ЖУРНАЛ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И МАТЕМАТИЧЕСКОЙ ФИЗИКИ, 2011, том 51, № 8, с. 1376–1389

УДК 519.658.4

ПРИМЕНЕНИЕ МЕТОДА НЕРАВНОМЕРНЫХ ПОКРЫТИЙ ДЛЯ ГЛОБАЛЬНОЙ ОПТИМИЗАЦИИ ЧАСТИЧНО ЦЕЛОЧИСЛЕННЫХ НЕЛИНЕЙНЫХ ЗАДАЧ[1)]

© 2011 г. Ю. Г. Евтушенко*, М. А. Посыпкин**

*(*119333 Москва, ул. Вавилова, 40, ВЦ РАН;*

***117312 Москва, пр-т 60-летия Октября, 9, ИСА РАН)*

e-mail: evt@ccas.ru; mposypkin@mail.ru

Поступила в редакцию 15.02.2011 г.

Метод неравномерных покрытий для поиска глобального экстремума функций многих переменных переносится на задачи нелинейного программирования. Показано, что метод можно использовать для решения задач, в которых помимо обычных ограничений наложены условия частичной целочисленности. Даны оценки точности решений и оценка числа шагов, необходимых для нахождения минимума с заданной точностью. Приведены новые миноранты, основанные на оценке спектра гессиана целевых функций и ограничений. Получены новые формулы для покрывающих множеств, повышающие эффективность метода. Приводятся примеры решения задач нелинейного программирования с помощью предложенного подхода. Библ. 15. Фиг. 3. Табл. 2.

Ключевые слова: глобальная оптимизация, нелинейное программирование, частично-целочисленные задачи, функция чувствительности, метод неравномерных покрытий, численные методы оптимизации.

1. ВВЕДЕНИЕ

Метод неравномерных покрытий для поиска глобального экстремума функций многих переменных был предложен в [1]. Дальнейшее развитие этот подход нашел в многочисленных работах. Укажем лишь некоторые из них (см. [2]–[6]). Различные варианты метода были распараллелены, программно реализованы и использовались для расчетов на многопроцессорных системах (см. [5], [6]). Метод был также перенесен на многокритериальные задачи (см. [7]).

В настоящей работе дано более общее, чем в [1], [2], трактование метода неравномерных покрытий. Приведены новые модификации метода, повышающие его эффективность. Отдельно рассмотрены случай минимизации на простом допустимом множестве и случай функциональных ограничений. Предлагаемый метод применим, в частности, для поиска глобального экстремума в задачах нелинейного программирования, в которых допустимое множество может быть неодносвязным и содержать изолированные точки. Метод расширен также на класс задач частично целочисленного нелинейного программирования. Даны оценки точности решений и оценка числа шагов, необходимых для нахождения минимума с заданной точностью. Приведены новые миноранты, основанные на оценке спектра гессиана целевых функций и ограничений. Получены новые формулы для покрывающих множеств, повышающие эффективность метода. Один из вариантов метода неравномерных покрытий программно реализован на базе пакета BNB-Solver (см. [8]) и адаптирован к использованию на многопроцессорной вычислительной технике.

В качестве примера решена задача, в которой глобальный минимум достигается в изолированной точке допустимого множества. Показано, что при использовании метода неравномерных покрытий введение дополнительного условия целочисленности существенно ускоряет расчеты. Метод также тестировался на задачах безусловной глобальной минимизации, в которых в качестве целевой функции брались полиномы с коэффициентами, сгенерированными случайным образом. Эти расчеты наглядно показывают эффективность лучших версий метода неравномер-

[1)] Работа выполнена при финансовой поддержке РФФИ (коды проектов 09-01-12098-офи_м, 10-01-900-11-БЕЛ_а), программы П-14 Президиума РАН, гранта НШ-4096.2010.1.

ных покрытий по сравнению с результатами, полученными программами поиска глобального экстремума BARON (см. [9]) и LINDOGLOBAL (см. [10]) из пакета GAMS [11].

В настоящей работе используются следующие обозначения. Компоненты n-мерного вектора x обозначаются верхним индексом, заключенным в круглые скобки: $x = \left(x^{(1)},..,x^{(n)}\right)$. Через $\|x\|$ обозначается евклидова норма в пространстве $\mathbb{R}^n$: $\|x\| = \sqrt{\Sigma_{i=1}^n (x^{(i)})^2}$. Запись $|x|$ обозначает вектор $\left(\left|x^{(1)}\right|,\ldots,\left|x^{(n)}\right|\right)$, составленный из абсолютных величин компонент x. Векторные неравенства и бинарные операции над векторами выполняются покомпонентно. Например, запись $a \le b$ означает, что $a_i^{(j)} \le b_i^{(j)}$ для всех $j = 1,2,\ldots,n$, а соотношение $c = a + b$ означает, что $c_i^{(j)} = a_i^{(j)} + b^{(j)}$ для всех $j = 1,2,\ldots,n$. Градиент функции $f(x)$ обозначается через $f_x(x)$, а $f_{xx}(x)$ – матрица Гессе. Для любого подмножества $A \subseteq \mathbb{R}^n$ определим его диаметр $d(A) = \sup\{\|x_1 - x_2\|, x_1, x_2 \in A\}$.

2. ВАРИАНТ МЕТОДА НЕРАВНОМЕРНЫХ ПОКРЫТИЙ ДЛЯ МНОЖЕСТВ ПРОСТОЙ СТРУКТУРЫ

Для непрерывной функции $f : \mathbb{R}^n \to \mathbb{R}$ ставится задача отыскания глобального минимума на компактном допустимом множестве $X \subseteq \mathbb{R}^n$:

$$f_* = \operatorname{glob}\min_{x\in X} f(x) = f(x_*), \tag{1}$$

где x_* – любая точка, в которой достигается глобальный минимум, равный f_*. Для этой задачи определим множество *глобальных решений* X_* и множество ε*-оптимальных решений* X_ε:

$$X_* = \{x \in X : f(x) = f_*\}, \quad X_\varepsilon = \{x \in X : f(x) \le f_* + \varepsilon\}, \quad \varepsilon > 0. \tag{2}$$

Предполагаем, что множество X_* из (2) не пусто. Требуется найти хотя бы одну точку из множества X_ε.

