

$$I_1^1(z) = \frac{(ab)^{n+1} (ia + 2b)^{n+1}}{(2b^2 + 4aib - a^2)n} e^{-ibz} (1 + O(1)). \quad (6)$$

Асимптотика интеграла (5) находится аналогично. И пользуясь тем фактом, что $Q_n(z) = e^{-2z/3} (1 + O(1))$, получаем первое равенство теоремы.

Перейдем к доказательству второго равенства теоремы. Рассмотрим

$$R_{n,2n}^2(z) = \frac{e^{(a-ib)z} z^{3n+1}}{(3n)!} \int_0^{a-ib} x^n (x-a-ib)^n (x-a+ib)^n e^{-zx} dx.$$

Преобразуем данный интеграл

$$R_{n,2n}^2(z) = \frac{e^{(a-ib)z} z^{3n+1}}{(3n)!} \left(\int_0^{-ib} x^n (x-a-ib)^n (x-a+ib)^n e^{-zx} dx + \int_{-ib}^{a-ib} x^n (x-a-ib)^n (x-a+ib)^n e^{-zx} dx \right).$$

Для определения интегралов, входящих в последнее равенство, поступаем аналогично первому случаю. И получаем требуемые теоремой равенства.

Список цитированных источников

1. Никишин, Е.М. Рациональные аппроксимации и ортогональность / Е.М. Никишин, В.Н. Сорокин. – М.: Наука, гл. ред. физ.-мат. лит., 1988. – 256с.
2. Сидоров, Ю.В. Лекции по теории функций комплексного переменного: учеб. для вузов. – 3-е изд. / Ю.В. Сидоров, М.В. Федорюк, М.И. Шабунин – М.: Наука, 1989. – 480 с.

УДК 519.853.3

ЧИСЛЕННЫЙ АНАЛИЗ ПРЯМОГО ОПОРНОГО МЕТОДА ДЛЯ МИНИМИЗАЦИИ ВЫПУКЛЫХ ФУНКЦИЙ

Бруцкий В.Р., Мокин А.А.

*Брестский государственный технический университет, г. Брест
Научный руководитель: Ракецкий В.М., к.ф.-м.н., доцент*

1. Введение. Целью настоящей работы является экспериментальное исследование прямого опорного метода для минимизации выпуклых функций [1]. В качестве объекта для проведения численного эксперимента рассматривается задача безусловной оптимизации

$$f(x) \rightarrow \min, x \in R^n, \quad (1)$$

где $f(x)$ – дважды дифференцируемая сильно выпуклая функция, т.е. в задаче (1) всегда существует единственное решение x^* – точка минимума функции $f(x)$, которая удовлетворяет необходимым и достаточным условиям оптимальности

$$\Delta_j(x) = \frac{\partial}{\partial x_j} f(x) = 0, \quad j = \overline{1, n}. \quad (2)$$

2. Алгоритм прямого опорного метода. Пусть $\{x, J_{ou}\}$ опорный план [1] задачи (1), известный к началу итерации, ε – заданная точность выполнения условий оптимальности (2), η – параметр метода, гарантирующий монотонное убывание целевой функции от итерации к итерации. Предположим, что опора целевой функции J_{ou} состоит из индексов $\{i_1, i_2, \dots, i_m\}$, множество неопорных индексов J_H состоит из индексов $\{j_1, j_2, \dots, j_{n-m}\}$, j_k –

неопорный индекс, для которого предстоит проверить условия оптимальности на текущей итерации, J_n^0 – множество индексов из J_n , для которых на предыдущих итерациях был установлен факт выполнения условий оптимальности с заданной точностью, т.е. $|\Delta_j(x)| \leq \varepsilon$, $j \in J_n^0$, и опорный план $\{x, J_{ou}\}$ не менялся.

