
C4 =
.62915455657885196167536367747443
Уравнения экстремали:
y=.67644949265860e-1*exp(-1.*x)+.12389604634005e-
2*exp(x)+.31832853810872*cos(x)-.69556054538131*sin(x)
z=.67644949265860e-1*exp(-1.*x)+.12389604634005e-2*exp(x)-
.31832853810872*cos(x)+.69556054538131*sin(x)
```

Заполняем таблицу для построения графика. Строим на трёх отдельных графиках результаты вычислений. Вначале строим двумерные графики функций $y(x)$: сплошной красной линией для примера 5b и пунктирной синей – для примера 2.

```
y5b = subs(Sol5bY,x,xpl);
z5b = subs(Sol5bZ,x,xpl);
plot ( xpl,y2a,':b', xpl,y5b,'-r' )
title ( '\bfExample 5b' ) % заголовок
xlabel('x')
ylabel('y(x)') % метки осей
```

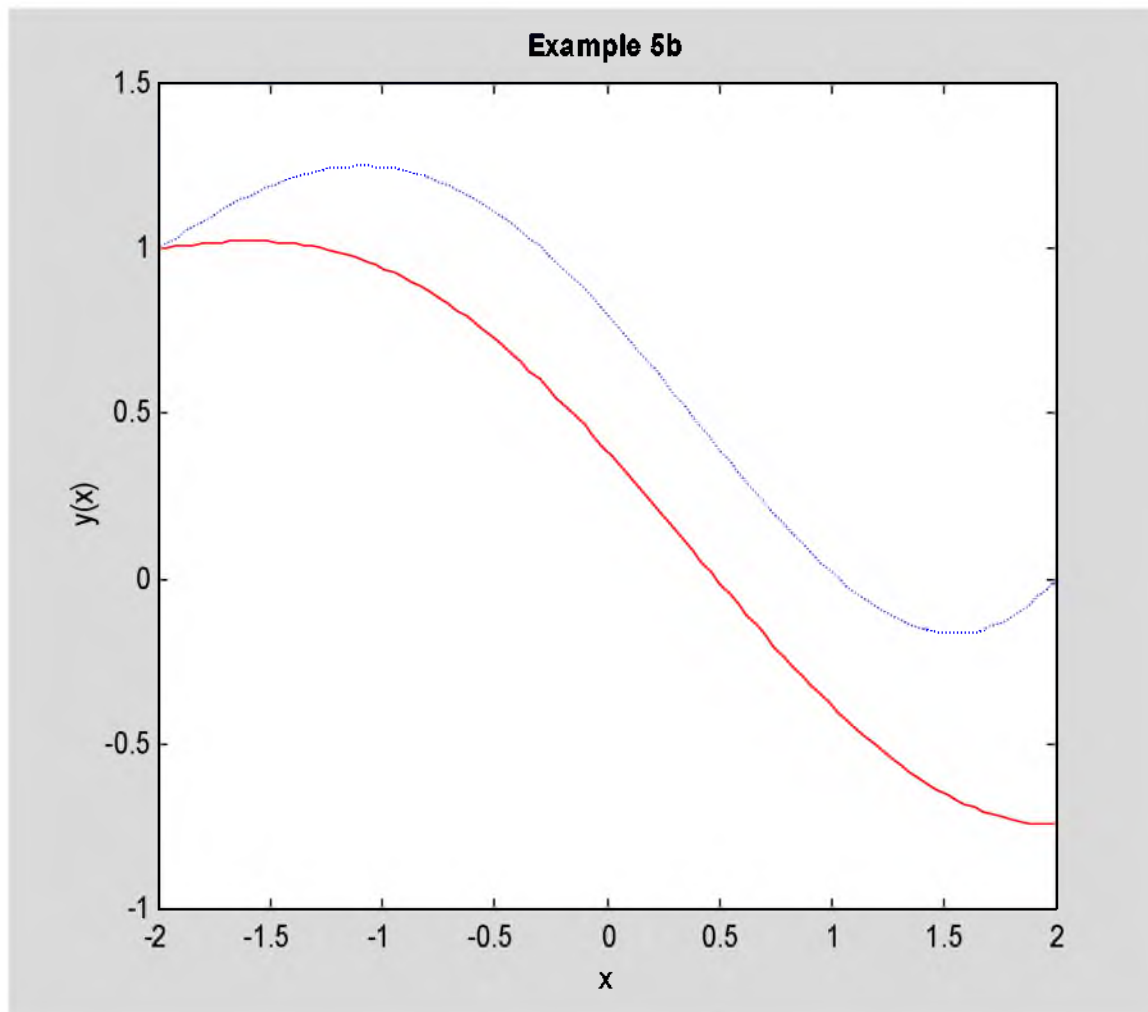



Рис. 5.2. Решение примера 5b – функция $y(x)$

Затем на новом рисунке строим двумерные графики функций $z(x)$, также сплошной красной линией для примера 5b и пунктирной синей – для примера 2.

```
plot ( xpl,z2a,':b', xpl,z5b,'-r' )  
title ( '\bfExample 5b' ) % заголовок  
xlabel('x')  
ylabel('z(x)') % метки осей
```

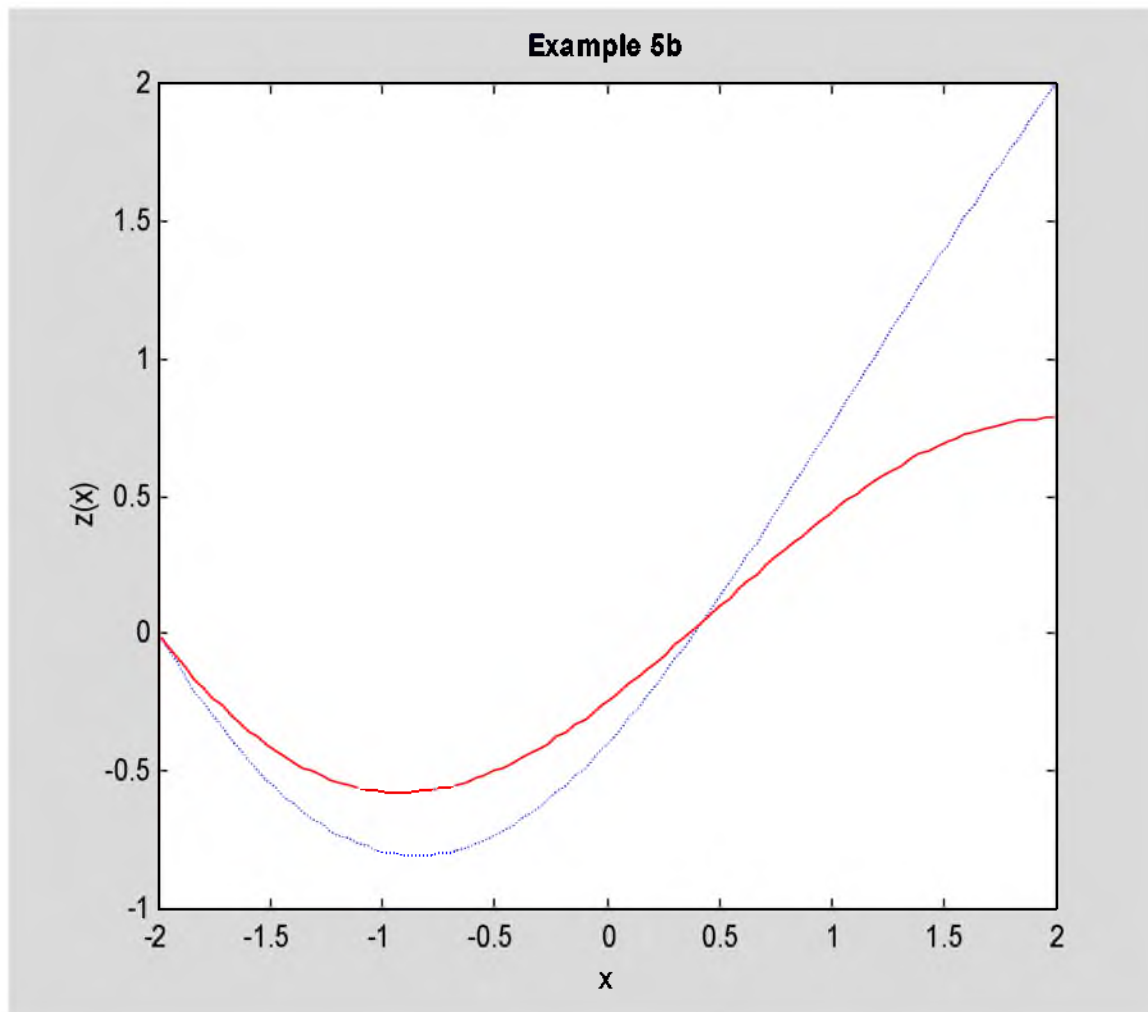



Рис. 5.3. Решение примера 5b – функция $z(x)$

И, наконец, строим трёхмерный график: сплошная красная линия – решение примера 5b, а пунктирная синяя – примера 2. Выбираем точку просмотра. Показываем сетку и контур.

```
plot3(xp1,y2a,z2a,':b',xp1,y5b,z5b,'-r')
title('\bfExample 5b') % заголовок
xlabel('x')
ylabel('y(x)')
zlabel('z(x)') % метки осей
view(205,30) % точка просмотра
grid on % показали сетку
box on % показали контур
```

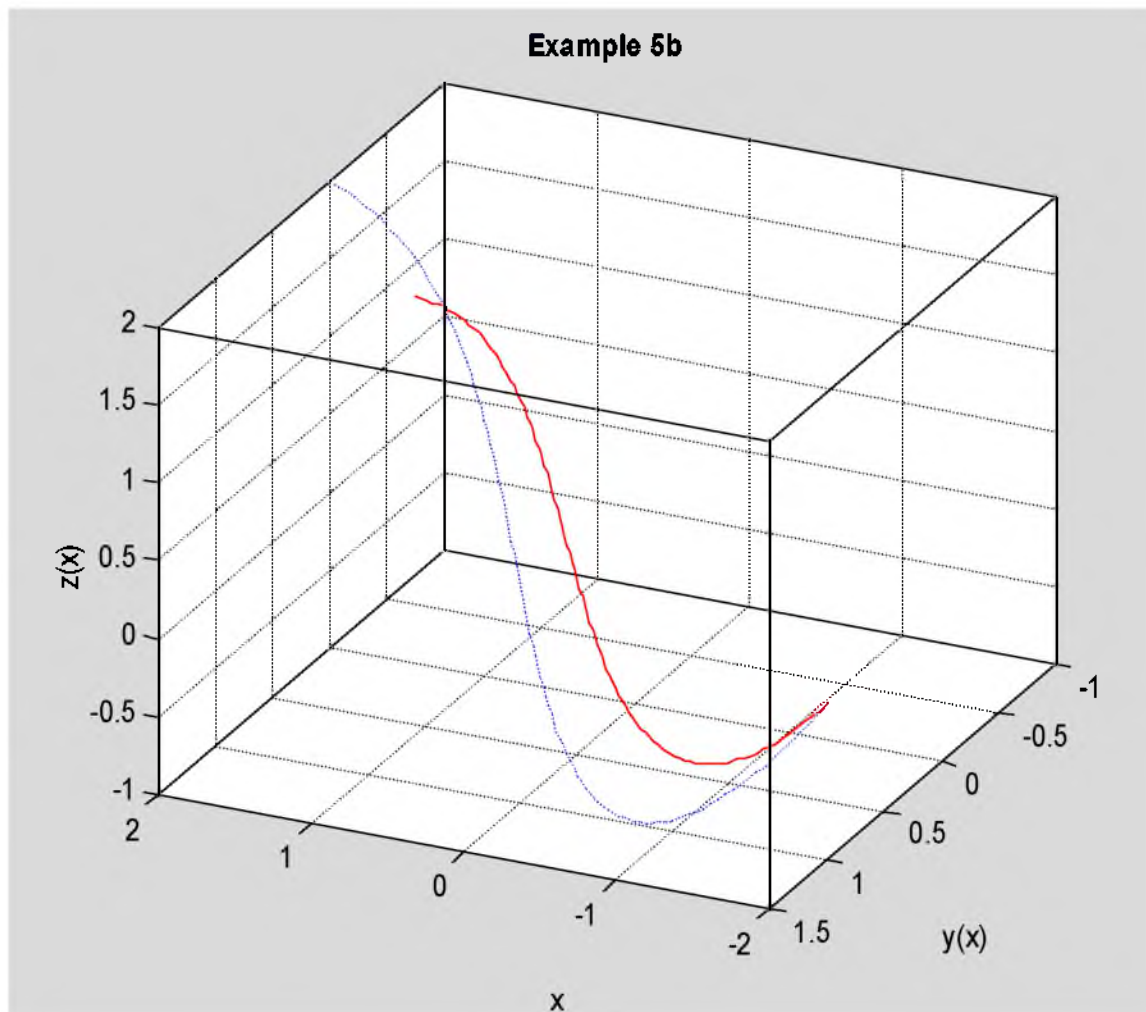


Рис. 5.4. Решение примера 5b в виде пространственной кривой

Ответ. Естественные граничные условия имеют вид

$$\begin{cases} F_{y'}(x_2) = 2y'(x_2) = 0, \\ F_{z'}(x_2) = 2z'(x_2) = 0. \end{cases} \quad (5.6)$$

После подстановки произвольных постоянных уравнения экстремали

$$\begin{cases} y = 0.067645e^{-x} + 0.00123896e^x + 0.3183385 \cos x - 0.6955605 \sin x; \\ z = 0.067645e^{-x} + 0.00123896e^x - 0.3183385 \cos x + 0.6955605 \sin x. \end{cases} \quad (5.7)$$

На рис.5.2–5.4 показаны соответственно функции $y(x)$, $z(x)$, и графики пространственных кривых.

5.3. Задание

Для своего варианта заданий 1а и 2 найти экстремали, если граничные условия на правом конце не заданы. Сравнить полученные решения с решениями примеров 1а и 2.

6. Условие трансверсальности

6.1. Краткие теоретические сведения

Рассмотрим задачу вариационного исчисления для функционала (1.1). Пусть на левом конце задано граничное условие: $y(x_1)=y_1$; а на правом конце x_2 график функции проходит через точку $M_2(x_2, y_2)$, лежащую на заданной линии $y=\varphi(x)$ (рис.6.1):

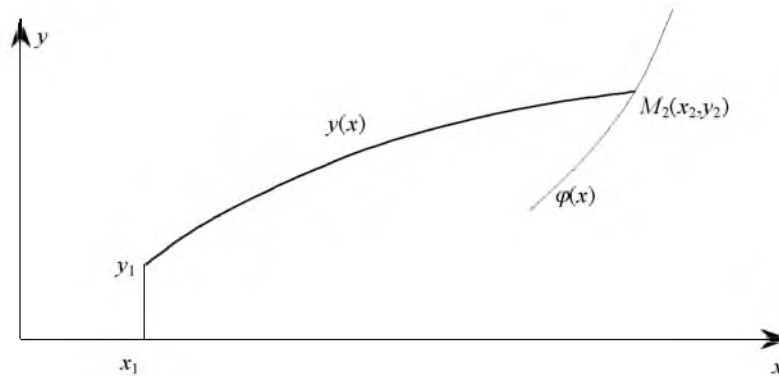


Рис. 6.1. Вариационная задача с подвижным правым концом

Выведем необходимое условие экстремума для такого функционала. Так как $J(y)$ достигает экстремума на классе функций с варьируемой правой точкой $M_2(x_2, y_2)$, то он тем более будет достигать экстремума на более узком классе функций – с неподвижной правой точкой. Следовательно, экстремаль должна удовлетворять уравнению Эйлера (1.5). Решение этого уравнения содержит 2 произвольные постоянные C_1 и C_2 , которые нужно определить. Неизвестны также 2 координаты точки сопряжения $M_2(x_2, y_2)$. Для нахождения этих 4-х неизвестных у нас есть условия:

1. граничное условие на левом конце: $y(x_1)=y_1$;
2. экстремаль проходит через неизвестную точку $M_2(x_2, y_2)$: $y(x_2)=y_2$;
3. точка $M_2(x_2, y_2)$ лежит на линии – ограничении: $y_2=\varphi(x_2)$.

Недостающее четвёртое уравнение выведем из необходимого условия экстремума $\delta J=0$. Вариация функционала вызывается вариацией функции $y(x)$ (вместе с её производной $y'(x)$) и вариацией правой точки x_2 :

$$\begin{aligned} \delta J(y_0) &= \int_{x_1}^{x_2+\delta x_2} (F(x, y_0+\delta y_0, y'_0+\delta y'_0) - F(x, y_0, y'_0)) dx = \\ &= \int_{x_1}^{x_2} (F(x, y_0+\delta y_0, y'_0+\delta y'_0) - F(x, y_0, y'_0)) dx + \int_{x_2}^{x_2+\delta x_2} F(x, y_0+\delta y_0, y'_0+\delta y'_0) dx. \end{aligned} \quad (6.1)$$

Первое слагаемое мы нашли в главе 5 – смотри (5.1):

$$\delta J_1 = (F_{y'} \delta y) \Big|_{x=x_2}. \quad (6.2)$$

Второй интеграл вычисляется на малом участке, поэтому воспользуемся теоремой о среднем. В силу непрерывности подынтегральной функции с точностью до бесконечно малых высшего порядка этот интеграл равен

$$\delta J_2 = F \Big|_{x=x_2} \delta x_2. \quad (6.3)$$

Таким образом, необходимое условие экстремума имеет вид

$$(F_{y'} \delta y) \Big|_{x=x_2} + F \Big|_{x=x_2} \delta x_2 = 0. \quad (6.4)$$

Из рис.6.2 видно, что величина $\delta y \Big|_{x=x_2}$ связана с δy_2 соотношением

$$\delta y \Big|_{x=x_2} = \delta y_2 - y'(x_2) \delta x_2. \quad (6.5)$$

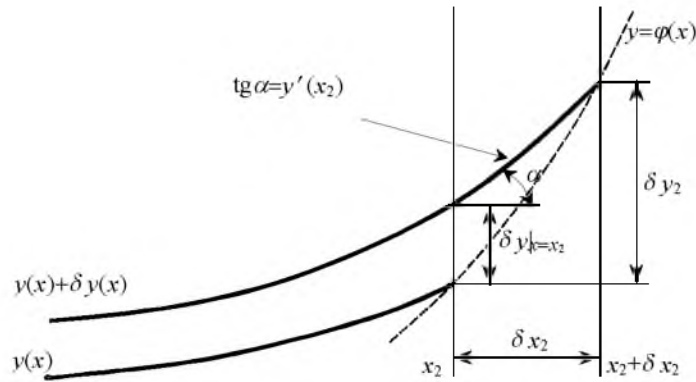


Рис. 6.2. Связь между $\delta y|_{x=x_2}$ и δy_2

Учитывая, что $\delta y_2 = \varphi'(x_2)\delta x_2$, запишем необходимое условие экстремума:

$$\delta J = F_{y'}|_{x=x_2}(\delta y_2 - y'(x_2)\delta x_2) + F|_{x=x_2}\delta x_2 = (F + (\varphi' - y')F_{y'})|_{x=x_2}\delta x_2 = 0. \quad (6.6)$$

В силу произвольности δx_2 получаем недостающее 4-е уравнение:

$$(F + (\varphi' - y')F_{y'})|_{x=x_2} = 0. \quad (6.7)$$

Оно называется условием трансверсальности. Смысл его такой: если двигать точку $M_2(x_2, y_2)$ по линии $y = \varphi(x)$, и по полученным двум точкам строить экстремали, то из всех экстремалей доставлять экстремум функционалу будет та, которая удовлетворяет условию трансверсальности (6.7).

Таким образом, для решения данной вариационной задачи нужно решить дифференциальное уравнение Эйлера, а затем найти произвольные постоянные C_1, C_2 и координаты точки $M_2(x_2, y_2)$ из решения системы 4-х нелинейных уравнений

$$\begin{cases} y(x_1, C_1, C_2) = y_1; \\ y(x_2, C_1, C_2) = y_2; \\ y_2 = \varphi(x_2); \\ (F + (\varphi' - y')F_{y'})|_{x=x_2} = 0. \end{cases} \quad (6.8)$$

Рассмотрим теперь функционал (2.1), зависящий от двух функций $y(x), z(x)$.

$$J(y, z) = \int_{x_1}^{x_2} F(x, y, z, y', z') dx \rightarrow \text{extr}. \quad (6.9)$$

Пусть для него задано только граничное условие при $x = x_1$, а при $x = x_2$ экстремаль (пространственная линия) проходит через точку $M_2(x_2, y_2, z_2)$, лежащую на заданной поверхности или линии.

И в этом случае, если какая-либо кривая доставляет экстремум функционалу на классе функций с подвижной правой точкой, то она тем более будет доставлять ему экстремум на более узком классе функций: с неподвижной правой точкой. Следовательно, экстремаль должна удовлетворять системе уравнений Эйлера (2.3). Однако граничных условий для нахождения всех 4-х произвольных постоянных недостаточно. Недостающие уравнения получим из необходимого условия экстремума: $\delta J = 0$.

$$\begin{aligned} \delta J &= \int_{x_1}^{x_2 + \delta x_2} (F(x, y_0 + \delta y_0, z_0 + \delta z_0, y'_0 + \delta y'_0, z'_0 + \delta z'_0) - F(x, y_0, z_0, y'_0, z'_0)) dx = \\ &= \int_{x_1}^{x_2} (F(x, y_0 + \delta y_0, z_0 + \delta z_0, y'_0 + \delta y'_0, z'_0 + \delta z'_0) - F(x, y_0, z_0, y'_0, z'_0)) dx + \\ &+ \int_{x_2}^{x_2 + \delta x_2} F(x, y_0 + \delta y_0, z_0 + \delta z_0, y'_0 + \delta y'_0, z'_0 + \delta z'_0) dx. \end{aligned} \quad (6.10)$$

В первом слагаемом разложим функцию F в ряд Тейлора в окрестности экстремали, удержим только линейные члены, и проинтегрируем по частям. Так как искомые функции удовлетворяют системе уравнений Эйлера и граничному условию на левом конце, то первое слагаемое будет:

$$\begin{aligned}\delta J_1 &= \int_{x_1}^{x_2} (F_y \delta y + F_z \delta z + F_{y'} \delta y' + F_{z'} \delta z') dx = \\ &= (F_{y'} \delta y + F_{z'} \delta z) \Big|_{x=x_1}^{x=x_2} - \int_{x_1}^{x_2} \left(F_y - \frac{dF_{y'}}{dx} \right) \delta y + \left(F_z - \frac{dF_{z'}}{dx} \right) \delta z dx = (F_{y'} \delta y + F_{z'} \delta z) \Big|_{x=x_2}.\end{aligned}\quad (6.11)$$

Для второго слагаемого используем теорему о среднем:

$$\delta J_2 = F \Big|_{x=x_2} \delta x_2. \quad (6.12)$$

Необходимое условие экстремума функционала (6.9) имеет вид

$$(F_{y'} \delta y + F_{z'} \delta z) \Big|_{x=x_2} + F \Big|_{x=x_2} \delta x_2 = 0. \quad (6.13)$$

Учитывая, что соотношение (6.5) имеет место и для функции $z(x)$:

$$\delta z \Big|_{x=x_2} = \delta z_2 - z'(x_2) \delta x_2, \quad (6.14)$$

запишем результат

$$\delta J = (F - y' F_{y'} - z' F_{z'}) \Big|_{x=x_2} \delta x_2 + F_{y'} \Big|_{x=x_2} \delta y_2 + F_{z'} \Big|_{x=x_2} \delta z_2. \quad (6.15)$$

Отсюда можно вывести условия трансверсальности для различных случаев граничных условий на правом конце.

Пусть, например, правая точка $M_2(x_2, y_2, z_2)$ лежит на заданной поверхности $z = \varphi(x, y)$. В этом случае в выражении (6.15) δx_2 и δy_2 независимы, а δz_2 находится дифференцированием уравнения поверхности: $\delta z_2 = \varphi_x'(x_2, y_2) \delta x_2 + \varphi_y'(x_2, y_2) \delta y_2$. Подставим эти выражения в (6.15), и приравняем нулю коэффициенты при независимых δx_2 и δy_2 . Получаем условия трансверсальности для этого случая:

$$\begin{cases} (F - y' F_{y'} + (\varphi'_x - z') F_{z'}) \Big|_{x=x_2} = 0; \\ (F_{y'} + F_{z'} \varphi'_y) \Big|_{x=x_2} = 0. \end{cases} \quad (6.16)$$

Для нахождения 4-х произвольных постоянных и 3-х координат точки M_2 мы имеем систему 7-ми нелинейных уравнений:

- 2 граничных условия на левом конце;
- 2 граничных условия на неизвестном правом конце;
- точка M_2 находится на поверхности $z = \varphi(x, y)$;
- 2 условия трансверсальности (6.16).

Пусть теперь правая точка $M_2(x_2, y_2, z_2)$ находится на заданной линии

$$\begin{cases} y = \varphi(x), \\ z = \psi(x). \end{cases} \quad (6.17)$$

В этом случае в (6.15) независимым будет только δx_2 , а δy_2 и δz_2 находятся дифференцированием уравнений (6.17): $\delta y_2 = \varphi'(x_2) \delta x_2$; $\delta z_2 = \psi'(x_2) \delta x_2$. Подставим эти выражения в (6.15). Приравняв нулю коэффициент при независимой δx_2 , получим условие трансверсальности для данного вида граничных условий на правом конце:

$$(F + (\varphi' - y') F_{y'} + (\psi' - z') F_{z'}) \Big|_{x=x_2} = 0. \quad (6.18)$$

Для нахождения 4-х произвольных постоянных и 3-х координат точки M_2 мы в этом случае будем иметь такую систему 7-ми нелинейных уравнений:

- 2 граничных условия на левом конце;
- 2 граничных условия на неизвестном правом конце;
- точка M_2 находится на линии (6.15) (2 уравнения);
- условие трансверсальности (6.18).

Задача для самостоятельного решения. Выведите условие трансверсальности для функционала (3.1), зависящего от функции и её производных 1-го и 2-го порядка, когда граничные условия (3.2) заданы только на левом конце интервала x_1 , а правый конец движется по заданной линии $y = \varphi(x)$, причём угол α между экстремалью и кривой $y = \varphi(x)$ в точке сопряжения x_2 задан (вариант 1) или произвольный (вариант 2).

6.2. Примеры выполнения заданий

6.2.а. Пример 6а

Найти экстремаль функционала, рассмотренного ранее в примере 1а. Граничное условие на левом конце то же, что и в примере 1а, а правый конец движется по заданной линии:

$$J(y) = \int_{-1}^{x_2} (x^2 + y^2 + y'^2) dx; \quad \begin{cases} y(-1) = 1; \\ y_2 = 5 - 2x_2^2. \end{cases} \quad (6.19)$$

Сравнить решение с 1а.

Программу для решения этого примера будем составлять на основе программы для примера 1а. Оставим полностью решение примера 1а, но без печатей.

```
clear all
format long
disp('Решаем пример 6а')
syms x y Dy D2y % описали символические переменные
F = x^2+y^2+Dy^2; % подынтегральная функция
x1=-1;
y1=1;
x2=1;
y2=2;
phi = 5-2*x^2; % функция на правом конце
fprintf('Подынтегральная функция: F=%s\n',char(F))
fprintf('Граничные условия: y(%d)=%d; y(%d)=%d\n',x1,y1,x2,y2)
fprintf('Ограничение на правом конце: y=%s\n',char(phi))
dFdy = diff(F,y);
dFdy1 = diff(F,Dy);
d_dFdy1_dx=diff(dFdy1,x);
d_dFdy1_dy=diff(dFdy1,y);
d_dFdy1_dyl = diff(dFdy1,Dy); % d(dF/dy)/dy
dFyldx = d_dFdy1_dx + d_dFdy1_dy*Dy + d_dFdy1_dyl*D2y;
Euler = simple(dFdy-dFyldx);
deqEuler = [ char(Euler) ' = 0 ' ]; % составили уравнение
Sol = dsolve(deqEuler,'x'); % решаем уравнение Эйлера
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
SolLeft=subs(Sol,x,sym(x1));
SolRight=subs(Sol,x,sym(x2));
EqLeft = [char(SolLeft) '=' char(sym(y1))];
EqRight = [char(SolRight) '=' char(sym(y2))];
Con=solve(EqLeft,EqRight);
C1=Con.C1;
C2=Con.C2;
Sol1a = vpa(eval(Sol),14); % подставляем C1, C2
xpl = linspace(x1,x2);
yla = subs(Sol1a,x,xpl);
Решаем пример 6а
Подынтегральная функция: F=x^2+y^2+Dy^2
Граничные условия: y(-1)=1; y(1)=2
Ограничение на правом конце: y=5-2*x^2
```

Приступим к решению примера 6а. Для нахождения произвольных постоянных C_1 , C_2 и правой точки x_2 , y_2 нужно решить систему нелинейных уравнений (6.8). Мы будем её решать методом Ньютона, описанным в приложении. Сформируем функцию для вычисления левых частей уравнений (6.8).

Вначале записываем заголовок функции, переобозначаем переменные. Формируем первые 3 уравнения. После этого выводим условие трансверсальности. Если записывать 4-е уравнение в одну строку, то эта строка получится очень длинной. Поэтому вычисляем предварительно функции $v(x_2)$, $v'(x_2)$, $\varphi'(x_2)$, $F(x_2)$, $F_y(x_2)$, а затем формируем строки с левой частью 4-го уравнения. Записываем всё в файл.

```
f{1} = 'function y = MyFunc(x)'; % заголовок
f{2} = 'C1=x(1); C2=x(2); xr=x(3); yr=x(4); y=x;'; % переобозначили переменные
f{3} = [ 'y(1)=' char(SolLeft) '-' char(sym(y1)) ' ' ];
f{4} = [ 'y(2)=' char(subs(Sol,x,sym('xr')))) '-yr;' ];
f{5} = [ 'y(3)=' char(subs(phi,x,sym('xr')))) '-yr;' ];
yxr = subs(Sol,x,sym('xr')); % x<=xr
dydx=diff(Sol,x);
dydxxr=subs(dydx,x,sym('xr'));
dphidx=diff(phi,x);
dphidxxr=subs(dphidx,x,sym('xr'));
Fxr = subs ( F, {x,y,Dy}, {sym('xr'),sym('yxr'),sym('dydxxr')} );
dFdy1xr = subs ( dFdy1, {x,y,Dy}, {sym('xr'),sym('yxr'),sym('dydxxr')} );
```

```

f{6} = [ 'yxr=', char(yxr), ';' ];
f{7} = [ 'dydxxr=', char(dydxr), ';' ];
f{8} = [ 'Fxr=' char(Fxr) ';' ];
f{9} = [ 'dphidxxr=', char(dphidxxr), ';' ];
f{10} = [ 'dFdylxr=', char(dFdylxr), ';' ];
f{11} = 'y(4)=Fxr+(dphidxxr-dydxxr)*dFdylxr;';
disp('Текст файла MyFunc.m')
fprintf('%s\n',f{:});
fid = fopen ( 'C:\Iglin\Matlab\MyFunc.m', 'w' );
fprintf(fid,'%s\n',f{:});
fclose(fid); % закрываем файл
Т е к с т   ф а й л а   М y F u n c . m
function y = MyFunc(x)
C1=x(1); C2=x(2); xr=x(3); yr=x(4); y=x;
y(1)=(C1*exp(-1)^2+C2)/exp(-1)-1;
y(2)=(C1*exp(xr)^2+C2)/exp(xr)-yr;
y(3)=5-2*xr^2-yr;
yxr=(C1*exp(xr)^2+C2)/exp(xr);
dydxxr=2*C1*exp(xr)-(C1*exp(xr)^2+C2)/exp(xr);
Fxr=xr^2+yxr^2+dydxxr^2;
dphidxxr=-4*xr;
dFdylxr=2*dydxxr;
y(4)=Fxr+(dphidxxr-dydxxr)*dFdylxr;

```

В качестве начального приближения задаём те значения констант, которые получились из решения примера 1а, точку $M_2(x_2, y_2)$ также берём из условия примера 1а. Решаем систему нелинейных уравнений, печатаем количество проведённых вычислений функций. Формируем аналитическое решение.

