

УДК 519.853

ПАРАЛЛЕЛЬНЫЕ МЕТОДЫ РЕШЕНИЯ ЗАДАЧ ГЛОБАЛЬНОЙ ОПТИМИЗАЦИИ ¹

Ю.Г. Евтушенко

Вычислительный центр им. А.А. Дородницына РАН
Россия, 119333, Москва, Вавилова 40

E-mail: evt@ccas.ru

М.А. Посыпкин

Институт системного анализа РАН
Россия, 117312 Москва, пр-т. 60-летия Октября, 9

E-mail: mposypkin@mail.ru

Ключевые слова: глобальная оптимизация, методы параллельных вычислений

Key words: global optimization, parallel computing

Рассматриваются методы решения задач глобальной оптимизации непрерывной функции многих переменных при условиях, когда целевая функция или ее градиент или гессиан удовлетворяют условиям Липшица. Для этих случаев получены опорные миноранты. Найдены аналитические выражения для минимумов минорант на шаре и параллелепипеде. Предложены дополнительные условия исключения подмножеств из области поиска для случая, когда градиент функции или его компоненты удовлетворяют условиям Липшица. Разработанные алгоритмы реализованы на многопроцессорных системах с распределенной памятью. Эффективность реализации иллюстрируется вычислительными экспериментами.

PARALLEL METHODS FOR SOLVING GLOBAL OPTIMIZATION PROBLEMS / Yu.G. Evtushenko (Dorodnicyn Computing Centre of the Russian Academy of Sciences, Vavilov str., 40, Moscow, 119333, GSP-1, Russia, E-mail: evt@ccas.ru), M.A. Posypkin (Institute for System Analysis of the RAS, pr-t. 60-letija Oktjabrja, Moscow, 117312, Russia, E-mail: mposypkin@mail.ru). Methods for solving global optimization problems with Lipschitzian objective function, Lipschitzian gradient and Hessian are considered. Minorants based on Taylor series are obtained and their analytical minima on ball and box regions are given. New elimination conditions for problems with Lipschitzian gradient are proposed. Developed algorithms are implemented for distributed memory multiprocessors. Numerical experiments confirming the efficiency of the implementation are presented.

1. Введение

Задачи глобальной оптимизации возникают во многих важных теоретических и прикладных областях. Эти задачи обычно чрезвычайно сложны и для

¹Работа выполнена при поддержке гранта РФФИ №08-01-00619, программы президиума РАН №14-2, гранта президента РФ НШ-5073.2008.1

их решения приходится применять современные многопроцессорные вычислительные комплексы. Развиваемый в данной работе подход поиска глобального экстремума основан на идеях метода неравномерных покрытий [1-3]. При выполнении определенных условий метод гарантирует нахождение глобального экстремума с заданной точностью.

2. Основная идея метода неравномерных покрытий

Задачу нахождения глобального минимума функции $f(x)$ на допустимом множестве $X \subset \mathbb{R}^n$ запишем в виде:

$$(1) \quad f_* = \mathit{glob} \min_{x \in X} f(x) = f(x_*).$$

Здесь вектор x_* принадлежит множеству решений X_* задачи (1). Определим множество ϵ -решений этой задачи:

$$(2) \quad X_*^\epsilon = \{x \in X : f(x) \leq f_* + \epsilon\}.$$

Для приближенного решения задачи (1) достаточно найти хотя бы одну точку из множества X_*^ϵ .

В процессе вычислений строятся последовательности $\{X_i\} = X_1, \dots, X_k$ подмножеств множества X и точек $x_i \in X_i$. В каждой точке x_i вычисляется значение функции $f(x)$, определяется рекорд f_r и запоминается соответствующая рекордная точка x_r по формулам:

$$(3) \quad f_r = \min_{1 \leq i \leq k} f(x_i) = f(x_r), 1 \leq r \leq k.$$

Для каждого подмножества X_i определяется *миноранта* $G_i(x)$ — функция такая что $f(x) \geq G_i(x)$ на X_i . Последовательности $\{X_i\}$ соответствует последовательность покрывающих множеств $\{Z_i\} = Z_1, \dots, Z_k$, определяемых из условия:

$$(4) \quad Z_i = \{x : x \in X_i, G_i(x) \geq f_r - \epsilon\}, r \leq i.$$

Всюду на множестве Z_i выполнено неравенство $f(x) \geq f_r - \epsilon$. Поэтому минимизация функции $f(x)$ на этом множестве не дает возможности уменьшить величину рекорда f_r более чем на ϵ . Следовательно множество Z_i может быть исключено из рассмотрения (отсеяно) и поиск глобального минимума продолжен на множестве $X_i \setminus Z_i$. Алгоритм завершает свою работу, когда все допустимое множество X оказывается покрытым множествами Z_i , т.е.

$$(5) \quad X \subseteq Z_1 \cup Z_2 \cup \dots \cup Z_k.$$

Имеет место

Теорема 1. Пусть для последовательности $\{Z_i\}$ выполнено условие (5). Тогда определяемая из (3) рекордная точка x_r принадлежит множеству X_*^ϵ .

Доказательство теоремы 1. Рассмотрим точку $x_* \in X_*$. Согласно формуле (5) найдется такое множество Z_i , что $x_* \in Z_i$. Тогда $f_* = f(x_*) \geq$

$G_i(x_*) \geq f_r - \epsilon$. Отсюда, по определению множества ϵ -оптимальных решений (2), следует что $x_r \in X_*^\epsilon$.

При применении на практике, рассмотренная общая схема метода неравномерных покрытий может модифицироваться с учетом особенностей решаемой задачи. В частности, могут применяться дополнительные, не основанные на минорантах, условия отсева подмножеств. Очевидно, что при уменьшении величины рекорда множества Z_i расширяются. Поэтому в процессе расчетов часто прибегают к помощи различных алгоритмов, уменьшающих рекорд. Для этих целей используются методы локального и эвристического поиска минимума.

3. Метод неравномерных покрытий для параллелепипеда

3.1. Общая схема организации вычислений

Рассмотрим частный случай задачи (1), когда допустимое множество — n -мерный параллелепипед с гранями, параллельными координатным осям:

$$X = \{x \in \mathbb{R}^n : a \leq x \leq b\}.$$

Здесь и далее неравенство $a \leq x$ означает, что $a^j \leq x^j$ для всех $1 \leq j \leq n$. Множества X_i представляют собой прямоугольные параллелепипеды $X_i = \{x \in \mathbb{R}^n : a_i \leq x \leq b_i\}$. При этом $a \leq a_i < b_i \leq b$, т.е. $X_i \subseteq X$. В дальнейшем потребуются понятие центра параллелепипеда $c_i = (a_i + b_i)/2$ и его радиуса $\rho(X_i) = \|c_i - a_i\|$. Здесь и далее в качестве нормы применяется евклидова норма для векторов и согласованная с ней операторная норма для матриц.

Алгоритм работает следующим образом. В процессе расчетов поддерживается список параллелепипедов $\{X_i\}$, первоначально состоящий из одного параллелепипеда X . Далее выбирается некоторый элемент списка — X_i и определяется множество Z_i по формуле (4). Если $X_i \subseteq Z_i$, то X_i удаляется из списка и становится элементом последовательности покрывающих множеств $\{Z_i\}$. В противном случае он подвергается дроблению на более мелкие параллелепипеды, которые замещают в списке параллелепипед X_i . В их центрах вычисляется значение функции и уточняется рекорд по формуле:

$$(6) \quad f_r = \min_{1 \leq i \leq k} f(c_i) = f(c_r), 1 \leq r \leq k.$$

Алгоритм завершит свою работу когда в последовательности X_i не останется ни одного элемента. В этом случае множества $\{Z_i\}$ покроют допустимое множество.

Условие $X_i \subseteq Z_i$ удобно проверять следующим образом. Пусть g_i — минимальное значение миноранты $G_i(x)$ на параллелепипеде X_i : $g_i = \min_{x \in X_i} G_i(x)$. Тогда, согласно соотношению (4), условие $X_i \subseteq Z_i$ выполняется, если выполнено неравенство

$$(7) \quad g_i \geq f_r - \epsilon.$$

Будем называть его *условием отсева* **RO** с минорантой G_i .

3.2. Миноранты

Приведем три случая, когда миноранты функции $f(x)$ строятся просто.