2.1. Проверим, выполняются ли условия (2) на индексе j_k с заданной точностью:

$$|\Delta_{j_k}(x)| \leq \varepsilon. \quad (3)$$

Если условие (3) не выполняется, то перейдем к п. 2.3. В противном случае положим $J_n^0 = J_n^0 \cup \{j_k\}$. Возможны 2 случая.

а) $J_n^0 = J_n$, т.е. все неопорные индексы удовлетворяют с заданной точностью условиям оптимальности. В этом случае перейдем к проверке согласованности опорного плана.

б) $J_n^0 \subset J_n$, $J_n^0 \neq J_n$, т.е. возможно, что среди неопорных индексов есть такие, для которых условия оптимальности не выполняются с заданной точностью. В этом случае положим $k = (k+1) \bmod (n-m)$ и снова проверим условие (3) для нового индекса.

2.2. Проверим условия согласования опорного плана

$$|\Delta_j(x)| \leq \varepsilon, \quad j \in J_{ou} \quad (4)$$

Если условия согласования (4) выполняются, то решение задачи (1) окончено, план x принимаем в качестве оптимального плана. В противном случае полагаем $J_{ou} = \emptyset$ и переходим к п.2.1. В качестве индекса j_k используем первый из индексов опорного плана, для которого не выполнены условия согласования (4).

2.3. Построим направление l для улучшения опорного плана $\{x, J_{ou}\}$

$$l_{j_k} = -\text{sign} \Delta_{j_k}, \quad l_j = 0, \quad j \in J_n \setminus \{j_k\}, \quad l_{ou} = G_{ou} p_{ou} \text{sign} \Delta_{j_k}, \quad p_{ou} = \{p_j, j \in J_{ou}\}, \quad p_j = \frac{\partial^2 f(x)}{\partial x_j \partial x_{j_k}} \quad (5)$$

2.1.4. Вычислим шаг θ вдоль направления l :

$$\theta = \begin{cases} |\Delta_{j_0}| / \alpha, & \text{если } \alpha > 0, \\ \infty, & \text{если } \alpha \leq 0, \end{cases}, \quad \alpha = l'_{ou} p_{ou} \text{sign} \Delta_{j_k} + d_{j_k j_k}, \quad d_{j_k j_k} = \frac{\partial^2 f(x)}{\partial x_{j_k} \partial x_{j_k}} \quad (6)$$

Если $\theta = \infty$, то перейдем к п.2.6.

2.5. Проверим условие

$$f(x + \theta l) < f(x) - \eta. \quad (7)$$

Возможны 2 варианта:

а) условие (7) выполняется. В этом случае построим новый опорный план $\{\tilde{x}, \tilde{J}_{ou}\}$, где $\tilde{x} = x + \theta l$, $\tilde{J}_{ou} = J_{ou} \cup \{j_k\}$, и пересчитаем матрицу G_{ou} :

$$\tilde{G}_{ou} = \begin{pmatrix} G_{ou} + \frac{l_{ou} l_{ou}^T}{\alpha} & -\frac{l_{ou}}{\alpha} \text{sign} \Delta_{j_0} \\ -\frac{l_{ou}^T}{\alpha} \text{sign} \Delta_{j_0} & \frac{1}{\alpha} \end{pmatrix}$$

Если $|\tilde{J}_{ou}| = n$, то $\tilde{J}_n = \emptyset$. Поэтому следующую итерацию начнем сразу с п. 2.2 – проверки условий согласования опорного плана $\{\tilde{x}, \tilde{J}_{ou}\}$. Если же $|\tilde{J}_{ou}| < n$, то перейдем к п. 2.1;

б) условие (7) не выполняется. Продолжим итерацию, для чего перейдем к п. 2.6.

2.6. Найдем шаг θ вдоль направления l с помощью метода одномерной оптимизации (например, метода «золотого» сечения») и проверим условие

$$f(x + \theta l) < f(x) - \eta / \delta \quad (8)$$

Возможны 2 исхода:

а) условие (8) выполняется. В этом случае уточним значение параметра $\eta = (f(x) - f(x + \theta l)) / 2$ и перейдем к следующей итерации по правилам пункта 2.5 а), приняв в качестве значения α величину $\alpha = |D_{j_0}| / \theta$.

