Министерство образования и науки Российской Федерации

Новосибирский государственный технический университет

Методы оптимизации

Методические указания
к лабораторным работам для студентов III курса ФПМИ
(направление 010500 – “Прикладная математика и информатика”
дневного отделения)

Новосибирск,
2011

УДК 519.85

Методические указания являются руководством при выполнении лабораторных занятий, проводимых по курсу "Методы оптимизации" со студентами (направление 010500 – “Прикладная математика и информатика”) в терминальном классе. Они охватывают ряд разделов математического программирования, и могут быть полезны студентам других специальностей.

Составители: д-р техн. наук, проф. Б.Ю. Лемешко,
канд. техн. наук С.Н. Постовалов,
канд. техн. наук В.С. Карманов
канд. техн. наук Е.В. Чимитова

Рецензент: д-р техн. наук Д.В. Лисицин

Работа подготовлена на кафедре прикладной математики

СОДЕРЖАНИЕ

Введение

Лабораторные работы по курсу "Методы оптимизации" связаны с методами поиска оптимальных решений и охватывают ряд разделов математического программирования. Это одномерные методы поиска, методы минимизации функций многих переменных, метод штрафных функций и статистические методы поиска.

При выполнении лабораторных работ предусмотрены как использование фрагментов готового программного обеспечения, так и самостоятельная программная реализация конкретных методов и их анализ, что позволяет глубже понять отдельные аспекты алгоритмов. Необходимое программное обеспечение находится на www-сайте по адресу http://ciu.nstu.ru/kaf/persons/20760/edu_actions/pcources/method.

В зависимости от темы лабораторной работы, доступности соответствующего материала в литературных источниках или полноты его изложения в курсе лекций, в тексте указаний могут присутствовать или отсутствовать сведения об алгоритмах используемых методов. В последнем случае предполагается, что студент может ознакомиться с необходимыми сведениями в литературном источнике, ссылка на который предлагается, или воспользоваться конспектом лекций.

При подготовке отчёта по каждой лабораторной работе основной упор должен быть сделан не на объём проделанной работы и обилие полученных результатов, а на анализ эффективности методов, сравнение их характеристик, определение области предпочтительного использования, на наглядность результатов, подтверждающих выводы по работе, что особенно важно при решении экономических задач. Отчет может быть представлен в электронном виде, но должен содержать всю необходимую информацию.

Лабораторная работа № 1

Методы одномерного поиска

Цель работы

Ознакомиться с методами одномерного поиска [3,12], используемыми в многомерных методах минимизации функций n переменных. Сравнить различные алгоритмы по эффективности на тестовых примерах.

Методические указания

1. *Общая схема методов поиска минимума на отрезке*

Пусть функция $f(x)$ унимодальна на отрезке $[a_0, b_0]$. Необходимо найти точку минимума функции на этом отрезке с заданной точностью ε. Все методы одномерного поиска базируются на последовательном уменьшении интервала, содержащего точку минимума.

Возьмем внутри отрезка $[a_0, b_0]$ две точки x_1 и x_2: $a_0 < x_1 < x_2 < b_0$, и вычислим значения функции в этих точках. Из свойства унимодальности функции можно сделать вывод о том, что минимум расположен либо на отрезке $[a_0, x_2]$, либо на отрезке $[x_1, b_0]$. Действительно, если $f(x_1) < f(x_2)$, то минимум не может находиться на отрезке $[x_2, b_0]$, а если $f(x_1) > f(x_2)$, то минимум не может находиться на отрезке $[a_0, x_1]$. Если же $f(x_1) = f(x_2)$, то минимум находится на интервале $[x_1, x_2]$.

Алгоритм заканчивается, когда длина интервала, содержащего минимум, становится меньше ε. Различные методы одномерного поиска отличаются выбором точек x_1, x_2. Об эффективности алгоритмов можно судить по числу вычислений функции, необходимому для достижения заданной точности.