1. Пусть функция $f(x)$ принадлежит классу C^0 , т.е. удовлетворяет условию Липшица на параллелепипеде X_i с константой $l_i : |f(x) - f(y)| \leq l_i \|x - y\|$. Тогда следующая функция будет опорной минорантой для функции $f(x)$ на параллелепипеде X_i :

$$G_i^0(x) = f(c_i) - l_i \|x - c_i\|.$$

Данная миноранта является вогнутой функцией и ее минимум на параллелепипеде X_i вычисляется по формуле:

$$(8) \quad g_i^0 = f(c_i) - l_i \|b_i - c_i\|.$$

Из неравенства $f(c_i) - l_i \|x - c_i\| \geq f_r - \epsilon$ следует $\|x - c_i\| \leq (f(c_i) - f_r + \epsilon) / l_i$. Поэтому, шар S_i радиуса $\rho_i = (f(c_i) - f_r + \epsilon) / l_i$ с центром в точке c_i может быть исключен из X_i . Параллелепипеды, радиус которых меньше или равен ρ_i , не подвергаются дроблению и могут быть исключены из дальнейшего рассмотрения.

Согласно (6) $f(c_i) \geq f_r$, поэтому $\rho_i \geq \rho_{im}$, где ρ_{im} — минимальный (по возможным значениям $f(c_i) \geq f_r$) радиус шара, определяемый по формуле:

$$(9) \quad \rho_{im} = \epsilon / l_i.$$

Равенство (9) применяется для быстрой проверки условия отсева, не требующей вычисления значений целевой функции. Всякий шар, в центре которого вычислено значение $f(x)$ и уточнен, если это возможно, рекорд может быть исключен из множества X_i , если его радиус меньше ρ_{im} .

2. Пусть функция $f(x)$ принадлежит классу C^1 , т.е. она дифференцируема, ее градиент $f_x(x)$ удовлетворяет условию Липшица с константой L_i на X_i , т.е. для любых x и y из X_i имеет место неравенство

$$(10) \quad \|f_x(x) - f_x(y)\| \leq L_i \|x - y\|.$$

Несложно показать (см. [4]), что если $f(x)$ дважды дифференцируема, то (10) имеет место тогда и только тогда, когда гессиан $f_{xx}(x)$ функции $f(x)$ удовлетворяет соотношению $\|f_{xx}(x)\| \leq L_i$ при любых $x \in X_i$. Это неравенство можно использовать для получения оценок константы Липшица. Введем следующие две опорные миноранты:

$$(11) \quad G_i^1(x) = f(c_i) + \langle f_x(c_i), x - c_i \rangle - \frac{L_i}{2} \|x - c_i\|^2,$$

$$\overline{G}_i^1(x) = f(c_i) - \|f_x(c_i)\| \cdot \|x - c_i\| - \frac{L_i}{2} \|x - c_i\|^2.$$

Согласно неравенству Коши-Буняковского для любых $x \in X_i$ справедливо $G_i^1(x) \geq \overline{G}_i^1(x)$. Множество $S_i = \{x \in \mathbb{R}^n : \overline{G}_i^1(x) \geq f_r - \epsilon\}$ является шаром с центром в точке c_i и радиусом

$$(12) \quad \rho_i = \left(\sqrt{\|f_x(c_i)\|^2 + 2L_i(\epsilon + f(c_i) - f_r)} - \|f_x(c_i)\| \right) / L_i.$$

Шар S_i может быть исключен из параллелепипеда X_i . Параллелепипеды, радиус которых меньше или равен ρ_i , не подвергаются дроблению и далее не рассматриваются.

В силу (6) справедливо $f(c_i) \geq f_r$, поэтому $\rho_i \geq \rho_{im}$, где ρ_{im} — минимальный (по возможным значениям $f(c_i) \geq f_r$) радиус шара, определяемый по формуле:

$$(13) \quad \rho_{im} = \left(\sqrt{\|f_x(c_i)\|^2 + 2L_i\epsilon} - \|f_x(c_i)\| \right) / L_i.$$

Миноранта $G_i^1(x)$ является вогнутой и ее минимум на параллелепипеде X_i достигается в точке z_i , j -я компонента которой задается соотношением:

$$z_i^j = \begin{cases} a_i^j, & \text{если } \partial f(c_i) / \partial x^j \geq 0, \\ b_i^j, & \text{если } \partial f(c_i) / \partial x^j < 0. \end{cases}$$

При этом значение минимума определяется по формуле

$$(14) \quad g_i^1 = G_i^1(z_i) = f(c_i) - \langle f_x(c_i), b_i - c_i \rangle - \frac{L_i}{2} \|b_i - c_i\|^2,$$

где $|f_x(c_i)| = (|\partial f(c_i) / \partial x^1|, \dots, |\partial f(c_i) / \partial x^n|)$.

3. Пусть $f \in C^2$ — дважды дифференцируемая функция с гессианом, удовлетворяющим условию Липшица на параллелепипеде X_i с константой M_i :

$$(15) \quad \|f_{xx}(x) - f_{yy}(y)\| \leq M_i \|x - y\|.$$

Рассмотрим следующие опорные миноранты функции $f(x)$ на параллелепипеде X_i :

$$\begin{aligned} G_i^2(x) &= f(c_i) + \langle f_x(c_i), x - c_i \rangle + \frac{1}{2} (x - c_i)^T f_{xx}(c_i) (x - c_i) - \\ &\quad - \frac{M_i}{6} \|x - c_i\|^3, \\ \overline{G}_i^2(x) &= f(c_i) + \langle f_x(c_i), x - c_i \rangle + \frac{1}{2} \lambda_{min} \|x - c_i\|^2 - \frac{M_i}{6} \|x - c_i\|^3, \end{aligned}$$

где λ_{min} — минимальное собственное значение матрицы $f_{xx}(c_i)$. Так как $(x - c_i)^T f_{xx}(c_i) (x - c_i) \geq \lambda_{min} \|x - c_i\|^2$, то $G^2(x) \geq \overline{G}^2(x)$.

Рассмотрим сферу S радиуса ρ с центром в точке c_i :

$$S = \{x \in \mathbb{R}^n : \|x - c_i\| = \rho\}.$$

Так как функция $\phi(x) = \frac{1}{2} \lambda_{min} \|x - c_i\|^2 - \frac{M_i}{6} \|x - c_i\|^3$ принимает одинаковые значения на поверхности сферы, то минимум миноранты $\overline{G}_i^2(x) = f(c_i) + \langle f_x(c_i), x - c_i \rangle + \phi(x)$ на ней лежит в точке пересечения луча $R(\tau) = c_i - \tau f_x(c_i) / \|f_x(c_i)\|$, $\tau \geq 0$ и сферы S . При этом, значение минимума определяется по формуле

$$H_i(\rho) = f(c_i) - \|f_x(c_i)\| \rho + \frac{1}{2} \lambda_{min} \rho^2 - \frac{M_i}{6} \rho^3.$$

Следовательно, максимальный радиус ρ_i шара S_i внутри которого выполняется неравенство $\overline{G}_i^2(x) \geq f_r - \epsilon$ является минимальным положительным корнем уравнения:

$$(16) \quad H_i(\rho) = f_r - \epsilon.$$

Существование положительного решения этого уравнения следует из непрерывности функции $H_i(\rho)$ и соотношений

$$H_i(0) = f(c_i) \geq f_r, \lim_{\rho \rightarrow +\infty} H_i(\rho) = -\infty.$$

Минимальный радиус ρ_i может быть легко получен с помощью формул Кардано или простым итерационным процессом. Параллелепипеды, радиусы которых меньше или равны ρ_i , не подвергаются дроблению и исключаются из дальнейшего рассмотрения.

3.3. Дополнительные правила отсева

Справедлива следующая

Лемма 1. Пусть функция $f(x)$ принадлежит классу C^1 , ее градиент удовлетворяет условию Липшица (10) с константой L и $\|f_x(c)\| \neq 0$. Тогда стационарная точка функции f может лежать только на границе или вне шара с центром в точке c и радиусом $\rho_m = \|f_x(c)\|/L$.

Доказательство. Предположим, что утверждение леммы не верно, т.е. существует стационарная точка \tilde{x} , такая что $\|\tilde{x} - c\| < \rho_m$. Применяя условие Липшица, имеем $\|f_x(c)\| \leq L\|c - \tilde{x}\| < L\rho_m = \|f_x(c)\|$. Получено противоречие и, тем самым, лемма доказана.