```

xinit = [ eval(C1); eval(C2); x2; y2 ]
[xzero,options] = fsolve('MyFunc',xinit);
fprintf('Количество вычислений функций: %d\n',options(10))
C1=vpa(sym(xzero(1)),14)
C2=vpa(sym(xzero(2)),14)
x2n=xzero(3);
y2n=xzero(4);
Sol6a = vpa(eval(Sol),14);
fprintf('Уравнение экстремали:\ny=%s\n',char(Sol6a))
fprintf('Правая точка x2=%12.6f\n',x2n)
xinit =
    0.69877023730774
    0.27331117318808
    1.00000000000000
    2.00000000000000
Количество вычислений функций: 48
C1 =
    .60110609424848e-1
C2 =
    .35974435482505
Уравнение экстремали:
y=(.60110609424848e-1*exp(x)^2+.35974435482505)/exp(x)
Правая точка x2=    1.524086

```

Заполним таблицу для построения графика функций $y(x)$ и $\varphi(x)$. Мы будем строить функцию $\varphi(x)$ на небольшом участке вблизи точки x_2 , полученной из решения системы нелинейных уравнений. Построим на одном рисунке три графика: решение примера 1а (синяя штриховая линия), функцию $\varphi(x)$ (зелёная пунктирная линия) и решение примера 6а (сплошная красная линия).

```

xphi = linspace(x2n-0.05,x2n+0.05);
yphi = subs(phi,x,xphi);
x6a = linspace(x1,x2n);
y6a = subs(Sol6a,x,x6a);
plot(xpl,y1a,'--b',xphi,yphi,':g',x6a,y6a,'-r')
title ( '\bfExample 6a' ) % заголовок

```



```
xlabel('x')
ylabel('y(x), phi(x)') % метки осей
```

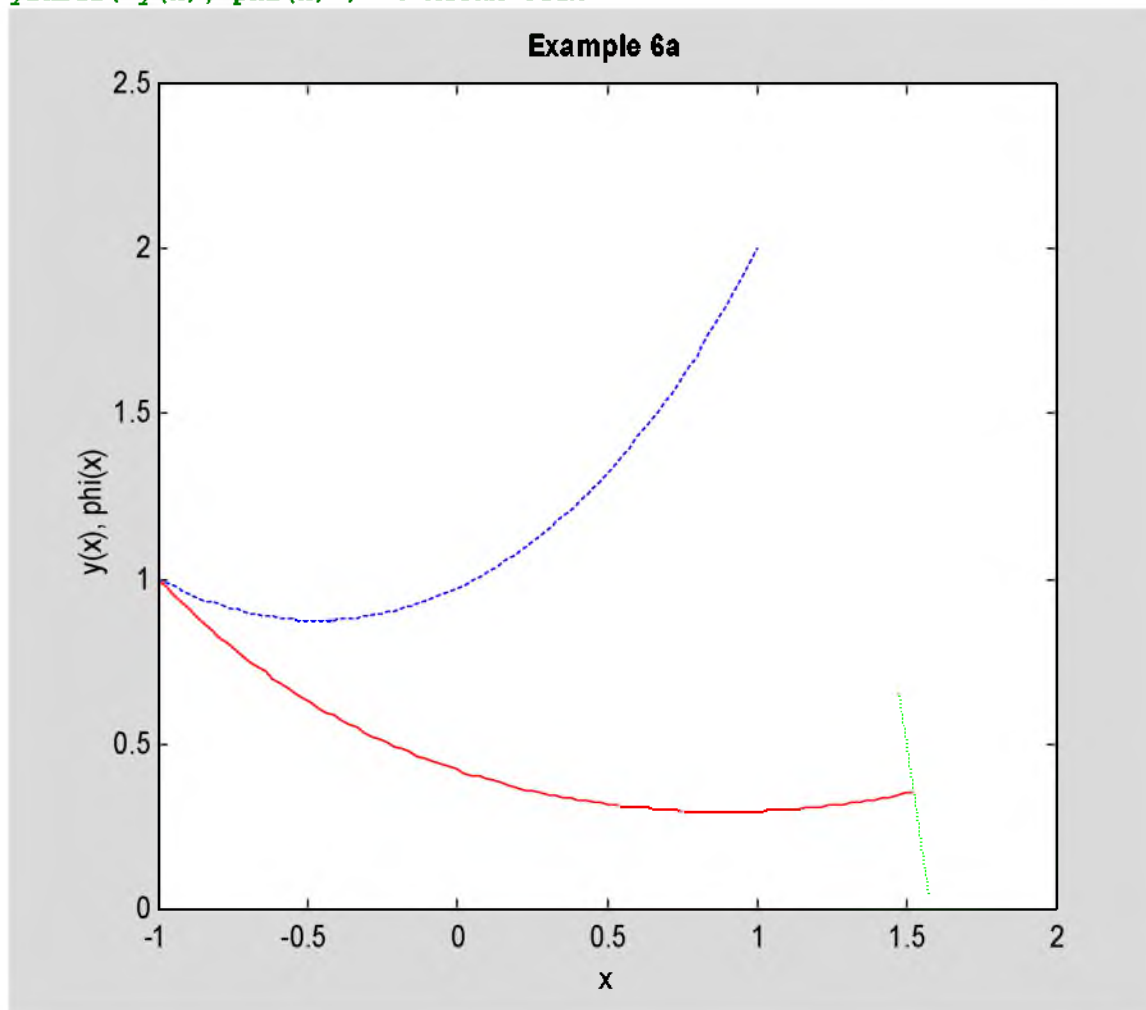


Рис. 6.3. Решение примера 6а

Ответ. Экстремаль функционала (6.19) имеет вид

$$y(x) = 0.0601106e^x + 0.359744e^{-x}; \quad x \in [-1; 1.524086]. \quad (6.20)$$

График решения изображён на рис.6.3 сплошной линией красного цвета. Решение примера 1а показано штриховой линией синего цвета. Функция $\varphi(x)$ нарисована зелёной пунктирной линией.

6.2.b. Пример 6b

Найти экстремаль функционала, рассмотренного ранее в примере 2, при условии, что правый конец находится на заданной поверхности.

$$J(y, z) = \int_{-2}^{x_2} (y'^2 + z'^2 + 2yz) dx; \quad \begin{cases} y(-2)=1; \\ z(-2)=0; \end{cases} \quad z_2 = 4 - x_2^2 - \frac{y_2^2}{2}. \quad (6.21)$$

Сравнить решение с решением примера 2.

Программу для решения примера 6b будем составлять на основе программ для примеров 2 и 6а. Оставим решение примера 2, которое нам нужно для сравнения. Мы его находим, заполняем таблицу для построения графика, но график пока не строим.

```
clear all
format long
disp('Решаем пример 6b')
syms x y z Dy D2y Dz D2z % описали переменные
F = Dy^2 + Dz^2 + 2*y*z; % подынтегральная функция
x1 = -2;
y1 = 1;
z1 = 0;
x2 = 2;
y2 = 0;
```



```

z2 = 2;
phi = 4-x^2-y^2/2;
fprintf('Подынтегральная функция: F=%s\n',char(F))
fprintf('Граничные условия слева: y(%d)=%d;    z(%d)=%d\n',x1,y1,x1,z1)
fprintf('Граничные условия справа: y(%d)=%d;    z(%d)=%d\n',x2,y2,x2,z2)
fprintf('Ограничивающая поверхность справа: z(x,y)=%s\n',char(phi))
dFdy = diff(F,y);
dFdy1 = diff(F,Dy);
d_dFdy1_dx = diff(dFdy1,x); % d(dF/dy')/dx
d_dFdy1_dy = diff(dFdy1,y);
d_dFdy1_dy1 = diff(dFdy1,Dy);
d_dFdy1_dz = diff(dFdy1,z);
d_dFdy1_dz1 = diff(dFdy1,Dz);
dFy1dx = d_dFdy1_dx + d_dFdy1_dy*Dy + d_dFdy1_dy1*D2y + d_dFdy1_dz*Dz +
d_dFdy1_dz1*D2z;
dFdz = diff(F,z);
dFdz1 = diff(F,Dz);
d_dFdz1_dx = diff(dFdz1,x); % d(dF/dz')/dx
d_dFdz1_dy = diff(dFdz1,y);
d_dFdz1_dy1 = diff(dFdz1,Dy);
d_dFdz1_dz = diff(dFdz1,z);
d_dFdz1_dz1 = diff(dFdz1,Dz);
dFz1dx = d_dFdz1_dx + d_dFdz1_dy*Dy + d_dFdz1_dy1*D2y + d_dFdz1_dz*Dz +
d_dFdz1_dz1*D2z;
EulerY = simple(dFdy-dFy1dx);
EulerZ = simple(dFdz-dFz1dx);
deqEulerY = [char(EulerY) ' = 0 ']; % уравнение Y
deqEulerZ = [char(EulerZ) ' = 0 ']; % уравнение Z
Sol = dsolve(deqEulerY,deqEulerZ,'x'); % решаем
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
SolLeftY = subs(Sol.y,x,sym(x1)); % x1 в y
SolLeftZ = subs(Sol.z,x,sym(x1)); % x1 в z
SolRightY = subs(Sol.y,x,sym(x2)); % x2 в y
SolRightZ = subs(Sol.z,x,sym(x2)); % x2 в z
EqLeftY = [char(vpa(SolLeftY,14)) ' = ' char(sym(y1))];
EqLeftZ = [char(vpa(SolLeftZ,14)) ' = ' char(sym(z1))];
EqRightY = [char(vpa(SolRightY,14)) ' = ' char(sym(y2))];
EqRightZ = [char(vpa(SolRightZ,14)) ' = ' char(sym(z2))];
Con = solve(EqLeftY,EqLeftZ,EqRightY,EqRightZ);
C1=Con.C1;
C2=Con.C2;
C3=Con.C3;
C4=Con.C4;
Sol2Y = vpa(eval(Sol.y),14);
Sol2Z = vpa(eval(Sol.z),14);
xpl = linspace(x1,x2); % массив абсцисс
y2a = subs(Sol2Y,x,xpl);
z2a = subs(Sol2Z,x,xpl);
Решаем пример 6b
Подынтегральная функция: F=Dy^2+Dz^2+2*y*z
Граничные условия слева: y(-2)=1;    z(-2)=0
Граничные условия справа: y(2)=0;    z(2)=2
Ограничивающая поверхность справа: z(x,y)=4-x^2-1/2*y^2

```

Нам нужно решить систему 7-ми нелинейных уравнений для нахождения 4-х произвольных постоянных и 3-х координат точки M_2 . Сформируем строки, описывающие уравнения этой системы. Вначале запишем заголовок функции и переобозначим переменные. Граничные условия слева и справа дают первые 4 уравнения. Точка M_2 находится на поверхности $z=\varphi(x,y)$ – это 5-е уравнение. 6-е и 7-е уравнения дают условия трансверсальности (6.16). Сформируем их. Вначале вычислим вспомогательные функции $y(x_2)$, $z(x_2)$, $y'(x_2)$, $z'(x_2)$, $\varphi_x'(x_2)$, $\varphi_y'(x_2)$, $F(x_2)$, $F_y(x_2)$, $F_z(x_2)$. Переведём эти символические величины в строки и сформируем из них 6-е и 7-е уравнения. Запишем все эти строки в файл строки для вычисления функций.

```

f{1} = 'function y = MyFunc(x); % заголовок
f{2} = 'C1=x(1); C2=x(2); C3=x(3); C4=x(4); xr=x(5); yr=x(6); zr=x(7); y=x;';
f{3} = [ 'y(1)= ' char(SolLeftY) '-' char(sym(y1)) ';' ] ;
f{4} = [ 'y(2)= ' char(SolLeftZ) '-' char(sym(z1)) ';' ] ;
f{5} = [ 'y(3)= ' char(subs(Sol.y,x,sym('xr')))) '-'yr;' ] ;
f{6} = [ 'y(4)= ' char(subs(Sol.z,x,sym('xr')))) '-'zr;' ] ;
LeftEq5 = subs ( phi, {x,y}, {sym('xr'),sym('yr')} );
f{7} = [ 'y(5)= ' char(LeftEq5) '-'zr;' ] ; % уравнение 5
yxr = subs(Sol.y,x,sym('xr'));
zxr = subs(Sol.z,x,sym('xr'));
dydx = diff(Sol.y,x);
dydxxr = subs(dydx,x,sym('xr'));
dzdx = diff(Sol.z,x);
dzdxxr = subs(dzdx,x,sym('xr'));
dphidx = diff(phi,x);
dphidy = diff(phi,y);
dphidxxr = subs(dphidx,{x,y},{sym('xr'),sym('yxr')});
dphidyxr = subs(dphidy,{x,y},{sym('xr'),sym('yxr')});
Fxr = subs ( F, {x,y,Dy,z,Dz}, {sym('xr'), sym('yxr'), sym('dydxxr'), sym('zxr'),
sym('dzdxxr')} );
dFdy1xr = subs ( dFdy1, {x,y,Dy,z,Dz}, {sym('xr'), sym('yxr'), sym('dydxxr'),
sym('zxr'), sym('dzdxxr')} );
dFdzl1xr = subs ( dFdzl1, {x,y,Dy,z,Dz}, {sym('xr'), sym('yxr'), sym('dydxxr'),
sym('zxr'), sym('dzdxxr')} );
f{8} = [ 'yxr= ' char(yxr), ';' ] ;
f{9} = [ 'zxr= ' char(zxr), ';' ] ;
f{10} = [ 'dydxxr= ' char(dydxxr), ';' ] ;
f{11} = [ 'dzdxxr= ' char(dzdxxr), ';' ] ;
f{12} = [ 'Fxr= ' char(Fxr), ';' ] ;
f{13} = [ 'dphidxxr= ' char(dphidxxr), ';' ] ;
f{14} = [ 'dphidyxr= ' char(dphidyxr), ';' ] ;
f{15} = [ 'dFdy1xr= ' char(dFdy1xr), ';' ] ;
f{16} = [ 'dFdzl1xr= ' char(dFdzl1xr), ';' ] ;
f{17} = 'y(6)=Fxr-dydxxr*dFdy1xr+(dphidxxr-dzdxxr)*dFdzl1xr;';
f{18} = 'y(7)=dFdy1xr+dphidyxr*dFdzl1xr;';
disp('Текст файла MyFunc.m')
fprintf('%s\n',f{:});
fid = fopen ( 'C:\Iglin\Matlab\MyFunc.m', 'w' );
fprintf(fid,'%s\n',f{:});
fclose(fid); % закрываем файл
Т е к с т   ф а й л а   М y F u n c . m
function y = MyFunc(x)
C1=x(1); C2=x(2); C3=x(3); C4=x(4); xr=x(5); yr=x(6); zr=x(7); y=x;
y(1)=1/4*C1*exp(2)+1/4*C1*exp(-2)+1/2*C1*cos(2)-1/4*C2*exp(2)+1/4*C2*exp(-2)-
1/2*C2*sin(2)+1/4*C3*exp(2)+1/4*C3*exp(-2)-
1/2*C3*cos(2)+1/2*C4*sin(2)+1/4*C4*exp(-2)-1/4*C4*exp(2)-1;
y(2)=1/4*C1*exp(2)+1/4*C1*exp(-2)-1/2*C1*cos(2)+1/2*C2*sin(2)+1/4*C2*exp(-2)-
1/4*C2*exp(2)+1/4*C3*exp(2)+1/4*C3*exp(-2)+1/2*C3*cos(2)-
1/4*C4*exp(2)+1/4*C4*exp(-2)-1/2*C4*sin(2)-0;
y(3)=1/4*C1*exp(-xr)+1/4*C1*exp(xr)+1/2*C1*cos(xr)-1/4*C2*exp(-
xr)+1/4*C2*exp(xr)+1/2*C2*sin(xr)+1/4*C3*exp(-xr)+1/4*C3*exp(xr)-1/2*C3*cos(xr)-
1/2*C4*sin(xr)+1/4*C4*exp(xr)-1/4*C4*exp(-xr)-yr;
y(4)=1/4*C1*exp(-xr)+1/4*C1*exp(xr)-1/2*C1*cos(xr)-
1/2*C2*sin(xr)+1/4*C2*exp(xr)-1/4*C2*exp(-xr)+1/4*C3*exp(-
xr)+1/4*C3*exp(xr)+1/2*C3*cos(xr)-1/4*C4*exp(-xr)+1/4*C4*exp(xr)+1/2*C4*sin(xr)-
zr;
y(5)=4-xr^2-1/2*yr^2-zr;
yxr=1/4*C1*exp(-xr)+1/4*C1*exp(xr)+1/2*C1*cos(xr)-1/4*C2*exp(-
xr)+1/4*C2*exp(xr)+1/2*C2*sin(xr)+1/4*C3*exp(-xr)+1/4*C3*exp(xr)-1/2*C3*cos(xr)-
1/2*C4*sin(xr)+1/4*C4*exp(xr)-1/4*C4*exp(-xr);
zxr=1/4*C1*exp(-xr)+1/4*C1*exp(xr)-1/2*C1*cos(xr)-1/2*C2*sin(xr)+1/4*C2*exp(xr)-
1/4*C2*exp(-xr)+1/4*C3*exp(-xr)+1/4*C3*exp(xr)+1/2*C3*cos(xr)-1/4*C4*exp(-
xr)+1/4*C4*exp(xr)+1/2*C4*sin(xr);

```

```

dydxxr=-1/4*C1*exp(-xr)+1/4*C1*exp(xr)-1/2*C1*sin(xr)+1/4*C2*exp(-
xr)+1/4*C2*exp(xr)+1/2*C2*cos(xr)-1/4*C3*exp(-xr)+1/4*C3*exp(xr)+1/2*C3*sin(xr)-
1/2*C4*cos(xr)+1/4*C4*exp(xr)+1/4*C4*exp(-xr);
dzdxxr=-1/4*C1*exp(-xr)+1/4*C1*exp(xr)+1/2*C1*sin(xr)-
1/2*C2*cos(xr)+1/4*C2*exp(xr)+1/4*C2*exp(-xr)-1/4*C3*exp(-xr)+1/4*C3*exp(xr)-
1/2*C3*sin(xr)+1/4*C4*exp(-xr)+1/4*C4*exp(xr)+1/2*C4*cos(xr);
Fxr=dydxxr^2+dzdxxr^2+2*yxr*zxr;
dphidxxr=-2*xr;
dphidyxr=-yxr;
dFdy1xr=2*dydxxr;
dFdzl1xr=2*dzdxxr;
y(6)=Fxr-dydxxr*dFdy1xr+(dphidxxr-dzdxxr)*dFdzl1xr;
y(7)=dFdy1xr+dphidyxr*dFdzl1xr;

```

Решаем систему нелинейных уравнений методом Ньютона (смотри приложение). В качестве начальных значений констант и координат точки M_2 задаём данные из решения примера 2. Печатаем решение.

```

xinit = [eval(C1);eval(C2);eval(C3);eval(C4);x2;y2;z2]
[xzero,options] = fsolve('MyFunc',xinit);
fprintf('Количество вычислений функций: %d\n',options(10))
C1 = vpa(sym(xzero(1)),14)
C2 = vpa(sym(xzero(2)),14)
C3 = vpa(sym(xzero(3)),14)
C4 = vpa(sym(xzero(4)),14)
x2n = xzero(5)
y2n = xzero(6)
z2n = xzero(7)
Sol6bY = vpa(eval(Sol.y),14);
Sol6bZ = vpa(eval(Sol.z),14);
fprintf('Уравнения экстремали: \nu(x)=%s\nz(x)=%s\n',char(Sol6bY),char(Sol6bZ))
xinit =
    0.80010116205604
   -0.75588248652808
   -0.40139781880493
    0.89374276891397
    2.00000000000000
           0
    2.00000000000000
Количество вычислений функций: 50
C1 =
.12795213276350
C2 =
-.64725967277369
C3 =
.45362068565709e-2
C4 =
.50897101383163
x2n =
    1.80444576439465
y2n =
   -0.57438130409517
z2n =
    0.57902894028817
Уравнения экстремали:
y(x)=.67694249640533e-1*exp(-1.*x)-.14500798304973e-2*exp(x)+.61707962953465e-
1*cos(x)-.57811534330266*sin(x)
z(x)=.67694249640533e-1*exp(-1.*x)-.14500798304973e-2*exp(x)-.61707962953465e-
1*cos(x)+.57811534330266*sin(x)

```

Составляем таблицу для графика поверхности $z=\varphi(x,y)$. Чтобы не затенять чертёж, будем рисовать эту поверхность только вблизи точки M_2 . Затем заполняем таблицу значений найденного решения и рисуем график.

```
[xi,yi] = meshgrid ( linspace(x2n-0.1, x2n+0.1), linspace(y2n-0.1, y2n+0.1));
```

```
z = subs(phi,{x,y},{xi,yi});
x6b = linspace(x1,x2n); % массив абсцисс
y6b = subs(Sol6bY,x,x6b);
z6b = subs(Sol6bZ,x,x6b);
plot3(xpl,y2a,z2a,'--b', xi,yi,z,':g',x6b,y6b,z6b,'-r')
title ( '\bfExample 6b' ) % заголовок
xlabel('x')
ylabel('y(x)')
zlabel('z(x)')
view(205,30)
grid on
box on
```

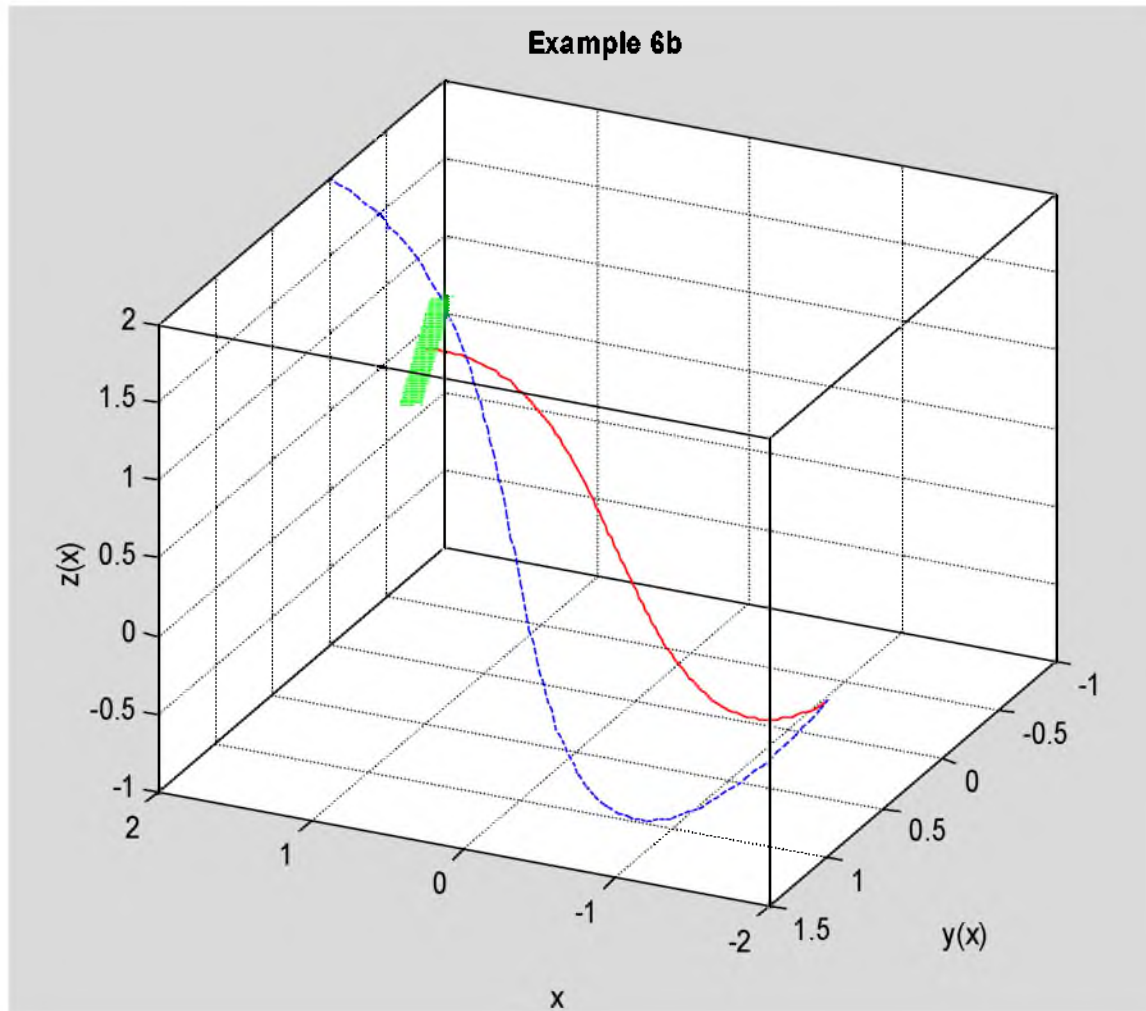


Рис. 6.4. Решение примера 6b

Ответ. Экстремаль функционала (6.21) имеет вид

$$\begin{cases} y(x) = 0.0676942e^{-x} - 0.001450856e^x + \\ \quad + 0.061707 \cos x - 0.578117 \sin x; \\ z(x) = 0.0676942e^{-x} - 0.001450856e^x - \\ \quad - 0.061707 \cos x + 0.578117 \sin x; \end{cases} \quad x \in [-2; 1.8044466]. \quad (6.22)$$

Её график приведен на рис.6.4 сплошной красной линией. Штриховой синей линией показано решение примера 2, зелёным цветом нарисован кусочек ограничивающей поверхности $z=\varphi(x,y)$.

6.2.с. Пример 6с

Найти экстремаль функционала, рассмотренного ранее в примерах 2 и 5b, при условии, что правый конец экстремали находится на заданной линии.

$$J(y,z) = \int_{-2}^y (y'^2 + z'^2 + 2yz) dx; \quad \begin{cases} y(-2)=1; \\ z(-2)=0; \end{cases} \quad \begin{cases} y_2 = 4 - x_2^2; \\ z_2 = 5 - e^{x_2}. \end{cases} \quad (6.23)$$

Сравнить решение с решением примера 2.

Напишем программу для примера 6с на основе программы для примера 6b. Изменятся данные. Решение примера 2 оставляем без изменения.

```
clear all
format long
syms x y z Dy D2y Dz D2z
disp('Решаем пример 6с')
F = Dy^2+Dz^2+2*y*z; % подынтегральная функция
x1 = -2;
y1 = 1;
z1 = 0;
x2 = 2;
y2 = 0;
z2 = 2;
phi = 4-x^2;
psi = 5-exp(x);
fprintf('Подынтегральная функция: F=%s\n',char(F))
fprintf('Граничные условия слева: y(%d)=%d; z(%d)=%d\n',x1,y1,x1,z1)
fprintf('Граничные условия справа: y(%d)=%d; z(%d)=%d\n',x2,y2,x2,z2)
fprintf('Ограничивающая линия справа: \ny=%s\nz=%s\n',char(phi),char(psi))
dFdy = diff(F,y);
dFdy1 = diff(F,Dy);
d_dFdy1_dx = diff(dFdy1,x); % d(dF/dy)/dx
d_dFdy1_dy = diff(dFdy1,y);
d_dFdy1_dy1 = diff(dFdy1,Dy);
d_dFdy1_dz = diff(dFdy1,z);
d_dFdy1_dz1 = diff(dFdy1,Dz);
dFy1dx = d_dFdy1_dx + d_dFdy1_dy*Dy + d_dFdy1_dy1*D2y + d_dFdy1_dz*Dz +
d_dFdy1_dz1*D2z;
dFdZ = diff(F,z);
dFdZ1 = diff(F,Dz);
d_dFdZ1_dx = diff(dFdZ1,x); % d(dF/dz)/dx
d_dFdZ1_dy = diff(dFdZ1,y);
d_dFdZ1_dy1 = diff(dFdZ1,Dy);
d_dFdZ1_dz = diff(dFdZ1,z);
d_dFdZ1_dz1 = diff(dFdZ1,Dz);
dFz1dx = d_dFdZ1_dx + d_dFdZ1_dy*Dy + d_dFdZ1_dy1*D2y + d_dFdZ1_dz*Dz +
d_dFdZ1_dz1*D2z;
EulerY = simple(dFdy-dFy1dx);
EulerZ = simple(dFdZ-dFz1dx);
deqEulerY = [char(EulerY) ' = 0 ']; % уравнение Y
deqEulerZ = [char(EulerZ) ' = 0 ']; % уравнение Z
Sol = dsolve(deqEulerY,deqEulerZ,'x'); % решаем
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
SolLeftY = subs(Sol.y,x,sym(x1)); % x1 в y
SolLeftZ = subs(Sol.z,x,sym(x1)); % x1 в z
SolRightY = subs(Sol.y,x,sym(x2)); % x2 в y
SolRightZ = subs(Sol.z,x,sym(x2)); % x2 в z
EqLeftY = [char(vpa(SolLeftY,14)) ' = ' char(sym(y1))];
EqLeftZ = [char(vpa(SolLeftZ,14)) ' = ' char(sym(z1))];
EqRightY = [char(vpa(SolRightY,14)) ' = ' char(sym(y2))];
EqRightZ = [char(vpa(SolRightZ,14)) ' = ' char(sym(z2))];
Con = solve(EqLeftY,EqLeftZ,EqRightY,EqRightZ);
C1=Con.C1;
C2=Con.C2;
C3=Con.C3;
C4=Con.C4;
Sol2Y = vpa(eval(Sol.y),14);
Sol2Z = vpa(eval(Sol.z),14);
xpl = linspace(x1,x2); % массив абсцисс
y2a = subs(Sol2Y,x,xpl);
z2a = subs(Sol2Z,x,xpl);
```


Решаем пример 6с

Подынтегральная функция: $F=Dy^2+Dz^2+2*y*z$

Граничные условия слева: $y(-2)=1$; $z(-2)=0$

Граничные условия справа: $y(2)=0$; $z(2)=2$

Ограничивающая линия справа:

$$y=4-x^2$$

$$z=5-\exp(x)$$

Заполняем систему нелинейных уравнений. Вначале записываем заголовок и переобозначаем переменные. Первые 4 уравнения – это граничные условия на концах. Следующие 2 уравнения: точка M_2 находится на линии (6.17). 7-е уравнение – это условие transversальности (6.18). Формируем его. Записываем функцию в файл.