Для множества $Z \subseteq \mathbb{R}^n$, функции $f(x) : \mathbb{R}^n \to \mathbb{R}$ и числа $\lambda \in \mathbb{R}$ определим *лебегово множество* $\mathcal{L}(f(\cdot), Z, \lambda) = \{x \in Z : f(x) \ge \lambda\}$ и *открытое лебегово множество* $\mathcal{L}'(f(\cdot), Z, \lambda) = \{x \in Z : f(x) > \lambda\}$. Используя понятие лебегова множества, можно определить необходимые и достаточные условия глобальной оптимальности для любой точки $x_* \in X$ следующим образом: $x_* \in X_* \Leftrightarrow$ $\Leftrightarrow \mathcal{L}(f(\cdot), X, f(x_*)) = X$.

Аналогично записывается критерий глобальной ε-оптимальности. Пусть $x_\varepsilon \in X$, тогда

$$x_\varepsilon \in X_\varepsilon \Leftrightarrow \mathcal{L}(f(\cdot), X, f(x_\varepsilon) - \varepsilon) = X. \tag{3}$$

Построение лебегова множества на всем допустимом множестве обычно бывает затруднительным. В таких случаях прибегают к дроблению множества X на подмножества и к построению на них минорант. Рассмотрим набор множеств $\{X_i\}$, $X_i \subseteq X$, $i = 1,2,\ldots,l$. На каждом множестве X_i определим миноранту $\mu_i(x)$ такую, что $f(x) \ge \mu_i(x)$ для всех $x \in X_i$. Пусть заданы совокупность допустимых точек $N_k = \{x_1,\ldots,x_k\}$ и совокупность множеств $M_k = \{\mathcal{L}_1,\ldots,\mathcal{L}_k\}$, удовлетворяющих условию

$$\mathcal{L}_i \subseteq \mathcal{L}(\mu_i(\cdot), X_i, f(x_i) - \varepsilon), \quad i = 1,2,\ldots,k. \tag{4}$$

Будем говорить, что совокупность множеств M_k *покрывает множество* X, если

$$X = \bigcup_{i=1}^{k} \mathcal{L}_i. \tag{5}$$

Множества $\mathcal{L}_i$ будем называть *покрывающими*.

Теорема 1. *Если для наборов допустимых точек* N_k *и покрывающих множеств* M_k *выполнено условие* (5), *то точка* $x_r = \operatorname{argmin}_{x\in N_k} f(x)$ *является* ε*-оптимальным решением задачи* (1) *и для него справедлива оценка*

$$f(x_r) \ge f_* \ge f(x_r) - \varepsilon. \tag{6}$$

Доказательство. Пусть x_* — одна из допустимых точек, в которых целевая функция принимает минимальное значение $f_* = f(x_*)$. Из условия (5) следует, что найдется по крайней мере одно покрывающее множество $\mathcal{L}_i$ такое, что $x_* \in \mathcal{L}_i$. Тогда из (4) следует $f_* = f(x_*) \ge \mu_i(x_*) \ge \ge f(x_i) - \varepsilon \ge f(x_r) - \varepsilon$. Тем самым показана справедливость правого неравенства в (6). Левое неравенство является следствием определения минимума функции $f(x)$ на X.

Теорема 1 является теоретической базой для реализации различных вычислительных схем метода неравномерных покрытий. Теорема 1 алгоритмически легко реализуется для множеств X простой структуры, когда нахождение допустимых точек и разбиение множества на подмножества осуществляется просто, например если X является параллелепипедом или задается системой линейных неравенств. Рассмотрим одну из возможных реализаций метода неравномерных покрытий.

Алгоритм COVER

Шаг 1. Поместить X в список S и положить $i := 1$. Выбрать в качестве x_0 произвольную точку из X.

Шаг 2. Взять из списка S некоторое множество, которое обозначим через X_i.

Шаг 3. Взять точку $c_i \in X_i$, вычислить $f(c_i)$ и положить

$$x_i = \begin{cases} c_i, & f(c_i) < f(x_{i-1}), \\ x_{i-1} & \text{в противном случае.} \end{cases}$$

Шаг 4. Согласно (4) определить множество $\mathcal{L}_i$ и его дополнение $X_i' = X_i \backslash \mathcal{L}_i$.

Шаг 5. Если $X_i' \ne \emptyset$, то разбить X_i' на p подмножеств $\mathcal{Y}_i = \left\{Y_1^i, \ldots, Y_p^i\right\}$ и добавить их в список S.

Шаг 6. Удалить X_i из списка S.

Шаг. 7. Если список S пуст, то завершить работу алгоритма, в противном случае положить $i := i + 1$ и перейти к шагу 2.

При $p = 2$ алгоритм COVER называется также *методом половинных делений* (или бисекций) (см. [2], [4]). Если алгоритм завершился за k итераций цикла 2–7, то найденная точка x_k будет ε-оптимальным решением задачи (1). Действительно, завершение работы алгоритма означает, что список S стал пустым, т.е. выполнено условие покрытия (5). По теореме 1, точка $x_r = = \operatorname{argmin}_{x \in N_k} f(x)$ будет ε-оптимальным решением задачи (1). Способ формирования последовательности точек x_i на шаге 3 обеспечивает выполнение равенства $x_k = x_r$. Точку x_i назовем *текущей рекордной точкой*, значение $f(x_i)$ — *текущим рекордом*, а множество X_i — *текущим подмножеством*.

Сделаем несколько замечаний в отношении реализации алгоритма COVER. Лебегово множество $\mathcal{L}(f(\cdot), X, \lambda)$ расширяется при уменьшении λ: $\mathcal{L}(f(\cdot), \lambda_1, X) \subseteq \mathcal{L}(f(\cdot), \lambda_2, X)$, если $\lambda_1 \ge \lambda_2$. Поэтому уменьшение текущего рекорда $f(x_i)$ расширяет множества $\mathcal{L}_i$, тем самым сокращая их количество, требуемое для выполнения условия покрытия (5). В процессе расчетов для уменьшения рекорда применяются методы локальной оптимизации, эвристические алгоритмы.