Согласно лемме 1 из параллелепипеда X_i можно исключить внутренность шара $S_i = \{x \in \mathbb{R}^n : \|x - c_i\| < \rho_i\}$, где

$$(17) \quad \rho_i = \|f_x(c_i)\|/L_i,$$

так как в нем нет точек минимума функции $f(x)$.

Обозначим через $\beta(X)$ границу множества X .

Условие отсева R1. Параллелепипед X_i исключаются из дальнейшего рассмотрения если

$$(18) \quad \rho(X_i) < \|f_x(c_i)\|/L_i \text{ и } X_i \cap \beta(X) = \emptyset.$$

Несложными алгебраическими преобразованиями показывается, что радиус покрывающего шара, определяемый формулой (17) больше радиуса, вычисляемого по формуле (12) в том и только том случае, когда справедливо неравенство

$$(19) \quad \|f_x(c_i)\| > \sqrt{2L_i(f(c_i) - f_r + \epsilon)}/3.$$

Неравенство (19) показывает, что правило отсева **R1** более эффективно по сравнению с правилом **R0** с минорантой $G_i^1(x)$ при относительно больших значениях градиента. Правило **R0** с минорантой $G_i^1(x)$ обеспечивает больший радиус покрывающего шара при больших ϵ и при $f(c_i) \gg f_r$. Таким образом, эти правила отсева взаимно дополняют друг друга и должны применяться совместно. В этом случае справедливо

Утверждение 1. Если параллелепипед X_i не удовлетворяет условию отсева **R0** с минорантой $G_i^1(x)$ и условию **R1**, то справедлива нижняя оценка радиуса параллелепипеда:

$$(20) \quad \rho(X_i) > \sqrt{\frac{2(f(c_i) - f_r + \epsilon)}{3L_i}} \geq \sqrt{\frac{2\epsilon}{3L_i}}.$$

Доказательство. Пусть параллелепипед не удовлетворяет условию отсева **R0** с минорантой $G_i^1(x)$. Используя (7) и (14), получим

$$f(c_i) - \langle |f_x(c_i)|, b_i - c_i \rangle - \frac{1}{2}L_i \|b_i - c_i\|^2 < f_r - \epsilon.$$

Применяя неравенство Коши-Буняковского, находим

$$\frac{1}{2}L_i \|b_i - c_i\|^2 + \|f_x(c_i)\| \cdot \|b_i - c_i\| + f_r - f(c_i) - \epsilon > 0.$$

Согласно предположению, для параллелепипеда X_i не выполняется неравенство (18), т.е. $\|f_x(c_i)\| \leq L_i \|b_i - c_i\|$. Следовательно,

$$\frac{3}{2}L_i \|b_i - c_i\|^2 + f_r - f(c_i) - \epsilon > 0.$$

Из этого соотношения и (6) непосредственно выводится неравенство (20). Утверждение доказано.

Условие R1 можно применять только по отношению к параллелепипедам, не имеющим общей границы с исходным параллелепипедом, либо если известно, что минимум достигается внутри исходного параллелепипеда. Последнее может иметь место, если решается задача безусловной оптимизации, а в качестве исходного выбирается параллелепипед, заведомо содержащий точку глобального экстремума. Если это не так, то требуется дополнительный анализ компонент градиента. Справедлива следующая

Лемма 2. Пусть $f(x) \in C^1$ и каждая j -я компонента ее градиента удовлетворяет условию Липшица с константой L^j :

$$\left| \frac{\partial f(x)}{\partial x^j} - \frac{\partial f(y)}{\partial x^j} \right| \leq L^j \cdot \|x - y\|, \quad j = 1, \dots, n.$$

Тогда если $\|f_x(c)\| \neq 0$, то стационарная точка функции $f(x)$ может лежать только вне или на границе шара $\{x \in \mathbb{R}^n : \|x - c\| \leq \rho\}$, где $\rho = \max_{1 \leq j \leq n} |\partial f(c)/\partial x^j|/L^j$.

Лемма 2 доказывается аналогично лемме 1.

Если x_* — точка минимума функции f на параллелепипеде X_i , то

$$\frac{\partial f(x_*)}{\partial x^j} = \begin{cases} = 0, & \text{если } a_i^j < x_*^j < b_i^j, \\ \geq 0, & \text{если } x_*^j = a_i^j, \\ \leq 0, & \text{если } x_*^j = b_i^j. \end{cases}$$

Если $|\partial f(c_i)/\partial x^j| > L_i^j \|c_i - a_i\|$, то компонента градиента $\partial f(c_i)/\partial x^j$ сохраняет знак на параллелепипеде X_i . Поэтому можно сформулировать

Условие отсева R2. Пусть $f(x) \in C^1$ и каждая j -я компонента ее градиента удовлетворяет условию Липшица с константой L_i^j на параллелепипеде X_i :

$$\left| \frac{\partial f(x)}{\partial x^j} - \frac{\partial f(y)}{\partial x^j} \right| \leq L_i^j \cdot \|x - y\|, \quad j = 1, \dots, n, x, y \in X_i.$$

Пусть также $|\partial f(c_i)/\partial x^j| > L_i^j \|c_i - a_i\|$. Тогда если $a_i^j > a^j$ и $\partial f(c_i)/\partial x^j > 0$, то параллелепипед X_i исключается из рассмотрения. Если $b_i^j < b^j$ и $\partial f(c_i)/\partial x^j <$

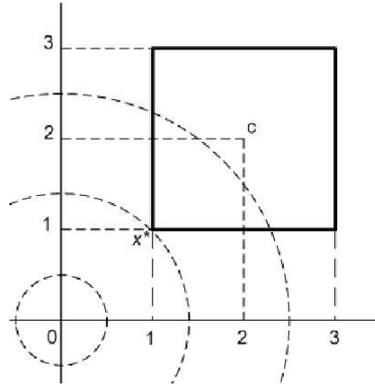


Рис. 1. Допустимая область $[1, 3] \times [1, 3]$ и линии уровня функции $f(x) = (x^1)^2 + (x^2)^2$.

0, то параллелепипед X_i также исключается из рассмотрения. Если $a_i^j = a^j$ и $\partial f(c_i)/\partial x^j > 0$, то минимум функции $f(x)$ может достигаться в точке x_* только при $x_*^j = a_i^j$. Из рассмотрения исключается область $\{[a_i^1, b_i^1] \times \dots \times [a_i^j, b_i^j] \times \dots \times [a_i^n, b_i^n]\}$. Аналогично, если $b_i^j = b^j$ и $\partial f(c_i)/\partial x^j < 0$, то минимум может достигаться в точке x_* только при $x_*^j = b_i^j$ и из рассмотрения исключается область $\{[a_i^1, b_i^1] \times \dots \times [a_i^j, b_i^j] \times \dots \times [a_i^n, b_i^n]\}$.

В случае $|\partial f(c_i)/\partial x^j| > L_i^j \|c_i - a_i\|$ условие отсева **R2** позволяет либо полностью отбросить параллелепипед X_i , когда его соответствующая граница не имеет общих точек с границей исходного параллелепипеда, либо сократить размерность исходного параллелепипеда в противном случае.

Применение условия **R2** в ряде случаев позволяет найти точный, а не приближенный минимум. В качестве иллюстрации рассмотрим задачу отыскания минимума функции $f(x) = (x^1)^2 + (x^2)^2$ на параллелепипеде $[1, 3] \times [1, 3]$. Линии уровня целевой функции и допустимая область представлены на рис. 1.

Минимум целевой функции достигается в вершине параллелепипеда $x_* = (1, 1)$. Применим к данному параллелепипеду условие отсева **R2**. Градиент функции $f(x)$ вычисляется по формуле $f_x(x) = (2x^1, 2x^2)$. Гессиан имеет вид $\begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$. Следовательно, константа Липшица для компонент градиента равняется 2. В центре параллелепипеда, точке $(2, 2)$, градиент принимает значение $(4, 4)$. Так как $2\|c - a\| = 2\sqrt{2} < 4$, то обе компоненты градиента сохраняют знак на параллелепипеде $[1, 3] \times [1, 3]$. В этом случае, в соответствии с условием **R2** можно положить $b^1 = a^1$ и $b^2 = a^2$. В результате получаем вырожденный параллелепипед $[1, 1] \times [1, 1]$, содержащий единственную точку $(1, 1)$, являющуюся решением задачи.