2. *Метод дихотомии* (деления отрезка пополам)

Точки x_1, x_2 выбираются на расстоянии $\delta < \varepsilon$ от середины отрезка:

$$\begin{aligned} x_1 &= (a_i + b_i - \delta)/2, \\ x_2 &= (a_i + b_i + \delta)/2 \end{aligned} \qquad (1)$$

За одну итерацию интервал неопределенности уменьшается примерно в два раза (рис. 1). За n итераций длина интервала будет примерно равна $\frac{(b_0 - a_0)}{2^n}$. Для достижения точности ε потребуется $n \geq \frac{\ln\left((b_0 - a_0)/\varepsilon\right)}{\ln 2}$ итераций. На каждой итерации минимизируемая функция вычисляется дважды.

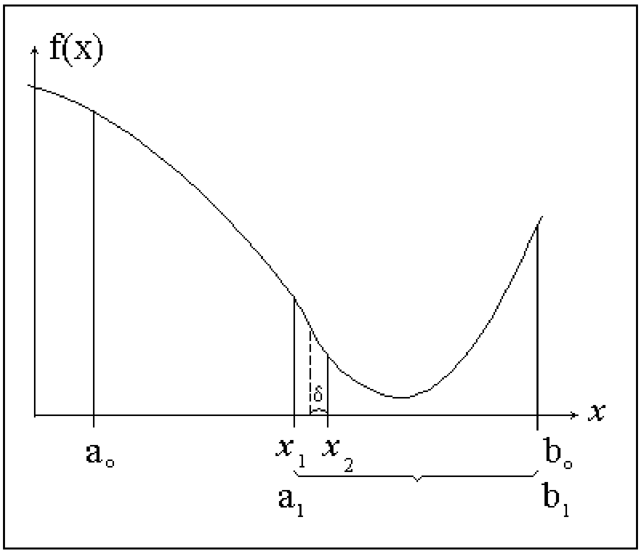

Рис. 1. Метод дихотомии

2. *Метод золотого сечения*

Точки x_1, x_2 находятся симметрично относительно середины отрезка $[a_0, b_0]$ и делят его в пропорции золотого сечения, когда длина всего отрезка относится к длине большей его части также, как длина большей части относится к длине меньшей части:

$$\frac{b_0 - a_0}{b_0 - x_1} = \frac{b_0 - x_1}{x_1 - a_0} \text{ и } \frac{b_0 - a_0}{x_2 - a_0} = \frac{x_2 - a_0}{b_0 - x_2}.$$

Отсюда

$$x_1 = a_i + \frac{(3 - \sqrt{5})}{2}(b_i - a_i) \approx a_i + 0.381966011 \times (b_i - a_i),$$

$$x_2 = a_i + \frac{(\sqrt{5} - 1)}{2}(b_i - a_i) \approx a_i + 0.618003399 \times (b_i - a_i) = \qquad (2)$$

$$= b_i - 0.381966011 \times (b_i - a_i).$$

За одну итерацию интервал неопределенности уменьшается в $\frac{\sqrt{5}+1}{2} = 1.618...$ раз, но на следующей итерации мы будем вычислять функцию только один раз, так как по свойству золотого сечения $\frac{x_2 - x_1}{b - x_1} = 0.381...$ и $\frac{b - x_2}{b - x_1} = 0.618....$ (рис. 2). Для достижения точности ε потребуется $n \geq \frac{\ln((b_0 - a_0)/\varepsilon)}{\ln \frac{\sqrt{5}+1}{2}}$ итераций.

Неточное задание величины $\sqrt{5}$ на ЭВМ уже при достаточно небольшом количестве итераций может приводить к погрешностям и потере точки минимума, так как она выпадает из интервала неопределенности. Поэтому, вообще говоря, при реализации алгоритма возможность такой ситуации должна быть предусмотрена.