Стандартным способом формирования множества $\mathcal{L}_i$, используемым в [2]–[5], является следующий:

$$\mathcal{L}_i = \begin{cases} X_i, & \min_{x \in X_i} \mu_i(x) \ge f(x_i) - \varepsilon, \\ \emptyset & \text{в противном случае.} \end{cases} \tag{7}$$

При таком способе множество X_i уменьшается только в случае, когда $\mathcal{L}(\mu_i(\cdot), X_i, f(x_i) - \varepsilon) = X_i$. Более эффективные способы построения покрывающего множества рассматриваются в разд. 5. Все они удовлетворяют следующему условию:

$$\text{если} \quad \mathcal{L}(\mu_i(\cdot), X_i, f(x_i) - \varepsilon) = X_i, \quad \text{то} \quad \mathcal{L}_i = X_i, \tag{8}$$

т.е. дают не меньшее сокращение множества X_i по сравнению со стандартным способом.

Предположим, что на итерации i алгоритма COVER также выполнены следующие свойства:

миноранты $\mu_i(\cdot)$ являются опорными в точках c_i:

$$\mu_i(c_i) = f(c_i); \tag{9}$$

миноранты $\mu_i(\cdot)$ удовлетворяют условию Липшица с константой K на множестве X_i:

$$|\mu_i(x_1) - \mu_i(x_2)| \le K\|x_1 - x_2\| \quad \forall x_1, x_2 \in X_i; \tag{10}$$

в результате разбиения диаметр множества сокращается таким образом что для любого множества Y_i^t из $\mathcal{Y}_i$ справедливо неравенство

$$d(Y_t^i) \le \alpha d(X_i'), \tag{11}$$

где $0 < \alpha < 1$.

Лемма 1. *Если выполнены свойства* (8)–(10), *то из* $d(X_i) \le \varepsilon/K$ *следует* $\mathcal{L}_i = X_i$.

Доказательство. Согласно (9) и (10) для любого $x \in X_i$ выполнены неравенства

$$\mu_i(x) \ge \mu_i(c_i) - K\|x - c_i\| \ge f(c_i) - Kd(X_i) \ge f(c_i) - \varepsilon \ge f(x_i) - \varepsilon.$$

Следовательно, $\mathcal{L}(\mu_i(\cdot), X_i, f(x_i) - \varepsilon) = X_i$. Согласно (8), из этого следует, что $\mathcal{L}_i = X_i$.

Теорема 2. *Если выполнены свойства* (8)–(11), *то алгоритм* COVER *завершится за число шагов, не превосходящее величины* $(p^{t+2} - 1)/(p - 1)$, *где* $t = \left\lfloor \log_\alpha \dfrac{\varepsilon}{Kd(X)} \right\rfloor$.

Доказательство. Согласно лемме 1, если текущее множество X_i имеет диаметр меньший или равный ε/K, то $\mathcal{L}_i = X_i$. В этом случае $X_i' = X_i \setminus \mathcal{L}_i = \emptyset$ и, следовательно, множество X_i удаляется из списка S, не подвергаясь разбиению. Поэтому любое подмножество, подвергнутое разбиению на шаге 5, было получено не более чем за t последовательных разбиений, где $t = \left\lfloor \log_\alpha \dfrac{\varepsilon}{Kd(X)} \right\rfloor$. Следовательно, любое подмножество, обработанное алгоритмом, было получено не более чем за $t + 1$ разбиение. Так как при разбиении образуется p новых подмножеств, то общее число обработанных подмножеств не превосходит суммы геометрической прогрессии $1 + p + p^2 + \dots + p^{t+1}$, равной $(p^{t+2} - 1)/(p - 1)$.

3. ПОИСК ГЛОБАЛЬНОГО МИНИМУМА В ЗАДАЧЕ НЕЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

Рассмотрим случай, когда в задаче (1) допустимое множество задано с помощью функциональных ограничений

$$X = \{x \in \mathbb{R}^n : g(x) \le 0_m\}, \tag{12}$$

где $g(x): \mathbb{R}^n \to \mathbb{R}^m$ – непрерывная вектор-функция. В этом случае разбиение множества X на подмножества X_i сложно алгоритмизуется. Поэтому предположим, что допустимое множество X содержится в некотором ограниченном множестве P из $\mathbb{R}^n$ простой структуры.

Для $\delta \in \mathbb{R}$ определим δ-допустимое множество следующим образом:

$$X^\delta = \{x \in \mathbb{R}^n : g^{(j)}(x) \le \delta, j = 1, 2, \dots, m\}. \tag{13}$$

Множество X^δ может быть эквивалентным образом определено с использованием функции $\phi(x) = \max\left(g^{(1)}(x), \dots, g^{(m)}(x)\right)$: $X^\delta = \{x \in \mathbb{R}^n : \phi(x) \le \delta\}$. Так как $g(x)$ – непрерывная функция, то функция $\phi(x)$ также непрерывна. Положим $\underline{\delta} = \inf\{\delta : X^\delta \ne \emptyset\}$, $\bar{\delta} = \sup\{\delta : X^\delta \subseteq P\}$.

При $\underline{\delta} < \delta < \bar{\delta}$ положим

$$f_*^\delta = \min_{x \in X^\delta} f(x).$$

Функция f_*^δ называется *функцией чувствительности*.