Для функций одной переменной можно ввести дополнительное правило отсева, основанное на следующей лемме

Лемма 3. Пусть вторая производная функции $f(x)$ удовлетворяет условию (15) Липшица с константой M : $|f_{xx}(x) - f_{xx}(y)| \leq M|x - y|$. Пусть $f_{xx}(c) \neq 0$. Обозначим $\rho = |f_{xx}(c)|/M$. Тогда если $f_{xx}(c) < 0$, то открытый интервал $(c - \rho, c + \rho)$ не содержит точек локального минимума функции $f(x)$. Если $f_{xx}(c) > 0$, то интервал $(c - \rho, c + \rho)$ содержит не более одной точки локаль-

ного минимума функции $f(x)$.

Доказательство. Из условия Липшицевости следует, что вторая производная сохраняет знак в пределах открытого интервала $(c - \rho, c + \rho)$. Следовательно, если $f_{xx}(c) < 0$, то функция вогнута в пределах этого интервала и, следовательно, не может принимать на нем минимального значения. Если $f_{xx}(c) > 0$, то на интервале $(c - \rho, c + \rho)$ функция выпукла, и, следовательно, может иметь не более одной точки локального минимума.

Из доказанной леммы следует

Условие отсева R3. Пусть $f(x) \in C^2$ — функция одной переменной и ее вторая производная удовлетворяет на отрезке $X_i = [a_i, b_i]$ условию Липшица с константой M_i : $|f_{xx}(x) - f_{xx}(y)| \leq M_i|x - y|$, $x, y \in X_i$. Положим $\rho_i = |f_{xx}(c_i)|/M_i$. Тогда, если $f_{xx}(c_i) < 0$, то интервал $(c_i - \rho_i, c_i + \rho_i)$ можно исключить из рассмотрения. Если $f_{xx}(c_i) > 0$, то интервал $(c_i - \rho_i, c_i + \rho_i)$ можно исключить из рассмотрения после того, как найден единственный локальный экстремум внутри него (если таковой имеется).

Для нахождения локального экстремума можно применять любой из алгоритмов локальной оптимизации (методы первого или второго порядка).

3.4. Определение констант Липшица

Рассмотренные в предыдущих разделах миноранты условия отсева опираются на значения констант Липшица для целевой функции и для ее градиента. Точное определение этих констант является оптимизационной задачей, возможно превосходящей по сложности исходную задачу. Но рассматриваемый алгоритм будет корректно работать при любом значении заданных величин, превосходящих минимальные.

Константу Липшица для функции можно сверху оценить как максимум нормы градиента на заданном параллелепипеде. Такая оценка может быть получена методами интервального анализа или другим образом. Один из способов получения оценки константы Липшица для градиента состоит в получении верхней оценки для нормы Гессiana. Эту оценку можно получить исходя из следующего известного соотношения для нормы матрицы: $\|(a_{ij})\| \leq \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{ij}|$. Тогда если известны верхние оценки модулей элементов Гессiana на параллелепипеде X_i : $\max_{x \in X_i} \left| \frac{\partial f(x)}{\partial x^j \partial x^k} \right| \leq G_i^{jk}$, то можно положить

$$(21) \quad L_i = \max_{1 \leq j \leq n} \sum_{k=1}^n |G_i^{jk}|.$$

В качестве константы Липшица для компонент $\partial f(x)/\partial x^j$ градиента функции на параллелепипеде X_i можно выбрать $\max_{x \in X_i} \left\| \nabla \frac{\partial f(x)}{\partial x^j} \right\|$. Т.е. можно положить $L_i^j = \sqrt{\sum_{k=1}^n (G_i^{jk})^2}$. Конкретные примеры вычисления констант Липшица приводятся в разделе, посвященном результатам эксперимента.

Константа Липшица для Гессiana легко оценивается в случае функции одной переменной как максимум абсолютной величины третьей производной: $M_i = \max_{x \in X_i} |f_{xxx}(x)|$.

При решении практических задач получение точных верхних оценок констант Липшица либо затруднено, либо найденные значения не позволяют найти минимум за приемлемое время. Тогда можно применить следующий подход. Первоначально значение константы Липшица полагается заведомо малым. С этим значением выполняется расчет и запоминается найденный приближенный глобальный минимум. Далее значение константы Липшица увеличивается и расчет повторяется. Полученное при этом новое значение глобального минимума, как правило, улучшает ранее найденное. Этот процесс повторяется до момента, ограниченного выделенными вычислительными и временными ресурсами. Такой подход хотя и не гарантирует оптимальности найденного решения, но в ряде практических задач позволяет успешно находить оптимум или хорошее приближение к нему [1-3]. Более сложные варианты адаптивного определения констант Липшица, основанные на стохастической модели, можно найти в работе [8].

3.5. Разбиение параллелепипеда

3.5.1. Общий случай. Если параллелепипед не может быть исключен из списка $\{X_i\}$ на основании рассмотренных выше правил отсева, то он дробится на более мелкие параллелепипеды. Это процесс может быть реализован различными способами. Наиболее употребительным является дихотомическое разбиение по наибольшему ребру при котором параллелепипед $X_i = \{x \in \mathbb{R}^n : a_i \leq x \leq b_i\}$ разбивается на два параллелепипеда $X_{i+1} = \{x \in \mathbb{R}^n : a_i \leq x \leq s_i\}$ и $X_{i+2} = \{x \in \mathbb{R}^n : t_i \leq x \leq b_i\}$. Векторы s_i и t_i определяются соотношениями:

$$s_i^j = \begin{cases} b_i^j, & \text{при } j \neq k, \\ c_i^j & \text{при } j = k, \end{cases} \quad t_i^j = \begin{cases} a_i^j, & \text{при } j \neq k, \\ c_i^j & \text{при } j = k, \end{cases}$$

где k — номер самого длинного ребра параллелепипеда X_i . Такой способ разбиения обеспечивает максимальное сокращение главной диагонали параллелепипеда.

Возможны также другие способы декомпозиции параллелепипеда: дихотомическое по ребру с максимальным значением компоненты градиента, на несколько параллелепипедов и др. Все алгоритмы, рассматриваемые ниже основаны на дихотомическом разбиении по максимальному ребру.

3.5.2. Функции одной переменной. В случае минимизации функции одной переменной параллелепипеды являются отрезками. Это позволяет более эффективно организовать процесс отсева подмножеств. Пусть $Z_i \subseteq X_i$ — отрезок, который можно исключить из рассмотрения. Непустое дополнение $X_i \setminus Z_i$ представляет собой один или два отрезка. Таким образом, в случае функции одной переменной не требуется разбивать отрезок пополам, так как это разбиение происходит в результате исключения некоторой окрестности центральной точки отрезка.

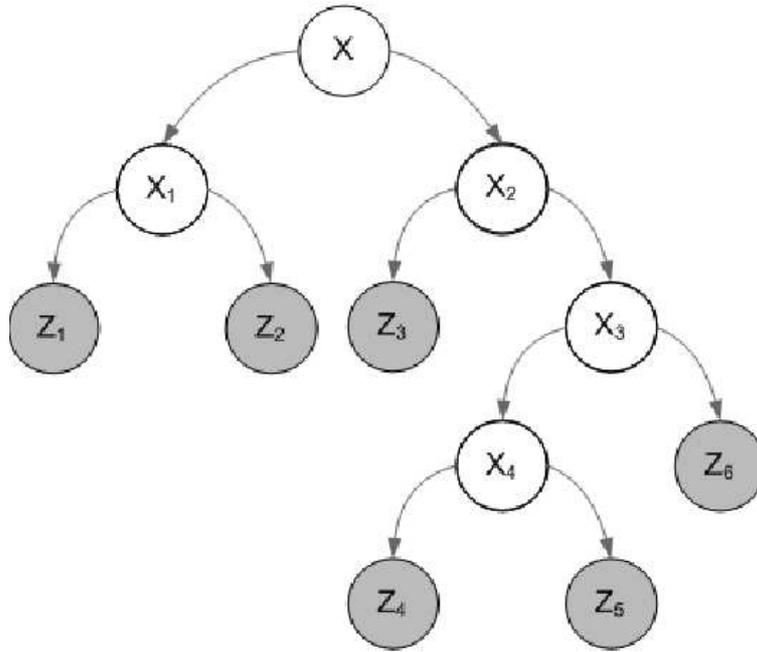


Рис. 2. Пример дерева ветвления.