```
f{1} = 'function y = MyFunc(x)'; % заголовок
f{2} = 'C1=x(1); C2=x(2); C3=x(3); C4=x(4); xr=x(5); yr=x(6); zr=x(7); y=x;';
f{3} = [ 'y(1)= ' char(SolLeftY) '-' char(sym(y1)) ';' ];
f{4} = [ 'y(2)= ' char(SolLeftZ) '-' char(sym(z1)) ';' ];
f{5} = [ 'y(3)= ' char(subs(Sol.y,x,sym('xr')))' '-yr;' ];
f{6} = [ 'y(4)= ' char(subs(Sol.z,x,sym('xr')))' '-zr;' ];
f{7} = [ 'y(5)= ' char(subs(phi,x,sym('xr')))' '-yr;' ];
f{8} = [ 'y(6)= ' char(subs(psi,x,sym('xr')))' '-zr;' ];
yxr = subs(Sol.y,x,sym('xr'));
zxr = subs(Sol.z,x,sym('xr'));
dydx = diff(Sol.y,x);
dydxxr = subs(dydx,x,sym('xr'));
dzdx = diff(Sol.z,x);
dzdxxr = subs(dzdx,x,sym('xr'));
dphidx = diff(phi,x);
dphidxxr = subs(dphidx,x,sym('xr'));
dpsidx = diff(psi,x);
dpsidxxr = subs(dpsidx,x,sym('xr'));
Fxr = subs ( F, {x,y,Dy,z,Dz}, {sym('xr'), sym('yxr'), sym('dydxxr'), sym('zxr'),
sym('dzdxxr')});
dFdylxr = subs ( dFdy1, {x,y,Dy,z,Dz}, {sym('xr'), sym('yxr'), sym('dydxxr'),
sym('zxr'), sym('dzdxxr')});
dFdzlxr = subs ( dFdZ1, {x,y,Dy,z,Dz}, {sym('xr'), sym('yxr'), sym('dydxxr'),
sym('zxr'), sym('dzdxxr')});
f{9} = [ 'yxr= ' char(yxr) ';' ];
f{10} = [ 'zxr= ' char(zxr) ';' ];
f{11} = [ 'dydxxr= ' char(dydxxr) ';' ];
f{12} = [ 'dzdxxr= ' char(dzdxxr) ';' ];
f{13} = [ 'Fxr= ' char(Fxr) ';' ];
f{14} = [ 'dphidxxr= ' char(dphidxxr) ';' ];
f{15} = [ 'dpsidxxr= ' char(dpsidxxr) ';' ];
f{16} = [ 'dFdylxr= ' char(dFdylxr) ';' ];
f{17} = [ 'dFdzlxr= ' char(dFdzlxr) ';' ];
f{18} = 'y(7) = Fxr + (dphidxxr-dydxxr)*dFdylxr + (dpsidxxr-dzdxxr)*dFdzlxr;';
disp('Текст файла MyFunc.m')
fprintf('%s\n',f{:});
fid = fopen ( 'C:\Iglin\Matlab\MyFunc.m', 'w' );
fprintf(fid,'%s\n',f{:});
fclose(fid); % закрываем файл

Текст файла MyFunc.m
function y = MyFunc(x)
C1=x(1); C2=x(2); C3=x(3); C4=x(4); xr=x(5); yr=x(6); zr=x(7); y=x;
y(1)=1/4*C1*exp(2)+1/4*C1*exp(-2)+1/2*C1*cos(2)-1/4*C2*exp(2)+1/4*C2*exp(-2)-
1/2*C2*sin(2)+1/4*C3*exp(2)+1/4*C3*exp(-2)-
1/2*C3*cos(2)+1/2*C4*sin(2)+1/4*C4*exp(-2)-1/4*C4*exp(2)-1;
y(2)=1/4*C1*exp(2)+1/4*C1*exp(-2)-1/2*C1*cos(2)+1/2*C2*sin(2)+1/4*C2*exp(-2)-
1/4*C2*exp(2)+1/4*C3*exp(2)+1/4*C3*exp(-2)+1/2*C3*cos(2)-
1/4*C4*exp(2)+1/4*C4*exp(-2)-1/2*C4*sin(2)-0;
```

```

y(3)=1/4*C1*exp(-xr)+1/4*C1*exp(xr)+1/2*C1*cos(xr)-1/4*C2*exp(-
xr)+1/4*C2*exp(xr)+1/2*C2*sin(xr)+1/4*C3*exp(-xr)+1/4*C3*exp(xr)-1/2*C3*cos(xr)-
1/2*C4*sin(xr)+1/4*C4*exp(xr)-1/4*C4*exp(-xr)-yr;
y(4)=1/4*C1*exp(-xr)+1/4*C1*exp(xr)-1/2*C1*cos(xr)-
1/2*C2*sin(xr)+1/4*C2*exp(xr)-1/4*C2*exp(-xr)+1/4*C3*exp(-
xr)+1/4*C3*exp(xr)+1/2*C3*cos(xr)-1/4*C4*exp(-xr)+1/4*C4*exp(xr)+1/2*C4*sin(xr)-
zr;
y(5)=4-xr^2-yr;
y(6)=5-exp(xr)-zr;
yxr=1/4*C1*exp(-xr)+1/4*C1*exp(xr)+1/2*C1*cos(xr)-1/4*C2*exp(-
xr)+1/4*C2*exp(xr)+1/2*C2*sin(xr)+1/4*C3*exp(-xr)+1/4*C3*exp(xr)-1/2*C3*cos(xr)-
1/2*C4*sin(xr)+1/4*C4*exp(xr)-1/4*C4*exp(-xr);
zxr=1/4*C1*exp(-xr)+1/4*C1*exp(xr)-1/2*C1*cos(xr)-1/2*C2*sin(xr)+1/4*C2*exp(xr)-
1/4*C2*exp(-xr)+1/4*C3*exp(-xr)+1/4*C3*exp(xr)+1/2*C3*cos(xr)-1/4*C4*exp(-
xr)+1/4*C4*exp(xr)+1/2*C4*sin(xr);
dydxxr=-1/4*C1*exp(-xr)+1/4*C1*exp(xr)-1/2*C1*sin(xr)+1/4*C2*exp(-
xr)+1/4*C2*exp(xr)+1/2*C2*cos(xr)-1/4*C3*exp(-xr)+1/4*C3*exp(xr)+1/2*C3*sin(xr)-
1/2*C4*cos(xr)+1/4*C4*exp(xr)+1/4*C4*exp(-xr);
dzdxxr=-1/4*C1*exp(-xr)+1/4*C1*exp(xr)+1/2*C1*sin(xr)-
1/2*C2*cos(xr)+1/4*C2*exp(xr)+1/4*C2*exp(-xr)-1/4*C3*exp(-xr)+1/4*C3*exp(xr)-
1/2*C3*sin(xr)+1/4*C4*exp(-xr)+1/4*C4*exp(xr)+1/2*C4*cos(xr);
Fxr=dydxxr^2+dzdxxr^2+2*yxr*zxr;
dphidxxr=-2*xr;
dpsidxxr=-exp(xr);
dFdy1xr=2*dydxxr;
dFdz1xr=2*dzdxxr;
y(7) = Fxr + (dphidxxr-dydxxr)*dFdy1xr + (dpsidxxr-dzdxxr)*dFdz1xr;

```

Задаём начальное приближение и решаем полученную систему. Найденные значения констант подставляем в общее решение.

```

xinit = [eval(C1);eval(C2);eval(C3);eval(C4);x2;y2;z2]
[xzero,options] = fsolve('MyFunc',xinit);
fprintf('Количество вычислений функций: %d\n',options(10))
C1 = vpa(sym(xzero(1)),14)
C2 = vpa(sym(xzero(2)),14)
C3 = vpa(sym(xzero(3)),14)
C4 = vpa(sym(xzero(4)),14)
x2n = xzero(5)
y2n = xzero(6)
z2n = xzero(7)
Sol6cY = vpa(eval(Sol.y),14);
Sol6cZ = vpa(eval(Sol.z),14);
fprintf('Уравнения экстремали: \nu(x)=%s\nz(x)=%s\n',char(Sol6cY),char(Sol6cZ))
xinit =
    0.80010116205604
   -0.75588248652808
   -0.40139781880493
    0.89374276891397
    2.00000000000000
         0
    2.00000000000000
Количество вычислений функций: 173
C1 =
-2.1131005775869
C2 =
.37848653989783
C3 =
2.2457391053786
C4 =
-.51661920847816
x2n =
1.76885916978923

```

```

y2n =
    0.87113720473800
z2n =
   -0.86415946062000
Уравнения экстремали:
y(x)=.67692799093008e-1*exp(-1.*x)-.13735351971575e-2*exp(x)-
2.1794198414828*cos(x)+.44755287418800*sin(x)
z(x)=.67692799093008e-1*exp(-1.*x)-.13735351971575e-
2*exp(x)+2.1794198414828*cos(x)-.44755287418800*sin(x)

```

Вычисляем таблицу значений функции (6.17) и решения нашей задачи. печатаем график. Ограничивающую линию рисуем на небольшом участке.

```

xb = linspace(x2n-0.2,x2n+0.2);
yi = subs(phi,x,xb);
zi = subs(psi,x,xb);
x6c = linspace(x1,x2n); % массив абсцисс
y6c = subs(Sol6cY,x,x6c);
z6c = subs(Sol6cZ,x,x6c);
plot3(xpl,y2a,z2a,'--b',xb,yi,zi,':g',x6c,y6c,z6c,'-r')
title('\bfExample 6c')
xlabel('x')
ylabel('y(x)')
zlabel('z(x)')
view(205,30)
grid on
box on

```

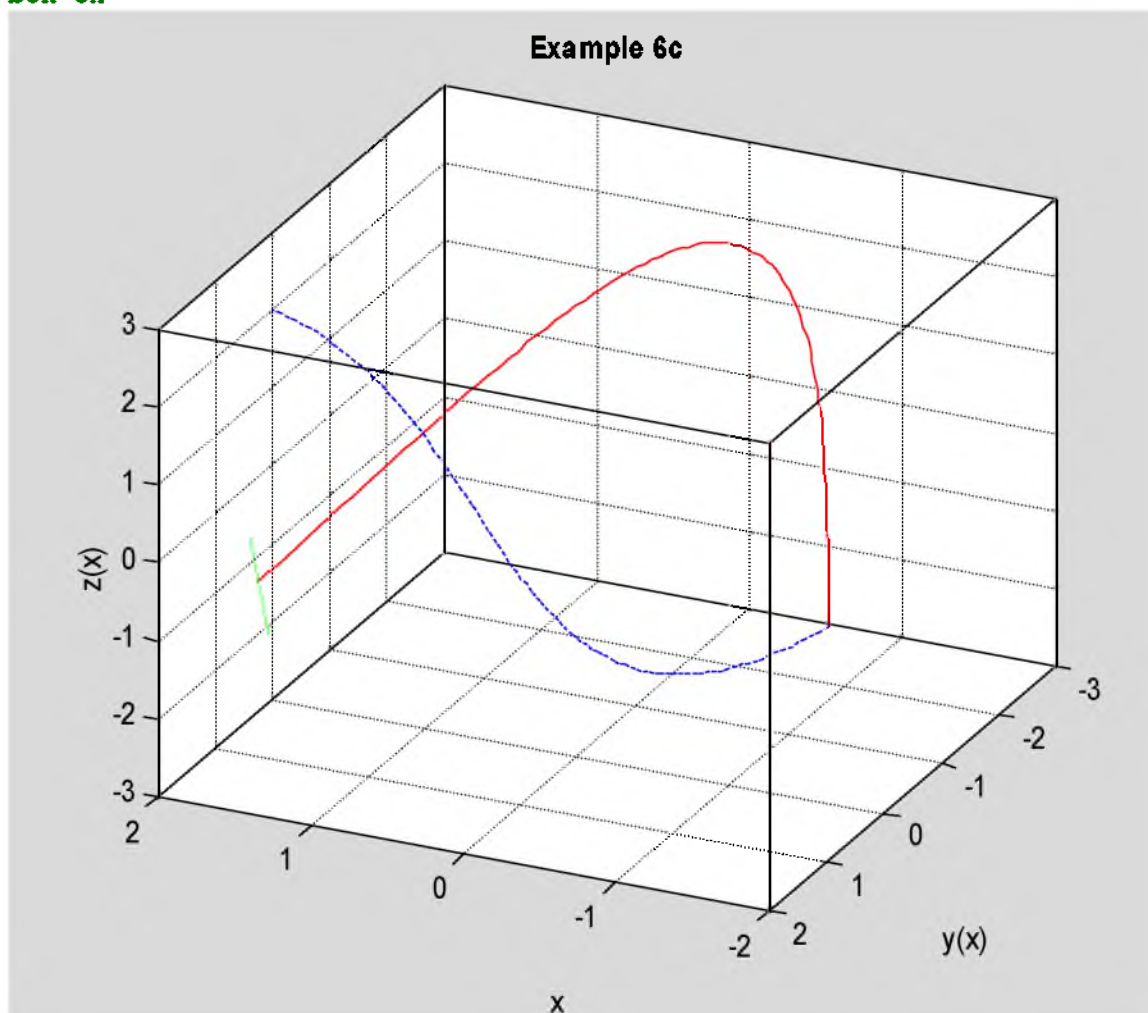


Рис. 6.5. Решение примера 6с

Ответ. Экстремаль функционала (6.23) имеет вид

$$\begin{cases} y(x)=0.0676928e^{-x}-0.0013735e^x- \\ -2.17942\cos x+0.447553\sin x; \\ z(x)=0.0676928e^{-x}-0.0013735e^x+ \\ +2.17942\cos x-0.447553\sin x; \end{cases} \quad x \in [-2; 1.76886]. \quad (6.24)$$

График экстремалей приведен на рис.6.5. Сплошной красной линией показано решение примера 6с, штриховой синей – решение примера 2, зелёным цветом показана ограничивающая линия (6.17).

6.3. Задание

- Для функционала 1а найти экстремаль, при условии, что правый конец движется по заданной линии.
- Для функционала 2 найти экстремаль, при условии, что правый конец движется по заданной поверхности.
- Для функционала 2 найти экстремаль, при условии, что правый конец движется по заданной линии.

6.4. Варианты заданий

Вариант 1.	a). $\varphi(x)=e^x-2$;	b). $\varphi(x,y)=e^x+y^2-4$;	c). $\varphi=2-x^2$; $\psi=3-e^x$;
Вариант 2.	a). $\varphi(x)=e^{2x}-9$;	b). $\varphi(x,y)=e^x+2y^2-4$;	c). $\varphi=3-x^2$; $\psi=2-e^x$;
Вариант 3.	a). $\varphi(x)=e^{x+0.5}-3$;	b). $\varphi(x,y)=2e^x+y^2-6$;	c). $\varphi=2-x^2$; $\psi=1-e^x$;
Вариант 4.	a). $\varphi(x)=e^x-10$;	b). $\varphi(x,y)=4-x^2-e^x$;	c). $\varphi=1-e^x$; $\psi=2-x^2$;
Вариант 5.	a). $\varphi(x)=2-e^{2x}$;	b). $\varphi(x,y)=5-x^2-e^{2y}$;	c). $\varphi=5-e^{2x}$; $\psi=1-x^2$;
Вариант 6.	a). $\varphi(x)=2-e^{2x}$;	b). $\varphi(x,y)=4-x^2-e^x$;	c). $\varphi=4-e^{2x}$; $\psi=2-x^2$;
Вариант 7.	a). $\varphi(x)=e^x-2$;	b). $\varphi(x,y)=4-2x^2-2y^2$;	c). $\varphi=4-2\sin x$; $\psi=2-x^2$;
Вариант 8.	a). $\varphi(x)=e^{x+0.5}-2$;	b). $\varphi(x,y)=4-2x^2-2\cos y$;	c). $\varphi=4-2\cos x$; $\psi=2-x^3$;
Вариант 9.	a). $\varphi(x)=e^{2x}-5$;	b). $\varphi(x,y)=5-2x^2-3\cos y$;	c). $\varphi=5-x^2$; $\psi=4-2\cos x$;
Вариант 10.	a). $\varphi(x)=0.5e^x-2$;	b). $\varphi(x,y)=3-e^x-\cos y$;	c). $\varphi=3-2x^2$; $\psi=3-\cos x$;
Вариант 11.	a). $\varphi(x)=e^x-5$;	b). $\varphi(x,y)=3-e^x-y^2/5$;	c). $\varphi=x^2-2$; $\psi=\sin x$;
Вариант 12.	a). $\varphi(x)=-e^{0.5x}$;	b). $\varphi(x)=100-99x$;	c). $\varphi=3-3x^2$; $\psi=3\sin x$;
Вариант 13.	a). $\varphi(x)=5-e^x$;	b). $\varphi(x,y)=3-e^x-2\cos(xy)$;	c). $\varphi=2-2x^2$; $\psi=3\sin 2x$;
Вариант 14.	a). $\varphi(x)=e^x-2$;	b). $\varphi(x,y)=3-e^x-2\cos y$;	c). $\varphi=3-3x^2$; $\psi=2\sin 2x$;
Вариант 15.	a). $\varphi(x)=2-e^{2x}$;	b). $\varphi(x,y)=e^{\frac{x}{y}}+2\cos y$;	c). $\varphi=\sin(x/2)-2$; $\psi=\sin x$;
Вариант 16.	a). $\varphi(x)=3-e^x$;	b). $\varphi(x,y)=e^x+2\sin(y/2)-5$;	c). $\varphi=\cos 2x$; $\psi=2\sin(x/2)$;
Вариант 17.	a). $\varphi(x)=4-e^x$;	b). $\varphi(x,y)=e^x+3\sin(y/2)-6$;	c). $\varphi=-2\sin(x/2)$; $\psi=\cos 2x$;
Вариант 18.	a). $\varphi(x)=6-e^x$;	b). $\varphi(x,y)=e^x+2\sin(y/2)-5$;	c). $\varphi=-2\sin(x/2)$; $\psi=2\cos 2x$;
Вариант 19.	a). $\varphi(x)=10-e^{2x}$;	b). $\varphi(x,y)=e^x+2\sin(y/2)-4$;	c). $\varphi=-3\sin(x/2)$; $\psi=2\cos 2x$;
Вариант 20.	a). $\varphi(x)=e^x-2$;	b). $\varphi(x,y)=e^{\frac{x}{2}}+\cos y-1$;	c). $\varphi=-2\sin(x/2)$; $\psi=3\cos 2x$;
Вариант 21.	a). $\varphi(x)=5-e^{0.5x}$;	b). $\varphi(x,y)=2-e^y+\cos x$;	c). $\varphi=-2e^{\frac{x}{2}}$; $\psi=3\cos 2x$;
Вариант 22.	a). $\varphi(x)=10-2e^{x-2}$;	b). $\varphi(x,y)=2-e^{x-4}+\cos y$;	c). $\varphi=2e^{\frac{x}{2}}$; $\psi=2\cos x$;
Вариант 23.	a). $\varphi(x)=5-e^x$;	b). $\varphi(x,y)=3-e^{x-5}+\cos y$;	c). $\varphi=4\sin x$; $\psi=2\cos(x/2)$;
Вариант 24.	a). $\varphi(x)=50-e^{x+3}$;	b). $\varphi(x,y)=1-e^{x-3}+5\cos(y/2)$;	c). $\varphi=3\sin x$; $\psi=3\cos(x/2)$;
Вариант 25.	a). $\varphi(x)=50-e^{x+3}$;	b). $\varphi(x,y)=100-50x$;	c). $\varphi=3\sin(x/2)$; $\psi=-4\cos x$;
Вариант 26.	a). $\varphi(x)=0.5e^x-2$;	b). $\varphi(x,y)=1-e^{x-3}+5\cos(y/2)$;	c). $\varphi=2\sin(3x/2)$; $\psi=-4\cos x$;
Вариант 27.	a). $\varphi(x)=50-e^x$;	b). $\varphi(x,y)=3-2e^{x-1}+5\cos(y/3)$;	c). $\varphi=-2\sin(3x/2)$; $\psi=4\cos(x/2)$;
Вариант 28.	a). $\varphi(x)=e^{2x-2}-5$;	b). $\varphi(x,y)=3-2e^{x-1}+5\cos(y/2)$;	c). $\varphi=100-50x$; $\psi=100-50x$;
Вариант 29.	a). $\varphi(x)=5-e^{2x}$;	b). $\varphi(x,y)=3-2e^{x-1}+5\cos(y/2)$;	c). $\varphi=2\sin(3x/2)$; $\psi=-2\cos x$;
Вариант 30.	a). $\varphi(x)=25-e^x$;	b). $\varphi(x,y)=3+\cos(y/2)-2e^{x-2}$;	c). $\varphi=2\sin(3x/2)$; $\psi=2\cos(x/2)$;

7. Отражение экстремалей

7.1. Краткие теоретические сведения

Рассмотрим задачу вариационного исчисления для функционала (1.1), зависящего от функции одной переменной и её производной. Пусть для этого функционала заданы граничные условия (1.2) и дополнительное требование: экстремаль $y(x)$ касается заданной кривой $y=\varphi(x)$ в неизвестной точке $M_0(x_0, y_0)$, где $x_0 \in [x_1, x_2]$ (рис. 7.1). В точке $M_0(x_0, y_0)$ экстремаль не обязательно должна быть гладкой.

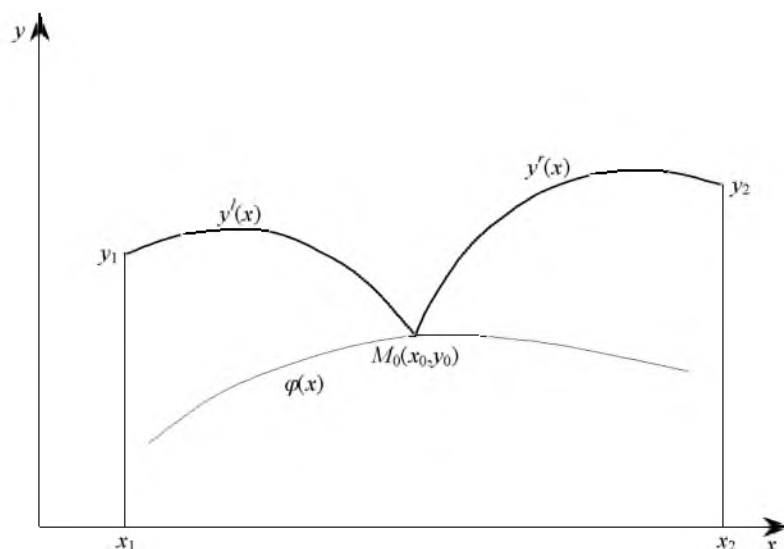


Рис. 7.1. Отражение экстремалей

Выведем необходимые условия экстремума для этой вариационной задачи. Так как функционал $J(y)$ достигает экстремума на интервале $[x_1, x_2]$, то он достигает также экстремума и на интервалах $[x_1, x_0]$ и $[x_0, x_2]$. Следовательно, на каждом из этих интервалов экстремаль должна удовлетворять дифференциальному уравнению Эйлера (1.5). Решения этого уравнения на левом $y'(x)$ и на правом интервалах $y''(x)$ содержат по 2 произвольные постоянные: C_1^l, C_2^l и C_1^r, C_2^r . Кроме того, неизвестны также 2 координаты точки касания x_0, y_0 . Для определения этих 6 неизвестных мы имеем уравнения:

1. граничное условие на левом конце $y'(x_1)=y_1$;
2. граничное условие на правом конце $y''(x_2)=y_2$;
3. $y'(x)$ проходит через точку M_0 : $y'(x_0)=y_0$;
4. $y''(x)$ также проходит через M_0 : $y''(x_0)=y_0$;
5. точка M_0 лежит на линии $y=\varphi(x)$: $y_0=\varphi(x_0)$.

Недостающее 6-е уравнение получим из необходимого условия экстремума функционала: $\delta J=0$. Вариация функционала на каждом из двух участков δJ_1 и δJ_2 вызывается вариацией функции $y(x)$ (вместе с её производной $y'(x)$) и вариацией точки δx_0 . В главе 6 было выведено условие трансверсальности (6.7) для функционала с подвижной правой границей. В нашей задаче это δJ_1 :

$$\delta J_1 = (F + (\varphi' - y')F_{y'}) \Big|_{x=x_0-0} \delta x_0. \quad (7.1)$$

Аналогично

$$\delta J_2 = -(F + (\varphi' - y'')F_{y''}) \Big|_{x=x_0+0} \delta x_0. \quad (7.2)$$

Приравняв сумму этих вариаций нулю и учитывая, что δx_0 — произвольная, получим недостающее 6-е уравнение:

$$(F + (\varphi' - y')F_{y'}) \Big|_{x=x_0-0} = (F + (\varphi' - y'')F_{y''}) \Big|_{x=x_0+0}. \quad (7.3)$$

Это условие является условием трансверсальности для задачи отражения. Его смысл следующий: если точку x_0 двигать по линии $y=\varphi(x)$ и находить пару экстремалей (решений уравнения Эйлера) $y'(x)$ и $y''(x)$, удовлетворяющих граничным условиям, то доставлять экстремум функционалу будет та пара функций, которая удовлетворяет условию трансверсальности (7.3).

Задача для самостоятельного решения. Выведите условия трансверсальности для задачи отражения функционала (2.1), зависящего от двух функций. Рассмотрите случаи, когда экстремаль отражается от заданной поверхности (вариант 1) и от заданной линии (вариант 2).

7.2. Пример выполнения заданий

Найти экстремаль функционала, рассмотренного ранее в примере 1а, при тех же граничных условиях, и при дополнительном требовании: экстремаль касается заданной линии в точке $M_0(x_0, y_0)$. Сравнить решение с 1а.

$$J(y) = \int_{-1}^1 (x^2 + y^2 + y'^2) dx; \quad \begin{cases} y(-1)=1; \\ y(1)=2; \end{cases} \quad y_0 = 0.8 - 0.2x_0^2. \quad (7.4)$$

Для составления программы воспользуемся программой для решения примера 6а, а также другими программами, составленными ранее. Как и в предыдущих задачах, вначале найдём решение примера 1а.

```
clear all
format long
disp('Решаем пример 7')
syms x y Dy D2y % описали символические переменные
F = x^2+y^2+Dy^2; % подынтегральная функция
x1 = -1;
y1 = 1;
x2 = 1;
y2 = 2;
phi = 0.8-0.2*x^2; % линия касания
fprintf('Подынтегральная функция: F=%s\n',char(F))
fprintf('Граничные условия: y(%d)=%d; y(%d)=%d\n',x1,y1,x2,y2)
fprintf('Линия отражения: y=%s\n',char(phi))
dFdy = diff(F,y);
dFdyl = diff(F,Dy);
d_dFdyl_dx = diff(dFdyl,x); % d(dF/dy')/dx
d_dFdyl_dy = diff(dFdyl,y); % d(dF/dy')/dy
d_dFdyl_dy1 = diff(dFdyl,Dy); % d(dF/dy')/dy'
dFyldx = d_dFdyl_dx + d_dFdyl_dy*Dy + d_dFdyl_dy1*D2y;
Euler = simple(dFdy-dFyldx);
deqEuler = [ char(Euler) '=0' ]; % составили уравнение
Sol = dsolve(deqEuler,'x'); % решаем уравнение Эйлера
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
SolLeft = subs(Sol,x,sym(x1)); % подставляем x1
SolRight = subs(Sol,x,sym(x2)); % подставляем x2
EqLeft = [char(SolLeft) '=' char(sym(y1))];
EqRight = [char(SolRight) '=' char(sym(y2))];
Con = solve(EqLeft,EqRight); % решаем систему
C1 = Con.C1;
C2 = Con.C2;
Sol1a = vpa(eval(Sol),14); % подставляем C1, C2
xpl = linspace(x1,x2);
yla = subs(Sol1a,x,xpl);
Решаем пример 7
Подынтегральная функция: F=x^2+y^2+Dy^2
Граничные условия: y(-1)=1; y(1)=2
Линия отражения: y=4/5-1/5*x^2
```

Начинаем формирование файла для вычисления левых частей системы нелинейных уравнений. Описываем заголовок функции. Присваиваем аргументы нужным параметрам. Восстанавливаем символические константы. Находим решения на левом и правом участках в нужных точках. Формируем первые 5 уравнений системы. Вычисляем необходимые функции для условия трансверсальности (7.3). Формируем строки для 6-го уравнения. Записываем функции в файл.

```
f{1} = 'function y = MyFunc(x)'; % заголовок
f{2} = 'C1l=x(1); C2l=x(2); C1r=x(3); C2r=x(4); x0=x(5); y0=x(6); y=x;';
syms C1 C2 C1l C2l C1r C2r % символические константы
y1 = subs(Sol,{C1,C2},{C1l,C2l});
y1x1 = subs(y1,x,sym(x1));
y1x0 = subs(y1,x,sym('x0'));
yr = subs(Sol,{C1,C2},{C1r,C2r});
yrx0 = subs(yr,x,sym('x0'));
```

```

yrx2 = subs(yr,x,sym(x2));
phix0 = subs(phi,x,sym('x0'));
f{3} = [ 'y(1)=' char(ylx1) '-' char(sym(y1)) ';' ];
f{4} = [ 'y(2)=' char(ylx0) '-y0;' ];
f{5} = [ 'y(3)=' char(yrx0) '-y0;' ];
f{6} = [ 'y(4)=' char(yrx2) '-' char(sym(y2)) ';' ];
f{7} = [ 'y(5)=' char(phix0) '-y0;' ];
dyl = diff(yl,x);
dyr = diff(yr,x);
dphi = diff(phi,x);
dyl0 = subs(dyl,x,sym('x0'));
dyr0 = subs(dyr,x,sym('x0'));
dphi0 = subs(dphi,x,sym('x0'));
F10 = subs(F,{x,y,Dy},{sym('x0'),sym('ylx0'),sym('dyl0')});
Fr0 = subs(F,{x,y,Dy},{sym('x0'),sym('yrx0'),sym('dyr0')});
dF10 = subs ( dFdy1, {x,y,Dy}, {sym('x0'),sym('ylx0'),sym('dyl0')} );
dFr0 = subs ( dFdy1, {x,y,Dy}, {sym('x0'),sym('yrx0'),sym('dyr0')} );
f{8} = [ 'ylx0=' char(ylx0) ';' ];
f{9} = [ 'yrx0=' char(yrx0) ';' ];
f{10} = [ 'dyl0=' char(dyl0) ';' ];
f{11} = [ 'dyr0=' char(dyr0) ';' ];
f{12} = [ 'dphi0=' char(dphi0) ';' ];
f{13} = [ 'F10=' char(F10) ';' ];
f{14} = [ 'Fr0=' char(Fr0) ';' ];
f{15} = [ 'dF10=' char(dF10) ';' ];
f{16} = [ 'dFr0=' char(dFr0) ';' ];
f{17} = 'y(6)=(F10+(dphi0-dyl0)*dF10)-(Fr0+(dphi0-dyr0)*dFr0)';
disp('Текст файла MyFunc.m')
fprintf('%s\n',f{:});
fid = fopen ( 'C:\Iglin\Matlab\MyFunc.m', 'w' );
fprintf(fid,'%s\n',f{:});
fclose(fid); % закрываем файл
Т е к с т   ф а й л а   М y F u n c . m
function y = MyFunc(x)
C1l=x(1); C2l=x(2); C1r=x(3); C2r=x(4); x0=x(5); y0=x(6); y=x;
y(1)=(C1l*exp(-1)^2+C2l)/exp(-1)-1;
y(2)=(C1l*exp(x0)^2+C2l)/exp(x0)-y0;
y(3)=(C1r*exp(x0)^2+C2r)/exp(x0)-y0;
y(4)=(C1r*exp(1)^2+C2r)/exp(1)-2;
y(5)=4/5-1/5*x0^2-y0;
ylx0=(C1l*exp(x0)^2+C2l)/exp(x0);
yrx0=(C1r*exp(x0)^2+C2r)/exp(x0);
dyl0=2*C1l*exp(x0)-(C1l*exp(x0)^2+C2l)/exp(x0);
dyr0=2*C1r*exp(x0)-(C1r*exp(x0)^2+C2r)/exp(x0);
dphi0=-2/5*x0;
F10=x0^2+ylx0^2+dyl0^2;
Fr0=x0^2+yrx0^2+dyr0^2;
dF10=2*dyl0;
dFr0=2*dyr0;
y(6)=(F10+(dphi0-dyl0)*dF10)-(Fr0+(dphi0-dyr0)*dFr0);

```

Для быстрой сходимости метода Ньютона нужно удачно задать начальное приближение. Зададим x_0 посередине интервала $[x_1, x_2]$, и $y_0 = \varphi(x_0)$. Начальные значения произвольных постоянных C_1^l , C_2^l и C_1^r , C_2^r найдём из условия, что линии $y^l(x)$ и $y^r(x)$ являются экстремальными каждая на своём участке. Запишем на каждом участке систему уравнений – граничных условий и решим её. Сформируем начальное приближение.