Известны следующие ее свойства (см. [12], [13]):

1) $f_*^0 = f_*$; $X^\delta \subseteq X$ при $\underline{\delta} < \delta \le 0$; $X \subseteq X^\delta$ при $0 \le \delta < \bar{\delta}$;

2) множество X^δ не сужается, а функция чувствительности монотонно не возрастает с увеличением δ:

$$X^{\delta_1} \subseteq X^{\delta_2}, \quad f_*^{\delta_1} \ge f_*^{\delta_2} \quad \text{для любых} \quad \delta_1, \delta_2, \quad \underline{\delta} < \delta_1 \le \delta_2 < \bar{\delta}; \tag{14}$$

3) для любых δ_0 таких, что $\underline{\delta} < \delta_0 < \bar{\delta}$, функция чувствительности непрерывна справа:

$$\lim_{\delta \to \delta_0 + 0} f_*^\delta = f_*^{\delta_0}. \tag{15}$$

Рассмотрим $\varepsilon \in \mathbb{R}_+$ и $\delta_1 \in \mathbb{R}$, $\delta_2 \in \mathbb{R}$, $\underline{\delta} < \delta_1 \le \delta_2 < \bar{\delta}$. Определим множество

$$X_\varepsilon^{\delta_1, \delta_2} = \left\{ x \in X^{\delta_2} : f(x) \le f_*^{\delta_1} + \varepsilon \right\}.$$

В частном случае, когда $\delta_1 = 0$, $\delta_2 = \delta$, это множество будем называть *множеством* ε δ*-оптимальных решений* и обозначать через X_ε^δ:

$$X_\varepsilon^\delta = \{ x \in X^\delta : f(x) \le f_* + \varepsilon \}.$$

По аналогии с (3) можно сформулировать следующий критерий для точки $x_\varepsilon^{\delta_2}$ из X^{δ_2}:

$$x_\varepsilon^{\delta_2} \in X_\varepsilon^{\delta_1, \delta_2} \Leftrightarrow \mathcal{L}(f(\cdot), P, f\left(x_\varepsilon^{\delta_2}\right) - \varepsilon) \cup \mathcal{L}'(\phi(\cdot), P, \delta_1) = P. \tag{16}$$

Действительно, если $x_\varepsilon^{\delta_2} \in X_\varepsilon^{\delta_1, \delta_2}$, то $X^{\delta_1} \subseteq \mathcal{L}(f(\cdot), P, f\left(x_\varepsilon^{\delta_2}\right))$. Так как $X^{\delta_1} \cup \mathcal{L}'(\phi(\cdot), P, \delta_1) = P$, то $\mathcal{L}(f(\cdot), P, f\left(x_\varepsilon^{\delta_2}\right) - \varepsilon) \cup \mathcal{L}'(\phi(\cdot), P, \delta_1) = P$. Пусть правая часть утверждения (16) выполнена. Рассмотрим точку $x_*^{\delta_1} \in X^{\delta_1}$ такую, что $f\left(x_*^{\delta_1}\right) = f_*^{\delta_1}$. Так как $X^{\delta_1} \cap \mathcal{L}'(\phi(\cdot), P, \delta_1) = \emptyset$, то $X^{\delta_1} \subseteq \mathcal{L}(f(\cdot), P, f\left(x_\varepsilon^{\delta_2}\right) - \varepsilon)$. Следовательно, $f_*^{\delta_1} = f\left(x_*^{\delta_1}\right) \ge f\left(x_\varepsilon^{\delta_2}\right) - \varepsilon$, т.е. $f\left(x_\varepsilon^{\delta_2}\right) \le f_*^{\delta_1} + \varepsilon$.

Рассмотрим совокупность множеств $X_1, \ldots, X_k$, где $X_i \subseteq P$. Определим миноранты $\mu_i(x)$ такие, что $f(x) \ge \mu_i(x)$ для всех x из X_i и миноранты $\nu_i(x)$ такие, что $\phi(x) \ge \nu_i(x)$ для всех x из X_i. Пусть заданы совокупность точек $N_k = \{x_1, \ldots, x_k\}$ из X^{δ_2} и совокупность множеств $M_k = \{\mathcal{L}_1, \ldots, \mathcal{L}_k\}$, удовлетворяющих условию

$$\mathcal{L}_i \subseteq \mathcal{L}(\mu_i(\cdot), X_i, f(x_i) - \varepsilon) \cup \mathcal{L}'(\nu_i(\cdot), X_i, \delta_1). \tag{17}$$

Условие покрытия в данном случае имеет вид

$$P = \bigcup_{i=1}^{k} \mathcal{L}_i. \tag{18}$$

Справедлива

Теорема 3. *Если для наборов точек N_k и покрывающих множеств M_k выполнено условие* (18), *то точка $x_r = \operatorname{argmin}_{x \in N_k} f(x)$ принадлежит множеству $X_\varepsilon^{\delta_1, \delta_2}$ и справедлива оценка*

$$f_*^{\delta_1} + \varepsilon \ge f(x_r) \ge f_*^{\delta_2}. \tag{19}$$

Доказательство. Неравенство $f(x_r) \ge f_*^{\delta_2}$ следует из определения $f_*^{\delta_2}$. Докажем левое неравенство. Пусть $f_*^{\delta_1} = f(x_1)$, где $x_1 \in X^{\delta_1}$. Из условия (18) следует, что $x_1 \in \mathcal{L}_i$ для некоторого $i : 1 \le i \le k$, т.е. $x_1 \in \mathcal{L}(\mu_i(\cdot), X_i, f(x_i) - \varepsilon) \cup \mathcal{L}'(\nu_i(\cdot), X_i, \delta_1)$. Так как $x_1 \in X^{\delta_1}$, то $\nu_i(x_1) \le \phi(x_1) \le \delta_1$, т.е. $x_1 \notin \mathcal{L}'(X_i, \nu_i(x), \delta_1)$. Следовательно, $x_1 \in \mathcal{L}(\mu_i(\cdot), X_i, f(x_i) - \varepsilon)$ и $f_*^{\delta_1} = f(x_1) \ge \mu_i(x_1) \ge f(x_i) - \varepsilon \ge f(x_r) - \varepsilon$. Отсюда следует левое неравенство в (19).

Выделим два частных случая теоремы 2. В первом $\delta_1 < 0$, $\delta_2 = 0$ точка x_r является приближенным допустимым решением, для которого справедливо неравенство $f_*^{\delta_1} + \varepsilon \geq f(x_r) \geq f_*$. Из-за отсутствия непрерывности слева в этом случае приближенное и точное решения могут сколь угодно отличаться. Кроме того, для задач, где $\underline{\delta} = 0$ данный подход неприменим, так как здесь внутренность множества X пуста.