4. Оценка трудоемкости метода

Рассмотрим вариант метода неравномерных покрытий, при котором производится дихотомическое разбиение параллелепипедов по максимальному ребру. Схематически работу алгоритма можно представить в виде ориентированного бинарного дерева (*дерева ветвления*), в котором дуги соединяют разбиваемый параллелепипед с параллелепипедами, построенными в результате разбиения (рис. 2). Концевыми вершинами такого дерева являются элементы последовательности $\{Z_i\}$, которые образуют покрытие допустимого множества X . Трудоемкость работы алгоритма характеризуется количеством вершин в дереве ветвления. Так как дерево бинарное, то общее число V вершин в нем связано с числом V_t концевых вершин соотношением $V = 2V_t - 1$. Поэтому в качестве меры трудоемкости можно взять число концевых вершин, которое совпадает с числом элементов в покрывающей последовательности $\{Z_i\}$.

Утверждение 2. Пусть $\rho_* > 0$ — нижняя граница радиуса параллелепипеда из множества $\{X_i\}$ внутренних (не концевых) вершин дерева ветвления. Пусть ρ_0 — радиус допустимого параллелепипеда X . Тогда число элементов в последовательности $\{Z_i\}$ не превосходит величины:

$$(22) \quad U(\rho_*, \rho_0, n) = 2(\rho_*/\rho_0)^{\theta(n)},$$

где $\theta(n) = 2/(\log_2(1 - \frac{3}{4n}))$.

Доказательство. Рассмотрим произвольный n -мерный параллелепипед $P = [a^1, b^1] \times \dots \times [a^n, b^n]$ с радиусом $\rho(P) = \frac{1}{2}\sqrt{(b^1 - a^1)^2 + \dots + (b^n - a^n)^2}$. Пусть j — номер самого длинного ребра параллелепипеда P . Радиус полученного в результате декомпозиции параллелепипеда P' вычисляется по следую-

щей формуле:

$$\rho(P') = \frac{1}{2} \sqrt{(b^1 - a^1)^2 + \dots + \frac{(b^j - a^j)^2}{4} + \dots + (b^n - a^n)^2}.$$

Так как $|b^j - a^j| \geq |b^i - a^i|$ для любого $1 \leq i \leq n$, то

$$\begin{aligned} \rho(P') &= \frac{1}{2} \sqrt{4\rho(P)^2 - \frac{3}{4}(b^j - a^j)^2} \leq \\ &\leq \frac{1}{2} \sqrt{4\rho(P)^2 - \frac{3}{4n}4\rho(P)^2} = \rho(P) \sqrt{1 - \frac{3}{4n}}. \end{aligned}$$

Следовательно, $\rho(P')/\rho(P) \leq \sqrt{1 - \frac{3}{4n}}$, минимально возможное уменьшение диагонали при декомпозиции составляет величину $\alpha_n = \sqrt{1 - \frac{3}{4n}}$. Таким образом, если некоторый параллелепипед Q соединен с корнем дерева путем длиной m , то $\rho(Q) \leq \alpha_n^m \rho_0$. По условию доказываемого утверждения если $Q \in \{X_i\}$, то $\alpha_n^m \rho_0 \geq \rho(Q) \geq \rho_*$. Следовательно, максимальная длина пути от корня дерева ветвления до конечной вершины не превосходит $\log_{\alpha_n}(\rho_*/\rho_0) + 1$.

Так как в рассматриваемом случае дерево ветвления — бинарное, то число конечных вершин (элементов последовательности $\{Z_i\}$) в нем не превосходит $2^{\beta(n)}$, где

$$\begin{aligned} \beta(n) &= \log_{\alpha_n}(\rho_*/\rho_0) + 1 = \log_2(\rho_*/\rho_0) / \log_2 \alpha_n + 1 = \\ &= \log_2(\rho_*/\rho_0) / \log_2(1 - 3/(4n)) + 1. \end{aligned}$$

Подставляя вместо $\beta(n)$ найденное выражение, получаем доказываемое утверждение.

Приведем приближенный вид функции $U(\rho_*, \rho_0, n)$ для $n = 1, 2, 3$:

$$\begin{aligned} U(\rho_*, \rho_0, 1) &= 2(\rho_0/\rho_*), U(\rho_*, \rho_0, 2) \simeq 2(\rho_0/\rho_*)^{2.95}, \\ U(\rho_*, \rho_0, 3) &\simeq 2(\rho_0/\rho_*)^{4.87}. \end{aligned}$$

Оценим асимптотическое поведение функции $U(\rho_*, \rho_0, n)$ при росте n . Заметим, что $\log_2(1 - \frac{3}{4n}) = \frac{1}{\ln 2} \ln(1 - \frac{3}{4n}) \simeq -\frac{1}{\ln 2} \frac{3}{4n}$. Поэтому, при $n \rightarrow \infty$ $U(\rho_*, \rho_0, n) \simeq 2(\rho_0/\rho_*)^{\frac{8 \cdot \ln 2}{3} n} \simeq 2(\rho_0/\rho_*)^{1.84 \cdot n}$. Полученное соотношение показывает, что с ростом n верхняя оценка экспоненциально растет.

Перейдем теперь к оценке скорости сходимости для конкретных минорант и правил отсева. Для этого надо подставить нижние оценки (9), (20) в формулу (22). В результате получается следующее утверждение.

Пусть ϵ — заданная точность вычислений. Пусть l_0 — константа Липшица для целевой функции на допустимом множестве X . Тогда трудоемкость метода неравномерных покрытий при применении миноранты $G_i^0(x)$ ограничена сверху величиной $U_1(\epsilon, l_0, \rho_0, n)$, асимптотическое поведение которой выражается формулой:

$$(23) \quad U_1(\epsilon, l_0, \rho_0, n) \simeq 2(l_0 \rho_0 / \epsilon)^{1.84 \cdot n}.$$

Пусть L_0 — константа Липшица для градиента целевой функции на допустимом множестве X . Тогда трудоемкость метода неравномерных покрытий при применении миноранты $G_i^1(x)$ и дополнительного правила отсева **R1**, ограничено сверху следующей величиной:

$$(24) \quad U_2(\epsilon, L_0, \rho_0, n) \simeq 2(3\rho_0^2 L_0 / 2\epsilon)^{0.92 \cdot n}.$$

Формулы (23) и (24) показывают, что верхняя оценка трудоемкости определяется следующими факторами:

- заданная точность;
- размерность переменных;
- значения констант Липшица;
- радиус допустимого параллелепипеда X .

Так как показатель степени в асимптотической формуле для величины U_2 вдвое меньше, чем для U_1 , то с ростом n верхняя оценка трудоемкости при использовании миноранты $G_i^1(x)$ существенно медленнее растет, чем при использовании миноранты $G_i^0(x)$. Из полученных оценок также следует, что при ограниченных сверху константах Липшица метод неравномерных покрытий решает задачу за конечное количество шагов.

5. Реализация метода неравномерных покрытий для многопроцессорных вычислительных систем

Полученные в предыдущих пунктах верхние оценки и данные вычислительных экспериментов показывают, что с ростом размерности задачи трудоемкость может существенно расти и зачастую для решения оптимизационных задач, возникающих на практике, нехватает мощности одной рабочей станции. В этом случае целесообразно переходить к расчетам на многопроцессорных вычислительных системах. Далее в разделе подробно рассматривается реализация метода неравномерных покрытий для многопроцессорных вычислительных систем.

Метод неравномерных покрытий имеет две основные составляющие: построение покрытия $\{Z_i\}$ и нахождение рекордной точки. Рассмотренная в предыдущих пунктах единая общая схема алгоритма включает в себя несколько процедур, которые можно рассматривать в качестве параметров:

1. процедура вычисления миноранты;
2. процедура отсева;
3. способ разбиения параллелепипеда.

Такая структура метода позволяет существенно сократить временные затраты на реализацию. Общая схема вычислений реализуется один раз, прочие процедуры подставляются в нее в качестве параметров.

Для нахождения рекордной точки могут применяться различные варианты эвристических оптимизационных алгоритмов. Базовая схема нахождения рекорда состоит в построении последовательности x_1, x_2, \dots, x_l допустимых точек и выбора минимального из значений целевой функции в этих точках:

$$f_r = \min_{i=1, \dots, l} f(x_i).$$

Для получения рекордов максимально близких к оптимуму можно применять различные алгоритмы локальной оптимизации: методы сопряженных градиентов, методы Ньютона, basin-hopping[5]. В этом случае формула для нахождения рекорда примет вид:

$$f_r = \min_{i=1,\dots,l} f(L_s(x_i)),$$

где L_s — алгоритм локальной оптимизации, отыскивающий минимум функции f , начиная с точки x_i .