```

initx0 = (x1+x2)/2; % середина интервала
inity0 = eval(subs(phi,x,sym(initx0)));
syms C1left C2left C1right C2right
yleft = subs ( Sol, {C1,C2}, {C1left,C2left} );
yleftx1 = subs(yleft,x,sym(x1));
yleftx0 = subs(yleft,x,sym(initx0));
Eq1 = [ char(yleftx1) '=' char(sym(y1)) ];

```

```
Eq2 = [ char(yleftx0) '=' char(sym(inity0)) ];
Con = solve(Eq1,Eq2); % решаем
C1left = Con.C1left;
C2left = Con.C2left;
yright = subs ( Sol, {C1,C2}, {C1right,C2right} );
yrightx2 = subs(yright,x,sym(x2));
yrightx0 = subs(yright,x,sym(inity0));
Eq1 = [ char(yrightx2) '=' char(sym(y2)) ];
Eq2 = [ char(yrightx0) '=' char(sym(inity0)) ];
Con = solve(Eq1,Eq2); % решаем
C1right = Con.C1right;
C2right = Con.C2right;
xinit = [eval(C1left);eval(C2left);eval(C1right);eval(C2right);initx0;inity0]
xinit =
    0.49975505008007
    0.30024494991993
    0.72570401403959
    0.07429598596041
         0
    0.800000000000000
```

Решаем нелинейную систему уравнений. Печатаем решения.

```
[xzero,options] = fsolve('MyFunc',xinit);
fprintf('Количество вычислений функций: %d\n',options(10))
C1l = vpa(sym(xzero(1)),14)
C2l= vpa(sym(xzero(2)),14)
C1r = vpa(sym(xzero(3)),14)
C2r = vpa(sym(xzero(4)),14)
x0 = xzero(5)
y0 = xzero(6)
y1 = vpa(eval(y1),14);
yr = vpa(eval(yr),14);
fprintf('Уравнения экстремали: \nYleft(x)=%s\nYright(x)=%s\n',char(y1),char(yr))
Количество вычислений функций: 44
C1l =
    .51086412707861
C2l =
    .29874147846989
C1r =
    .71006201975852
C2r =
    .18987550963062
x0 =
    -0.30200362642249
y0 =
    0.78176677852392
Уравнения экстремали:
Yleft(x)=(.51086412707861*exp(x)^2+.29874147846989)/exp(x)
Yright(x)=(.71006201975852*exp(x)^2+.18987550963062)/exp(x)
```

Заполняем таблицы для графиков функций $y=\varphi(x)$ и решения задачи. Строим графики. Функцию $y=\varphi(x)$ строим на небольшом участке.

```
xi = linspace(x0-0.2,x0+0.2);
yi = subs(phi,x,xi);
x1 = xpl(find(xpl<x0));
xr = xpl(find(xpl>=x0));
y7 = [subs(y1,x,x1),subs(yr,x,xr)]; % функция
plot(xpl,y1a,'--b',xi,yi,':g',xpl,y7,'-r')
title('Example 7') % заголовок
xlabel('x')
ylabel('y(x), phi(x)') % метки осей
```

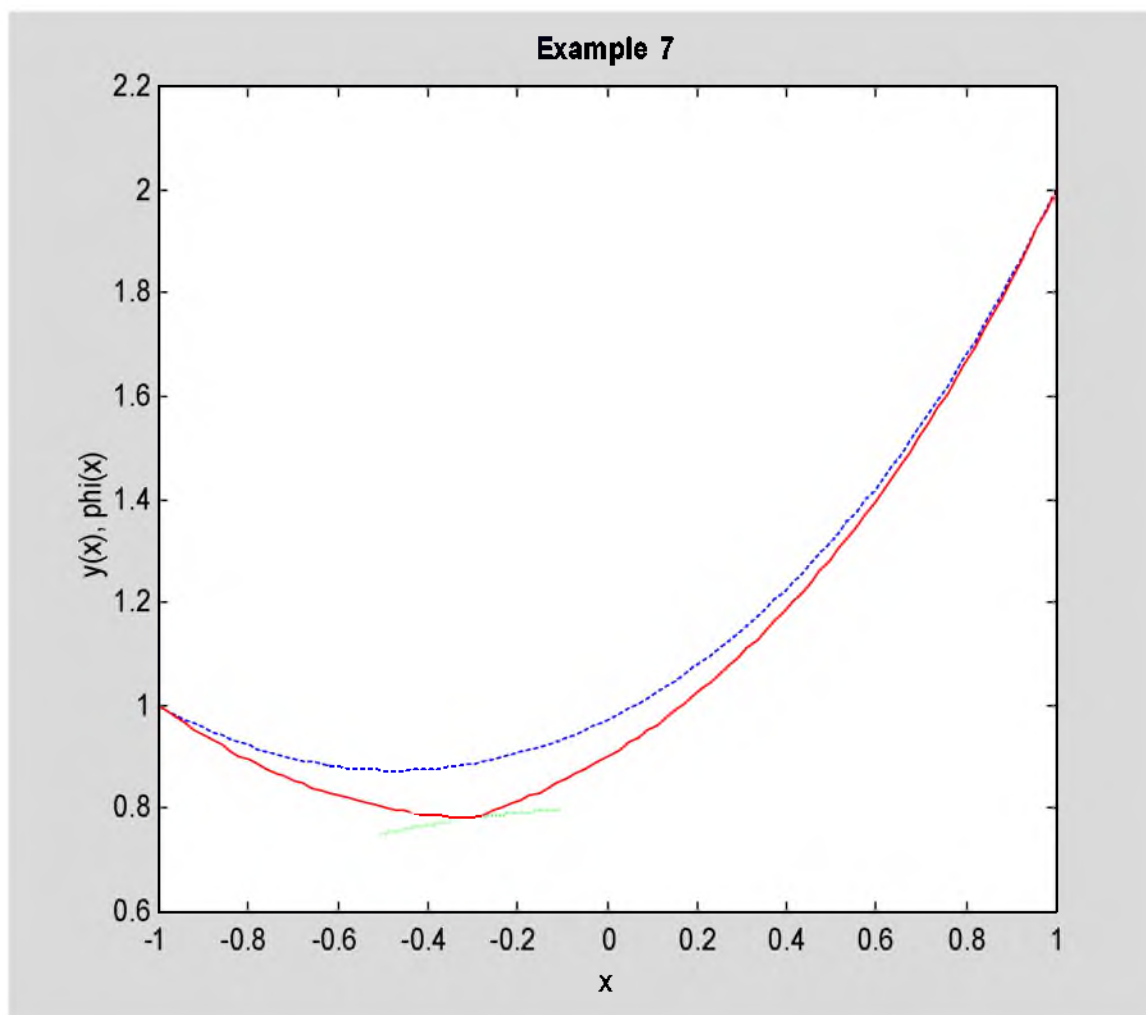


Рис. 7.2. Решение примера 7

Ответ. Экстремаль функционала (7.4) имеет вид

$$y(x) = \begin{cases} 0.510828e^x + 0.298746e^{-x}; & x < -0.302079; \\ 0.710065e^x + 0.189857e^{-x}; & x \geq -0.302079. \end{cases} \quad (7.5)$$

График экстремали показан на рис.7.2 сплошной красной линией. Штриховая синяя линия – решение примера 1а, зелёным цветом показан участок линии, которой касается экстремаль.

7.3. Задание

Для функционала 1а найти экстремаль при дополнительном условии, что эта экстремаль касается заданной линии.

7.4. Варианты заданий

Вариант 1. $\varphi(x) = 0.4 - x^2$;

Вариант 3. $\varphi(x) = 2x^2 + 1$;

Вариант 5. $\varphi(x) = -(x+1)^2 - 0.6$;

Вариант 7. $\varphi(x) = -(x-0.5)^2 - 0.2$;

Вариант 9. $\varphi(x) = -x^2 - 1$;

Вариант 11. $\varphi(x) = 0.5 - (x-1)^2$;

Вариант 13. $\varphi(x) = -3x^2 - 2$;

Вариант 15. $\varphi(x) = 0.5 - (x-0.5)^2$;

Вариант 17. $\varphi(x) = 2 - x^2$;

Вариант 19. $\varphi(x) = -2(x-1)^2 - 2$;

Вариант 2. $\varphi(x) = -7 - 3(x-0.5)^2$;

Вариант 4. $\varphi(x) = 2(x-1)^2 + 0.5$;

Вариант 6. $\varphi(x) = -2x^2 - 1$;

Вариант 8. $\varphi(x) = -(x-0.5)^2$;

Вариант 10. $\varphi(x) = 1 - (x-0.5)^2$;

Вариант 12. $\varphi(x) = 3 + 2x^2$;

Вариант 14. $\varphi(x) = (x-1)^2 + 4$;

Вариант 16. $\varphi(x) = 1 - (x+1)^2$;

Вариант 18. $\varphi(x) = 2 - (x-0.5)^2$;

Вариант 20. $\varphi(x) = -x^2 - 1$;

Вариант 21. $\varphi(x)=0.5-(x-1)^2$;

Вариант 23. $\varphi(x)=-2x^2-3$;

Вариант 25. $\varphi(x)=-3(x+1)^2$;

Вариант 27. $\varphi(x)=-(x-3)^2-6$;

Вариант 29. $\varphi(x)=-3x^2-2$;

Вариант 22. $\varphi(x)=-(x+1)^2-3$;

Вариант 24. $\varphi(x)=-3(x+1)^2$;

Вариант 26. $\varphi(x)=0.5-(x-0.5)^2$;

Вариант 28. $\varphi(x)=-2-x^2$;

Вариант 30. $\varphi(x)=-(x-1)^2-2$;

8. Преломление экстремалей

8.1. Краткие теоретические сведения

Пусть кривая $y=\varphi(x)$ разделяет плоскость xOy на две области, и точки $M_1(x_1, y_1)$ и $M_2(x_2, y_2)$ расположены по разные стороны от этой кривой (рис.8.1).

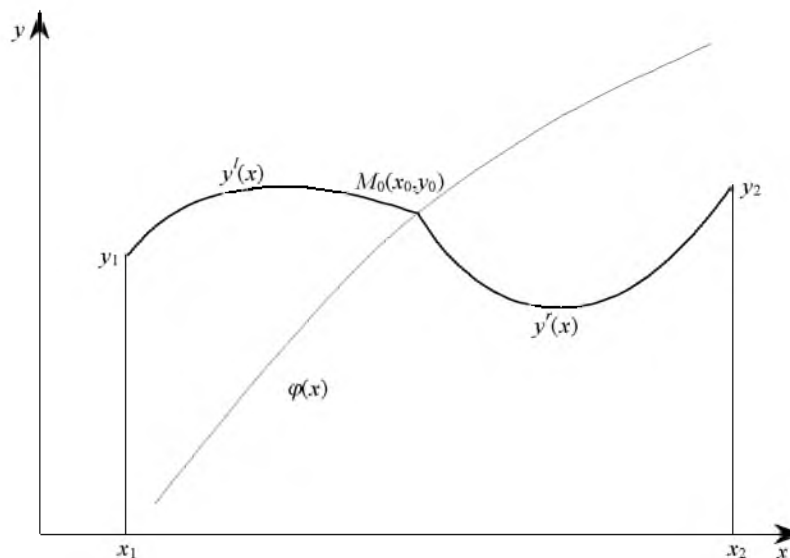


Рис. 8.1. Преломление экстремалей

Рассмотрим задачу вариационного исчисления для функционала

$$J(y) = \int_{x_1}^{x_0} F^l(x, y, y') dx + \int_{x_0}^{x_2} F^r(x, y, y') dx. \quad (8.1)$$

Он отличается от функционала, рассмотренного в главе 7 тем, что в нём подынтегральные функции слева и справа от точки $M_0(x_0, y_0)$ – разные. Для вывода необходимого условия экстремума этого функционала воспользуемся результатами главы 7. Каждая из кривых $y'1(x)$, $y'2(x)$ является решением уравнения Эйлера для своей функции F и содержит по 2 произвольные постоянные: C_1^l , C_2^l и C_1^r , C_2^r . Для нахождения этих 4-х констант и 2-х координат точки преломления $M_0(x_0, y_0)$ у нас есть уравнения:

1. граничное условие на левом конце $y'1(x_1)=y_1$;
2. граничное условие на правом конце $y'2(x_2)=y_2$;
3. $y'1(x)$ проходит через точку M_0 : $y'1(x_0)=y_0$;
4. $y'2(x)$ также проходит через M_0 : $y'2(x_0)=y_0$;
5. точка M_0 лежит на линии $y=\varphi(x)$: $y_0=\varphi(x_0)$.

Недостающее 6-е уравнение получим из необходимого условия экстремума функционала: $\delta J=0$. Оно отличается от (7.3) тем, что в левую и правую части равенства входят разные функции F :

$$\left(F^l + (\varphi' - y') F_{y'}^l \right) \Big|_{x=x_0-0} = \left(F^r + (\varphi' - y') F_{y'}^r \right) \Big|_{x=x_0+0}. \quad (8.2)$$

Уравнение (8.2) является условием трансверсальности для задачи преломления. Его смысл: если точку x_0 двигать по линии $y=\varphi(x)$ и находить пару экстремалей (решений уравнения Эйлера) $y'1(x)$ и $y'2(x)$, удовлетворяющих граничным условиям, то доставлять экстремум функционалу будет та пара функций, которая удовлетворяет условию трансверсальности (8.2).

Задача для самостоятельного решения. Выведите условие трансверсальности для задачи преломления функционала (2.1), зависящего от двух функций, когда экстремаль преломляется на заданной поверхности.

8.2. Пример выполнения задания

Найти экстремаль функционала

$$J(y) = \int_{-1}^{x_0} (x^2 + y^2 + y'^2) dx + \int_{x_0}^1 (y'^2 + 2y' \operatorname{sh} x - 5x^2) dx, \quad \begin{cases} y(-1)=1; \\ y(1)=3; \end{cases} \quad y_0 = 5 - 2e^{x_0}. \quad (8.3)$$

Будем составлять программу для решения примера 8 на основе программы для примера 7 с использованием других программ. Описываем необходимые переменные и вводим исходные данные.


```
clear all
format long
disp('Решаем пример 8')
syms x y Dy D2y % описали символические переменные
F1 = x^2+y^2+Dy^2;
x1 = -1;
y1 = 1;
Fr = Dy^2+2*Dy*sinh(x)-5*x^2;
x2 = 1;
y2 = 3;
phi = 5-2*exp(x); % линия преломления
fprintf('Подынтегральная функция на левом участке: Fleft=%s\n',char(F1))
fprintf('Граничное условие слева: y(%d)=%d\n',x1,y1)
fprintf('Подынтегральная функция на правом участке: Fright=%s\n',char(Fr))
fprintf('Граничное условие справа: y(%d)=%d\n',x2,y2)
fprintf('Линия преломления: phi=%s\n',char(phi))
Решаем пример 8
Подынтегральная функция на левом участке:
Fleft=x^2+y^2+Dy^2
Граничное условие слева: y(-1)=1
Подынтегральная функция на правом участке:
Fright=Dy^2+2*Dy*sinh(x)-5*x^2
Граничное условие справа: y(1)=3
Линия преломления: phi=5-2*exp(x)
```

Составляем и решаем дифференциальное уравнение Эйлера на левом участке. Для этого воспользуемся фрагментом программы для примера 1а, в котором во всех переменных добавим суффикс 1, так как решение находится на левом участке.

```
dF1dy = diff(F1,y);
dF1dy1 = diff(F1,Dy);
d_dF1dy1_dx = diff(dF1dy1,x); % d(dF1/dy')/dx
d_dF1dy1_dy = diff(dF1dy1,y); % d(dF1/dy')/dy
d_dF1dy1_dy1 = diff(dF1dy1,Dy); % d(dF1/dy')/dy'
dFly1dx = d_dF1dy1_dx + d_dF1dy1_dy*Dy + d_dF1dy1_dy1*D2y;
EulerL = simple(dF1dy-dFly1dx);
deqEulerL = [char(EulerL) '=0']; % уравнение
SolL = dsolve(deqEulerL,'x'); % решаем уравнение Эйлера
if length(SolL)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
fprintf('Решение на левом участке:\nYleft=%s\n',char(SolL))
Решение на левом участке:
Yleft=(C1*exp(x)^2+C2)/exp(x)
```

Для формирования первого интеграла уравнения Эйлера на правом участке используем фрагмент программы для примера 1b, во всех переменных добавим суффикс r.

```
dFr1dy = simple(diff(Fr,Dy)); % dFr/dy'
deqEulerR = [char(dFr1dy) '=C']; % уравнение
SolR = dsolve(deqEulerR,'x'); % решаем
if length(SolR)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
fprintf('Решение на правом участке:\nYright=%s\n',char(SolR))
Решение на правом участке:
Yright=-cosh(x)+1/2*C*x+C1
```

Начинаем формирование системы нелинейных уравнений для определения произвольных постоянных и координат точки преломления. Записываем заголовок функции, переобозначаем переменные. Описываем необходимые символические переменные. Находим аналитические выражения для решений на обоих интервалах в необходимых точках. Первые 4 уравнения – это граничные условия. 5-е уравнение – точка лежит

на кривой $y=\varphi(x)$. Далее формируем условие transversальности (8.2). Для этого вычисляем входящие в (8.2) функции в нужных точках, переводим в строки 9 вспомогательных функций, и записываем само уравнение. Записываем систему уравнений в файл.

```
f{1} = 'function y = MyFunc(x); % заголовок
f{2} = 'C1l=x(1); C2l=x(2); C1r=x(3); C2r=x(4); x0=x(5); y0=x(6); y=x;';
syms C C1 C2 C1l C2l C1r C2r % описали переменные
y1 = subs(SolL,{C1,C2},{C1l,C2l});
ylx1 = subs(y1,x,sym('x0'));
ylx0 = subs(y1,x,sym('x0'));
yr = subs(SolR,{C1,C},{C1r,C2r});
yrx0 = subs(yr,x,sym('x0'));
yrx2 = subs(yr,x,sym('x2'));
phix0 = subs(phi,x,sym('x0'));
f{3} = [ 'y(1)= ' char(ylx1) '-' char(sym(y1)) ';' ];
f{4} = [ 'y(2)= ' char(ylx0) '-' y0; ' ]; % уравнение 2
f{5} = [ 'y(3)= ' char(yrx0) '-' y0; ' ]; % уравнение 3
f{6} = [ 'y(4)= ' char(yrx2) '-' char(sym(y2)) ';' ];
f{7} = [ 'y(5)= ' char(phix0) '-' y0; ' ]; % уравнение 5
dyldx = diff(y1,x);
dyrdx = diff(yr,x);
dphidx = diff(phi,x);
dphi0 = subs(dphidx,x,sym('x0'));
dy10 = subs(dyldx,x,sym('x0'));
dyr0 = subs(dyrdx,x,sym('x0'));
F10 = subs(F1,{x,y,Dy},{sym('x0'),sym('ylx0'),sym('dy10')});
Fr0 = subs(Fr,{x,y,Dy},{sym('x0'),sym('yrx0'),sym('dyr0')});
dF10 = subs ( dF1dy1, {x,y,Dy}, {sym('x0'),sym('ylx0'),sym('dy10')} );
dFr0 = subs ( dFrDy1, {x,y,Dy}, {sym('x0'),sym('yrx0'),sym('dyr0')} );
f{8} = [ 'ylx0= ' char(ylx0) , ' ' ];
f{9} = [ 'yrx0= ' char(yrx0) , ' ' ];
f{10} = [ 'dy10= ' char(dy10) , ' ' ];
f{11} = [ 'dyr0= ' char(dyr0) , ' ' ];
f{12} = [ 'dphi0= ' char(dphi0) , ' ' ];
f{13} = [ 'F10= ' char(F10) , ' ' ];
f{14} = [ 'Fr0= ' char(Fr0) , ' ' ];
f{15} = [ 'dF10= ' char(dF10) , ' ' ];
f{16} = [ 'dFr0= ' char(dFr0) , ' ' ];
f{17}='y(6)=(F10+(dphi0-dy10)*dF10)-(Fr0+(dphi0-dyr0)*dFr0)';
disp('Текст файла MyFunc.m')
fprintf('%s\n',f{:});
fid = fopen ( 'C:\Iglin\Matlab\MyFunc.m', 'w' );
fprintf(fid,'%s\n',f{:});
fclose(fid); % закрываем файл

Т е к с т   ф а й л а   MyFunc.m
function y = MyFunc(x)
C1l=x(1); C2l=x(2); C1r=x(3); C2r=x(4); x0=x(5); y0=x(6); y=x;
y(1)=(C1l*exp(-1)^2+C2l)/exp(-1)-1;
y(2)=(C1l*exp(x0)^2+C2l)/exp(x0)-y0;
y(3)=-cosh(x0)+1/2*C2r*x0+C1r-y0;
y(4)=-cosh(1)+1/2*C2r+C1r-3;
y(5)=5-2*exp(x0)-y0;
ylx0=(C1l*exp(x0)^2+C2l)/exp(x0);
yrx0=-cosh(x0)+1/2*C2r*x0+C1r;
dy10=2*C1l*exp(x0)-(C1l*exp(x0)^2+C2l)/exp(x0);
dyr0=-sinh(x0)+1/2*C2r;
dphi0=-2*exp(x0);
F10=x0^2+ylx0^2+dy10^2;
Fr0=dyr0^2+2*dyr0*sinh(x0)-5*x0^2;
dF10=2*dy10;
dFr0=2*dyr0+2*sinh(x0);
y(6)=(F10+(dphi0-dy10)*dF10)-(Fr0+(dphi0-dyr0)*dFr0);
```

Начальную точку $M_0(x_0, y_0)$ берём посередине интервала $[x_1, x_2]$ на линии $y=\varphi(x)$. Начальные значения произвольных постоянных C_1^l, C_2^l и C_1^r и C_2^r находим из условий, что каждая из кривых $y^l(x), y^r(x)$ является экстремалью на своём участке. Запишем на каждом участке систему уравнений – граничных условий и решим её. Сформируем начальное приближение.

```

initx0 = (x1+x2)/2; % середина интервала
inity0 = eval(subs(phi,x,sym(initx0)));
syms C1left C2left C1right C2right
yleft = subs ( SolL, {C1,C2}, {C1left,C2left} );
yleftx1 = subs(yleft,x,sym(x1));
yleftx0 = subs(yleft,x,sym(initx0));
Eq1 = [ char(yleftx1) '=' char(sym(y1)) ];
Eq2 = [ char(yleftx0) '=' char(sym(inity0)) ];
Con = solve(Eq1,Eq2); % решаем
C1left = Con.C1left;
C2left = Con.C2left;
yright = subs ( SolR, {C1,C}, {C1right,C2right} );
yrightx2 = subs(yright,x,sym(x2));
yrightx0 = subs(yright,x,sym(initx0));
Eq1 = [ char(yrightx2) '=' char(sym(y2)) ];
Eq2 = [ char(yrightx0) '=' char(sym(inity0)) ];
Con = solve(Eq1,Eq2); % решаем
C1right = Con.C1right;
C2right = Con.C2right;
xinit = [eval(C1left);eval(C2left);eval(C1right);eval(C2right);initx0;inity0]
xinit =
    3.04409386412934
   -0.04409386412934
    4.00000000000000
    1.08616126963049
         0
    3.00000000000000

```

Решаем систему нелинейных уравнений. Печатаем количество итераций. Находим аналитические решения на интервалах.

```

[xzero,options] = fsolve('MyFunc',xinit);
fprintf('Количество вычислений функций: %d\n',options(10))
C1l = vpa(sym(xzero(1)),14)
C2l = vpa(sym(xzero(2)),14)
C1r = vpa(sym(xzero(3)),14)
C2r = vpa(sym(xzero(4)),14)
x0 = xzero(5)
y0 = xzero(6)
y1 = vpa(eval(y1),14);
yr = vpa(eval(yr),14);
fprintf('Уравнения экстремали: \nYleft(x)=%s\nYright(x)=%s\n',char(y1),char(yr))
Количество вычислений функций: 83
C1l =
1.7154497130289
C2l =
.13571904256960
C1r =
2.9671894157856
C2r =
3.1517827551092
x0 =
0.27612757268543
y0 =
2.36396963006479
Уравнения экстремали:
Yleft(x)=(1.7154497130289*exp(x)^2+.13571904256960)/exp(x)
Yright(x)=-1.*cosh(x)+1.5758913775546*x+2.9671894157856

```

Заполняем таблицы для построения графиков функции $y=\varphi(x)$ и решения. Строим графики. Функцию $\varphi(x)$ строим на небольшом участке.

```
xpl = linspace(x1,x2); % массив абсцисс для графика
xi = linspace(x0-0.2,x0+0.2);
yi = subs(phi,x,xi); % phi(x)
xl = xpl(find(xpl<x0));
xr = xpl(find(xpl>=x0));
y8 = [subs(y1,x,xl),subs(yr,x,xr)]; % функция
plot(xi,yi,'g',xpl,y8,'-r') % рисуем график
title ( '\bfExample 8' ) % заголовок
xlabel('x')
ylabel('y(x), phi(x)') % метки осей
```

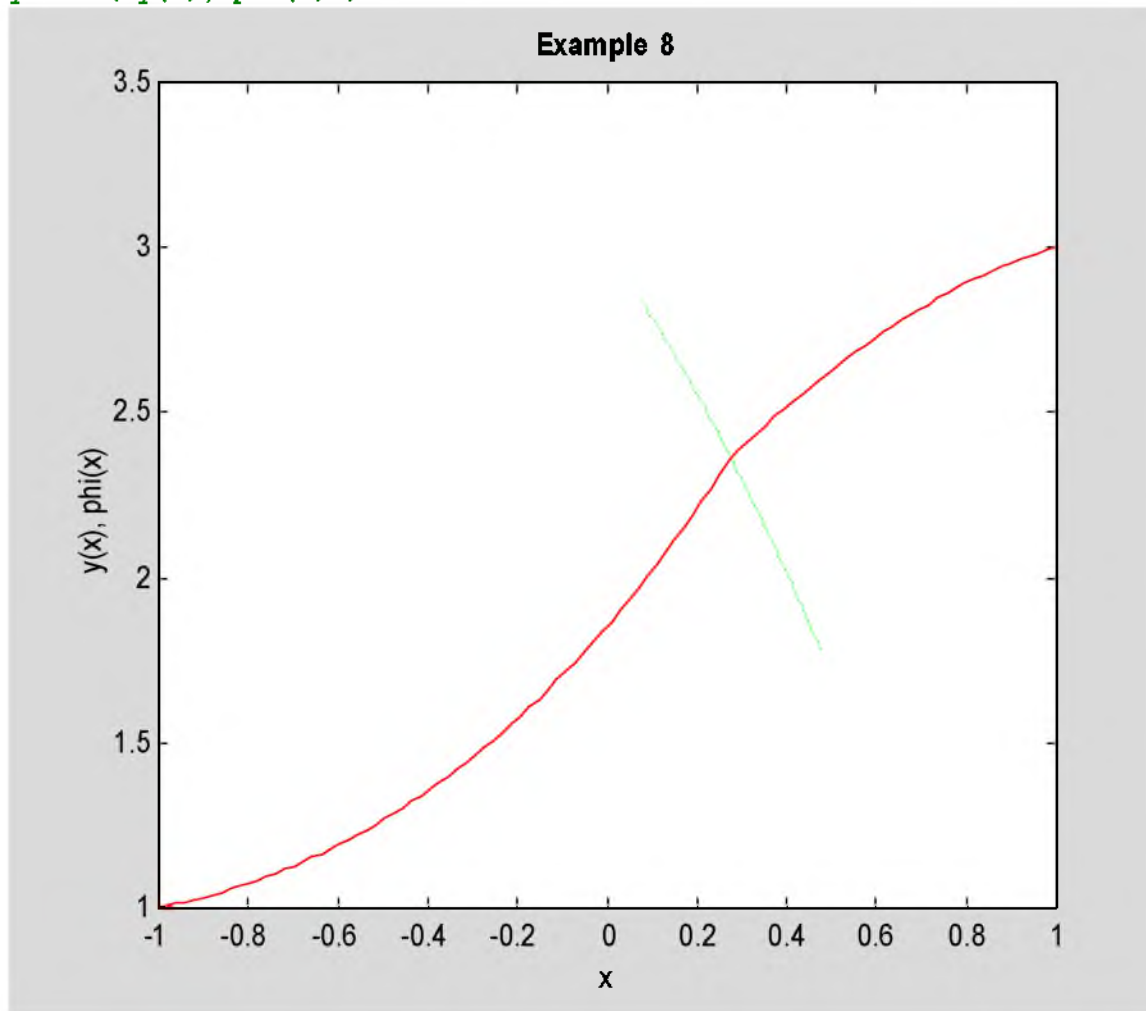


Рис. 8.2. Решение примера 8

Ответ. Экстремаль функционала (8.3) имеет вид

$$y(x) = \begin{cases} 1.7154526e^x + 0.1357182e^{-x}; & x < 0.276128; \\ -\operatorname{ch} x + 1.575897x + 2.967183; & x \geq 0.276128. \end{cases} \quad (8.4)$$

График экстремали показан на рис.8.2 сплошной красной линией. Штриховая зелёная линия – линия преломления $y=\varphi(x)$.

8.3. Задание

Решить задачу преломления экстремалей для функционала вида (1.1). Подынтегральную функцию на участке $[x_1, x_0]$ и граничное условие на левом конце $M_1(x_1, y_1)$ взять из варианта 1а, а функцию на правом участке $[x_0, x_2]$ и правое граничное условие $M_2(x_2, y_2)$ – из варианта 1б. Линия преломления задана.

8.4. Варианты заданий

Вариант 1. $\varphi(x)=2(x+1)^2-10$;

Вариант 3. $\varphi(x)=2(x+1)^2-3$;

Вариант 5. $\varphi(x)=10-2(x+2)^2$;

Вариант 7. $\varphi(x)=4-2(x+1)^2$;

Вариант 9. $\varphi(x)=3-2(x+1)^2$;

Вариант 11. $\varphi(x)=x^2-2$;

Вариант 13. $\varphi(x)=x^2$;

Вариант 15. $\varphi(x)=(x+1)^2$;

Вариант 17. $\varphi(x)=(x+1)^2$;

Вариант 19. $\varphi(x)=x^2-1$;

Вариант 21. $\varphi(x)=2-\frac{x^2}{8}$;

Вариант 23. $\varphi(x)=x^2-14$;

Вариант 25. $\varphi(x)=8-10x^2$;

Вариант 27. $\varphi(x)=2(x-2)^2-8$;

Вариант 29. $\varphi(x)=6-\frac{x^2}{2}$;

Вариант 2. $\varphi(x)=(x+1)^2-5$;

Вариант 4. $\varphi(x)=x^2-4$;

Вариант 6. $\varphi(x)=4-5x^2$;

Вариант 8. $\varphi(x)=4-2(x+1)^2$;

Вариант 10. $\varphi(x)=5-3x^2$;

Вариант 12. $\varphi(x)=2-x^2$;

Вариант 14. $\varphi(x)=x^2-3$;

Вариант 16. $\varphi(x)=(x+1)^2-2$;

Вариант 18. $\varphi(x)=(x+1)^2$;

Вариант 20. $\varphi(x)=10-3x^2$;

Вариант 22. $\varphi(x)=5-2e^{x-2}$;

Вариант 24. $\varphi(x)=6-\frac{(x+1)^2}{2}$;

Вариант 26. $\varphi(x)=4x^2-10$;

Вариант 28. $\varphi(x)=2x^2-10$;

Вариант 30. $\varphi(x)=x^2-3$;

9. Односторонние вариации

9.1. Краткие теоретические сведения

Рассмотрим задачу вариационного исчисления для функционала (1.1) при граничных условиях (1.2) и при дополнительном условии: допустимая кривая не должна попасть в область D , ограниченную линией $L: y=\varphi(x)$ (рис.9.1). Будем предполагать, что решение задачи нетривиальное, то есть если область D не запретная, то экстремаль проходит через неё (штриховая линия на рис.9.1).

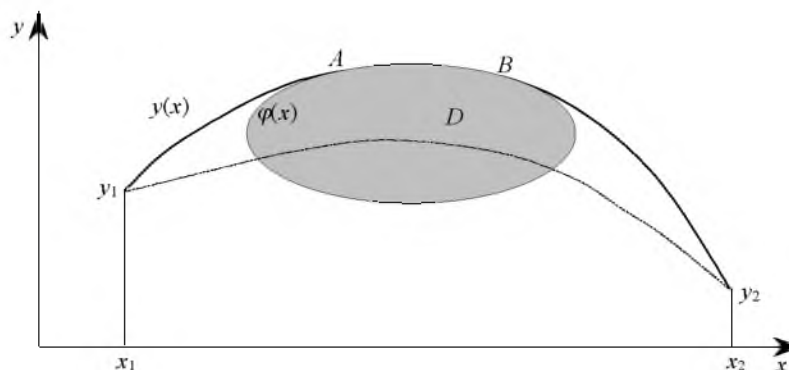


Рис. 9.1. Односторонние вариации

Для этой задачи экстремаль состоит из дуг, лежащих вне D , и частей L . При этом на границе (дуга AB) возможны только односторонние вариации функции $y(x)$: в сторону вне области D . Выведем необходимое условие экстремума для этой вариационной задачи. Для упрощения выкладок рассмотрим случай, когда имеется только одна точка сопряжения M_0 (рис.9.2).

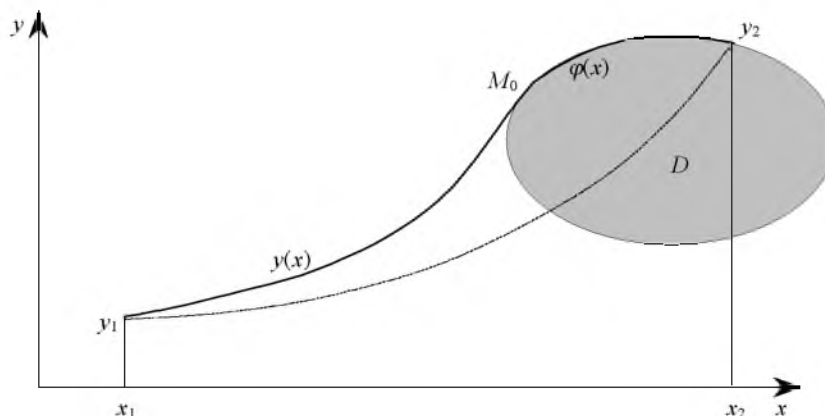


Рис. 9.2. Одна точка сопряжения

Необходимым условием экстремума функционала является равенство нулю его вариации: $\delta J=0$. Вариация функционала вызывается вариацией функции $y(x)$ вместе с её производной на участке M_1M_0 и вариацией точки M_0 . Вычислим вариации функционала: δJ_1 на участке M_1M_0 и δJ_2 на участке M_0M_2 . δJ_1 мы находили ранее при выводе условий трансверсальности. Это формула (6.6), в которой вместо x_2 стоит x_0 .

$$\delta J_1 = \left(F + (\varphi' - y') F_{y'} \right) \Big|_{x=x_0} \delta x_0. \quad (9.1)$$

На втором участке функция не варьируется, варьируется только положение точки x_0 . Поэтому

$$\delta J_2 = \int_{x_0 + \delta x_0}^{x_2} F(x, y, y') dx - \int_{x_0}^{x_2} F(x, y, y') dx = - \int_{x_0}^{x_0 + \delta x_0} F(x, \varphi(x), \varphi'(x)) dx. \quad (9.2)$$

Интервал интегрирования малый, по теореме о среднем с точностью до малых высшего порядка

$$\delta J_2 = -F(x, \varphi(x), \varphi'(x)) \Big|_{x=x_0} \delta x_0. \quad (9.3)$$

Сложим (9.1) и (9.3). Так как δx_0 произвольная, то множитель при δx_0 равен 0:

$$\delta J = \left(F(x, y, y') - F(x, \varphi, \varphi') + (\varphi' - y') F_{y'} \right) \Big|_{x=x_0} = 0. \quad (9.4)$$

Упростим полученное выражение. Разложим первое слагаемое в (9.4) в ряд Тейлора по 3-му аргументу в окрестности φ' , и удержим малые только 1-го порядка.

$$F(x, y, y') = F(x, y, \varphi') + F_{y'}(x, y, \varphi')(y' - \varphi'). \quad (9.5)$$

Подставим (9.5) в (9.4) и учтём, что в точке сопряжения x_0 выполняется условие $y(x) = \varphi(x)$.

$$\delta J = \left(F_{y'}(x, y, \varphi')(y' - \varphi') - (y' - \varphi') F_{y'}(x, y, y') \right) \Big|_{x=x_0} = 0. \quad (9.6)$$

Применим ещё раз теорему о среднем к выражению (9.6). С точностью до малых высшего порядка

$$\delta J = \left((y' - \varphi')^2 F_{y'y'}(x, y, y') \right) \Big|_{x=x_0} = 0. \quad (9.7)$$

Если экстремум на данной экстремали действительно достигается, то выполняется достаточное условие Лежандра, и $F_{y'y'} \neq 0$. Тогда из (9.7) получаем условие сопряжения

$$y'(x_0) = \varphi'(x_0). \quad (9.8)$$

Условие сопряжения (9.8) является условием трансверсальности для нашей задачи. Его смысл следующий. Если на кривой $y = \varphi(x)$ выбирать различные точки x_0 , и по полученным точкам x_0 строить экстремали на участке $[x_1, x_0]$ и добавлять к ним участки дуг $[x_0, x_2]$ функции $\varphi(x)$, то доставлять экстремум функционалу будет та линия, для которой выполняется условие (9.8).

Для нахождения двух произвольных констант, входящих в выражение для экстремали, и координат точки сопряжения $M_0(x_0, y_0)$ у нас есть 4 уравнения:

1. граничное условие на левом конце $y(x_1) = y_1$;
2. граничное условие на правом конце $y(x_0) = y_0$;
3. точка M_0 лежит на линии $y = \varphi(x)$: $y_0 = \varphi(x_0)$;
4. условие сопряжения (9.8).

Задача для самостоятельного решения. Выведите условия сопряжения функционала (2.1), зависящего от двух функций, когда в пространстве задана поверхность, ограничивающая некоторую запретную область.

9.2. Пример выполнения задания

Найти экстремаль функционала

$$J(y) = \int_{-1}^1 (x^2 + y^2 + y'^2) dx; \quad \begin{cases} y(-1) = 1; \\ y(1) = 2; \end{cases} \quad y \geq 2 - (x-1)^2. \quad (9.9)$$

Сравнить решение с решением примера 1а. Построить графики.

Составим программу для решения данного примера на основе программы для примера 8 с использованием других программ. Описываем необходимые переменные и вводим исходные данные. Оставляем решение примера 1а.