Во втором случае $\delta_1 = 0$, $\delta_2 > 0$. Положим $\delta = \delta_2$. Точка x_r будет приближенным, но, возможно, недопустимым решением, для которого справедливо неравенство $f_* + \varepsilon \geq f(x_r) \geq f_*^{\delta}$. При стремлении ε и δ к нулю значение $f(x_r)$ будет стремиться к f_*. Рассмотрим модификацию базового алгоритма бисекций для задач нелинейного программирования в этом случае.

Алгоритм COVERNLP

Шаг. 1. Поместить P в список S и положить $i := 1$.

Шаг 2. Взять из списка S некоторое множество, которое обозначим через X_i.

Шаг 3. Взять точку $c_i \in X_i$; если $\phi(c_i) \leq \delta$, то вычислить $f(c_i)$ и положить

$$x_i = \begin{cases} c_i, & i = 1 \quad \text{или} \quad f(c_i) < f(x_{i-1}), \\ x_{i-1} & \text{в противном случае.} \end{cases}$$

Шаг 4. Определить множество $\mathcal{L}_i$ согласно (17) и его дополнение $X_i' = X_i \backslash \mathcal{L}_i$.

Шаг 5. Если $X_i' \neq \emptyset$, то разбить X_i' на p подмножеств $\mathcal{Y}_i = \{Y_1^i, \ldots, Y_p^i\}$ и добавить их в список S.

Шаг 6. Удалить X_i из списка S.

Шаг 7. Если список S пуст, то завершить работу алгоритма, в противном случае положить $i := i + 1$ и перейти к шагу 2.

Если приведенный алгоритм завершился за k итераций цикла 2–7, то найденная точка x_k будет ε, δ-оптимальным решением задачи (1), где допустимое множество задается формулой (12). Если алгоритм COVERNLP завершился и не было найдено ни одной точки c_i такой, что $\phi(c_i) \leq \delta$, то, значит, множество X пусто. Будем предполагать, что на i-м шаге выполнены следующие условия:

миноранты $\mu_i(\cdot)$ и $\nu_i(\cdot)$ являются опорными в точке c_i:

$$\mu_i(c_i) = f(c_i), \quad \nu_i(c_i) = \phi(c_i); \tag{20}$$

миноранты $\mu_i(\cdot)$, $\nu_i(\cdot)$ удовлетворяют условию Липшица с константами K_1 и K_2 на множестве X_i, т.е. для любых $x_1, x_2 \in X_i$

$$|\mu_i(x_1) - \mu_i(x_2)| \leq K_1 \|x_1 - x_2\|, \quad |\nu_i(x_1) - \nu_i(x_2)| \leq K_2 \|x_1 - x_2\|; \tag{21}$$

$$\text{если} \quad \mathcal{L}(\mu_i(\cdot), X_i, f(x_i) - \varepsilon) = X_i \quad \text{или} \quad \mathcal{L}'(\nu_i(\cdot), X_i, \delta_1) = X_i, \quad \text{то} \quad \mathcal{L}_i = X_i. \tag{22}$$

Лемма 2. *Если выполнены свойства* (20)–(22), *то из* $d(X_i) \leq \min(\varepsilon/K_1, \delta/K_2)$ *следует* $\mathcal{L}_i = X_i$.

Доказательство. Пусть выполнено $d(X_i) \leq \min(\varepsilon/K_1, \delta/K_2)$. Рассмотрим два случая: $\phi(c_i) \leq \delta$ и $\phi(c_i) > \delta$. В первом случае, согласно (20) и (21), для любого $x \in X_i$ выполнено неравенство

$$\mu_i(x) \geq \mu_i(c_i) - K_1 \|x - c_i\| \geq f(c_i) - K_1 d(X_i) \geq f(c_i) - \varepsilon \geq f(x_i) - \varepsilon.$$

Следовательно, $\mathcal{L}(\mu_i(\cdot), X_i, f(x_i) - \varepsilon) = X_i$ и, согласно (22), имеем $L_i = X_i$. Если $\phi(c_i) > \delta$, то, согласно (20) и (21), для любого $x \in X_i$ выполнено неравенство

$$\nu_i(x) \geq \nu_i(c_i) - K_2 \|x - c_i\| \geq \phi(c_i) - K_2 d(X_i) \geq \phi(c_i) - \delta > 0.$$

Следовательно, $\mathcal{L}'(\nu_i(\cdot), X_i, \delta_1) = X_i$ и, согласно (22), имеем $\mathcal{L}_i = X_i$.

Аналогично теореме 2 доказывается

Теорема 4. *Если выполнены свойства* (11), (20)–(22), *то алгоритм* COVERNLP *завершится за число шагов, не превосходящее величины* $(p^{t+2} - 1)/(p - 1)$, *где* $t = \left\lfloor \log_\alpha \dfrac{\min(\varepsilon/K_1, \delta/K_2)}{d(P)} \right\rfloor$.

4. МИНОРАНТЫ

Если функция $f(x)$ удовлетворяет условию Липшица с константой l_i на множестве X_i, то, согласно [1], следующая функция будет минорантой для функции $f(x)$:

$$\mu_i(x) = f(c_i) - l_i\|x - c_i\|. \tag{23}$$

Если градиент функции $f(x)$ удовлетворяет условию Липшица, то, согласно [5], миноранта для функции $f(x)$ имеет вид

$$\mu_i(x) = f(c_i) + \langle f_x(c_i), x - c_i\rangle - \frac{L_i}{2}\|x - c_i\|^2, \tag{24}$$

где $\|f_x(x) - f_x(y)\| \le L_i\|x - y\|$ для любых $x, y \in X_i$.