В качестве точек x_1, x_2, \dots, x_l могут выбираться центры параллелепипедов последовательности $\{X_i\}$. Возможны также варианты, при которых процесс нахождения рекордной точки полностью независим от процесса построения покрывающей последовательности. Например, точки x_1, x_2, \dots, x_l могут выбираться случайно в исходном параллелепипеде.

Метод неравномерных покрытий работает в соответствии с общей схемой метода ветвей и границ. Поэтому для его реализации была выбрана библиотека BNB-Solver[6], предназначенной для решения задач методами ветвей и границ на многопроцессорных системах с распределенной памятью. Эта объектно-ориентированная библиотека имеет модульную параметризованную структуру. Потребовалось реализовать только алгоритмы-параметры, перечисленные выше. Логика последовательных и параллельных вычислений была реализована ранее при решении дискретных оптимизационных задач и не подвергалась модификации.

Схема организации параллельных вычислений в библиотеке BNB-Solver представлена на рис. 3. Введены следующие обозначения: M - процесс, управляющий работой всего приложения; B_i - процессы, которые выполняют операции метода ветвей и границ (в рассматриваемом случае строят покрытие); BM - процесс, координирующий работу процессов B_i ; H_i - процессы, которые выполняют построение последовательности $\{x_i\}$ и нахождение рекордной точки; HM - процесс, координирующий их работу.

Процесс BM поддерживает список параллелепипедов, который в начальный момент состоит только из допустимого параллелепипеда X . Выполнив несколько шагов метода неравномерных покрытий, он переходит к рассылке полученных при этом параллелепипедов по процессам B_i . Если список не пуст и рабочий процесс свободен, то BM пересылает ему параллелепипед. Работа процесса B_i управляется двумя параметрами — T_i и S_i . Получив параллелепипед для обработки, процесс выполняет заданное число разбиений (T_i), после чего пересылает некоторое число (S_i) параллелепипедов на управляющий процесс и продолжает расчеты. По окончании покрытия процесс B_i посылает процессу BM соответствующее сообщение, получив которое, процесс BM посылает B_i новый параллелепипед. С помощью параметров T_i и S_i процесс BM управляет интенсивностью поступления подзадач от рабочих процессов. Вычисления завершаются, когда список параллелепипедов на процессе BM оказался пустым и все процессы B_i закончили вычисления.

Поиск рекордной точки производится следующим образом. Процесс HM принимает допустимые точки от процессов H_i и B_i . Получив очередную допустимую точку. Процесс HM рассылает полученную точку всем либо некоторым (в зависимости от выбранного алгоритма) процессам H_i , выполняющим

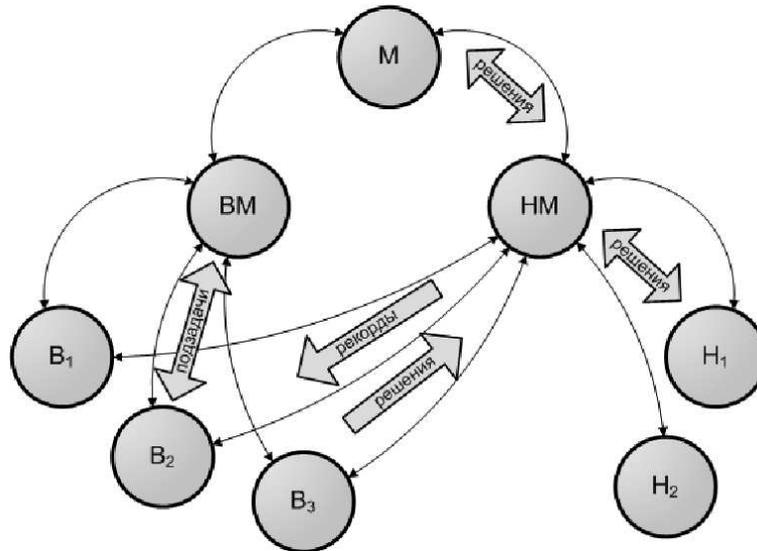


Рис. 3. Организация параллельных вычислений.

локальную оптимизацию. Если значение целевой функции в полученной точке меньше рекорда, то рекорд и рекордное решения обновляются.

В рассматриваемой схеме возможны различные варианты распределения процессоров между группами B_i и H_i и различные схемы взаимодействия этих групп. Например, если последовательность $\{x_i\}$ состоит из центров параллелепипедов, создаваемых при построении покрытия, то процессы B_i периодически пересылают их процессу $НМ$. Возможен также вариант при котором процесс $НМ$ не использует информации от процессов B_i . Такая схема имеет место, когда группа, состоящая из процессов $НМ$ и H_i , автономно реализует некоторый эвристический алгоритм. Пример такой реализации алгоритма basin-horping рассматривается в работе [7].

6. Результаты экспериментов

6.1. Результаты, полученные на однопроцессорной рабочей станции

Первая серия экспериментов была направлена на изучение эффективности применения минорант и дополнительных правил отсева и проводилась на рабочей станции Pentium IV 2.8 GHz.

6.1.1. Функции одной переменной. Функции одной переменной являются важным частным случаем, для которого предлагаемые методы работают особенно быстро, в силу более эффективной организации отсева и возможности использования миноранты G^3 и правила отсева **R3**. В табл. 6.1.1. приведены функции, для которых проводились эксперименты. Функции взяты из работы [9]. В качестве допустимого множества для всех примеров был выбран отрезок $[-10, 10]$, вычисления проводились с точностью $\epsilon = 10^{-6}$. Верхние оценки констант Липшица были получены методами интервального анализа.

В таблице 6.1.1. приводятся значения трудоемкости (числа вершин в дереве ветвления) для различных минорант и правил отсева.

Таблица 1. Функции для тестирования

№	функция	x_*	f_*
1	$3x^4 - 16x^3 + 18x^2$	3	27
2	$x^6 - 15x^4 + 27x^2 + 250$	3	7
3	$x^4 - 10x^3 + 35x^2 - 50x + 24$	1.3819	-1
4	$\sin(x)$	7.8579	-1

Таблица 2. Результаты экспериментов для различных минорант и правил отсева

№	G_i^0	G_i^1	$G_i^1 + R1$	G_i^2	$G_i^2 + R1$	$G_i^2 + R1 + R3$
1	129	29	25	12	12	9
2	139	57	41	19	19	17
3	203	45	35	14	14	10
4	65	35	13	13	11	10

При анализ данных таблицы 6.1.1. необходимо учитывать, что применение правила отсева **R0** с минорантой G_i^1 требует приблизительно вдвое, а с минорантой G_i^2 — втрое больше вычислений, чем в случае миноранты G_i^0 . Тем не менее на основании проведенных экспериментов можно сказать, что для рассмотренных функций использование минорант более высокого порядка дает заметное ускорение. Заметим, что применение дополнительных правил отсева **R1** и **R3** практически не увеличивает вычислительные затраты, т.к. не требует дополнительных вычислений значений функции и оценок, поэтому их применение оправдано, несмотря на относительно незначительное сокращение количества шагов алгоритма.

6.1.2. Обобщенная функция Розенброка. Обобщенная функция Розенброка является классическим тестовым примером для алгоритмов оптимизации:

$$f(x) = \sum_{i=1}^{n-1} (100 \cdot ((x^i)^2 - (x^{i+1})^2 + (x^i - 1)^2).$$

Эта функция имеет в точности один глобальный минимум в пространстве \mathbb{R}^n , который достигается в точке x_* , где $x_*^1 = 1, \dots, x_*^n = 1$. Получим оценки констант Липшица для обобщенной функции Розенброка и ее градиента. Компоненты градиента вычисляются по формуле:

$$\frac{\partial f(x)}{\partial x^i} = \alpha_{in} \cdot (400(x^i)^3 - 400x^{i+1}x^i + 2x^i - 2) + \alpha_{i1} \cdot (200(x^i - (x^{i-1})^2)),$$

$$i = 1, 2, \dots, n,$$

где $\alpha_{ij} = \begin{cases} 0, & i = j; \\ 1, & i \neq j. \end{cases}$

Рассмотрим параллелепипед $X_j = \{x \in \mathbb{R}^n : a_j \leq x \leq b_j\}$. Обозначим через m_j^i величину $\max_{x \in X_j} |x^i| = \max(|a_j^i|, |b_j^i|)$. Тогда,

$$\begin{aligned} \max_{x \in X_j} \left| \frac{\partial f(x)}{\partial x^i} \right| &\leq \\ &\leq \alpha_{in} \cdot (400(m_j^i)^3 + 400m_j^{i+1}m_j^i + 2m_j^i - 2) + \alpha_{i1} \cdot (200(m_j^i + (m_j^{i-1})^2)). \end{aligned}$$

Полученное выражение позволяет производить оценку сверху максимума нормы градиента, и, соответственно, константы Липшица для функции.