б) условие (8) не выполняется. В этом случае прервем текущую итерацию и проверим условия согласования опорного плана $\{x, J_{ou}\}$, т.е. перейдем к п. 2.2.

3. Эксперименты. В эксперименте целевая функция (1) имела вид

$$f(x) = \frac{1}{2} x^T D x + c^T x + \sum_{i=1}^m e^{\sum_{j=1}^n a_{ij} x_j}, \quad (9)$$

где D – симметричная положительно определенная ($n \times n$) – матрица, c – n -вектор, $a_{ij}, i = \overline{1, m}, j = \overline{1, n}$ – известные величины. Функция (9) является дифференцируемой (существуют любые ее производные) и сильно выпуклой. При этом имеют место следующие соотношения

$$\Delta_j(x) = \frac{\partial f(x)}{\partial x_j} = \sum_{k=1}^n d_{jk} x_k + c_j + \sum_{k=1}^m a_{kj} e^{\sum_{l=1}^n (a_{kl} x_l - b_k)}, \quad \frac{\partial f(x)}{\partial x_i \partial x_j} = d_{ij} + \sum_{k=1}^m a_{ki} a_{kj} e^{\sum_{l=1}^n (a_{kl} x_l - b_k)} \quad (10)$$

Для автоматизации процесса тестирования был разработан генератор задач вида (9). В ходе эксперимента генерировались задачи одного размера сериями по 10 штук. Контролировались два параметра: время решения и количество итераций. В качестве результатов эксперимента принимались средние результаты по серии, из которых отбрасывалось одно наибольшее и одно наименьшее значение.

Эксперимент 1. Исследовалось влияние на вычислительный процесс способа вычисления шага. Рассмотрены 2 варианта: а) вычисление шага по правилам, изложенным в п. 2; б) вычисление шага с заданной точностью на каждой итерации алгоритма (методом «золотого сечения»).

Эксперимент 2. Исследовалось, как влияет на вычислительный процесс способ вычисления производных. Рассмотрены 2 варианта: а) точное вычисление производных по формулам (10); б) приближенное вычисление производных по простейшим разностным формулам.

В табл. 1 приведены результаты эксперимента по решению задач с размерами от 10 до 30 переменных (буквы аа, ба, аб, бб обозначают сочетания вариантов экспериментов 1,2).

Таблица 1 – Результаты эксперимента

Количество переменных	Количество итераций				Время решения (мс)			
	аа	ба	аб	бб	аа	ба	аб	бб
10	30	30	30	30	17	30	24	38
15	60	59	62	65	34	45	47	65
20	79	68	79	65	32	83	81	114
25	81	82	81	93	40	120	102	162
30	120	90	120	90	71	177	286	310

Результаты эксперимента позволяют сделать выводы:

1) вычисление шага с заданной точностью приводит к уменьшению количества итераций, однако в 2-3 раза увеличивает время решения задачи;

2) приближенное вычисление производных не приводит к существенному увеличению количества итераций, однако увеличивает время решения задачи. При этом разрыв по времени решения увеличивается с ростом размеров задачи

Список цитированных источников

1. Ракецкий, В.М. К минимизации выпуклых функций с простыми ограничениями // Вестник БрГТУ. – 2011. – № 5(71): Физика, математика, информатика. – С. 108–110

УДК 519.24

**ПРИМЕНЕНИЕ СОСТОЯТЕЛЬНЫХ ОЦЕНОК СПЕКТРАЛЬНЫХ ПЛОТНОСТЕЙ
ДЛЯ АНАЛИЗА ЭРГОДИЧЕСКИХ СТАЦИОНАРНЫХ СЛУЧАЙНЫХ ПРОЦЕССОВ**

Бусько Н.В.