Заметим, что $L_i \ge \sup_{x\in X_i}\|f_{xx}(x)\|$. Обычно принимают $L_i \ge \max(|k_i|, |K_i|)$, где k_i и K_i – такие числа, что спектр матрицы Гессе $f_{xx}(x)$ при $x \in X_i$ полностью лежит в пределах отрезка $[k_i, K_i]$. Использование чисел k_i, K_i позволяет строить более точные миноранту $\mu_i(x)$ и мажоранту $\mathcal{M}_i(x)$ для функции $f(x)$:

$$\mu_i(x) = f(c_i) + \langle f_x(c_i), x - c_i\rangle + \frac{k_i}{2}\|x - c_i\|^2, \tag{25}$$

$$\mathcal{M}_i(x) = f(c_i) + \langle f_x(c_i), x - c_i\rangle + \frac{K_i}{2}\|x - c_i\|^2. \tag{26}$$

Лебегово множество $\mathcal{L}(\mu_i(x), X_i, f(x_i) - \varepsilon)$ состоит из точек X_i, удовлетворяющих условию

$$\alpha(x) = f(c_i) + \langle f_x(c_i), x - c_i\rangle + \frac{k_i}{2}\|x - c_i\|^2 - f(x_i) + \varepsilon \ge 0. \tag{27}$$

Заметим, что $\alpha_x(x) = f_x(c_i) + k_i(x - c_i)$. При $k_i \ne 0$ уравнение $\alpha_x(x) = 0$ имеет единственное решение $z_i = c_i - \frac{1}{k_i}f_x(c_i)$. После замены переменных $x = y + z_i$ неравенство (27) примет вид

$$\frac{k_i}{2}\|y\|^2 \ge \frac{1}{2k_i}\|f_x(c_i)\|^2 - f(c_i) + f(x_i) - \varepsilon.$$

Лебегово множество имеет различный вид в зависимости от значения k_i. Рассмотрим три случая.

1. $k_i < 0$. Лебегово множество является пересечением X_i и шара с центром в точке c_i с радиусом ρ_i, вычисляемым по формуле

$$\rho_i = \sqrt{\frac{2}{k_i}\left(\frac{1}{2k_i}\|f_x(c_i)\|^2 - f(c_i) + f(x_i) - \varepsilon\right)}. \tag{28}$$

2. $k_i = 0$. Лебегово множество есть пересечение X_i и полупространства, определяемого неравенством

$$\langle f_x(c_i), x - c_i\rangle + f(c_i) - f(x_i) + \varepsilon \ge 0.$$

3. $k_i > 0$. Если $\frac{1}{2k_i}\|f_x(c_i)\|^2 - f(x_i) + f(x_r) - \varepsilon < 0$, то лебегово множество совпадает с X_i. В противном случае оно является пересечением X_i и внешности (с границей) шара радиуса, определяемого формулой (28).

Числа k_i и K_i можно оценить с помощью теоремы Гершгорина (см. [14]). Согласно этой теореме, для любого действительного собственного значения γ матрицы (a_{ij}) справедливы неравенства

$$\min_{i=1,2,\ldots,n}\left(a_{ii} - \sum_{j=1, j\ne i}^{n}|a_{ij}|\right) \le \lambda \le \max_{i=1,2,\ldots,n}\left(a_{ii} + \sum_{j=1, j\ne i}^{n}|a_{ij}|\right).$$

Следовательно, для оценки границ спектра можно использовать неравенства

$$k_i \geq \min_{i=1,2,..,n}\left(\underline{u_{ii}} - \sum_{j=1,j\neq i}^{n} v_{ij}\right), \quad K_i \leq \max_{i=1,2,..,n}\left(\overline{u_{ii}} + \sum_{j=1,j\neq i}^{n} v_{ij}\right), \tag{29}$$

где числа $\underline{u_{ij}}$, $\overline{u_{ij}}$ удовлетворяют соотношениям

$$\underline{u_{ij}} \leq \min_{x\in X_i}\frac{\partial f(x)}{\partial x^i \partial x^j}, \quad \max_{x\in X_i}\frac{\partial f(x)}{\partial x^i \partial x^j} \leq \overline{u_{ij}}, \quad v_{ij} = \max(|\underline{u_{ij}}|, |\overline{u_{ij}}|).$$

Оценки $\underline{u_{ij}}$, $\overline{u_{ij}}$ могут быть получены различными способами, например при помощи техники интервального анализа.

Заметим, что теорема 1 останется справедливой, если несколько ослабить ее условие. Во-первых, достаточно, чтобы $\mu_i(x)$ была минорантой функции $f(x)$ только на множестве $X_* \cap X_i$. Во-вторых, условие покрытия (5) может быть заменено условием $X_* \cap (\cup_{i=1}^{k} \mathcal{L}_i) \neq \emptyset$.

Пусть $c_i \in X_i$, $x_* \in X_* \cap X_i$ и X_i содержит только внутренние точки допустимого множества X. Тогда $f_x(x_*) = 0$ и мажоранта (26), построенная в точке x_*, примет вид

$$\mathcal{M}(x) = f(x_*) + \frac{K_i}{2}\|x - x_*\|^2.$$

Для всех $x \in X_i$ выполняется оценка $f(x) \leq \mathcal{M}(x)$. Поэтому $f(c_i) \leq \mathcal{M}(c_i)$. Отсюда получим, что $f(x_*) \geq f(c_i) - \frac{K_i}{2}\|x_* - c_i\|^2$. Таким образом, построена новая миноранта

$$\hat{\mu}_i(x) = f(c_i) - \frac{K_i}{2}\|x_* - c_i\|^2 \tag{30}$$

для функции $f(x)$ на множестве $X_* \cap X_i$. Лебегово множество определяется следующим образом:

$$\mathcal{L}(\hat{\mu}_i(\cdot), X_i, f(x_i) - \varepsilon) = \left\{x \in X_i : f(c_i) - \frac{K_i}{2}\|x_* - c_i\|^2 \geq f(x_i) - \varepsilon\right\}.$$

Заметим, что если $K_i \leq 0$, то $\mathcal{L}(\mu_i(\cdot), X_i, f(x_i) - \varepsilon) = X_i$, поскольку $f(x_i) \leq f(c_i)$. Если $K_i > 0$, то лебегово множество есть пересечение X_i с шаром, который имеет центр в точке c_i и радиус

$$\rho_i = \sqrt{\frac{2}{K_i}\big(f(c_i) - f(x_i) + \varepsilon\big)} \geq \sqrt{2\varepsilon/K_i}.$$

Так как достаточно показать, что $X_* \cap \cup_{i=1}^{k}\mathcal{L}_i \neq \emptyset$, то подмножества, заведомо не содержащие точек из X_*, могут быть исключены из рассмотрения. Если X_i содержит только внутренние точки множества X_i и градиент целевой функции удовлетворяет условию Липшица с константой L_i, то шар с центром в точке c_i и радиусом $\|f_x(c_i)\|/L_i$ может быть исключен из рассмотрения, так как в точках этого шара градиент отличен от нуля.