```
clear all
format long
disp('Решаем пример 9')
syms x y Dy D2y % описали символические переменные
F = x^2 + y^2 + Dy^2;
x1 = -1;
y1 = 1;
x2 = 1;
y2 = 2;
phi = 2 - (x-1)^2;
fprintf('Подынтегральная функция: F=%s\n', char(F))
fprintf('Граничные условия: y(%d)=%d; y(%d)=%d\n', x1, y1, x2, y2)
fprintf('Ограничение на правом конце: y=%s\n', char(phi))
dFdy = diff(F, y);
dFdy1 = diff(F, Dy);
d_dFdy1_dx = diff(dFdy1, x);
d_dFdy1_dy = diff(dFdy1, y);
d_dFdy1_dy1 = diff(dFdy1, Dy); % d(dF/dy')/dy'
dFyldx = d_dFdy1_dx + d_dFdy1_dy*Dy + d_dFdy1_dy1*D2y;
Euler = simple(dFdy - dFyldx);
deqEuler = [ char(Euler) '=0' ]; % уравнение
Sol = dsolve(deqEuler, 'x'); % решаем уравнение Эйлера
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
SolLeft = subs(Sol, x, sym(x1));
SolRight = subs(Sol, x, sym(x2));
```



```
EqLeft = [char(SolLeft) '=' char(sym(y1))];
EqRight = [char(SolRight) '=' char(sym(y2))];
Con = solve(EqLeft,EqRight); % решаем систему
C1 = Con.C1;
C2 = Con.C2;
Sol1a = vpa(eval(Sol),14);
xpl = linspace(x1,x2);
y1a = subs(Sol1a,x,xpl);
Решаем пример 9
Подынтегральная функция:  $F=x^2+y^2+Dy^2$ 
Граничные условия:  $y(-1)=1$ ;  $y(1)=2$ 
Ограничение на правом конце:  $y=2-(x-1)^2$ 
```

Сохраняем численные значения найденных произвольных постоянных. Они будут нам нужны как начальное приближение при решении системы нелинейных уравнений. Формируем систему нелинейных уравнений для нахождения произвольных постоянных и точки сопряжения M_0 . Записываем её в файл.

```
C1e = eval(C1);
C2e = eval(C2);
f{1} = 'function y = MyFunc(x)'; % заголовок
f{2} = 'C1=x(1); C2=x(2); x0=x(3); y0=x(4); y=x;';
syms C1 C2 % описали символические константы
yx1 = subs(Sol,x,sym(x1));
yx0 = subs(Sol,x,sym('x0'));
dydx = diff(Sol,x);
dphidx = diff(phi,x);
dydxx0 = subs(dydx,x,sym('x0'));
phidx0 = subs(phi,x,sym('x0'));
dphidx0 = subs(dphidx,x,sym('x0'));
f{3} = [ 'y(1)=' char(yx1) '-' char(sym(y1)) ';' ];
f{4} = [ 'y(2)=' char(yx0) '-y0;' ]; % уравнение 2
f{5} = [ 'y(3)=' char(phidx0) '-y0;' ]; % уравнение 3
f{6} = [ 'dydxx0=', char(dydxx0), ';' ];
f{7} = [ 'dphidx0=', char(dphidx0), ';' ];
f{8} = 'y(4)=dydxx0-dphidx0;';
disp('Текст файла MyFunc.m')
fprintf('%s\n',f{:});
fid = fopen('C:\Iglin\Matlab\MyFunc.m','w');
fprintf(fid,'%s\n',f{:});
fclose(fid); % закрываем файл
Текст файла MyFunc.m
function y = MyFunc(x)
C1=x(1); C2=x(2); x0=x(3); y0=x(4); y=x;
y(1)=(C1*exp(-1)^2+C2)/exp(-1)-1;
y(2)=(C1*exp(x0)^2+C2)/exp(x0)-y0;
y(3)=2-(x0-1)^2-y0;
dydxx0=2*C1*exp(x0)-(C1*exp(x0)^2+C2)/exp(x0);
dphidx0=-2*x0+2;
y(4)=dydxx0-dphidx0;
```

В качестве начального приближения для произвольных постоянных берём их значения из решения примера 1a. Начальное приближение для точки M_0 – точка M_2 . Решаем полученную систему нелинейных уравнений. Печатаем аналитическое решение.

```
xinit = [ C1e; C2e; x2; y2 ] % начальное приближение
[xzero,options]=fsolve('MyFunc',xinit);
fprintf('Количество вычислений функций: %d\n',options(10))
C1 = vpa(sym(xzero(1)),14)
C2 = vpa(sym(xzero(2)),14)
x0 = xzero(3)
y0 = xzero(4)
Sol9 = vpa(eval(Sol),14); % аналитическое решение
fprintf('Уравнение экстремали:\nu=%s\n',char(Sol9))
```

```
xinit =  
    0.69877023730774  
    0.27331117318808  
    1.00000000000000  
    2.00000000000000  
Количество вычислений функций: 27  
C1 =  
.99615664942608  
C2 =  
.23306493394882  
x0 =  
    0.36410789260554  
y0 =  
    1.59565485618002  
Уравнение экстремали:  
y=(.99615664942608*exp(x)^2+.23306493394882)/exp(x)
```

Заполняем таблицы для функции $\varphi(x)$ и для решения. Строим график.

```
xi = linspace(x0-0.4,x2);  
yi = subs(phi,x,xi); % phi(x)  
xl=xpl(find(xpl<x0));  
xr=xpl(find(xpl>=x0));  
y9 = [subs(Sol9,x,xl),subs(phi,x,xr)]; % функция  
plot(xpl,y1a,'--b', xi,yi,':g',xpl,y9,'-r') % график  
title('\bfExample 9')  
xlabel('x')  
ylabel('y(x), phi(x)')
```

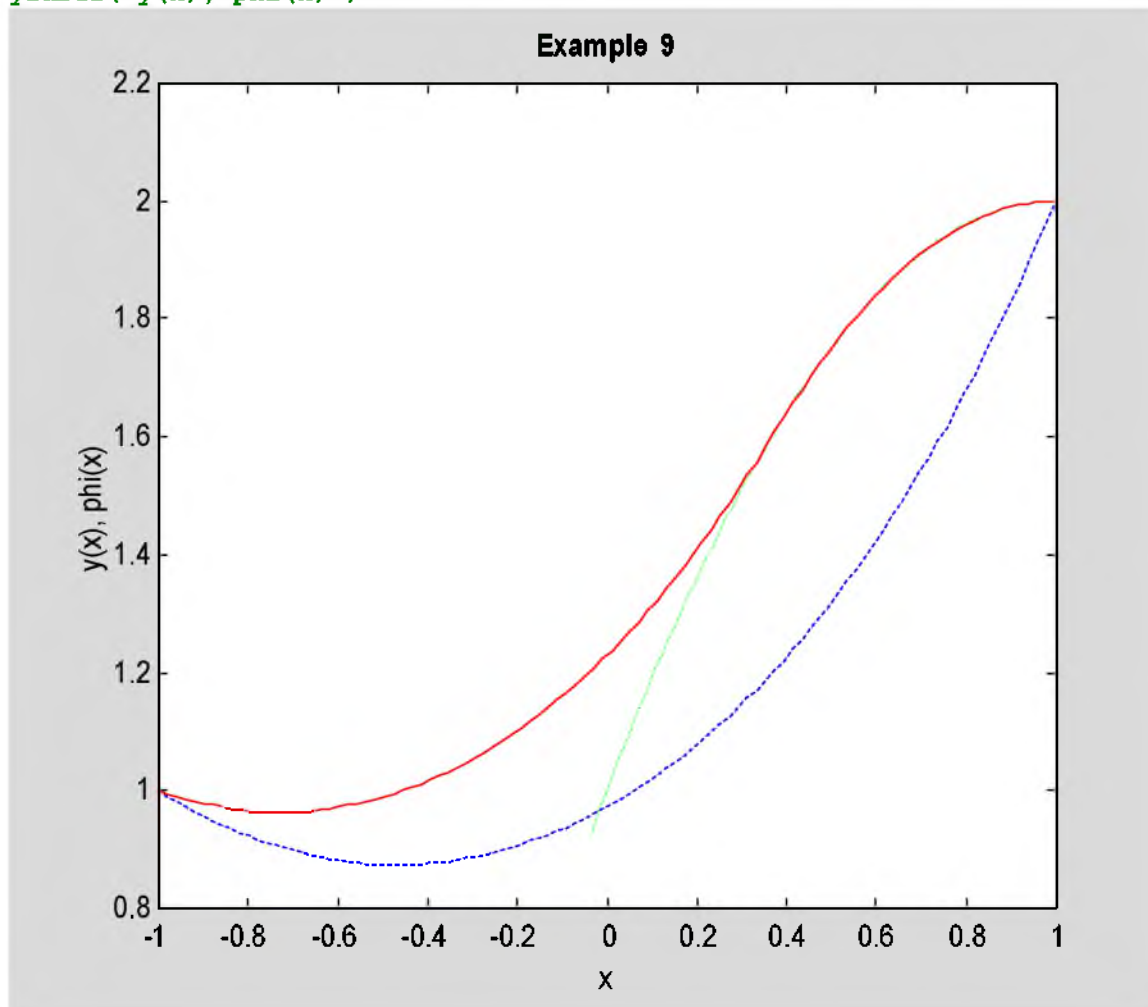


Рис. 9.3. Решение примера 9

Ответ. Экстремаль функционала (9.9) имеет вид

$$y(x) = \begin{cases} 0.99616e^x + 0.23306e^{-x}; & x < 0.36411; \\ 2 - (x-1)^2; & x \geq 0.36411. \end{cases} \quad (9.10)$$

Она показана на рис.9.3 сплошной красной линией. Штриховая синяя линия – решение примера 1а. Показана также линия сопряжения $y=\varphi(x)$ пунктирной линией зелёного цвета.

9.3. Задание

Решить пример 1а при условии, что экстремаль не может заходить в запрещённую область D . Уравнение границы области задано. Сравнить результат с решением примера 1а.

9.4. Варианты заданий

Вариант 1. $y \geq 1 - (x-1)^2$;

Вариант 2. $y \geq 4 - 40(x-1)^2$;

Вариант 3. $y \geq \frac{1}{2} - (x-1)^2$;

Вариант 4. $y \geq 2 - (x-2)^2$;

Вариант 5. $y \geq 1 - 3x^2$;

Вариант 6. $y \geq -(x-1)^2 - 1$;

Вариант 7. $y \geq 3 - (x-1)^2$;

Вариант 8. $y \geq 3 - (x-1)^2$;

Вариант 9. $y \geq 3 - (x-1)^2$;

Вариант 10. $y \geq 2 - (x-2)^2$;

Вариант 11. $y \geq 2 - (x-2)^2$;

Вариант 12. $y \leq (x-1.5)^2 + 3.75$;

Вариант 13. $y \leq (x-2)^2 + 2$;

Вариант 14. $y \leq (x-2)^2 + 3$;

Вариант 15. $y \geq 2 - (x-1)^2$;

Вариант 16. $y \geq 1 - (x-1)^2$;

Вариант 17. $y \geq 3 - (x-1)^2$;

Вариант 18. $y \geq 4 - (x-1)^2$;

Вариант 19. $y \leq 10(x-1.8)^2 + 2.6$;

Вариант 20. $y \geq 3 - (x-2)^2$;

Вариант 21. $y \geq 3 - (x-2)^2$;

Вариант 22. $y \leq 6\cos(3x-12) - 2$;

Вариант 23. $y \geq 3\sqrt{2x}\cos(3x-6) - 3$;

Вариант 24. $y \leq 20 - 20\cos(3x)$;

Вариант 25. $y \leq 22 - 20\cos(3x-3)$;

Вариант 26. $y \geq 3 - (x-2)^2$;

Вариант 27. $y \leq 6\cos(3x-12) - 3$;

Вариант 28. $y \geq 3x\cos(3x-6) - 3$;

Вариант 29. $y \leq 26 - 25\cos(3x-3)$;

Вариант 30. $y \geq 2\sqrt{3x}\cos(2x-6) - 2$;

10. Изопериметрическая задача

10.1. Краткие теоретические сведения

Изопериметрической задачей в узком смысле слова называется задача нахождения экстремума функционала (1.1) при граничных условиях (1.2) и при дополнительном ограничении-равенстве на длину дуги M_1M_2 :

$$\int_{x_1}^{x_2} \sqrt{1+y'^2} dx = \text{const}. \quad (10.1)$$

Изопериметрическими являются две классические задачи вариационного исчисления: задача Дидоны и задача о цепной линии.

Задача Дидоны. По преданию, Дидона была дочерью одного из древних царей и его наследницей. Однако братья решили лишить её престола. Чтобы посмеяться над ней, они вывели Дидону на пустынный берег моря, дали её шкуру быка и сказали: «Построй себе город на этой шкуре, и царствуй в нём». Не растерялась Дидона. Порезала она шкуру на тонкие полоски, связала их в длинную верёвку, и охватила этой верёвкой большой кусок прибрежной полосы. Удивились братья мудрости и находчивости Дидоны, и не стали препятствовать строительству нового города. Так, по преданию, был основан Карфаген. А задача, которую решила Дидона, стала классической задачей вариационного исчисления: линией заданной длины ограничить максимально возможную площадь. Если концы линии закреплены на береговой линии (оси Ox), то решением задачи будет сегмент круга. Если же один из концов может скользить по береговой линии, то решение задачи Дидоны – полукруг.

Задача о цепной линии. Найти форму провисания цепи, состоящей из большого числа мелких звеньев. Или, что то же самое, найти форму провисания тонкой абсолютно гибкой нерастяжимой нити. Математически эта задача сводится к минимизации потенциальной энергии нити (вертикальной координаты центра тяжести) при ограничении-равенстве на длину линии. Решением этой задачи является цепная линия – гиперболический косинус.

Изопериметрическая задача в широком смысле слова – это задача исследования на экстремум функционала вида (1.1), (2.1) или (3.1) с соответствующими граничными условиями и с ограничениями-равенствами в виде интегралов такого же вида. Эти ограничения называются условиями изопериметричности. В частности, для простейшей задачи вариационного исчисления (1.1, 1.2) условия изопериметричности имеют вид

$$J_k = \int_{x_1}^{x_2} F_k(x, y, y') dx = l_k = \text{const}; \quad k = \overline{1, n}. \quad (10.2)$$

Рассмотрим применение метода неопределённых множителей Лагранжа для решения задачи (1.1, 1.2, 10.2). Эта задача решается так же, как и для функции нескольких переменных при ограничениях-равенствах.

Составляется вспомогательный функционал (функционал Лагранжа)

$$L(y) = \int_{x_1}^{x_2} \left(F(x, y, y') + \sum_{k=1}^n \lambda_k F_k(x, y, y') \right) dx, \quad (10.3)$$

и для этого функционала решается вариационная задача. Неизвестные произвольные постоянные уравнения Эйлера C_1, C_2 и неопределённые множители Лагранжа λ_k находятся из граничных условий (1.2) и условий изопериметричности (10.2).

10.2. Пример выполнения задания

Найти экстремаль функционала

$$J(y) = \int_{-1}^1 (x^2 + y^2 + y'^2) dx; \quad \begin{cases} y(-1) = 1; \\ y(1) = 2; \end{cases} \quad \int_{x_1}^{x_2} y dx = 2. \quad (10.4)$$

Сравнить решение с решением примера 1а. Построить графики.

Составим программу для этого примера на основе программы для примера 1а с использованием других программ. Описываем необходимые переменные и вводим исходные данные. Повторяем решение примера 1а.

```
clear all
format long
disp('Решаем пример 10')
syms x y Dy D2y lambda % символические переменные
F = x^2 + y^2 + Dy^2; % подынтегральная функция
x1 = -1;
y1 = 1;
x2 = 1;
```

```

y2 = 2;
F1 = y;
J1 = 2; % функция-ограничение
fprintf('Подынтегральная функция: F=%s\n',char(F))
fprintf('Граничные условия: y(%d)=%d; y(%d)=%d\n',x1,y1,x2,y2)
fprintf('Условие изопериметричности:
Int(%s, 'x', %d, %d)=%d\n',char(F1),x1,x2,J1)
dFdy = diff(F,y);
dFdy1 = diff(F,Dy);
d_dFdy1_dx = diff(dFdy1,x);
d_dFdy1_dy = diff(dFdy1,y);
d_dFdy1_dy1 = diff(dFdy1,Dy); % d(dF/dy')/dy'
dFyldx = d_dFdy1_dx + d_dFdy1_dy*Dy + d_dFdy1_dy1*D2y;
Euler = simple(dFdy-dFyldx);
deqEuler = [ char(Euler) '=0' ]; % составили уравнение
Sol = dsolve(deqEuler,'x'); % решаем уравнение Эйлера
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
SolLeft = subs(Sol,x,sym(x1));
SolRight = subs(Sol,x,sym(x2));
EqLeft = [char(vpa(SolLeft,14)) '=' char(sym(y1))];
EqRight = [char(vpa(SolRight,14)) '=' char(sym(y2))];
Con = solve(EqLeft,EqRight); % решаем систему
C1 = Con.C1;
C2 = Con.C2;
Sol1a = vpa(eval(Sol),14); % подставляем C1, C2
xpl = linspace(x1,x2);
yla = subs(Sol1a,x,xpl);
Решаем пример 10
Подынтегральная функция: F=x^2+y^2+Dy^2
Граничные условия: y(-1)=1; y(1)=2
Условие изопериметричности: Int(y,'x',-1,1)=2

```

Формируем и решаем уравнение Эйлера для модифицированного функционала с учётом условия изопериметричности.

```

L = F+lambda*F1; % модифицированный функционал
fprintf('Модифицированная подынтегральная функция L=%s\n',char(L))
dLdy = diff(L,y);
dLdy1 = diff(L,Dy);
d_dLdy1_dx = diff(dLdy1,x);
d_dLdy1_dy = diff(dLdy1,y);
d_dLdy1_dy1 = diff(dLdy1,Dy); % d(dL/dy')/dy'
dLyldx = d_dLdy1_dx + d_dLdy1_dy*Dy + d_dLdy1_dy1*D2y;
EulerL = simple(dLdy-dLyldx);
deqEulerL = [char(EulerL) '=0']; % составили уравнение
fprintf('Уравнение Эйлера для неё:\n%s\n',deqEulerL)
disp('Решение уравнения Эйлера:')
SolL = dsolve(deqEulerL,'x') % решаем уравнение Эйлера
if length(SolL)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
Модифицированная подынтегральная функция
L=x^2+y^2+Dy^2+lambda*y
Уравнение Эйлера для неё:
2*y+lambda-2*D2y=0
Решение уравнения Эйлера:
SolL =
-1/2*(lambda*exp(x)-2*C1*exp(x)^2-2*C2)/exp(x)

```

Вычисляем левую часть условия изопериметричности.

```
dydx = diff(SolL,x); % dy/dx
F1_y = subs ( F1, {y,Dy}, {SolL,dydx} );
intF1 = vpa(int(F1_y,x,x1,x2),14) % вычислили интеграл
intF1 =
2.3504023872876*C1-1.*lambda+2.3504023872876*C2
```

Формируем систему уравнений и решаем её. В данных примерах система получается линейной, и мы решаем её с помощью команды `solve`.

```
SolLleft = vpa ( subs(SolL,x,sym(x1)), 14 );
SolLright = vpa ( subs(SolL,x,sym(x2)), 14 );
LeftL = [ char(SolLleft) '=' char(sym(y1)) ];
RightL = [ char(SolLright) '=' char(sym(y2)) ];
intF1J1 = [ char(intF1) '=' char(sym(J1)) ];
ConL = solve(LeftL,RightL,intF1J1);
C1 = vpa(ConL.C1,14)
C2 = vpa(ConL.C2,14)
lambda = vpa(ConL.lambda,14) % множитель Лагранжа
Sol10 = vpa(eval(SolL),14); % аналитическое решение
fprintf('Уравнение экстремали:\nu=%s\n',char(Sol10))
C1 =
.89229998917453
C2 =
.46684092505488
lambda =
1.1945280494651
Уравнение экстремали:
y=(-.59726402473255*exp(x)+.89229998917453*exp(x)^2+.46684092505488)/exp(x)
```

Формируем решение, заполняем таблицу и строим график.