*Гродненский государственный университет им. Я. Купалы, г. Гродно
Научный руководитель: Семенчук Н.В., к. ф.-м. н., доцент*

Эргодическим стационарным случайным процессом называется такой процесс, любая вероятностная характеристика которого, полученная на ансамбле реализаций в какой-либо момент времени t , равна, с вероятностью, сколь угодно близкой к единице, аналогичной характеристике, полученной на одной единственной реализации процесса путем усреднения по времени за достаточно большой промежуток времени T [1].

Использование эргодического стационарного случайного процесса позволяет строить оценки спектральной плотности по одной его реализации, при этом зачастую избавляя исследователей от проведения многочисленных экспериментов, связанных с затратами материальных и временных ресурсов.

Расширенной периодограммой называется периодограмма вида [2]

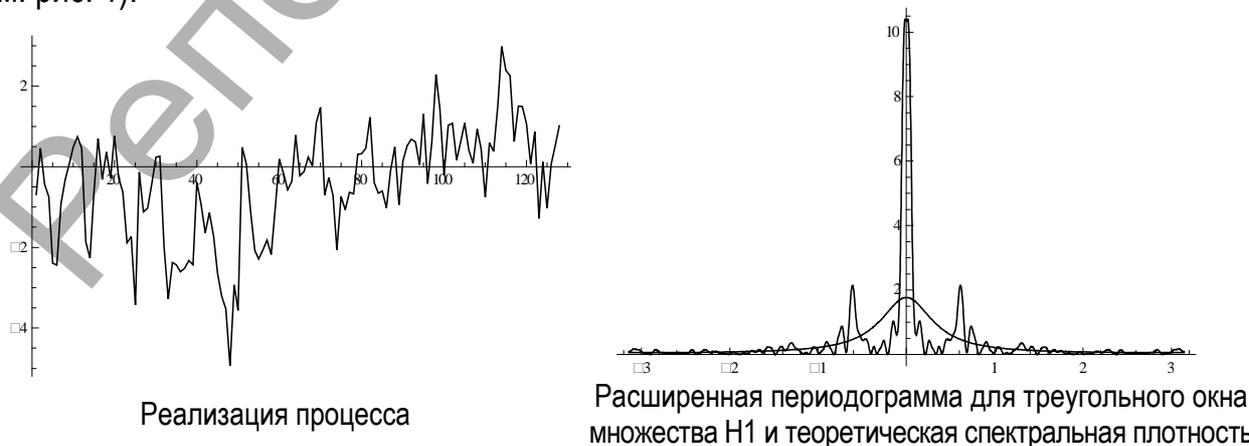
$$I_T^{(h)}(\lambda) = \frac{1}{2\pi H_2^{(T)}(0)} d_T(\lambda) d_T(-\lambda), \tag{1}$$

где $d_T(\lambda) = \sum_{t=0}^{T-1} h_T(t) X(t) e^{-i\lambda t}$,

$$H_k^{(T)}(\lambda) = \sum_{t=0}^{T-1} (h_T(t))^k e^{-i\lambda t},$$

функция $h_T(t) = h\left(\frac{t}{T}\right)$, $h: [0, 1] \rightarrow \mathbf{R}$ – функция окна просмотра данных, $k \in \mathbf{N}$, $T \in \mathbf{N}$.

Разработан алгоритм построения расширенной периодограммы в СКА «Mathematica 8.0» с использованием 10 окон просмотра данных. Алгоритм позволяет строить расширенные периодограммы за приемлемое время для рядов длиной до 10000 наблюдений (см. рис. 1).



**Рисунок 1а – Результаты для процесса авторегрессии первого порядка
 $AR(1)$ $T=128$, с параметром $\beta_1 = 0.7$, $\varepsilon_t \sim N(0,1)$. ($M = 3$)**