5. НАХОЖДЕНИЕ ПОКРЫВАЮЩИХ МНОЖЕСТВ И ИХ ДОПОЛНЕНИЙ

Далее ограничимся случаем, когда множества X, P и X_i в алгоритмах COVER и COVERNLP являются n-мерными параллелепипедами. Для корректной работы алгоритма необходимо, чтобы множества X_i' также были n-мерными параллелепипедами. Положим $X_i = \{x \in \mathbb{R}^n : a_i \leq x \leq b_i\}$, $c_i = (a_i + b_i)/2$.

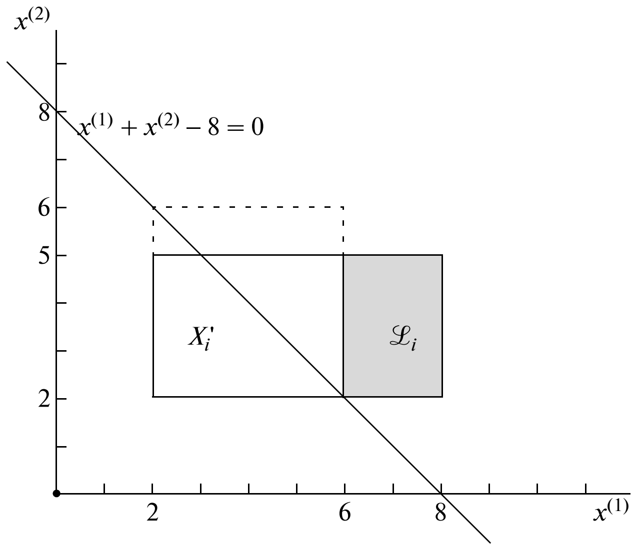
Для рассмотренных выше минорант возможны три вида лебеговых множеств.

1. Пересечение полупространства и параллелепипеда X_i.

2. Дополнение шара до параллелепипеда X_i.

3. Пересечение шара и параллелепипеда X_i.

Рассмотрим способы построения покрывающих множеств для каждого из этих случаев.



Фиг. 1. Исключение части параллелепипеда с учетом линейных неравенств.

1. Рассмотрим лебегово множество, которое является пересечением полупространства $l(x) = \langle p, x \rangle + q \ge 0$, где $p, x \in \mathbb{R}^n, q \in \mathbb{R}$, и параллелепипеда X_i. Минимум функции $l(x)$ на X_i достигается в точке z_i, определяемой соотношениями

$$z_i^{(j)} = \begin{cases} a_i^{(i)}, & p^{(j)} \ge 0, \\ b_i^{(j)}, & p^{(j)} < 0, \quad j = 1, 2, \dots, n. \end{cases}$$

Положим $v_i = \min_{x \in X_i} l(x) = \langle p, z_i \rangle + q$. Если $v_i \ge 0$, то положим $\mathcal{L}_i = X_i$.

Если $v_i < 0$, то определим параллелепипед $I_i = I_i^1 \times \ldots \times I_i^n$, где

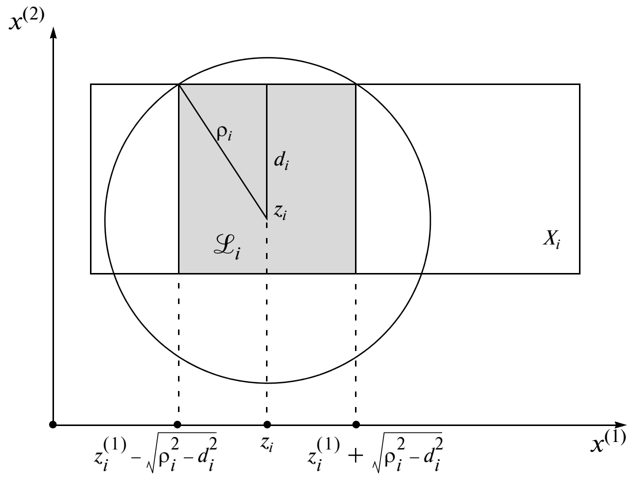
$$I_i^j = \begin{cases} [b_i^{(j)} - v_i/p^{(j)}, b_i^{(j)}], & p^{(j)} < 0, \\ [a_i^{(j)}, a_i^{(j)} - v_i/p^{(j)}], & p^{(j)} > 0, \\ [a_i^{(j)}, b_i^{(j)}], & p^{(j)} = 0. \end{cases}$$

Несложно показать, что $l(x) > 0$ для любых $x \in X_i \backslash I_i$. Поэтому можно положить $X_i' = X_i \cap I_i$, $\mathcal{L}_i = X_i \backslash X_i'$. При этом множество X_i' является параллелепипедом.

Фиг. 1 иллюстрирует этот случай для параллелепипеда $X_i = [(2,2), (8,5)]$ и $l(x) = x^{(1)} + x^{(2)} - 8 \ge 0$.

2. Пусть лебегово множество является дополнением шара B_i с центром z_i и радиусом ρ_i до параллелепипеда X_i. Рассмотрим куб $C_i = \{x \in \mathbb{R}^n : z_i^{(j)} - \rho_i \le x_i^{(j)} \le z_i^{(j)} + \rho_i,\ j = 1, 2, .., n\}$, описанный около шара B_i. Так как $X_i \backslash C_i \subseteq X_i \backslash B_i$, то положим $X_i' = C_i \cap X_i$, $\mathcal{L}_i = X_i \backslash X_i'$. Множество X_i' – параллелепипед, границы которого легко находятся.

3. Пусть лебегово множество представляет собой пересечение параллелепипеда X_i и шара B_i с центром в точке z_i и радиусом ρ_i. Выберем базисный вектор $e_s = \{\underbrace{0, \dots, 0}_{s-1}, 1, 0, \dots, 0\} \in \mathbb{R}^n$, перпендикулярно которому будет производиться дробление параллелепипеда, где $s = \arg\max_{j=1,2,..,n} \left(b_i^{(j)} - a_i^{(j)}\right)$.