```
y10 = subs(Sol10,x,xpl); % вычислили функцию
plot(xpl,y1a,'--b', xpl,y10,'-r') % рисуем график
title('\bfExample 10')
xlabel('x')
ylabel('y(x)')
```

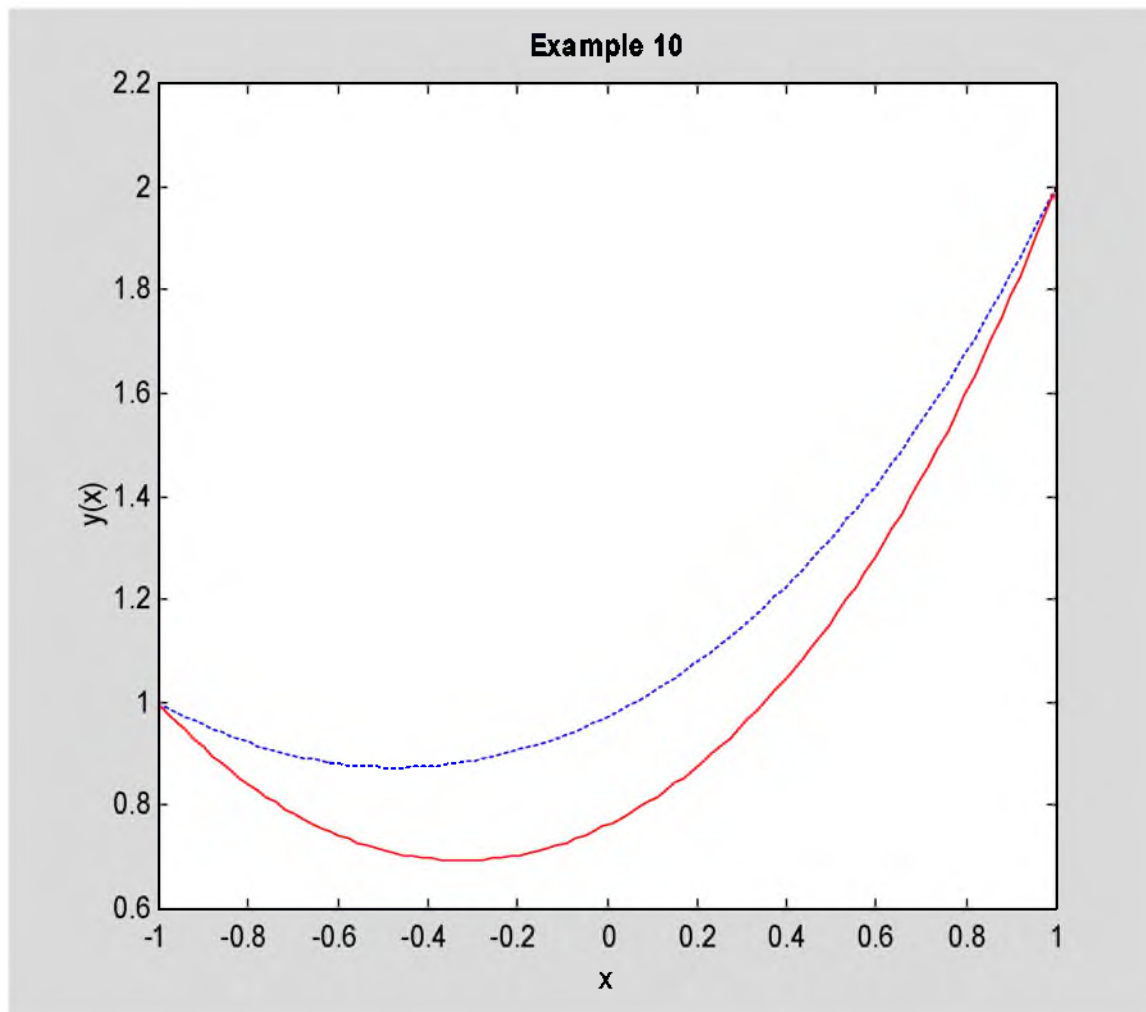


Рис. 10.1. Решение примера 10

Ответ. Экстремаль функционала (10.4) имеет вид

$$y(x) = 0.8923e^x + 0.46684e^{-x} - 0.597264. \quad (10.5)$$

Неопределённый множитель Лагранжа $\lambda = 1.194528$. График экстремали показан на рис.10.1 сплошной красной линией. Штриховая синяя линия – решение примера 1а.

10.3. Задание

Решить пример 1а при заданном изопериметрическом условии. Сравнить результат с решением примера 1а.

10.4. Варианты заданий

Вариант 1. $\int_{x_1}^{x_2} y dx = 1$

Вариант 2. $\int_{x_1}^{x_2} y dx = 4$

Вариант 3. $\int_{x_1}^{x_2} y dx = 0.5$

Вариант 4. $\int_{x_1}^{x_2} y dx = 2$

Вариант 5. $\int_{x_1}^{x_2} y dx = 1$

Вариант 6. $\int_{x_1}^{x_2} y dx = 1$

Вариант 7. $\int_{x_1}^{x_2} y dx = 3$

Вариант 8. $\int_{x_1}^{x_2} y dx = 3$

Вариант 9. $\int_{x_1}^{x_2} y dx = 2$

Вариант 10. $\int_{x_1}^{x_2} y dx = 2$

Вариант 11. $\int_{x_1}^{x_2} y dx = 2$

Вариант 13. $\int_{x_1}^{x_2} y dx = 4$

Вариант 15. $\int_{x_1}^{x_2} y dx = 1$

Вариант 17. $\int_{x_1}^{x_2} y dx = 6$

Вариант 19. $\int_{x_1}^{x_2} (x^2 + y) dx = 5$

Вариант 21. $\int_{x_1}^{x_2} (\sin x + y) dx = 5$

Вариант 23. $\int_{x_1}^{x_2} (y + y') dx = 3$

Вариант 25. $\int_{x_1}^{x_2} (y + y') dx = 16$

Вариант 27. $\int_{x_1}^{x_2} (x + y + y') dx = 8$

Вариант 29. $\int_{x_1}^{x_2} (\sin x + y + y') dx = 10$

Вариант 12. $\int_{x_1}^{x_2} y dx = 6$

Вариант 14. $\int_{x_1}^{x_2} y dx = 4$

Вариант 16. $\int_{x_1}^{x_2} (x + y) dx = 4$

Вариант 18. $\int_{x_1}^{x_2} (x^2 + y) dx = 4$

Вариант 20. $\int_{x_1}^{x_2} (\sin x + y) dx = 5$

Вариант 22. $\int_{x_1}^{x_2} (x^2 + y) dx = 8$

Вариант 24. $\int_{x_1}^{x_2} (y + y') dx = 3$

Вариант 26. $\int_{x_1}^{x_2} (x + y + y') dx = 8$

Вариант 28. $\int_{x_1}^{x_2} (x^2 + y + y') dx = 6$

Вариант 30. $\int_{x_1}^{x_2} (\sin x + y + y') dx = 8$

11. Численные методы 1. Метод начальных параметров

11.1. Краткие теоретические сведения

В методе начальных параметров краевую задачу для системы дифференциальных уравнений пытаются заменить задачей Коши. При этом предполагается, что в распоряжении расчётчика имеется программное обеспечение для решения начальной задачи для нормальной системы дифференциальных уравнений.

Рассмотрим метод начальных параметров для решения нормальной системы n дифференциальных уравнений 1-го порядка

$$\frac{dy}{dx} = F(x, y). \quad (11.1)$$

Если эта система дополнена начальными условиями $y(x_1) = y_1$, то её можно решить с помощью стандартного программного обеспечения, например, методов Рунге-Кутты. Рассмотрим случай, когда эта система дополнена граничными условиями: часть координат вектора y задана на левом конце x_1 , а часть – на правом x_2 . Пусть, для определённости, на правом конце заданы m первых координат вектора y , а на левом конце – $n-m$ последних. Этого всегда можно добиться перенумерацией переменных. Если бы нам удалось найти недостающие координаты y на левом конце, то можно было бы решить начальную задачу. Будем рассматривать m недостающих координат y в начальной точке x_1 как неизвестные t_1, t_2, \dots, t_m . Для их нахождения имеется m уравнений – граничные условия на правом конце x_2 . Таким образом, задача нахождения неизвестных начальных условий сводится к решению системы из m в общем случае нелинейных уравнений.

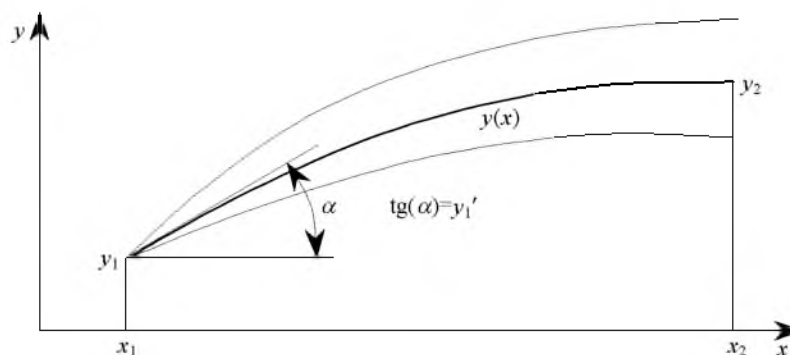


Рис. 11.1. Геометрическая интерпретация метода начальных параметров

Если $n=2$, а $m=1$, то геометрическая интерпретация метода начальных параметров показана на рис.11.1. Задавая различные значения для недостающего начального условия $y'(x_1)$, будем получать различные $y(x_2)$. Чтобы получить нужное значение y_2 , нужно угадать угол наклона экстремали в начальной точке. Это напоминает процесс стрельбы из артиллерийского орудия. Поэтому метод начальных параметров называется также методом стрельбы.

Если исходная система дифференциальных уравнений (11.1) линейная (даже с переменными коэффициентами), то вектор решений в любой точке $y(x_2)$ также будет линейно зависеть от начальных условий:

$$y(x_2) = Ay(x_1) + b. \quad (11.2)$$

Оставим в системе (11.2) только m первых строк, соответствующих граничным условиям при $x=x_2$, и только m первых столбцов, соответствующих неизвестным начальным параметрам:

$$\begin{cases} a_{11}t_1 + a_{12}t_2 + \dots + a_{1m}t_m + b_1 = y_1(x_2); \\ a_{21}t_1 + a_{22}t_2 + \dots + a_{2m}t_m + b_2 = y_2(x_2); \\ \dots \\ a_{m1}t_1 + a_{m2}t_2 + \dots + a_{mm}t_m + b_m = y_m(x_2) \end{cases} \quad (11.3)$$

Для нахождения компонентов матрицы A и координат вектора b нужно решить начальную задачу $m+1$ раз с такими вариантами неизвестных начальных условий:

$$\begin{aligned} t &= \{1, 0, \dots, 0\} \text{ – для формирования } \{a_{11}+b_1, a_{21}+b_2, \dots, a_{m1}+b_m\}; \\ t &= \{0, 1, \dots, 0\} \text{ – для формирования } \{a_{12}+b_1, a_{22}+b_2, \dots, a_{m2}+b_m\}; \\ &\dots \\ t &= \{0, 0, \dots, 1\} \text{ – для формирования } \{a_{1m}+b_1, a_{2m}+b_2, \dots, a_{mm}+b_m\}; \end{aligned}$$

а также

$$t = \{0, 0, \dots, 0\} \text{ – для формирования } \{b_1, b_2, \dots, b_m\}.$$

Поэтому метод начальных параметров называют также методом матричной прогонки.

Если $n=2$, а $m=1$, то геометрическая интерпретация метода начальных параметров показана на рис.11.1.

Задавая различные значения для недостающего начального условия $y'(x_1)$, будем получать различные $y(x_2)$. Чтобы получить нужное значение y_2 , нужно угадать угол наклона экстремали в начальной точке. Это напоминает процесс стрельбы из артиллерийского орудия. Поэтому метод начальных параметров называется также методом стрельбы.

11.2. Примеры выполнения заданий

11.2.а. Пример 11а

Решить методом начальных параметров пример 1а. Сравнить решение с аналитическим. Построить графики.

Используем программу для примера 1а для составления этой программы. Вводим исходные данные. Решаем пример 1а и заполняем таблицу.

```
clear all
format long
disp('Решаем пример 11а')
nnp = 10; % число точек интегрирования
syms x y Dy D2y % описали символические переменные
F = x^2+y^2+Dy^2; % подынтегральная функция
x1 = -1;
y1 = 1;
x2 = 1;
y2 = 2;
fprintf('Подынтегральная функция: F=%s\n',char(F))
fprintf('Граничные условия: y(%d)=%d; y'(%d)=%d\n',x1,y1,x2,y2)
dFdy = diff(F,y);
dFdyl = diff(F,Dy);
d_dFdyl_dx = diff(dFdyl,x);
d_dFdyl_dy = diff(dFdyl,y);
d_dFdyl_dyl = diff(dFdyl,Dy); % d(dF/dy')/dy'
dFyldx = d_dFdyl_dx + d_dFdyl_dy*Dy + d_dFdyl_dyl*D2y;
Euler = simple(dFdy-dFyldx);
degEuler = [ char(Euler) '=0' ]; % составили уравнение
Sol = dsolve(degEuler,'x'); % решаем уравнение Эйлера
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
SolLeft = subs(Sol,x,sym(x1));
SolRight = subs(Sol,x,sym(x2));
EqLeft = [char(SolLeft) '=' char(sym(y1))];
EqRight = [char(SolRight) '=' char(sym(y2))];
Con = solve(EqLeft,EqRight); % решаем систему
C1 = Con.C1;
C2 = Con.C2;
Sol1a = vpa(eval(Sol),14);
xpl = linspace(x1,x2);
yla=subs(Sol1a,x,xpl);
Решаем пример 11а
Подынтегральная функция: F=x^2+y^2+Dy^2
Граничные условия: y(-1)=1; y(1)=2
```

Для применения решателей систем дифференциальных уравнений приводим уравнение Эйлера 2-го порядка к системе 2-х дифференциальных уравнений 1-го порядка путём замены

$$\begin{cases} y_1 = y; \\ y_2 = y'. \end{cases} \quad (11.4)$$

Для этого решаем дифференциальное уравнение Эйлера относительно y'' и формируем правые части системы дифференциальных уравнений. Записываем их в файл.

```
f2 = solve(degEuler,D2y); % решаем относительно y''
f2 = subs ( f2, {y,Dy}, {sym('y(1)'),sym('y(2)')} );
rp{1} = 'function dydx = MyRightPart(x,y)';
rp{2} = 'dydx=zeros(2,1)';
```

```

rp{3} = 'dydx(1)=y(2);';
rp{4} = [ 'dydx(2)= ' char(f2) ' '];
disp('Текст файла MyRightPart.m')
fprintf ( '%s\n', rp{:} );
fid = fopen ( 'C:\Iglin\Matlab\MyRightPart.m', 'w' );
fprintf ( fid, '%s\n', rp{:} );
fclose(fid); % закрываем файл
Т е к с т   ф а й л а   MyRightPart.m
function dydx = MyRightPart(x,y)
dydx=zeros(2,1);
dydx(1)=y(2);
dydx(2)=y(1);

```

Мы сформировали систему дифференциальных уравнений

$$\begin{cases} y_1' = y_2; & y_1(x_1) = 1; \\ y_2' = y_1; & y_1(x_2) = 2. \end{cases} \quad (11.5)$$

где в нашем случае $x_1 = -1$; $x_2 = 1$. Если бы было известно начальное условие $y_2(x_1)$, то эту систему можно было бы решить с помощью стандартных численных методов. Обозначим $y_2(x_1)$ как неизвестную величину: $t = y_2(x_1)$. Присвоив ей какое-либо пробное значение, можно решить систему дифференциальных уравнений и найти функцию $f = y_1(x_2) - 2$. Очевидно, f можно рассматривать как функцию от t . То есть нужно решить уравнение $f(t) = 0$. В нашем случае, когда исходная система дифференциальных уравнений является **линейной**, уравнение относительно t также будет **линейным**. То есть функция $f(t)$ имеет структуру $f(t) = at + b$. Чтобы построить эту функцию, нужно решить 2 начальные задачи для $t = 0$ и $t = 1$. Решаем эти задачи.

```

xr = linspace(x1,x2,nnp+1); % точки для численного решения
y0 = [y1,0]; % решаем СДУ при y'(x1)=0;
[xx,YY] = ode45('MyRightPart',xr,y0);
yend0 = YY(nnp+1,1)-y2
y0 = [y1,1]; % решаем СДУ при y'(x1)=1;
[xx,YY] = ode45('MyRightPart',xr,y0);
yend1 = YY(nnp+1,1)-y2
yend0 =
    1.76219612254200
yend1 =
    5.38905701685360

```

Система линейных уравнений (11.3) в данном случае состоит из одного уравнения. Для нахождения неизвестного $t = y_2(x_1)$ проводим линейную интерполяцию.

```

y0 = [y1,yend0/(yend0-yend1)]
y0 =
    1.000000000000000    -0.48587364497652

```

Решаем систему дифференциальных уравнений при найденных действительных начальных условиях. Строим график.

```

[xx,YY] = ode45('MyRightPart',xr,y0);
plot ( xpl,y1a,'--b', xr,YY(:,1),'-r' )
title ( '\bfExample 11a' ) % заголовок
xlabel( 'x' )
ylabel('y(x)') % метки осей

```

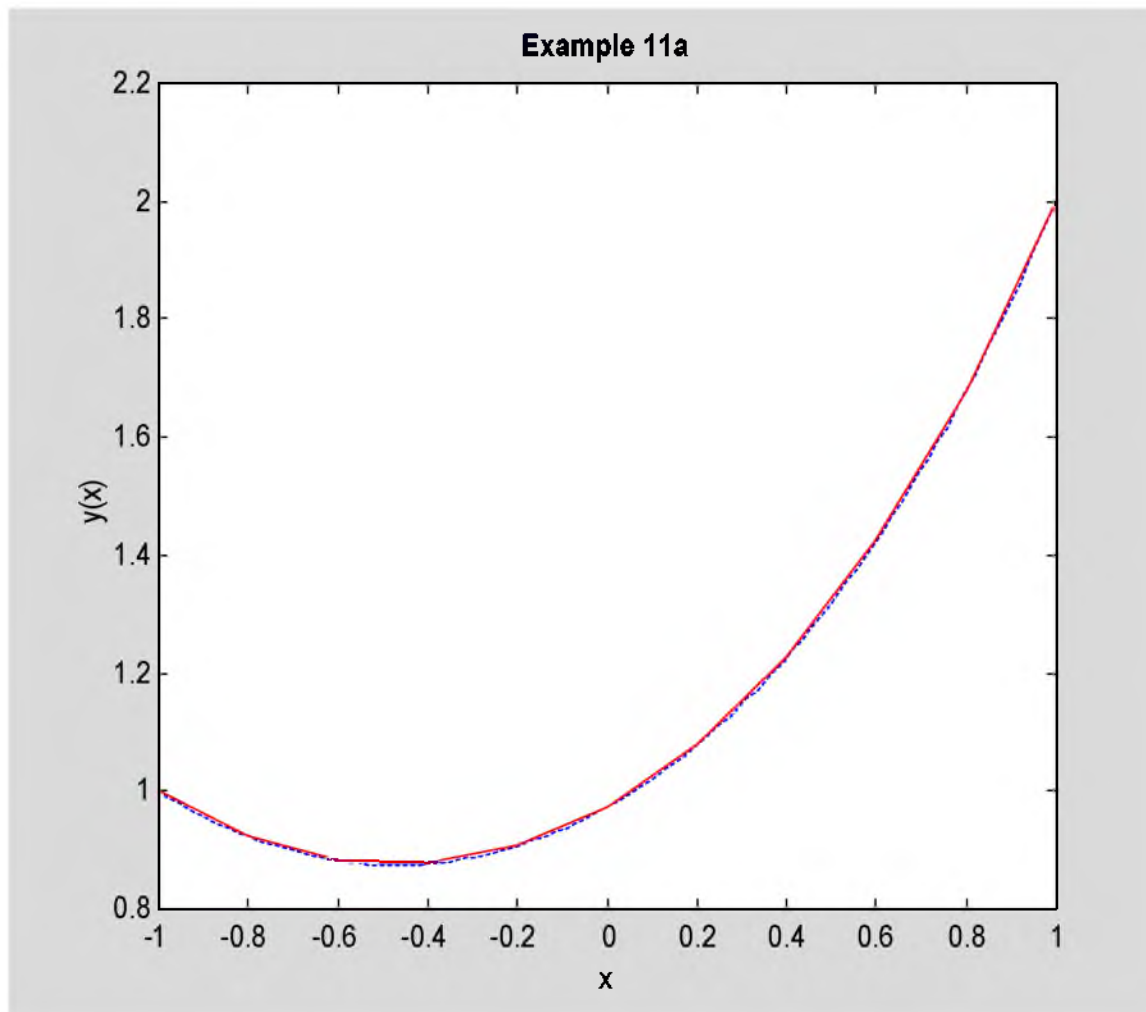


Рис. 11.2. Решение примера 11a

Ответ. График экстремали показан на рис.11.2 сплошной красной линией. Штриховая синяя линия – решение примера 1a. Видно, что в точках, где печатается численное решение, оно сливается с аналитическим. Неизвестное начальное условие: $y'(x_1) = -0.48587364$.

11.2.b. Пример 11b

Решить методом начальных параметров пример 2. Сравнить решение с аналитическим. Построить графики.

Программу для этого примера напишем на основе программы для примера 2 с использованием программы для примера 11a. Находим аналитическое решение примера 2. Заполняем таблицу.

```
clear all
format long
disp('Решаем пример 11b')
nnp = 10; % число точек интегрирования
syms x y z Dy D2y Dz D2z % описали переменные
F = Dy^2 + Dz^2 + 2*y*z; % подынтегральная функция
x1 = -2;
y1 = 1;
z1 = 0;
x2 = 2;
y2 = 0;
z2 = 2;
fprintf('Подынтегральная функция: F=%s\n', char(F))
fprintf('Граничные условия слева: y(%d)=%d; z(%d)=%d\n', x1, y1, x1, z1)
fprintf('Граничные условия справа: y(%d)=%d; z(%d)=%d\n', x2, y2, x2, z2)
dFdy = diff(F, y);
dFdyl = diff(F, Dy);
d_dFdyl_dx = diff(dFdyl, x); % d(dF/dy')/dx
```

```

d_dFdy1_dy = diff(dFdy1,y);
d_dFdy1_dy1 = diff(dFdy1,Dy);
d_dFdy1_dz = diff(dFdy1,z);
d_dFdy1_dz1 = diff(dFdy1,Dz);
dFy1dx = d_dFdy1_dx + d_dFdy1_dy*Dy + d_dFdy1_dy1*D2y + d_dFdy1_dz*Dz +
d_dFdy1_dz1*D2z;
dFdZ = diff(F,z);
dFdZ1 = diff(F,Dz);
d_dFdZ1_dx = diff(dFdZ1,x); % d(dF/dz')/dx
d_dFdZ1_dy = diff(dFdZ1,y);
d_dFdZ1_dy1 = diff(dFdZ1,Dy);
d_dFdZ1_dz = diff(dFdZ1,z);
d_dFdZ1_dz1 = diff(dFdZ1,Dz);
dFz1dx = d_dFdZ1_dx + d_dFdZ1_dy*Dy + d_dFdZ1_dy1*D2y + d_dFdZ1_dz*Dz +
d_dFdZ1_dz1*D2z;
EulerY = simple(dFdy-dFy1dx);
EulerZ = simple(dFdZ-dFz1dx);
deqEulerY = [char(EulerY) '=0']; % уравнение Y
deqEulerZ = [char(EulerZ) '=0']; % уравнение Z
Sol = dsolve(deqEulerY,deqEulerZ,'x'); % решаем
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
SolLeftY = subs(Sol.y,x,sym(x1)); % x1 в y
SolLeftZ = subs(Sol.z,x,sym(x1)); % x1 в z
SolRightY = subs(Sol.y,x,sym(x2)); % x2 в y
SolRightZ = subs(Sol.z,x,sym(x2)); % x2 в z
EqLeftY = [char(vpa(SolLeftY,14)) '=' char(sym(y1))];
EqLeftZ = [char(vpa(SolLeftZ,14)) '=' char(sym(z1))];
EqRightY = [char(vpa(SolRightY,14)) '=' char(sym(y2))];
EqRightZ = [char(vpa(SolRightZ,14)) '=' char(sym(z2))];
Con = solve(EqLeftY,EqLeftZ,EqRightY,EqRightZ);
C1 = Con.C1;
C2 = Con.C2;
C3 = Con.C3;
C4 = Con.C4;
Sol2Y = vpa(eval(Sol.y),14);
Sol2Z = vpa(eval(Sol.z),14);
xpl = linspace(x1,x2); % массив абсцисс
y2a = subs(Sol2Y,x,xpl);
z2a = subs(Sol2Z,x,xpl);
Решаем пример 11b
Подынтегральная функция: F=Dy^2+Dz^2+2*y*z
Граничные условия слева: y(-2)=1; z(-2)=0
Граничные условия справа: y(2)=0; z(2)=2

```

Сведём систему 2-х дифференциальных уравнений Эйлера 2-го порядка к системе 4-х нормальных дифференциальных уравнений 1-го порядка вида (11.1). Решим уравнения Эйлера относительно y'' , z'' . Подставим в оба уравнения y , z , y' , z' . Сформируем правые части для системы дифференциальных уравнений и запишем их в файл.

```

f2yz = solve(deqEulerY,deqEulerZ,D2y,D2z);
f2=subs(f2yz,{D2y,{y,Dy,z,Dz},{sym('y(1)'),sym('y(2)'),sym('y(3)'),sym('y(4)')}});
f4=subs(f2yz,{D2z,{y,Dy,z,Dz},{sym('y(1)'),sym('y(2)'),sym('y(3)'),sym('y(4)')}});
rp{1} = 'function dydx = MyRightPart(x,y)';
rp{2} = 'dydx=zeros(4,1)';
rp{3} = 'dydx(1)=y(2)';
rp{4} = [ 'dydx(2)= ' char(f2) ' '];
rp{5} = 'dydx(3)=y(4)';
rp{6} = [ 'dydx(4)= ' char(f4) ' '];
disp('Текст файла MyRightPart.m')
fprintf('%s\n',rp{:});
fid = fopen('C:\Iglin\Matlab\MyRightPart.m','w');

```

```
fprintf(fid, '%s\n', rp{:});
fclose(fid); % закрываем файл
Текст файла MyRightPart.m
function dydx = MyRightPart(x,y)
dydx=zeros(4,1);
dydx(1)=y(2);
dydx(2)=y(3);
dydx(3)=y(4);
dydx(4)=y(1);
```

Применим метод начальных параметров для решения задачи. Неизвестные у нас обозначены

$$y_1 = y; \quad y_2 = y'; \quad y_3 = z; \quad y_4 = z'. \quad (11.6)$$

В начальной точке x_1 неизвестны $y_2(x_1)=t_1$ и $y_4(x_1)=t_2$. Найдём их из решения системы 2-х уравнений

$$\begin{cases} f_1(t_1, t_2) = y(x_2) - y_2 = 0; \\ f_2(t_1, t_2) = z(x_2) - z_2 = 0. \end{cases} \quad (11.7)$$

Система дифференциальных уравнений Эйлера у нас является линейной, поэтому и система уравнений (11.7) также будет линейной.

$$\begin{cases} f_1(t_1, t_2) = a_{11}t_1 + a_{12}t_2 + b_1 = 0; \\ f_2(t_1, t_2) = a_{21}t_1 + a_{22}t_2 + b_2 = 0. \end{cases} \quad (11.8)$$

Найдём коэффициенты этой системы и правые части.

```
xr = linspace(x1,x2,nnp+1); % точки для численного решения
A = zeros(2,2); % матрица системы уравнений
b = zeros(2,1); % вектор правых частей
y0 = [y1;0;z1;0]; % решаем СДУ при y'(x1)=0; z'(x1)=0;
[xx,YY] = ode45('MyRightPart',xr,y0);
b=YY(nnp+1,[1 3])' - [y2;z2]
y0 = [y1;1;z1;0]; % решаем СДУ при y'(x1)=1; z'(x1)=0;
[xx,YY] = ode45('MyRightPart',xr,y0);
A(:,1) = YY(nnp+1,[1 3])' - [y2;z2] - b;
y0 = [y1;0;z1;1]; % решаем СДУ при y'(x1)=0; z'(x1)=1;
[xx,YY] = ode45('MyRightPart',xr,y0);
A(:,2) = YY(nnp+1,[1 3])' - [y2;z2] - b
b =
13.32736606398895
11.98099902570117
A =
13.26662430099436 14.02342479230629
14.02342479230629 13.26662430099436
```

Решаем систему линейных уравнений (11.8), находим недостающие начальные условия. Для этих начальных условий решаем систему дифференциальных уравнений вида (11.1) и строим график полученного решения.

```
yz0 = -A\b; % нашли начальные условия
y0 = [y1;yz0(1);z1;yz0(2)] % истинные начальные условия
[xx,YY] = ode45('MyRightPart',xr,y0); % решаем
plot3(xp1,y2a,z2a,'--b',xr,YY(:,1),YY(:,3),'-r')
title('Example 11b') % заголовок
xlabel('x')
ylabel('y(x)')
zlabel('z(x)')
view(205,30)
grid on
box on
y0 =
1.000000000000000
0.42582034231673
0
-1.35320471612876
```

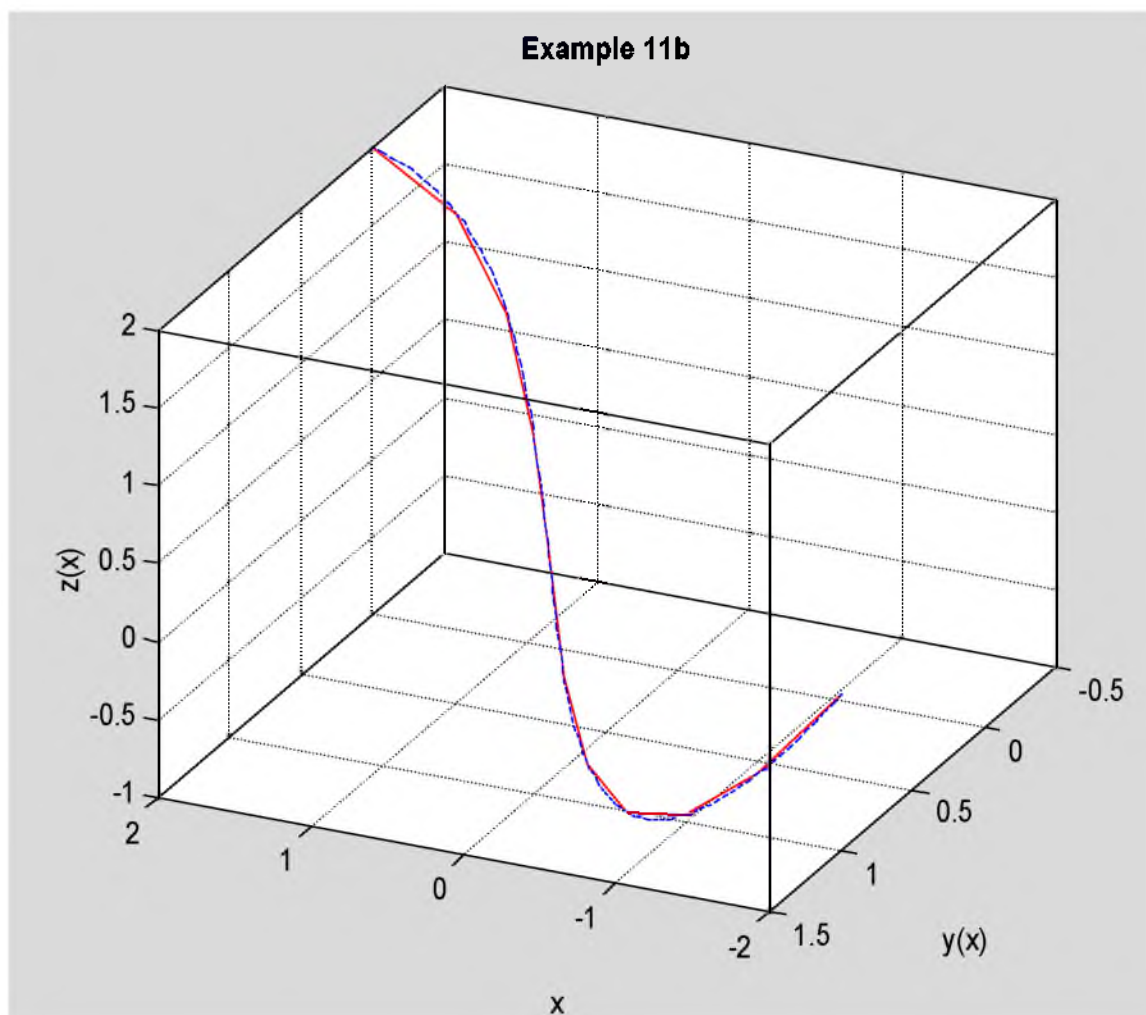



Рис. 11.3. Решение примера 11b

Ответ. График экстремали показан на рис.11.3 сплошной красной линией. Он практически сливается с решением примера 2, которое показано штриховой синей линией. Неизвестные начальные условия: $y'(x_1)=0.42582$; $z'(x_1)=-1.35320$.

11.2.с. Пример 11с

Решить методом начальных параметров пример 3. Сравнить решение с аналитическим. Построить графики.

Уравнение Эйлера (3.6) является уравнением 4-го порядка с 2-мя граничными условиями на левом конце и 2-мя на правом. Сведём его к нормальной системе 4-х уравнений 1-го порядка заменой

$$\begin{cases} y_1 = y; & y_1(x_1) = y_{10}; \\ y_2 = y'; & y_2(x_1) = y'_{10}; \\ y_3 = y''; & y_1(x_2) = y_{20}; \\ y_4 = y'''; & y_2(x_2) = y'_{20}. \end{cases} \quad (11.9)$$

Неизвестные начальные условия $t_1=y_3(x_1)$ и $t_2=y_4(x_1)$ найдём из решения системы уравнений

$$\begin{cases} f_1(t_1, t_2) = y_1(x_2) - y_{20} = 0; \\ f_2(t_1, t_2) = y_2(x_2) - y'_{20} = 0. \end{cases} \quad (11.10)$$

Так как уравнение (3.6) линейное, то и система (11.10) будет линейной. Решая её, найдём начальные условия, а затем и решение уравнения Эйлера.

Для составления программы используем программы для примеров 3 и 11b. Вначале повторяем решение примера 3.

```
clear all
format long
disp('Решаем пример 11с')
nnp = 10;
syms x y Dy D2y D3y D4y % описали переменные
```

```

F = D2y^2-2*Dy^2+4*y*Dy+y^2-2*y*sin(x);
x1 = -1;
y1 = 1;
Dy1 = -1;
x2 = 1;
y2 = 2;
Dy2 = 1;
fprintf('Подынтегральная функция: F=%s\n',char(F))
fprintf('Граничные условия слева: y(%d)=%d;   y'(%d)=%d\n',x1,y1,x1,Dy1)
fprintf('Граничные условия справа: y(%d)=%d;   y'(%d)=%d\n',x2,y2,x2,Dy2)
dFdy = diff(F,y);
dFdy1 = diff(F,Dy);
dFdy2 = diff(F,D2y); % dF/dy'
d_dFdy1_dx = diff(dFdy1,x); % d(dF/dy')/dx
d_dFdy1_dy = diff(dFdy1,y); % d(dF/dy')/dy
d_dFdy1_dy1 = diff(dFdy1,Dy); % d(dF/dy')/dy'
d_dFdy1_dy2 = diff(dFdy1,D2y); % d(dF/dy')/dy''
dFy1dx = d_dFdy1_dx + d_dFdy1_dy * Dy + d_dFdy1_dy1 * D2y + d_dFdy1_dy2*D3y;
d_dFdy2_dx = diff(dFdy2,x); % d(dF/dy'')/dx
d_dFdy2_dy = diff(dFdy2,y); % d(dF/dy'')/dy
d_dFdy2_dy1 = diff(dFdy2,Dy); % d(dF/dy'')/dy'
d_dFdy2_dy2 = diff(dFdy2,D2y); % d(dF/dy'')/dy''
dFy2dx = d_dFdy2_dx + d_dFdy2_dy * Dy + d_dFdy2_dy1 * D2y + d_dFdy2_dy2 * D3y;
d_dFdy2dx_dx = diff(dFy2dx,x); % d((dFy'')/dx)/dx
d_dFdy2dx_dy = diff(dFy2dx,y); % d((dFy'')/dx)/dy
d_dFdy2dx_dy1 = diff(dFy2dx,Dy); % d((dFy'')/dx)/dy'
d_dFdy2dx_dy2 = diff(dFy2dx,D2y); % d((dFy'')/dx)/dy''
d_dFdy2dx_dy3 = diff(dFy2dx,D3y); % d((dFy'')/dx)/dy'''
d2Fy2dx2 = d_dFdy2dx_dx + d_dFdy2dx_dy * Dy + d_dFdy2dx_dy1 * D2y +
d_dFdy2dx_dy2 * D3y + d_dFdy2dx_dy3 * D4y;
Euler = simple(dFdy-dFy1dx+d2Fy2dx2);
deqEuler = [char(Euler) '=0']; % составили уравнение
Sol = dsolve ( deqEuler, 'x' );
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
dydx = diff(Sol,x); % нашли производную
slY = subs(Sol,x,sym(x1));
slDY = subs(dydx,x,sym(x1));
srY = subs(Sol,x,sym(x2));
srDY = subs(dydx,x,sym(x2));
elY = [char(vpa(slY,14)) '=' char(sym(y1))];
elDY = [char(vpa(slDY,14)) '=' char(sym(Dy1))];
erY = [char(vpa(srY,14)) '=' char(sym(y2))];
erDY = [char(vpa(srDY,14)) '=' char(sym(Dy2))];
Con = solve(elY,elDY,erY,erDY);
C1 = Con.C1;
C2 = Con.C2;
C3 = Con.C3;
C4=Con.C4;
Sol3 = vpa(eval(Sol),14); % подставляем C1-C4;
xpl = linspace(x1,x2);
y3 = subs(Sol3,x,xpl);
Решаем пример 11с
Подынтегральная функция: F=D2y^2-2*Dy^2+4*y*Dy+y^2-2*y*sin(x)
Граничные условия слева: y(-1)=1;   y'(-1)=-1
Граничные условия справа: y(1)=2;   y'(1)=1

```

Разрешаем уравнение Эйлера относительно y^{IV} и подставляем в него обозначения (11.9). Формируем правые части для системы дифференциальных уравнений, к которой сводится уравнение Эйлера. Записываем их в файл.