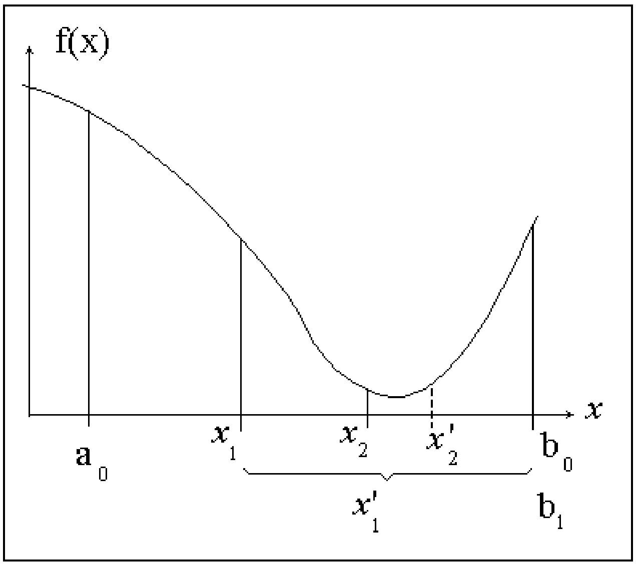

Рис. 2. Метод золотого сечения

3. *Метод Фибоначчи*

Числа Фибоначчи определяются соотношениями:

$$F_{n+2} = F_{n+1} + F_n,\ n = 1,2,3...,\ F_1 = F_2 = 1.$$

С помощью индукции можно показать, что n-е число Фибоначчи представимо в виде (*формула Бинэ*):

$$F_n = \left[\left((1+\sqrt{5})/2\right)^n - \left((1-\sqrt{5})/2\right)^n\right]/\sqrt{5},\quad n = 1,2,...$$

Из этой формулы видно, что при больших n: $F_n \approx \left((1+\sqrt{5})/2\right)^n/\sqrt{5}$, так что числа Фибоначчи с увеличением n растут очень быстро.

На начальном интервале вычисляют точки

$$x_1 = a_0 + \frac{F_n}{F_{n+2}}(b_0 - a_0),$$
$$x_2 = a_0 + \frac{F_{n+1}}{F_{n+2}}(b_0 - a_0), \tag{3}$$

где n выбирается исходя из точности и начальной длины интервала (см. ниже соотношение (5)).

На k-м шаге метода будет получена тройка чисел $a_k, b_k, \overline{x_k}$, локализирующая минимум $f(x)$, такая, что

$$\Delta_k = b_k - a_k = (b_0 - a_0)\frac{F_{n-k+3}}{F_{n+2}},\ 1 \le k \le n,\ a_1 = a_0,\ b_1 = b_0,$$

а точка $\overline{x_k}$, $a_k < \overline{x_k} < b_k$, с вычисленным значением

$$f(\overline{x_k}) = \min_{1 \le i \le k} f(x_i),$$

совпадает с одной из точек

$$x_1 = a_k + \frac{F_{n-k+1}}{F_{n-k+3}}(b_k - a_k) = a_k + \frac{F_{n-k+1}}{F_{n+2}}(b_0 - a_0),$$
$$x_2 = a_k + \frac{F_{n-k+2}}{F_{n-k+3}}(b_k - a_k) = a_k + \frac{F_{n-k+2}}{F_{n+2}}(b_0 - a_0), \tag{4}$$

расположенных на отрезке $[a_k, b_k]$ симметрично относительно его середины (рис. 3). При $k = n$ процесс заканчивается. В этом случае длина отрезка

$$\Delta_n = b_n - a_n = (b_0 - a_0)/F_{n+2},$$

а точки

$$x_1 = a_n + \frac{F_1}{F_{n+2}}(b_0 - a_0),$$
$$x_2 = a_n + \frac{F_2}{F_{n+2}}(b_0 - a_0)$$

совпадают и делят отрезок пополам.