Для получения оценки константы Липшица для градиента получим оценку нормы Гессиана. Гессиан рассматриваемой функции является трехдиагональной матрицей, компоненты которой имеют вид:

$$\begin{aligned} \frac{\partial^2 f(x)}{\partial (x^i)^2} &= \alpha_{in} \cdot (1200(x^i)^2 - 400x^{i+1} + 2) + \alpha_{i1} \cdot 200, \\ \frac{\partial^2 f(x)}{\partial x^i \partial x^{i-1}} &= -400x^{i-1}. \end{aligned}$$

Для абсолютных величин компонент Гессиана справедливы следующие оценки:

$$\begin{aligned} \left| \frac{\partial^2 f(x)}{\partial (x^i)^2} \right| &\leq \max(\alpha_{in} \cdot (1200(m_j^i)^2 - 400a_j^{i+1} + 2) + \alpha_{i1} \cdot 200, \\ &\quad \alpha_{in} \cdot (1200(m_j^i)^2 - 400b_j^{i+1} + 2) + \alpha_{i1} \cdot 200), \\ \left| \frac{\partial^2 f(x)}{\partial x^i \partial x^{i-1}} \right| &\leq 400 \cdot m_j^{i-1}. \end{aligned}$$

Пользуясь этими неравенствами и соотношением (21), можно получить оценки константы Липшица для градиента рассматриваемой функции.

Первая серия экспериментов была направлена на поиск минимума обобщенной функции Розенброка на параллелепипеде $X = [-n, n]^n$. Вычисления проводились в предположении, что минимум достигается во внутренней точке параллелепипеда. В таблице 3 приводится трудоемкость (число вершин в дереве ветвления) и в скобках — время расчета в секундах. Во всех экспериментах использовалось правило отсева R0. При этом использование правила отсева R0 с минорантой $G_i^0(x)$ не позволяет отыскивать минимум за приемлемое время при малых ϵ . Существенно лучший результат достигается при применении миноранты $G_i^1(x)$. Использование условия отсева R1 ускоряет работу алгоритма еще приблизительно в 1.5 раза. Применение условия R2 дает увеличение скорости работы на два порядка.

Второй пример состоит в отыскании минимума обобщенной функции Розенброка на параллелепипеде $[-n, 0]^n$. Так как компоненты градиента отрицательны на этом параллелепипеде, то минимум достигается в вершине параллелепипеда, точке x_* , $x_*^1 = \dots = x_*^n = 0$. Из таблицы 4 видно, что применение правила отсева R2 позволяет радикально сократить перебор. При этом число вершин в дереве ветвления не зависит от заданной точности, так как найденный минимум является точным.

Таблица 3. Трудоемкость (число вершин в дереве ветвления) и время расчетов (сек.) для обобщенной функции Розенброка на параллелепипеде $[-n, n]^n$ при применении различных условий отсева в методе неравномерных покрытий.

задача	условия отсева	$\epsilon = 0.1$	$\epsilon = 0.01$	$\epsilon = 0.001$
$n = 2$	G_i^0	4927255 (46.08)	-	-
	G_i^1	2717 (0.04)	4637 (0.03)	6613 (0.05)
	$G_i^1 + R1$	1809 (0.03)	2963 (0.02)	4145 (0.04)
	$G_i^1 + R2$	389 (0.01)	613 (0.01)	887 (0.01)
$n = 3$	G_i^1	40963 (0.36)	60233 (0.48)	86383 (0.82)
	$G_i^1 + R1$	30285 (0.25)	47247 (0.38)	61487 (0.51)
	$G_i^1 + R2$	893 (0.02)	1127 (0.02)	1183 (0.03)
$n = 4$	G_i^1	447221 (3.81)	571713 (4.95)	689793 (5.74)
	$G_i^1 + R1$	342179 (2.98)	414177 (3.54)	478515 (4.09)
	$G_i^1 + R2$	6321 (0.05)	7787 (0.06)	9205 (0.08)

6.1.3. Перенос заряда в ДНК. Следующая задача решается при расчете переноса заряда в молекуле ДНК [10]. Данный пример демонстрирует применение метода неравномерных покрытий для задач с ограничениями типа равенства. Для фрагмента цепочки ДНК длины n требуется решить задачу оптимизации с ограничением типа равенства:

$$(25) \quad \begin{aligned} f(x) &= \sum_{i=1}^n \sum_{j=1}^n \eta_{ij} x^i x^j - \frac{1}{2} \sum_{i=1}^n \kappa_i (x^i)^4 \rightarrow \min, \\ \sum_{i=1}^n (x^i)^2 &= 1, \text{ где} \\ \eta_{ij} &= \eta_{ji} \geq 0 \text{ при } 1 \leq i, j \leq n, \\ \eta_{ij} &= 0 \text{ при } |i - j| > 1, \\ \kappa_i &> 0 \text{ при } 1 \leq i \leq n. \end{aligned}$$

Заметим, что все точки, удовлетворяющие ограничению задачи, лежат внутри n -мерного параллелепипеда $X = [-1, 1]^n$, который выберем в качестве допустимого множества. Введем дополнительное правило отсева для учета ограничения задачи: исключаем параллелепипед, не имеющих общих точек со сферой $\|x\| = 1$. Это условие допускает эффективную проверку. Так как равенство нулю градиента не является необходимым условием минимума в случае задач с ограничением, то правила **R1** и **R2** неприменимы в рассматриваемом примере. В остальном метод неравномерных покрытий работает обычным образом.

Компоненты градиента и гессиана функции f вычисляются следующим образом:

$$\frac{\partial f(x)}{\partial x^i} = 2 \sum_{j=1}^n \eta_{ij} x^j - 2\kappa_i (x^i)^3, \quad \frac{\partial^2 f(x)}{\partial x^i \partial x^j} = 2\eta_{ij} - 6\rho_{ij}\kappa_i (x^i)^2,$$

где $\rho_{ij} = \begin{cases} 0, & i \neq j \\ 1, & i = j. \end{cases}$

Таблица 4. Трудоемкость (число вершин в дереве ветвления) и время расчетов (сек.) для обобщенной функции Розенброка на параллелепипеде $[-n, 0]^n$ при применении различных правил отсева.

задача	условия отсева	$\epsilon = 0.1$	$\epsilon = 0.01$	$\epsilon = 0.001$
$n = 2$	G_i^0	173 (0.01)	205 (0.01)	233 (0.01)
	G_i^1	153 (0.01)	213 (0.01)	301 (0.01)
	$G_i^1 + R1$	147 (0.01)	235 (0.01)	325 (0.01)
	$G_i^1 + R2$	17 (0.0)	17 (0.0)	17 (0.0)
$n = 3$	G_i^0	1575 (0.02)	1907 (0.02)	3239 (0.03)
	G_i^1	981 (0.01)	1173 (0.02)	1983 (0.02)
	$G_i^1 + R1$	1441 (0.03)	1491 (0.03)	14939 (0.14)
	$G_i^1 + R2$	25 (0.0)	25 (0.0)	25 (0.0)
$n = 4$	G_i^0	19277 (0.17)	23489 (0.23)	731653 (6.37)
	G_i^1	10199 (0.09)	18465 (0.18)	46401 (0.43)
	$G_i^1 + R1$	6375 (0.05)	7533 (0.06)	36769 (0.28)
	$G_i^1 + R2$	59 (0.0)	59 (0.0)	59 (0.0)

Оценим константу Липшица для функции f на параллелепипеде X_k :

$$l_k \leq \max_{x \in X_k} \|\nabla f(x)\| \leq \max_{x \in X_k} \sqrt{\sum_{i=1}^n \left(2 \sum_{j=1}^n \eta_{ij} x^j - 2\kappa_i (x^i)^3 \right)^2}.$$

Верхняя оценка выражения в правой части может быть вычислена методами интервального анализа [11].