```
f4 = solve(deqEuler,D4y); % находим D4y
```

```

f4 = subs(f4,{y,Dy,D2y,D3y},{sym('y(1)'),sym('y(2)'),sym('y(3)'),sym('y(4)')});
rp{1} = 'function dydx = MyRightPart(x,y)';
rp{2} = 'dydx=zeros(4,1)';
rp{3} = 'dydx(1)=y(2)';
rp{4} = 'dydx(2)=y(3)';
rp{5} = 'dydx(3)=y(4)';
rp{6} = [ 'dydx(4)= ' char(f4) ' '];
disp('Текст файла MyRightPart.m')
fprintf('%s\n',rp{:});
fid = fopen('C:\Iglin\Matlab\MyRightPart.m','w');
fprintf(fid,'%s\n',rp{:});
fclose(fid); % закрываем файл
Т е к с т   ф а й л а   MyRightPart.m
function dydx = MyRightPart(x,y)
dydx=zeros(4,1);
dydx(1)=y(2);
dydx(2)=y(3);
dydx(3)=y(4);
dydx(4)=-y(1)+sin(x)-2*y(3);

```

Формируем коэффициенты и свободные члены системы линейных уравнений (11.3, 11.10). Для этого 3 раза решаем начальную задачу при значениях неизвестных начальных параметров: $\{t_1, t_2\}=\{0,0\}$; $\{t_1, t_2\}=\{1,0\}$; и $\{t_1, t_2\}=\{0,1\}$. Решаем систему (11.10) – находим неизвестные начальные параметры. Решаем начальную задачу при этих значениях начальных параметров. Рисуем график.

```

xr = linspace(x1,x2,nnr+1); % точки для численного решения
A = zeros(2,2);
b = zeros(2,1);
y0 = [y1;Dy1;0;0]; % начальные условия (0,0)
[xx,YY] = ode45('MyRightPart',xr,y0); % решаем СДУ
b = YY(nnr+1,[1 2])' - [y2;Dy2] % правые части
y0 = [y1;Dy1;1;0]; % начальные условия (1,0)
[xx,YY] = ode45('MyRightPart',xr,y0); % решаем СДУ
A(:,1) = YY(nnr+1,[1 2])' - [y2;Dy2] - b; % 1-й столбец матрицы A
y0 = [y1;Dy1;0;1]; % начальные условия (0,1)
[xx,YY] = ode45('MyRightPart',xr,y0); % решаем СДУ
A(:,2) = YY(nnr+1,[1 2])' - [y2;Dy2] - b % 2-й столбец матрицы A
yz0 = -A\b; % нашли начальные условия
y0 = [y1;Dy1;yz0] % истинные начальные условия
[xx,YY] = ode45('MyRightPart',xr,y0); % решаем
plot(xp1,y3,'--b',xr,YY(:,1),'-r')
title('Example 11c') % заголовок
xlabel('x')
ylabel('y(x)')
b =
-3.54609584741614
-2.69446117363805
A =
0.90929722937057 0.87079594142352
0.03850131148804 0.90929725087375
y0 =
1.000000000000000
-1.000000000000000
1.10693966258040
2.91636485466376

```

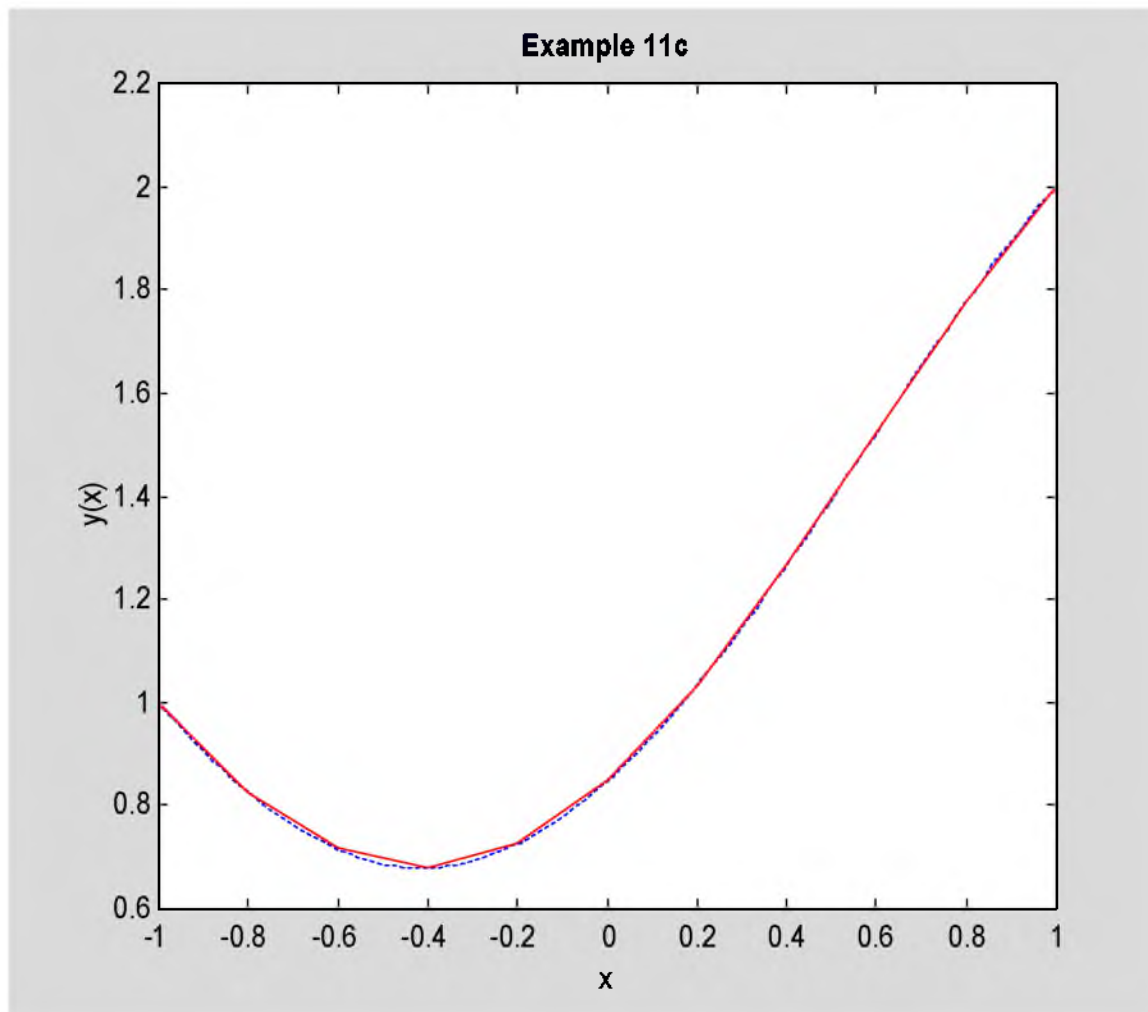


Рис. 11.4. Решение примера 11с

Ответ. График экстремали показан на рис.11.4 сплошной красной линией. Он практически сливается с решением примера 3, которое показано штриховой синей линией. Неизвестные начальные условия: $y''(x_1)=1.10694$; $y'''(x_1)=2.91636$.

11.3. Задание

Решить примеры 1а, 2 и 3 методом начальных параметров. Сравнить решения с аналитическими.

12. Численные методы 2. Метод конечных разностей

12.1. Краткие теоретические сведения

В методе конечных разностей (МКР) экстремаль аппроксимируется кусочно-линейной функцией. Варьируются ординаты $y(x)$ в заданных точках, и функционал становится функцией этих неизвестных ординат y_1, y_2, \dots .

Рассмотрим применение МКР к вариационной задаче для функционала (1.1, 1.2). Разобьём интервал $[x_1, x_2]$ на n участков одинаковой длины $\Delta x = (x_2 - x_1)/n$. Будем обозначать точки деления x_k . То есть граничные точки в этом примере в дальнейшем будем обозначать x_0 и x_n . Функционал J будет равен сумме функционалов на отдельных участках J_k . Так как участки малые, применим для вычисления интегралов формулу касательных: каждый из J_k равен подинтегральной функции, вычисленной в средней точке, и умноженной на ширину интервала:

$$J_k = \int_{x_{k-1}}^{x_k} F(x, y, y') dx = F\left(\frac{x_{k-1} + x_k}{2}, \frac{y_{k-1} + y_k}{2}, \frac{y_k - y_{k-1}}{\Delta x}\right) \Delta x. \quad (12.1)$$

Функционал (1.1) будет функцией неизвестных ординат y_1, y_2, \dots, y_{n-1} :

$$J(y_1, y_2, \dots, y_{n-1}) = \sum_{k=1}^n J_k = \Delta x \sum_{k=1}^n F\left(\frac{x_{k-1} + x_k}{2}, \frac{y_{k-1} + y_k}{2}, \frac{y_k - y_{k-1}}{\Delta x}\right). \quad (12.2)$$

Значения y_0 и y_n известны – это заданные граничные условия. Для нахождения экстремума продифференцируем J по переменным y_1, y_2, \dots, y_{n-1} , приравняем производные нулю и решим полученную систему уравнений. При вычислении $\partial J / \partial y_k$ учтём, что от y_k зависят только 2 слагаемых формулы (12.2): J_k и J_{k+1} .

$$\begin{aligned} \frac{\partial J}{\partial y_k} &= \frac{\partial J_k}{\partial y_k} + \frac{\partial J_{k+1}}{\partial y_k} = \\ &= F_y\left(\frac{x_{k-1} + x_k}{2}, \frac{y_{k-1} + y_k}{2}, \frac{y_k - y_{k-1}}{\Delta x}\right) \frac{\Delta x}{2} + F_y\left(\frac{x_k + x_{k+1}}{2}, \frac{y_k + y_{k+1}}{2}, \frac{y_{k+1} - y_k}{\Delta x}\right) \frac{\Delta x}{2} + \\ &+ F_y\left(\frac{x_k + x_{k+1}}{2}, \frac{y_k + y_{k+1}}{2}, \frac{y_{k+1} - y_k}{\Delta x}\right) \frac{\Delta x}{2} - F_y\left(\frac{x_{k-1} + x_k}{2}, \frac{y_{k-1} + y_k}{2}, \frac{y_k - y_{k-1}}{\Delta x}\right) \frac{\Delta x}{2}. \end{aligned} \quad (12.3)$$

После сокращения на Δx получим систему линейных алгебраических уравнений, которая своей структурой напоминает уравнение Эйлера (1.5):

$$\begin{cases} \left[\frac{1}{2} \left(F_y\left(\frac{x_{k-1} + x_k}{2}, \frac{y_{k-1} + y_k}{2}, \frac{y_k - y_{k-1}}{\Delta x}\right) + F_y\left(\frac{x_k + x_{k+1}}{2}, \frac{y_k + y_{k+1}}{2}, \frac{y_{k+1} - y_k}{\Delta x}\right) \right) - \right. \\ \left. - \frac{1}{\Delta x} \left(F_y\left(\frac{x_k + x_{k+1}}{2}, \frac{y_k + y_{k+1}}{2}, \frac{y_{k+1} - y_k}{\Delta x}\right) - F_y\left(\frac{x_{k-1} + x_k}{2}, \frac{y_{k-1} + y_k}{2}, \frac{y_k - y_{k-1}}{\Delta x}\right) \right) \right] = 0, \\ k \in [1, n-1] \end{cases} \quad (12.4)$$

Вместо F_y , которая фигурирует в уравнении Эйлера (1.5), в уравнении (12.4) – полусумма значений F_y на интервалах, примыкающих к точке x_k . Вместо dF_y/dx – отношение разности величин F_y справа и слева от точки x_k к ширине интервала Δx . Таким образом, (12.4) можно рассматривать как конечноразностный аналог уравнения Эйлера (1.5).

Аналогично выглядит конечноразностная аппроксимация уравнения Эйлера-Остроградского (4.6). Разобьём область D на n участков вдоль оси Ox : $\Delta x = (x_2 - x_1)/n$ и на m участков вдоль оси Oy : $\Delta y = (y_2 - y_1)/m$. Будем обозначать узлы сетки x_i, y_k . Граничные точки в этом примере будут x_0, x_m, y_0, y_m . МКР-аппроксимация уравнения Эйлера-Остроградского (4.6) имеет вид

$$\begin{cases} \frac{1}{4} (F_z(i, k) + F_z(i+1, k) + F_z(i, k+1) + F_z(i+1, k+1)) - \\ - \frac{1}{\Delta x} (F_p(i+1, k) - F_p(i, k)) - \frac{1}{\Delta y} (F_q(i, k+1) - F_q(i, k)) = 0, \\ i \in [1, n-1], \quad k \in [1, m-1] \end{cases} \quad (12.5)$$

где аргументы (i, k) для любой функции означают

$$F(i, k) = F\left(\frac{x_{ik} + x_{i-1, k} + x_{i, k-1} + x_{i-1, k-1}}{4}, \frac{y_{ik} + y_{i-1, k} + y_{i, k-1} + y_{i-1, k-1}}{4}, \frac{z_{ik} + z_{i-1, k} + z_{i, k-1} + z_{i-1, k-1}}{4}, \frac{z_{ik} - z_{i-1, k}}{\Delta x}, \frac{z_{ik} - z_{i, k-1}}{\Delta y}\right). \quad (12.6)$$

12.2. Примеры выполнения заданий

12.2.а. Пример 12а

Решить методом конечных разностей пример 1а. Сравнить решение с аналитическим. Построить графики.

На основе программы для примера 1а составим программу для данного примера. Вводим исходные данные. Решаем пример 1а.

```
clear all
format long
disp('Решаем пример 12а')
nf = 10; % число интервалов разбиения
syms x y Dy D2y % описали символические переменные
F = x^2+y^2+Dy^2; % подынтегральная функция
x1 = -1;
y1 = 1;
x2 = 1;
y2 = 2;
fprintf('Подынтегральная функция: F=%s\n',char(F))
fprintf('Граничные условия: y(%d)=%d; y(%d)=%d\n',x1,y1,x2,y2)
dFdy = diff(F,y);
dFdy1 = diff(F,Dy);
d_dFdy1_dx = diff(dFdy1,x);
d_dFdy1_dy = diff(dFdy1,y);
d_dFdy1_dy1 = diff(dFdy1,Dy); % d(dF/dy')/dy'
dFyldx = d_dFdy1_dx + d_dFdy1_dy*Dy + d_dFdy1_dy1*D2y;
Euler = simple(dFdy-dFyldx);
degEuler = [ char(Euler) ' = 0 ' ]; % составили уравнение
Sol = dsolve(degEuler,'x'); % решаем уравнение Эйлера
if length(Sol)~=1 % решений нет или более одного
    error('Нет решений или более одного решения!');
end
SolLeft = subs(Sol,x,sym(x1));
SolRight = subs(Sol,x,sym(x2));
EqLeft = [char(SolLeft) '=' char(sym(y1))];
EqRight = [char(SolRight) '=' char(sym(y2))];
Con = solve(EqLeft,EqRight); % решаем систему
C1 = Con.C1;
C2 = Con.C2;
Sol1a = vpa(eval(Sol),14);
xpl = linspace(x1,x2);
yla = subs(Sol1a,x,xpl);
Решаем пример 12а
Подынтегральная функция: F=x^2+y^2+Dy^2
Граничные условия: y(-1)=1; y(1)=2
```

Формируем и печатаем конечноразностное уравнение (12.4).

```
syms xim1 xi xip1 yim1 yi yip1
dx = sym((x2-x1)/nf)
dFdyP = subs ( dFdy, {x,y,Dy}, {(xim1+xi)/2, (yim1+yi)/2, (yi-yim1)/dx});
dFdyN = subs ( dFdy, {x,y,Dy}, {(xi+xipl)/2, (yi+yip1)/2, (yip1-yi)/dx});
dFdy1P = subs ( dFdy1, {x,y,Dy}, {(xim1+xi)/2, (yim1+yi)/2, (yi-yim1)/dx});
dFdy1N = subs ( dFdy1, {x,y,Dy}, {(xi+xipl)/2, (yi+yip1)/2, (yip1-yi)/dx});
EqF = simple((dFdyN+dFdyP)/2-(dFdy1N-dFdy1P)/dx)
dx =
1/5
EqF =
101*yi-99/2*yip1-99/2*yim1
```

Это уравнение зависит от x_{i-1} , x_i , x_{i+1} , y_{i-1} , y_i , y_{i+1} . Сформируем из него 3-диагональную систему уравнений. Коэффициенты при неизвестных этой системы – это коэффициенты при y_{i-1} , y_i , y_{i+1} ; правые части – это свободные члены. Заполним матрицу коэффициентов и вектор правых частей.

```
neq = nf-1 % количество уравнений
```



```

edx = eval(dx) % шаг сетки
xiC=[x1+edx:(x2-x1-2*edx)/neq:x2-edx]'; % текущая точка
xiP=xiC-edx;
xiN=xiC+edx; % предыдущая и последующая точки
e0 = subs(EqF,{xim1,xi,xip1,yim1,yi,yip1},{xiP,xiC,xiN,0,0,0}); % своб. члены
e0 = ones(neq,1).*e0; % если скаляр, сделали вектором
eiP = subs ( EqF, {xim1,xi,xip1,yim1,yi,yip1}, {xiP,xiC,xiN,1,0,0} ) - e0;
eiC = subs ( EqF, {xim1,xi,xip1,yim1,yi,yip1}, {xiP,xiC,xiN,0,1,0} ) - e0;
eiN = subs ( EqF, {xim1,xi,xip1,yim1,yi,yip1}, {xiP,xiC,xiN,0,0,1} ) - e0;
b = -eval(e0); % правые части
A = diag(eval(eiC)) + diag(eval(eiN(1:neq-1)),1) + diag(eval(eiP(2:neq)),-1);
b(1) = b(1)-eiP(1)*y1; % учли граничные условия
b(neq) = b(neq)-eiN(1)*y2;
neq =
    9
edx =
    0.200000000000000

```

Решаем полученную систему и строим графики.

```

y = A\b; % решаем систему
x12a = linspace(x1,x2,nf+1); % абсциссы
y12a = zeros(nf+1,1);
y12a(2:nf) = y; % решение
y12a(1) = y1;
y12a(nf+1) = y2; % граничные условия
plot(xp1,y1a,'--b',x12a,y12a,'-r')
title ( '\bfExample 12a' ) % заголовок
xlabel('x')
ylabel('y(x)') % ось OY

```

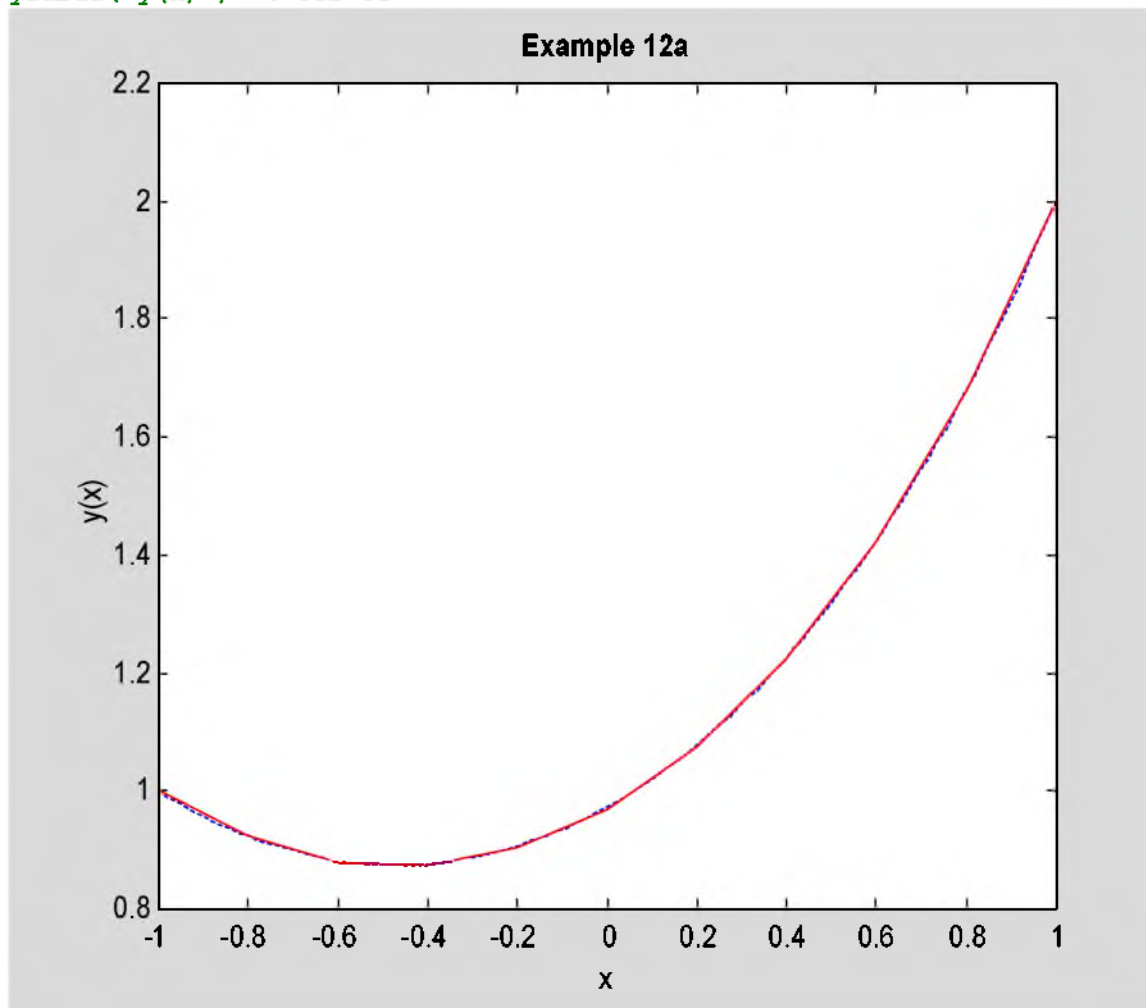


Рис. 12.1. Решение примера 12а

Ответ. Для 10 интервалов разбиения система уравнений МКР имеет вид

$$\begin{cases} -49.5y_{i-1} + 101y_i - 49.5y_{i+1} = 0, \\ i \in [1, 9] \end{cases} \quad (12.7)$$

при этом в 1-е уравнение входит $y_0=1$, а в 9-е уравнение $y_{10}=2$. График экстремали показан на рис. 12.1 сплошной красной линией, он практически сливается с теоретическим решением – штриховой синей линией.

12.2.b. Пример 12b

Решить МКР пример 4. Сравнить с решением МКЭ. Построить график.

Программу для этого примера напишем на основе программы для примера 4 с использованием программы для примера 12а. Введём исходные данные и вычислим частные производные F_z , F_p , F_q .

```
clear all
format long
disp('Решаем пример 12b')
n = 20; % интервалов разбиения по меньшей стороне
syms x y z Dz x Dz y % описали переменные
F = Dz x^2 - 2*Dz y^2 + 2*y*z*(sin(pi*x)+x/5); % подынтегральная функция
zc = x/10 + y^2/50; % функция граничных условий
x1 = 0;
x2 = 1;
y1 = 0;
y2 = 2;
fprintf('Подынтегральная функция: F=%s\n', char(F))
fprintf('Граничное условие на контуре: z=%s\n', char(zc))
fprintf('Область: %d<=x<=%d; %d<=y<=%d\n', x1, x2, y1, y2)
dFd z = diff(F, z);
dFd p = diff(F, Dz x);
dFd q = diff(F, Dz y);
Решаем пример 12b
Подынтегральная функция: F=Dz x^2-2*Dz y^2+2*y*z*(sin(pi*x)+1/5*x)
Граничное условие на контуре: z=1/10*x+1/50*y^2
Область: 0<=x<=1; 0<=y<=2
```

Строим квадратную сетку разбиения области.

```
del = min(x2-x1, y2-y1)/n % шаг сетки
nx = round((x2-x1)/del)
ny = round((y2-y1)/del)
del =
0.0500000000000000
nx =
20
ny =
40
```

Описываем символические переменные: x, y, z в необходимых точках.

```
syms xim1 xi xip1 ykm1 yk ykp1 zim1km1 zim1k zim1kp1 zikm1 zik zikp1 zip1km1
zip1k zip1kp1
dx = sym((x2-x1)/nx)
dy = sym((y2-y1)/ny)
dx =
1/20
dy =
1/20
```

Формируем конечноразностное уравнение (12.5).

```
Zik = subs(dFd z, {x, y, z, Dz x, Dz y}, {(xim1+xi)/2, (ykm1+yk)/2,
(zik+zim1k+zikm1+zim1km1)/4, (zik-zim1k)/dx, (zik-zikm1)/dy});
```

```

Ziplk = subs ( dFdz, {x,y,z,Dzx,Dzy}, {(xi+xipl)/2, (ykm1+yk)/2,
(ziplk+zik+ziplkm1+zikml)/4, (ziplk-zik)/dx, (ziplk-ziplkm1)/dy});
Zikpl = subs ( dFdz, {x,y,z,Dzx,Dzy}, {(xim1+xi)/2, (yk+ykp1)/2,
(zikpl+zimlkp1+zik+zimlk)/4, (zikpl-zimlkp1)/dx, (zikpl-zik)/dy});
Ziplkp1 = subs ( dFdz, {x,y,z,Dzx,Dzy}, {(xi+xipl)/2, (yk+ykp1)/2,
(ziplkp1+zikpl+zik+ziplk+zikml)/4, (ziplkp1-zikpl)/dx, (ziplkp1-ziplk)/dy});
Pik = subs ( dFdp, {x,y,z,Dzx,Dzy}, {(xim1+xi)/2, (ykm1+yk)/2,
(zik+zimlk+zikml+zimlkm1)/4, (zik-zimlk)/dx, (zik-zikml)/dy});
Piplk = subs ( dFdp, {x,y,z,Dzx,Dzy}, {(xi+xipl)/2, (ykm1+yk)/2,
(ziplk+zik+ziplkm1+zikml)/4, (ziplk-zik)/dx, (ziplk-ziplkm1)/dy});
Qik = subs ( dFdq, {x,y,z,Dzx,Dzy}, {(xim1+xi)/2, (ykm1+yk)/2,
(zik+zimlk+zikml+zimlkm1)/4, (zik-zimlk)/dx, (zik-zikml)/dy});
Qikpl = subs ( dFdq, {x,y,z,Dzx,Dzy}, {(xim1+xi)/2, (yk+ykp1)/2,
(zikpl+zimlkp1+zik+zimlk)/4, (zikpl-zimlkp1)/dx, (zikpl-zik)/dy});
EqF = (Zik+Ziplk+Zikpl+Ziplkp1)/4 - (Piplk-Pik)/dx - (Qikpl-Qik)/dy;
EqF = simple(EqF)
EqF =
1/4*(ykm1+yk)*(sin(pi*(1/2*xim1+1/2*xi))+1/10*xim1+1/10*xi)+1/4*(ykm1+yk)*(sin(p
i*(1/2*xi+1/2*xipl))+1/10*xi+1/10*xipl)+1/4*(yk+ykp1)*(sin(pi*(1/2*xim1+1/2*xi)
)+1/10*xim1+1/10*xi)+1/4*(yk+ykp1)*(sin(pi*(1/2*xi+1/2*xipl))+1/10*xi+1/10*xipl)-
800*ziplk-1600*zik-800*zimlk+1600*zikpl+1600*zikml

```

Записываем функцию **EqF** в файл.

```

f{1} = 'function y = MyEqF ( xy, zimlkm1, zikml, ziplkm1, zimlk, zik, ziplk,
zimlkp1, zikpl, ziplkp1 )';
f{2} = 'xim1=xy(1); xi=xy(2); xipl=xy(3); ykm1=xy(4); yk=xy(5); ykp1=xy(6);';
f{3} = ['y=' char(EqF) '']; % формируем функцию
disp('Текст файла MyEqF.m')
fprintf ( '%s\n', f{:} );
fid = fopen ( 'C:\Iglin\Matlab\MyEqF.m', 'w' );
fprintf ( fid, '%s\n', f{:} );
fclose(fid); % закрываем файл
Т е к с т   ф а й л а   М у Е q F . m
function y = MyEqF ( xy, zimlkm1, zikml, ziplkm1, zimlk, zik, ziplk, zimlkp1,
zikpl, ziplkp1 )
xim1=xy(1); xi=xy(2); xipl=xy(3); ykm1=xy(4); yk=xy(5); ykp1=xy(6);
y=1/4*(ykm1+yk)*(sin(pi*(1/2*xim1+1/2*xi))+1/10*xim1+1/10*xi)+1/4*(ykm1+yk)*(sin
(pi*(1/2*xi+1/2*xipl))+1/10*xi+1/10*xipl)+1/4*(yk+ykp1)*(sin(pi*(1/2*xim1+1/2*xi
))+1/10*xim1+1/10*xi)+1/4*(yk+ykp1)*(sin(pi*(1/2*xi+1/2*xipl))+1/10*xi+1/10*xipl
)-800*ziplk-1600*zik-800*zimlk+1600*zikpl+1600*zikml;

```

Формируем систему конечноразностных уравнений. Из-за большого размера системы уравнений применяем разрежённую матрицу. Нумеруем уравнения так, чтобы быстрее менялась координата y . Для каждого уравнения вычисляем входящие в него величины: правые части и коэффициенты при всех неизвестных. Заполняем матрицу коэффициентов и вектор правых частей. Если точка расположена на боковой стороне или в углу, учитываем граничное условие.