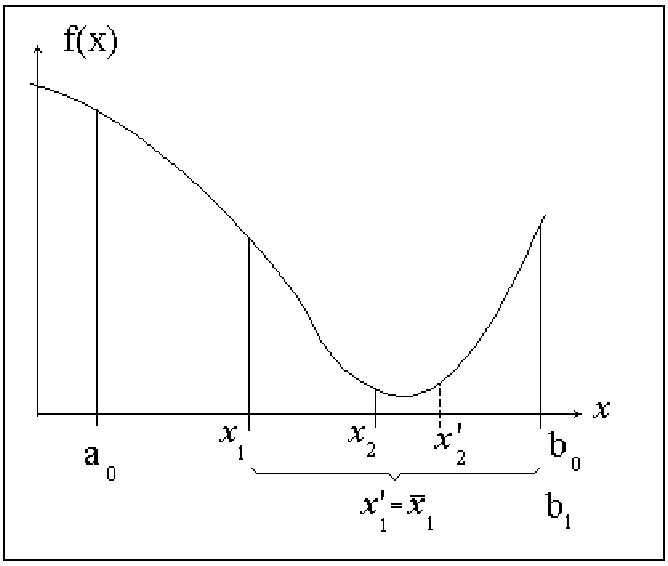

Рис. 3. Метод Фибоначчи

Следовательно

$$\frac{b_n - a_n}{2} = \frac{b_0 - a_0}{F_{n+2}} < \varepsilon.$$

Отсюда можно выбрать n из условия

$$\frac{b_0 - a_0}{\varepsilon} < F_{n+2}. \tag{5}$$

С ростом n, из-за того, что F_n / F_{n+2} – бесконечная десятичная дробь, происходит искажение метода. Поэтому на очередном шаге в качестве новой точки берут из (4) наиболее удалённую от $\overline{x_{k-1}}$ на предыдущем шаге.

4. *Поиск интервала, содержащего минимум функции*

В рассмотренных методах требуется знать начальный отрезок, содержащий точку минимума. Поиск отрезка на прямой заключатся в том, что возрастающие по величине шаги осуществляются до тех пор, пока не будет пройдена точка минимума функции, т.е. убывание функции сменится на возрастание.

Например, интервал может быть выделен с помощью следующего алгоритма. На первом шаге выбираем начальную точку x_0 и определяем направление убывания функции.

Шаг 1. Если $f(x_0) > f(x_0 + \delta)$, то полагаем: k=1, $x_1 = x_0 + \delta$, $h = \delta$. Иначе, если $f(x_0) > f(x_0 - \delta)$, то $x_1 = x_0 - \delta$, $h = -\delta$.

Шаг 2. Удваиваем h и вычисляем $x_{k+1} = x_k + h$.

Шаг 3. Если $f(x_k) > f(x_{k+1})$, то полагаем $k = k+1$ и переходим к шагу 2. Иначе – поиск прекращаем, т.к. отрезок $[x_{k-1}, x_{k+1}]$ содержит точку минимума.

5. *Поиск минимума функции* n *переменных в заданном направлении*

Пусть требуется найти минимум функции n переменных $f(\overline{x})$, где $\overline{x} = (x_1, x_2, ..., x_n)$, в направлении вектора $\overline{s}$. Для этого нужно найти минимум функции $g(\lambda) = f(\overline{x} + \lambda \cdot \overline{s})$ рассмотренными выше методами, λ – величина шага в заданном направлении.

Порядок выполнения работы

1. Реализовать методы дихотомии, золотого сечения и Фибоначчи, исследовать их сходимость и провести сравнение по числу вычислений функции для достижения заданной точности. Построить график зависимости количества вычислений минимизируемой функции от логарифма задаваемой точности ε.
2. Реализовать алгоритм поиска интервала, содержащего минимум функции.

Варианты заданий

1. $f(x) = \sin(x)$, $x \in [-\pi/2, \pi/2]$.
2. $f(x) = \cos(x)$, $x \in [0, \pi]$.

3. $f(x)=(x-2)^2$, $x\in[-2,20]$.
4. $f(x)=(x-15)^2+5$, $x\in[2,200]$.
5. $f(x)=(x+5)^4$, $x\in[-10,15]$.
6. $f(x)=e^x$, $x\in[0,100]$.
7. $f(x)=x^2+2x-4$, $x\in[-10,20]$.
8. $f(x)=x^3-x$, $x\in[0,1]$.
9. $f(x)=x^5-x^2$, $x\in[0,1]$.
10. $f(x)=-x/e^x$, $x\in[0,3]$.
11. $f(x)=x^4-x$, $x\in[0,1]$.
12. $f(x)=x^4/\ln x$, $x\in[1.1, 1.5]$.