Оценим константу Липшица для градиента функции. Максимум модуля элемента Гессiana на параллелепипеде X_k можно оценить следующим образом:

$$\begin{aligned} \max_{x \in X_k} \left| \frac{\partial^2 f(x)}{\partial x^i \partial x^j} \right| &= \max_{x \in X_k} |2\eta_{ij} - 6\rho_{ij}\kappa_i(x^i)^2| = \\ &= \begin{cases} 2|\eta_{ij}|, & \text{при } i \neq j, \\ \max_{x^i \in [a_k^i, b_k^i]} |2\eta_{ii} - 3\kappa(x^i)^2|. \end{cases} \end{aligned}$$

При этом

$$\begin{aligned} \max_{x^i \in [a_k^i, b_k^i]} |\eta_{ii} - 3\kappa x_i^2| &= \\ &= \begin{cases} \max(|\eta_{ii}|, |\eta_{ii} - 3\kappa a_k^i|^2, |\eta_{ii} - 3\kappa (b_k^i)^2|), & \text{при } 0 \in [a_k^i, b_k^i], \\ \max(|\eta_{ii} - 3\kappa (a_k^i)^2|, |\eta_{ii} - 3\kappa (b_k^i)^2|), & \text{при } 0 \notin [a_k^i, b_k^i]. \end{cases} \end{aligned}$$

Оценка константы Липшица для градиента может быть легко получена на основании формулы (21).

Трудоемкость и время решения (в скобках) задачи (25) для трех цепочек ДНК длиной 4,6 и 8 приведены в таблице (5). Так как правила отсева **R1** и **R2** неприменимы, то сравниваются только результаты для минорант

G_i^0 и G_i^1 . Приведенные данные показывают, что применение миноранты G_i^1 позволяет существенно ускорить процесс вычислений и это ускорение растет с увеличением размерности задачи.

Таблица 5. Трудоемкость (число вершин в дереве ветвления) и время расчетов (сек.) для задачи о переносе заряда в ДНК при применении различных правил отсева.

задача	условия отсева	$\epsilon = 0.5$	$\epsilon = 0.1$
actg	G_i^0	857 (0.12)	5463 (0.22)
	G_i^1	497 (0.05)	3843 (0.12)
actgac	G_i^0	26823 (0.7)	248749 (5.24)
	G_i^1	12325 (0.33)	112929 (3.36)
actgactg	G_i^0	2420151 (47.83)	-
	G_i^1	945079 (14.87)	43294105 (771.3)
задача	условия отсева	$\epsilon = 0.01$	$\epsilon = 0.001$
actg	G_i^0	100447 (2.3)	2927629 (59.6)
	G_i^1	88009 (1.7)	2659613 (39.5)
actgac	G_i^0	26676869 (434.52)	-
	G_i^1	8656429 (147.1)	-
actgactg	G_i^0	-	-
	G_i^1	-	-

6.2. Результаты, полученные на параллельных системах

При росте числа переменных функции наблюдается значительный рост времени счета. Поэтому для решения задачи за приемлемое время необходимо использовать многопроцессорные вычислительные комплексы. В таблице 6 приведено время решения задачи для обобщенной функции Розенброка с числом переменных 5 и точностью 0.001 на суперкомпьютере МВС-6000IM для различного числа процессоров:

Время решения сокращается практически пропорционально числу рабочих процессоров, что говорит о высокой эффективности распараллеливания.

В рассмотренных тестовых примерах использовались точные верхние оценки констант Липшица. На практике часто получить такие оценки затруднительно, либо значения оценок настолько велики, что метод работает неприемлемо долго. В таких случаях можно оценивать константы эвристически. В работе [3] показано, что такой подход позволяет численно находить решение

Таблица 6. Зависимость времени работы метода неравномерных покрытий от числа процессоров для обобщенной функции Розенброка.

число процессоров	1	2	4	8	16	32
время работы (сек.)	28,02	14,23	7,13	3,94	2,51	1,82

задачи поиска конфигурации молекулы с минимальной энергией [5]. Эта задача заключается в отыскании минимума функции потенциальной энергии взаимодействия N атомов, задаваемой формулой: $F(x) = \sum_{i=1}^N \sum_{j=i+1}^N v(r_{ij})$, где $x \in \mathbb{R}^{3N}$ — вектор декартовых координат атомов в трехмерном пространстве, r_{ij} — расстояние между атомами i и j , а $v(r)$ — потенциальная энергия взаимодействия двух атомов, находящихся на расстоянии r (потенциал).

Для решения этой задачи был разработан и реализован параллельный эвристический алгоритм, основанный на идеях метода монотонного случайного спуска (Monotonic Basin-Hopping) [5]. Алгоритм начинает работу с некоторой точки x_0 локального минимума целевой функции. Формируется список точек, состоящий на первом шаге только из одного элемента — точки x_0 . На очередном шаге из списка выбирается точка x с наименьшим среди элементов списка значением целевой функции. Далее заданное число раз выполняется следующая последовательность действий: точка x подвергается возмущению и в результате последующей локальной оптимизации получается точка x' . Если удастся улучшить значение целевой функции, то есть если $f(x') < f(x)$, то точка x' добавляется в список. Выбор точек из списка производится до момента выполнения условия останова, которое может быть задано различными способами. Выбранные точки помечаются для того чтобы избежать их повторного обхода.

Рассмотренный алгоритм допускает эффективную параллельную реализацию, так как просмотр окрестности можно выполнять различными процессорами независимо. Вычисления удобно организовать по схеме управляющий-рабочие: один из процессоров, называемый управляющим, поддерживает список точек, остальные процессоры выполняют поиск локальных минимумов. В результате вычислительных экспериментов были получены все наиболее сложные оптимальные конфигурации атомных кластеров для потенциалов Морса и Леннарда-Джонса, что не удавалось ранее сделать без применения специальных алгоритмов, учитывающих геометрическую природу задачи.

В таблице 7 приведены найденные минимумы энергии f^* для наиболее сложных конфигураций с потенциалом Морса $v(r) = e^{\rho(1-r)} (e^{\rho(1-r)} - 2)$, $\rho = 10$ и 14 . Для расчетов использовались 256 процессоров суперкомпьютера MVS-50000BM. Число атомов N выбиралось равным 80, 85, 90, что соответствовало задачам минимизации в пространствах размерностей 240, 255 и 270 соответственно. В таблице приведено время за которое был получен окончательный результат t_1 и общее время расчетов t_2 (в минутах):

Список литературы

1. Евтушенко Ю.Г. Численный метод поиска глобального экстремума функций (перебор на неравномерной сетке) // ЖВМиМФ. 1971. Т. 11, № 6. С. 1390-1403.
2. Евтушенко Ю.Г., Ратькин В.А. Метод половинных делений для глобальной оптимизации функции многих переменных // Техническая кибернетика. 1987. № 1. С. 119-127.
3. Евтушенко Ю.Г., Малкова В.У., Станевичюс А.А. Распараллеливание процесса поиска глобального экстремума // АИТ. 2007. № 5. С. 46-58.
4. Nesterov Yu. Introductory lectures on Convex Optimization. A Basic Course. Boston: Kluwer, 2004.
5. Leary R.H. 2000. Global Optimization on Funneling Landscapes // J. of Global Optimization. 2000. Vol. 18, No 4. P. 367-383.

Таблица 7. Расчет минимума потенциальной энергии молекулы с помощью параллельной реализации МВН.

ρ	N	f^*	t_1	t_2
10	80	-344.911381	73	120
10	85	-368.440724	3	120
10	90	-392.509997	76	120
14	80	-340.811371	47	120
14	85	-363.438687	30	120
14	90	-387.973912	23	120

6. Посыпкин М.А., Сигал И.Х. Применение параллельных эвристических алгоритмов для ускорения параллельного метода ветвей и границ // ЖВМиМФ. 2007. Т. 47, № 9. С. 1524-1537.
7. Посыпкин М.А. Параллельный эвристический алгоритм глобальной оптимизации // Труды ИСА РАН. Проблемы вычислений в распределенной среде. 2007.
8. Strongin R.G., Sergeyev Ya.D. Global optimization with non-convex constraints: Sequential and parallel algorithms. Dordrecht: Kluwer Academic Publishers, 2000. 728 p.
9. Casado L.G., Garcí'a I., Martí'nez J.A., Sergeyev Ya.D. New interval analysis support functions using gradient information in a global minimization algorithm // Journal of Global Optimization. 2003. Vol. 25, No 4. P. 345-362.
10. Компьютеры и суперкомпьютеры в биологии / Под. ред. В.Д. Лахно и М.Н. Устинина. Москва-Ижевск: Институт компьютерных исследований, 2002. 528 с.
11. Eldon Hansen. Global Optimization Using Interval Analysis. New York: Marcel Dekker, 1992.