```

neq = (nx-1)*(ny-1) % число уравнений
A = sparse(neq,neq);
b = zeros(neq,1);
edx = eval(dx);
edy = eval(dy);
for i=1:nx-1,
    xi = x1+i*edx;
    xim1 = xi-edx;
    xipl = xi+edx;
    for k=1:ny-1,
        yk = y1+k*edy;
        ykm1 = yk-edy;
        ykp1 = yk+edy;
        numeq = (i-1)*(ny-1)+k; % номер уравнения
        xy = [xim1,xi,xipl,ykm1,yk,ykp1];

```

```

ei0 = MyEqF(xy,0,0,0,0,0,0,0,0,0);
eim1km1 = MyEqF(xy,1,0,0,0,0,0,0,0,0)-ei0;
eikm1 = MyEqF(xy,0,1,0,0,0,0,0,0,0)-ei0;
eiplkm1 = MyEqF(xy,0,0,1,0,0,0,0,0,0)-ei0;
eim1k = MyEqF(xy,0,0,0,1,0,0,0,0,0)-ei0;
eik = MyEqF(xy,0,0,0,0,1,0,0,0,0)-ei0;
eiplk = MyEqF(xy,0,0,0,0,0,1,0,0,0)-ei0;
eim1kp1 = MyEqF(xy,0,0,0,0,0,0,1,0,0)-ei0;
eikp1 = MyEqF(xy,0,0,0,0,0,0,0,1,0)-ei0;
eiplkp1 = MyEqF(xy,0,0,0,0,0,0,0,0,1)-ei0;
b(numeq) = -ei0; % правая часть
A(numeq,numeq) = eik; % диаг. элемент (i,k)
if k<ny-1 % точки на серединах сторон
    A(numeq,numeq+1) = eikp1; % (i,k+1)
else
    b(numeq) = b(numeq)-eikp1*subs(zc,{x,y},{xi,ykp1});
end
if k>1
    A(numeq,numeq-1) = eikm1; % (i,k-1)
else
    b(numeq) = b(numeq)-eikm1*subs(zc,{x,y},{xi,ykm1});
end
if i<nx-1
    A(numeq,numeq+ny-1) = eiplk; % (i+1,k)
else
    b(numeq) = b(numeq)-eiplk*subs(zc,{x,y},{xip1,yk});
end
if i>1
    A(numeq,numeq-ny+1) = eim1k; % (i-1,k)
else
    b(numeq) = b(numeq)-eim1k*subs(zc,{x,y},{xim1,yk});
end
if (i<nx-1)&(k>1) % угловые точки
    A(numeq,numeq+ny-2) = eiplkm1; % (i+1,k-1)
else
    b(numeq) = b(numeq) - eiplkm1 * subs(zc,{x,y},{xip1,ykm1});
end
if (i<nx-1)&(k<ny-1)
    A(numeq,numeq+ny) = eiplkp1; % (i+1,k+1)
else
    b(numeq) = b(numeq) - eiplkp1 * subs(zc,{x,y},{xip1,ykp1});
end
if (i>1)&(k>1)
    A(numeq,numeq-ny) = eim1km1; % (i-1,k-1)
else
    b(numeq) = b(numeq) - eim1km1 * subs(zc,{x,y},{xim1,ykm1});
end
if (i>1)&(k<ny-1)
    A(numeq,numeq-ny+2) = eim1kp1; % (i-1,k+1)
else
    b(numeq) = b(numeq) - eim1kp1 * subs(zc,{x,y},{xim1,ykp1});
end
end
end
neq =
741

```

Решаем систему уравнений метода конечных разностей и преобразовываем вектор-столбец решений в двумерный массив нужной размерности. Строим массив увеличенных размеров, заполняем его крайние элементы граничными условиями, а средние – полученным решением. Формируем сетку (x,y), соответствующую полученному решению. Рисуем график в виде трёхмерной поверхности с затенением. Надписываем оси. Выравниваем масштабы по осям Ox и Oy . Выбираем палитру.

```
zeq = A\b; % решили СЛАУ
```

```

zsqr = reshape(zsq,ny-1,nx-1); % переформатировали
zfull = zeros(ny+1,nx+1); % полная сетка нулей
[X,Y] = meshgrid(linspace(x1,x2,nx+1),linspace(y1,y2,ny+1));
zfull = subs(zc,{x,y},{X,Y}); % граничные условия
zfull(2:ny,2:nx) = zsqr; % решение
surf1(X,Y,zfull) % строим поверхность
title('\bfExample 12b')
xlabel('x')
ylabel('y')
zlabel('z(x,y)')
v = axis;
da = daspect;
da(1:2) = min(da(1:2));
daspect(da);
axis(v);
colormap(gray) % палитра

```

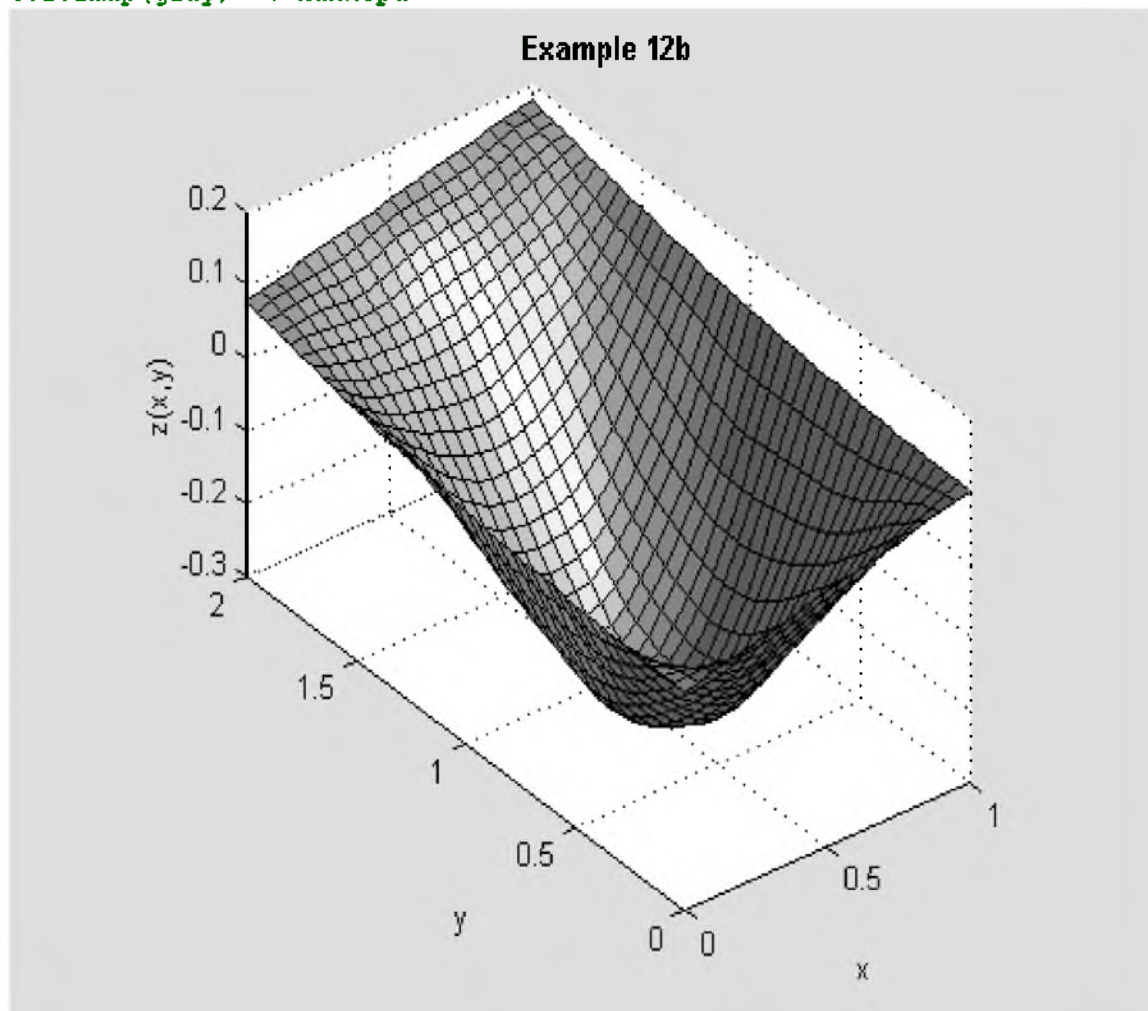


Рис. 12.2. Решение примера 12b

Ответ. Для сетки 20×40 система уравнений МКР вида (12.3) имеет вид

$$\begin{aligned}
 & \left[\frac{1}{4} \left((y_{k-1} + y_k) \sin \left(\pi \frac{x_{i-1} + x_i}{2} + \frac{x_{i-1} + x_i}{10} \right) + (y_{k-1} + y_k) \sin \left(\pi \frac{x_i + x_{i+1}}{2} + \frac{x_i + x_{i+1}}{10} \right) + \right. \right. \\
 & \left. \left. + (y_k + y_{k+1}) \sin \left(\pi \frac{x_{i-1} + x_i}{2} + \frac{x_{i-1} + x_i}{10} \right) + (y_k + y_{k+1}) \sin \left(\pi \frac{x_i + x_{i+1}}{2} + \frac{x_i + x_{i+1}}{10} \right) \right) - \right. \\
 & \left. - 800z_{i+1,k} - 1600z_{i,k} - 800z_{i-1,k} + 1600z_{i,k+1} + 1600z_{i,k-1} = 0; \right. \\
 & \left. i \in [1, 19], \quad k \in [1, 39] \right]
 \end{aligned} \tag{12.8}$$

при этом входящие в (12.8) величины z при $i=1, i=20, k=1, k=40$ находятся из граничного условия в соответствующей точке. График экстремальной поверхности показан на рисунке в зоне вывода. Видно, что он очень похож на поверхность, построенной по МКЭ в главе 4.

12.3. Задание

Решить примеры 1а и 4 методом конечных разностей. Сравнить решения с аналитическими.

13. Численные методы 3. Метод Рунге

13.1. Краткие теоретические сведения

В методе Рунге решение вариационной задачи ищется в виде линейной комбинации известных, заданных заранее функций (они называются базисными). После подстановки такой линейной комбинации в функционал он становится функцией неизвестных коэффициентов этой комбинации. Коэффициенты подбираются так, чтобы функционал принимал экстремальное значение. Таким образом, вариационная задача сводится к задаче исследования на экстремум функции нескольких переменных.

Линейная комбинация базисных функций должна удовлетворять граничным условиям при любых значениях коэффициентов. Если граничные условия однородные (нулевые), то и базисные функции должны удовлетворять однородным граничным условиям. Если же граничные условия неоднородные, то можно применить следующий приём. Будем искать решение в виде линейной комбинации базисных функций, удовлетворяющих однородным граничным условиям, но прибавим к этой комбинации какую-либо функцию, удовлетворяющую заданным граничным условиям. Коэффициент при этой функции не варьируется, он всегда равен 1. Таким образом, полученная линейная комбинация при любых значениях варьируемых параметров будет удовлетворять заданным граничным условиям.

13.2. Примеры выполнения заданий

13.2.а. Пример 13а

Решить методом Рунге пример 1а. Взять базисные функции: одну полуволну синуса и две полуволны синуса. Сравнить решение с аналитическим. Построить графики.

Будем искать решение в виде

$$y(x) = \varphi_0(x) + \alpha_1 \varphi_1(x) + \alpha_2 \varphi_2(x), \quad (13.1)$$

где $\varphi_0(x)$ – любая функция, удовлетворяющая граничным условиям, а $\varphi_1(x)$ и $\varphi_2(x)$ – базисные функции, удовлетворяющие однородным (нулевым) граничным условиям. Выберем $\varphi_0(x)$ в виде прямой, соединяющей граничные точки $M_1(x_1, y_1)$ и $M_2(x_2, y_2)$

$$\varphi_0(x) = y_1 + \frac{y_2 - y_1}{x_2 - x_1}(x - x_1). \quad (13.2)$$

Базисные функции:

$$\varphi_1(x) = \sin \pi \frac{x - x_1}{x_2 - x_1}; \quad \varphi_2(x) = \sin 2\pi \frac{x - x_1}{x_2 - x_1}. \quad (13.3)$$

Составим программу для этого примера на основе программы для примера 1а. Вводим исходные данные. Решаем пример 1а.

```
clear all
format long
disp('Решаем пример 13а')
syms x y Dy D2y a1 a2 % описали переменные
F = x^2+y^2+Dy^2; % 
x1 = -1;
y1 = 1;
x2 = 1;
y2 = 2;
fprintf('Подынтегральная функция: F=%s\n', char(F))
fprintf('Граничные условия: y(%d)=%d; y(%d)=%d\n', x1, y1, x2, y2)
dFdy = diff(F, y);
dFdy1 = diff(F, Dy);
d_dFdy1_dx = diff(dFdy1, x);
d_dFdy1_dy = diff(dFdy1, y);
d_dFdy1_dy1 = diff(dFdy1, Dy); % d(dF/dy')/dy'
dFyldx = d_dFdy1_dx + d_dFdy1_dy*Dy + d_dFdy1_dy1*D2y;
Euler = simple(dFdy-dFyldx);
deqEuler = [ char(Euler) '=0' ]; % составили уравнение
Sol = dsolve(deqEuler, 'x'); % решаем уравнение Эйлера
if length(Sol)~=1 %решений нет или более одного
    error('Нет решений или более одного решения!');
end
SolLeft = subs(Sol, x, sym(x1));
```

```
SolRight = subs(Sol,x,sym(x2));
EqLeft = [char(SolLeft) '=' char(sym(y1))];
EqRight = [char(SolRight) '=' char(sym(y2))];
Con = solve(EqLeft,EqRight); % решаем систему
C1 = Con.C1;
C2 = Con.C2;
Sol1a = vpa(eval(Sol),14);
xpl = linspace(x1,x2);
y1a = subs(Sol1a,x,xpl);
Решаем пример 13а
Подынтегральная функция: F=x^2+y^2+Dy^2
Граничные условия: y(-1)=1; y(1)=2
```

Переводим граничные условия в символические константы. Описываем базисные функции. Формируем решение вида (13.1) и производную от него.

```
x1s = sym(x1);
x2s = sym(x2);
y1s = sym(y1);
y2s = sym(y2);
fi0 = y1s+(y2s-y1s)*(x-x1s)/(x2s-x1s)
fi1 = sin(pi*(x-x1s)/(x2s-x1s))
fi2 = sin(2*pi*(x-x1s)/(x2s-x1s))
yritz = fi0+a1*fi1+a2*fi2
dyritzdx = diff(yritz,x);
fi0 =
3/2+1/2*x
fi1 =
sin(1/2*pi*(x+1))
fi2 =
sin(pi*(x+1))
yritz =
3/2+1/2*x+a1*sin(1/2*pi*(x+1))+a2*sin(pi*(x+1))
```

Подставляем эти функции в подынтегральную функцию, упрощаем и интегрируем.

```
Fritz = simple(subs(F,{y,Dy},{yritz,dyritzdx}));
Jritz = simple(int(Fritz,x,x1s,x2s))
Jritz =
(1/4*a1^2+a2^2)*pi^2+35/6+a2^2+a1^2+(12*a1-2*a2)/pi
```

Найдём экстремум функционала Jritz, который является уже функцией 2-х переменных: a1 и a2. Дифференцируем её по α_1 и α_2 , составляем систему уравнений и решаем её. Преобразовываем полученные символические константы в числа.

```
eq1 = diff(Jritz,a1);
eq2 = diff(Jritz,a2);
eq1 = [char(eq1), '=0'];
eq2 = [char(eq2), '=0'];
S = solve(eq1,eq2,a1,a2); % решаем относительно a1, a2
a1r = eval(S.a1)
a2r = eval(S.a2)
a1r =
-0.55080426575187
a2r =
0.02928440396155
```

Заполняем таблицу для решения методом Рунга. Строим график.

```
yr=subs(subs(yritz,{a1,a2},{a1r,a2r}),x,xpl);
plot(xpl,y1a,'--b',xpl,yr,'-r')
title('\bfExample 13a')
xlabel('x')
```



```
ylabel('y(x)')
```

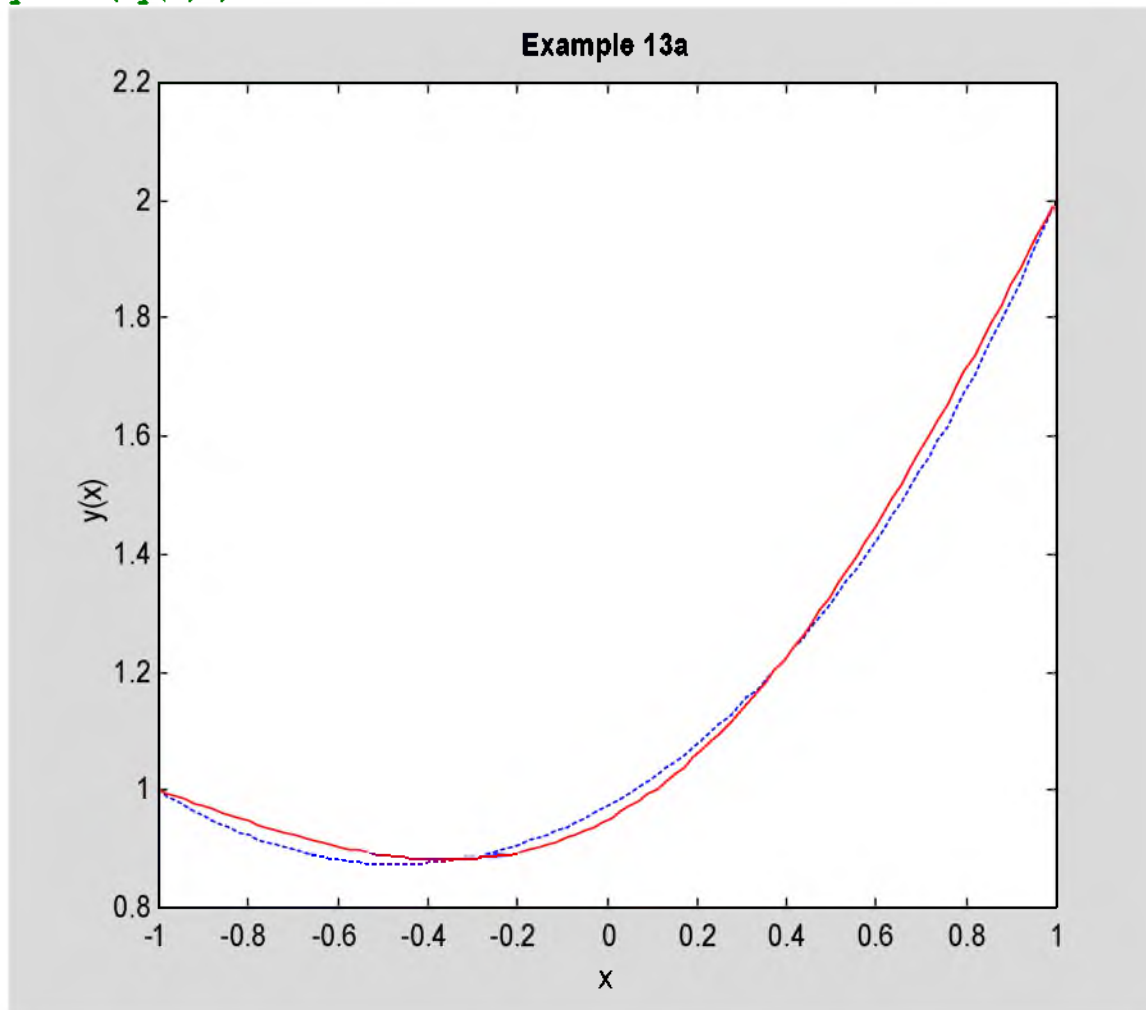


Рис. 13.1. Решение примера 13а

Ответ. Решение задачи методом Рунге имеет вид

$$y(x) = \frac{x+3}{2} - 0.550804 \sin \pi \frac{x+1}{2} + 0.0292844 \sin \pi (x+1). \quad (13.4)$$

График этого решения показан на рис.13.1 сплошной красной линией. Штриховая синяя линия – аналитическое решение.

13.2.b. Пример 13b

Решить методом Рунге пример 4. Взять в качестве базисных функций одну и две полуволны синуса по обеим координатам. Построить график. Сравнить с решением МКР 12b и МКЭ 4. Удачно ли подобраны базисные функции?

Будем искать решение в виде

$$z(x, y) = \varphi_0(x, y) + \alpha_{11} \varphi_{11}(x, y) + \alpha_{21} \varphi_{21}(x, y) + \alpha_{12} \varphi_{12}(x, y) + \alpha_{22} \varphi_{22}(x, y), \quad (13.5)$$

где $\varphi_0(x, y)$ – функция, удовлетворяющая граничным условиям, а $\varphi_{11}(x, y)$, $\varphi_{21}(x, y)$, $\varphi_{12}(x, y)$ и $\varphi_{22}(x, y)$ – базисные функции, удовлетворяющие однородным (нулевым) граничным условиям. В качестве $\varphi_0(x, y)$ возьмём заданное граничное условие:

$$\varphi_0(x) = z|_c = \frac{x}{10} + \frac{y^2}{50}. \quad (13.6)$$

Базисные функции $\varphi_{ik}(x, y)$ – это функции, имеющие i полуволн синуса в направлении x и k в направлении y .

$$\varphi_{ik}(x, y) = \sin i \pi \frac{x-x_1}{x_2-x_1} \sin k \pi \frac{y-y_1}{y_2-y_1}; \quad i, k = [1, 2] \quad (13.7)$$

Используем программы для решения примеров 13а и 4. Описываем символические константы и вводим исходные данные.

```
clear all
```

```
format long
disp('Решаем пример 13b')
n = 20; % число точек для построения графика
syms x y z Dz x Dz y a11 a21 a12 a22 % переменные
F = Dz x^2-2*Dz y^2+2*y*z*(sin(pi*x)+x/5); % подынтегральная функция
zc = x/10+y^2/50; % функция граничных условий
x1 = 0;
x2 = 1;
y1 = 0;
y2 = 2;
fprintf('Подынтегральная функция: F=%s\n',char(F))
fprintf('Граничное условие на контуре: z=%s\n',char(zc))
fprintf('Область: %d<=x<=%d; %d<=y<=%d\n',x1,x2,y1,y2)
Решаем пример 13b
Подынтегральная функция: F=Dz x^2-2*Dz y^2+2*y*z*(sin(pi*x)+1/5*x)
Граничное условие на контуре: z=1/10*x+1/50*y^2
Область: 0<=x<=1; 0<=y<=2
```

Переводим границы в символические константы. Задаём базисные функции. Формируем решение в виде (13.5) и частные производные от него.

```
x1s = sym(x1);
x2s = sym(x2);
y1s = sym(y1);
y2s = sym(y2);
fi11 = sin(pi*(x-x1s)/(x2s-x1s))*sin(pi*(y-y1s)/(y2s-y1s))
fi21 = sin(2*pi*(x-x1s)/(x2s-x1s))*sin(pi*(y-y1s)/(y2s-y1s))
fi12 = sin(pi*(x-x1s)/(x2s-x1s))*sin(2*pi*(y-y1s)/(y2s-y1s))
fi22 = sin(2*pi*(x-x1s)/(x2s-x1s))*sin(2*pi*(y-y1s)/(y2s-y1s))
zritz = zc+a11*fi11+a21*fi21+a12*fi12+a22*fi22
dzritzdx = diff(zritz,x);
dzritzdy = diff(zritz,y);
fi11 =
sin(pi*x)*sin(1/2*pi*y)
fi21 =
sin(2*pi*x)*sin(1/2*pi*y)
fi12 =
sin(pi*x)*sin(pi*y)
fi22 =
sin(2*pi*x)*sin(pi*y)
zritz =
1/10*x+1/50*y^2+a11*sin(pi*x)*sin(1/2*pi*y)+a21*sin(2*pi*x)*sin(1/2*pi*y)+a12*sin(pi*x)*sin(pi*y)+a22*sin(2*pi*x)*sin(pi*y)
```

Вычисляем подынтегральную функцию $F(x,y,z,p,q)$ для этого решения. Интегрируем её.

```
Fritz = simple(subs(F,{z,Dz x,Dz y},{zritz,dzritzdx,dzritzdy}));
Jx = simple(int(Fritz,x,x1s,x2s));
Jritz = simple(int(Jx,y,y1s,y2s))
Jritz =
(1/4*a11^2-1/2*a12^2+a22^2+7/4*a21^2)*pi^2+203/3750+(4*a11+18/25-2*a12)/pi+(2/5*a22+72/25*a11-4/5*a12-4/5*a21)/pi^2
```

Символическая переменная $Jritz$ является функцией аргументов $a11$, $a21$, $a12$ и $a22$. Для нахождения экстремума дифференцируем функцию по этим переменным и решаем полученную систему уравнений. Находим численные значения полученных символических констант.

```
eq11 = diff(Jritz,a11);
eq21 = diff(Jritz,a21);
eq12 = diff(Jritz,a12);
eq22 = diff(Jritz,a22);
eq11 = [char(eq11),'=0'];
eq21 = [char(eq21),'=0'];
```

```
eq12 = [char(eq12), '=0'];
eq22 = [char(eq22), '=0'];
S = solve(eq11, eq21, eq12, eq22, a11, a21, a12, a22);
r11 = eval(S.a11)
r21 = eval(S.a21)
r12 = eval(S.a12)
r22 = eval(S.a22)
r11 =
    -0.31714433325258
r21 =
    0.00234651022964
r12 =
    -0.07271585467015
r22 =
    -0.00205319645094
```

Строим сетку для рисования графика. Вычисляем во всех узлах сетки функцию $\varphi_0(x,y)$. Прибавляем остальные слагаемые согласно формуле (13.5). Строим поверхность. Надписываем заголовок и оси. Выравниваем масштабы по осям Ox и Oy . Выбираем палитру.

```
[X,Y]=meshgrid(linspace(x1,x2),linspace(y1,y2));
zr = subs ( subs(zritz,{a11,a12,a21,a22},{r11,r12,r21,r22}), {x,y}, {X,Y} );
surf1(X,Y,zr)
title('\bfExample 13b')
xlabel('x')
ylabel('y')
zlabel('z(x,y)')
v = axis;
da = daspect;
da(1:2) = min(da(1:2));
daspect(da);
axis(v);
colormap(gray) % палитра
```

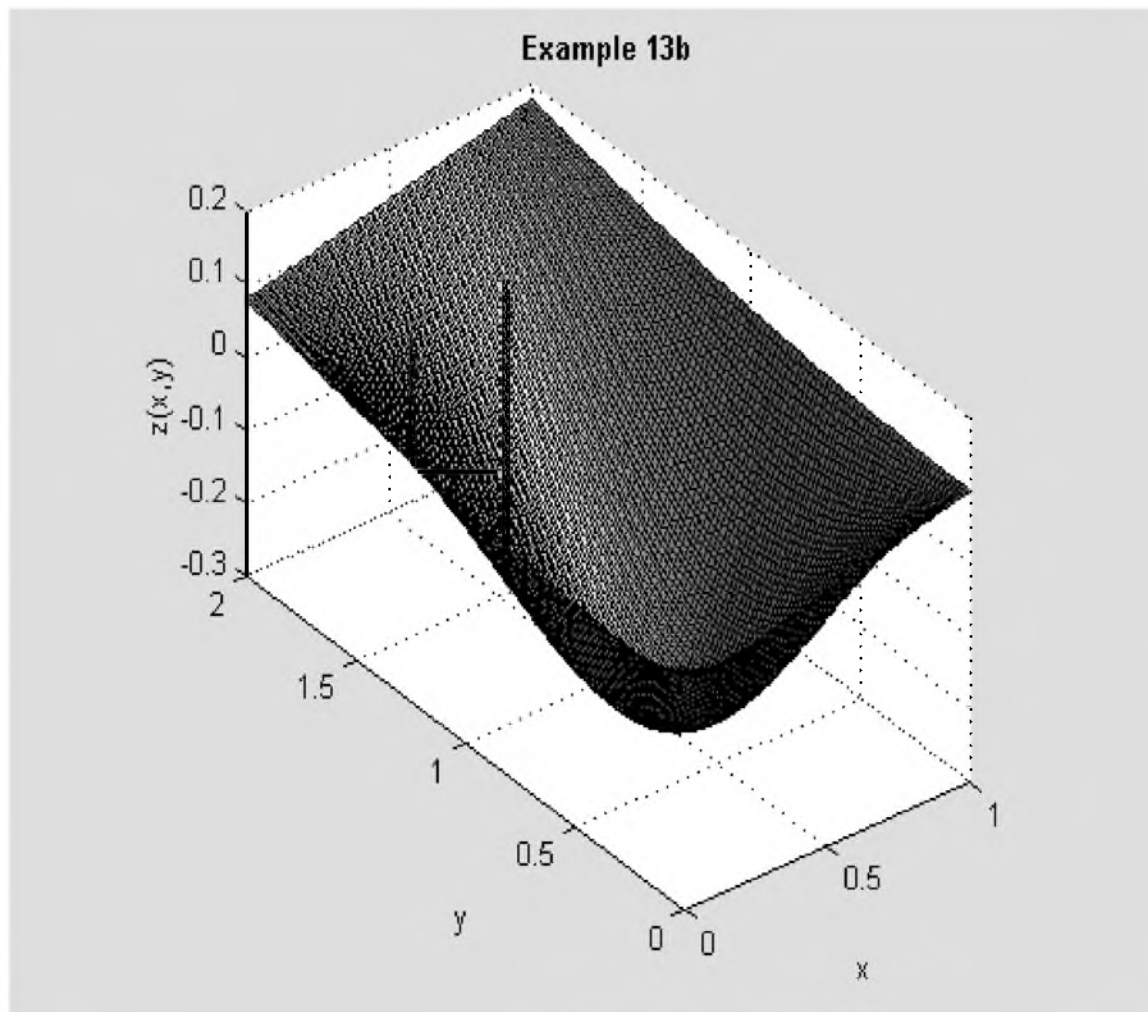


Рис. 13.2. Решение примера 13b

Ответ. Решение задачи методом Ритца имеет вид

$$z(x, y) = \frac{x}{10} + \frac{y^2}{50} - 0.31714433 \sin \pi x \sin \frac{\pi y}{2} + 0.00234651 \sin 2\pi x \sin \frac{\pi y}{2} - 0.072716 \sin \pi x \sin \pi y - 0.0020532 \sin 2\pi x \sin \pi y. \quad (13.8)$$

Решение показано на рис.13.2. Оно хорошо согласуется с результатами решения примеров 4 и 12b. Значит, выбор базисных функций (13.7) оказался удачным.

13.3. Задание

Решить примеры 1a и 4 методом Ритца. Решение примера 1a сравнить с аналитическим. График решения примера 4 сравнить с графиком решения этого примера МКР и МКЭ. Удачно ли выбраны базисные функции?

Литература

1. Ахиезер Н. И. Лекции по вариационному исчислению. – М., Гостехиздат, 1955.
2. Гельфанд И. М., Фомин С. В. Вариационное исчисление. – М., Наука, 1969.
3. Краснов Л. М., Макаренко Г. И., Киселёв А. И. Вариационное исчисление. – М., Наука, 1973.
4. Потёмкин В.Г. Система инженерных и научных расчётов MATLAB 5.x. В 2-х томах. М., Диалог-МИФИ, 1999.
5. Цлаф Л. Я. Вариационное исчисление и интегральные уравнения. – М., Наука, 1970.
6. Эльсгольц Л. Э. Дифференциальные уравнения и вариационное исчисление. – М., Наука, 1965.
7. Optimization Toolbox User's Guide. – © COPYRIGHT 1990 – 1997 by The MathWorks, Inc.
8. Partial Differential Equation Toolbox User's Guide. – © COPYRIGHT 1984 – 1997 by The MathWorks, Inc.
9. Symbolic Math Toolbox User's Guide. – © COPYRIGHT 1993 – 1997 by The MathWorks, Inc.

Оглавление

Введение	Introduction_M.doc
1 Элементарная задача вариационного исчисления	Part_01_M.doc
2 Экстремаль функционала, зависящего от нескольких функций	Part_02_M.doc
3 Экстремаль функционала, зависящего от производных высших порядков	Part_03_M.doc
4 Экстремаль функционала, зависящего от функции нескольких переменных	Part_04_M.doc
5 Естественные граничные условия	Part_05_M.doc
6 Условие трансверсальности	Part_06_M.doc
7 Отражение экстремалей	Part_07_M.doc
8 Преломление экстремалей	Part_08_M.doc
9 Односторонние вариации	Part_09_M.doc
10 Изопериметрическая задача	Part_10_M.doc
11 Численные методы 1. Метод начальных параметров	Part_11_M.doc
12 Численные методы 2. Метод конечных разностей	Part_12_M.doc
13 Численные методы 3. Метод Рунге	Part_13_M.doc
Литература	Summary_m.doc