Содержание отчета

Отчет должен содержать: титульный лист; цель работы; задание; таблицы с результатами исследований по каждому методу, где должны быть отражены границы и длины интервалов на каждой итерации, соотношение длины интервала на $k-1$ итерации к длине интервала на k итерации; график зависимости количества вычислений целевой функции от логарифма задаваемой точности ε; выводы по всем пунктам задания.

Контрольные вопросы

1. Метод дихотомии.
2. Метод золотого сечения.
3. Метод Фибоначчи.
4. Метод квадратичной интерполяции (*метод парабол*)
5. Алгоритм поиска интервала, содержащего минимум функции.

Лабораторная работа № 2

Методы спуска (0-го, 1-го и 2-го порядка и переменной метрики)

Цель работы

Ознакомиться с методами поиска минимума функции n переменных в оптимизационных задачах без ограничений [1,5,7].

Методические указания

1. Общая схема методов спуска

Пусть дана функция $f(\bar{x})$, где $\bar{x} = (x_1, x_2, ..., x_n)^T$, и задана начальная точка $\bar{x}_0$. Требуется найти минимум функции $f(\bar{x})$ с точностью ε_f – по функции, ε_i – по переменным x_i, где $i = 1,...,n$.

На k-м шаге ($k > 0$) определяем вектор $\bar{s}_k$, в направлении которого функция $f(\bar{x})$ уменьшается. В этом направлении делаем шаг величиной λ_k и получаем новую точку $\bar{x}^{k+1} = \bar{x}^k + \lambda_k \bar{s}^k$, в которой $f\left(\bar{x}^{k+1}\right) < f\left(\bar{x}^k\right)$. Поиск прекращаем как только $\left| f\left(\bar{x}^{k+1}\right) - f\left(\bar{x}^k\right) \right| < \varepsilon_f$ или для всех i верно $\left| \bar{x}_i^{k+1} - \bar{x}_i^k \right| < \varepsilon_i$.

Различные методы спуска отличаются выбором направления и величины шага. Как правило, для нахождения λ_k используется процедура одномерного поиска.

2. Методы 0-го порядка (прямые методы)

К методам нулевого порядка относятся методы, не использующие производные для выбора направления спуска: метод вращающихся координат; метод деформируемого многогранника; метод Хука и Дживса; метод Гаусса; метод Пауэлла.

3. Методы 1-го порядка

К методам первого порядка относятся методы, использующие производные первого порядка для выбора направления спуска: метод наискорейшего спуска; метод сопряженных градиентов в модификации Данилина-Пшеничного (Полака-Рибьера); метод сопряженных градиентов в модификации Флетчера-Ривса.

4. Методы 2-го порядка

К методам второго порядка относятся методы, использующие производные первого и второго порядка для выбора направления спуска: метод Ньютона и его модификации.

5. Методы переменной метрики

К методам переменной метрики относятся методы первого порядка, в которых при работе алгоритма на квадратичных функциях аппроксимируется матрица, обратная к матрице вторых частных производных. Как и методы сопряженных градиентов они имеют квадратичную за *n* шагов скорость сходимости. К ним относятся: метод Бройдена, метод Флетчера; метод Пирсона и др.

Порядок выполнения работы

1. С использованием программного обеспечения исследовать алгоритмы на заданной тестовой функции, осуществляя спуск из различных исходных точек (не менее трех). Исследовать сходимость алгоритма, фиксируя точность определения минимума/максимума, количество итераций метода и количество вычислений функции в зависимости от задаваемой точности поиска. Результатом выполнения данного пункта должны быть выводы об объёме вычислений в зависимости от задаваемой точности и начального приближения.
2. Построить траекторию спуска различных алгоритмов из одной и той же исходной точки с одинаковой точностью. В отчете наложить эту траекторию на рисунок с линиями равного уровня заданной функции.
3. Реализовать метод поиска экстремума функции, проанализировать его работу на квадратичной функции (линии равного уровня не должны быть окружностями) и функции Розенброка ($f_1(\bar{x}) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$). Включить в реализуемый алгоритм собственную процедуру, реализующую одномерный поиск по направлению.

   Метод поиска для самостоятельной реализации выбирается студентом в зависимости от уровня сложности. Максимальное количество баллов за выполнение данной лабораторной работы равно 8 + уровень сложности (от 9 до 12).

Метод поиска для самостоятельной реализации	**Уровень сложности**
метод Гаусса	1
метод Хука и Дживса	1
метод Пауэлла	3
метод деформируемого многогранника	4
метод наискорейшего спуска	1
метод Ньютона	3
метод Пирсона	3
метод вращающихся координат (Розенброка)	3

метод сопряженных градиентов в модификации Данилина-Пшеничного (Полака-Рибьера);	2
метод сопряженных градиентов в модификации Флетчера-Ривса	2
метод Девидона-Флетчера-Пауэлла.	4
метод Бройдена.	4

Варианты заданий

Условие задачи:
Найти **максимум** заданной функции:
Для нечетных вариантов целевая функция имеет вид:

$$f(x,y) = A_1 \exp\left\{-\left(\frac{x-a_1}{b_1}\right)^2 - \left(\frac{y-c_1}{d_1}\right)^2\right\} + A_2 \exp\left\{-\left(\frac{x-a_2}{b_2}\right)^2 - \left(\frac{y-c_2}{d_2}\right)^2\right\}$$

Для четных вариантов целевая функция имеет вид:

$$f(x,y) = \frac{A_1}{1+\left(\frac{x-a_1}{b_1}\right)^2 + \left(\frac{y-c_1}{d_1}\right)^2} + \frac{A_2}{1+\left(\frac{x-a_2}{b_2}\right)^2 + \left(\frac{y-c_2}{d_2}\right)^2}$$

Варианты:

№ варианта	A_1	A_2	a_1	a_2	b_1	b_2	c_1	c_2	d_1	d_2
1	2	3	1	2	2	3	1	3	1	2
2	1	3	2	1	3	1	2	1	3	2
3	1	2	3	2	1	2	1	2	3	1
4	2	1	1	3	2	3	2	1	1	3
5	3	1	2	1	1	2	3	1	2	1
6	2	1	2	3	3	1	2	1	1	3
7	2	3	1	1	2	3	1	3	2	3
8	3	2	2	2	1	3	2	3	2	1
9	2	3	3	1	1	1	2	1	1	3
10	1	2	3	2	1	2	2	2	1	1
11	2	1	1	2	2	2	1	1	3	2
12	3	1	3	3	2	1	2	1	1	3

Содержание отчета

Отчет должен содержать: титульный лист; цель работы; задание; таблицы с результатами проведенных исследований, где должны быть отражены начальное приближение $\bar{x}_0$, задаваемая точность, количество итераций, число вычислений целевой функции, найденная точка и значение функции в ней, а также выводы о сходимости алгоритмов в зависимости от точности и начального приближения с указанием преимуществ и недостатков.

В отчет необходимо включить текст разработанной программы поиска, результаты ее тестирования.

Контрольные вопросы

1. Метод Гаусса.
2. Метод Хука и Дживса.
3. Метод Розенброка (вращающихся координат).
4. Метод Пауэлла.
5. Метод деформируемого многогранника.
6. Метод наискорейшего спуска.
7. Метод сопряженных градиентов и его модификации.
8. Метод Ньютона и его модификации.
9. Методы переменной метрики.

Лабораторная работа № 3

Метод штрафных функций

Цель работы

Ознакомиться с методами штрафных функций при решении задач нелинейного программирования [5,12]. Изучить типы штрафных и барьерных функций, их особенности, способы и области применения, влияние штрафных функций на сходимость алгоритмов, зависимость точности решения задачи нелинейного программирования от величины коэффициента штрафа.

Методические указания

С помощью методов штрафных функций и барьеров (их еще называют *методы внешней и внутренней штрафной точки*) задача нелинейного программирования решается путём исследования *последовательности задач* без ограничений. Вследствие того, что методы штрафных функций и барьеров не оперируют ограничениями в явном виде, они оказываются эффективными в вычислительном отношении для задач нелинейного программирования.

Методы штрафных функций и барьеров аппроксимируют исходную задачу нелинейного программирования последовательностью связанных с ней задач без ограничений, каждая из которых может быть решена с помощью имеющихся алгоритмов оптимизации.

В **методе штрафных функций** исходную задачу

$$\min f(\overline{x})$$

при ограничениях

$$h_j(\overline{x}) = 0, \ \ j = \overline{1,m};$$

$$q_j(\overline{x}) \le 0, \ \ j = \overline{1,k},$$

сводят к задаче без ограничений

$$\min Q(x) = \min \left\{ f(\overline{x}) + r_0 [\sum_{j=1}^{m} r_j \Phi_j (h_j(\overline{x})) + \sum_{l=1}^{k} r_l S_l (q_l(\overline{x}))] \right\}. \qquad (6)$$

где $\Phi(\cdot)$, $S(\cdot)$ – функции штрафа, которые накладываются при нарушении ограничений. Обычно функция штрафа выбирается такой, чтобы штраф был равен нулю, если ограничение выполняется, и больше нуля, если нарушено.

Барьерные функции отличаются от штрафных тем, что в допустимой области они всегда не равны нулю и, кроме того, резко возрастают, стремясь к бесконечности, при приближении к границе допустимой области. В отличие от штрафных барьерные функции требуют специальной адаптации алгоритмов оптимизации, так как при случайном нарушении ограничений в процессе поиска может произойти переполнение разрядной сетки.

Стратегия выбора коэффициентов штрафа. Эффективность применения метода штрафных функций существенно зависит от выбора функции штрафа и правильно подобранной стратегии корректировки коэффициентов штрафа r_j. Как правило, алгоритм подбора коэффициентов штрафа заключается в следующем. На начальном этапе фиксируем точку $\overline{x}_0$, а также начальные значения коэффициентов штрафа и находим минимум функции (6) в точке $\overline{x}_1$. Далее проверяем величину штрафа: если штраф больше заданной точности ε, то изменяем величину штрафа (для штрафных функций коэффициенты штрафа увеличиваются, а для барьерных функций – уменьшаются) и повторяем поиск из точки $\overline{x}_1$. Так продолжаем до тех пор, пока величина штрафа не станет меньше ε.

Порядок выполнения работы

1. Применяя методы поиска 0-го порядка на основании исходных текстов программ, реализующих соответствующие алгоритмы, построить программу для решения задачи нелинейного программирования с использованием барьерных и штрафных функций.
2. Исследовать сходимость методов штрафных и барьерных функций в зависимости от выбора штрафных и барьерных функций соответственно и стратегии выбора коэффициентов штрафа, осуществляя спуск из различных исходных точек. Исследовать сходимость, фиксируя точность определения минимума, количество итераций метода и степень нарушения ограничений в зависимости от задаваемой величины коэффициента штрафа.

Варианты заданий

Условие: целевая функция из лабораторной работы № 2.

Варианты ограничений:

1	$x-y\le 1,\ 0\le x\le 4$	**7**	$x+y\le 1$
2	$x+y\le 1$	**8**	$x+2y\le 1$
3	$x+3y\le 3$	**9**	$2x+2y\le -1$
4	$3x+y\le 3$	**10**	$3x+2y\le 6$

5	$2x - 2y \le 1$	**11**	$2x + y \le 2$
6	$2x + y \le 1$	**12**	$2x + 3y \le 6$

Содержание отчета

Отчет должен содержать: титульный лист; цель работы; задание; таблицы с результатами проведенных исследований, где должны быть отражены начальное приближение $\bar{x}_0$, задаваемая точность, начальное значение коэффициента штрафа, количество итераций, число вычислений целевой функции, найденная точка и значение функции в ней, а также выводы об эффективности метода штрафных функций, рекомендации о выборе функций штрафа и стратегии выбора коэффициентов штрафа в зависимости от используемого метода оптимизации и вида задачи нелинейного программирования.

Контрольные вопросы

1. Метод штрафных функций.
2. Метод барьерных функций.
3. Стратегии изменения коэффициентов штрафа.
4. Виды штрафных функций для ограничений равенств.
5. Виды штрафных функций для ограничений неравенств.
6. Виды барьерных функций.

Лабораторная работа № 4

Статистические методы поиска

Цель работы

Ознакомиться со статистическими методами поиска при решении задач нелинейного программирования [12]. Изучить методы случайного поиска при определении локальных и глобальных экстремумов функций.

Методические указания

Статистические методы поиска иногда делят на 2 вида: ненаправленный и направленный поиск. Ненаправленный случайный поиск используется чаще всего для определения глобального экстремума задачи нелинейного программирования. В этом случае последующие испытания проводятся совершенно независимо от результатов предыдущих. В допустимой области генерируются случайные точки, в которых вычисляются значения целевой функции. В простейшем случае генерация осуществляется по равномерному закону в n-мерном гиперпрямоугольнике. Если задача с ограничениями, то допустимая область вписывается в гиперпрямоугольник, и оставляются только те точки, которые попадают в допустимую область. В направленном случайном поиске отдельные испытания связаны между собой. Результаты уже проведенных испытаний используются для проведения последующих. Сходимость таких методов значительно выше, но приводят они только к локальным решениям. Примерами таких методов являются алгоритм с парной пробой, алгоритм наилучшей пробы, в котором генерируются случайные точки на сфере и спуск осуществляется в "наилучшем" направлении, алгоритм статистического градиента, алгоритм наилучшей пробы с направляющим гиперквадратом.

При тестировании реализованных алгоритмов желательно фиксировать начальное значение генератора случайных чисел. Это позволит повторить работу алгоритма при повторном запуске программы.

Порядок выполнения работы

1. Для поиска экстремума заданной функции реализовать простейший случайный поиск экстремума в гиперпрямоугольнике. Желательно результаты поиска отображать в графическом режиме.
2. Реализовать алгоритм наилучшей пробы с направляющим гиперквадратом. Размер гиперквадрата должен меняться в процессе поиска по следующему алгоритму: вдали от точки экстремума размер гиперквадрата должен увеличиваться, вблизи точки экстремума - уменьшаться.

3. Сравнить результаты работы алгоритмов случайного поиска с результатами работы методов штрафных функций.

Варианты заданий совпадают с вариантами заданий к лабораторной работе № 3.

Содержание отчета

Отчет должен содержать: титульный лист; цель работы; задание; таблицы с результатами проведенных исследований, где должны быть отражены задаваемая точность, количество случайных проб, найденная точка и значение функции в ней, а также выводы об эффективности методов случайного поиска, их трудоёмкости, возможностях при определении локального и глобального экстремума.

Контрольные вопросы

1. Алгоритм с парной пробой.
2. Алгоритм статистического градиента.
3. Алгоритм наилучшей пробы с направляющим гиперквадратом.
4. Алгоритмы глобального поиска.

Литература

1. Васильев В.П. Численные методы решения экстремальных задач. - М.: Наука, 1980. - 518 с.
2. Зайченко Ю.П. Исследование операций. - Киев: Вища школа, 1975. - 320 с.
3. Карманов В.П. Математическое программирование. - М.: Наука, 1975. - 272 с.
4. Кюнци Г.П., Крелле В., Нелинейное программирование, Изд - во "Советское радио", М., 1965.
5. Моисеев Н.Н., Иванилов Ю.П., Столярова Е.М. Методы оптимизации. - М.: Наука, 1978. - 352 с.
6. Растригин Л.А. Статистические методы поиска. - М.: Наука, 1968. - с. 82-121.
7. Химмельблау Д. Прикладное нелинейное программирование. – М.:Мир, 1975. – 534 с.