Фиг. 2. Исключение части параллелепипеда, лежащей внутри шара.

Максимальное из расстояний от точки z_i до прямых, проходящих вдоль ребер параллелепипеда X_i параллельно вектору e_s, определяется по формуле

$$d_i = \sqrt{\sum_{j=1}^{n} (d_i^{(j)})^2}, \quad \text{где} \quad d_i^{(j)} = \begin{cases} \max(|a_i^{(j)} - z_i^{(j)}|, |b_i^{(j)} - z_i^{(j)}|), & j \neq s, \\ 0, \quad j = s. \end{cases} \tag{31}$$

При $d_i < \rho_i$ в качестве покрывающего множества возьмем параллелепипед $\mathscr{L}_i = I_i^1 \times \ldots \times I_i^n$, где

$$I_i^j = \begin{cases} [a_i^{(j)}, b_i^{(j)}], \quad j \neq s, \\ [z_i^{(s)} - \sqrt{\rho_i^2 - d_i^2}, z_i^{(s)} + \sqrt{\rho_i^2 - d_i^2}], \quad j = s. \end{cases}$$

Несложно показать, что $\mathscr{L}_i \subseteq B_i \cap X_i$. Этот случай проиллюстрирован на фиг. 2 при $n = 2$.

Дополнение $X_i' = X_i \backslash \mathscr{L}_i$ может состоять из одного или более параллелепипедов. Если X_i' – параллепипед, то он разбивается на два равных параллелепипеда вдоль наибольшего ребра.

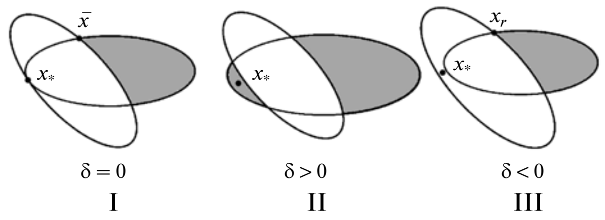
В задачах частично целочисленного программирования на допустимое множество X помимо функциональных ограничений (12) накладывается требование целочисленности вектора x либо некоторых его компонент: $x^{(j)} \in \mathbb{Z}$, $j \in J$, где J – подмножество индексов целочисленных компонент вектора x. Без ограничения общности можно считать, что $a^{(j)} \in \mathbb{Z}$, $b^{(j)} \in \mathbb{Z}$ при $j \in J$.

Обозначим через $\lfloor x \rfloor$ наибольшее целое число, не превосходящее x, а через $\lceil x \rceil$ – наименьшее целое число, большее или равное x. На шаге 5 алгоритма COVERNLP целочисленность позволяет уменьшить размер множеств, добавляемых в список, следующим образом: $a_i^{(j)} := \lceil a_i^{(j)} \rceil$, $b_i^{(j)} := \lfloor b_i^{(j)} \rfloor$ для всех индексов j из множества J.

Целочисленность также учитывается при определении рекордной точки x_i на шаге 3. Перед ним точка c_i подвергается следующему преобразованию:

$$c_i^{(j)} := \begin{cases} \lfloor c_i^{(j)} \rfloor, \quad j \in J, \\ c_i^{(j)}, \quad j \notin J. \end{cases}$$

Остальные шаги алгоритма остаются без изменений. Таким образом, учет условия целочисленности позволяет уменьшить параллелепипеды, получаемые в результате разбиения, и, соответственно, сократить число итераций алгоритма. Ниже это свойство демонстрируется на примере.



Фиг. 3. Множество X^δ (заштриховано) при различных δ.

6. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ

Алгоритмы COVER и COVERNLP программно реализованы на базе пакета BNB-Solver (см. [8]). При решении конкретной задачи требуется реализовать модули, вычисляющие значения функции $f(x)$, ее градиент, функции $g^{(i)}(x)$, их градиенты, оценки констант Липшица и границ спектра для целевой функции и ограничений. Для полиномов от нескольких переменных перечисленные объекты могут быть автоматически вычислены. Поэтому были добавлены модули, поддерживающие полиномиальные целевые функции и ограничения.

Метод бисекций эффективно распараллеливается. На многопроцессорных системах с общей памятью несколько потоков независимо выполняют итерации 2–7 алгоритма, обращаясь к общему списку S. Для предотвращения потерь, связанных с синхронизацией при доступе к общему списку, каждый поток также поддерживает локальный список подмножеств. Часть подмножеств из этого списка периодически копируется в общий. В момент, когда локальный список исчерпывается, поток берет новое подмножество для обработки из общего списка. Реализация для систем с распределенной памятью следует той же самой схеме, но вместо копирования применяется передача сообщений по сети. Ключевым для управления процессом вычислений является выбор параметров, определяющих частоту обменов и число передаваемых данных. Более подробно вопросы параллельной реализации метода неравномерных покрытий рассмотрены в [5], [6].

7. ВЫЧИСЛИТЕЛЬНЫЙ ЭКСПЕРИМЕНТ

7.1. Поиск изолированного минимума

Рассмотрим следующий пример из [15]:

$$f(x) = x^{(1)} \to \min, \tag{32}$$

$$g^{(1)}(x) = (x^{(1)} - 5)^2 + 2(x^{(2)} - 5)^2 + (x^{(3)} - 5)^2 - 18 \le 0, \tag{33}$$

$$g^{(2)}(x) = 100 - (x^{(1)} + 7 - 2x^{(2)})^2 - 4(2x^{(1)} + x^{(2)} - 11)^2 - 5(x^{(3)} - 5)^2 \le 0. \tag{34}$$

Особенностью задачи (32)–(34) является тот факт, что глобальный минимум достигается в изолированной допустимой точке $x_* = (1, 4, 5)$, где $f(x_*) = 1$, $g^{(1)}(x_*) = g^{(2)}(x_*) = 0$ (в [15] ошибочно указана точка $(1, 3, 5)$). Использованный в [15] метод поиска глобального минимума определил допустимую точку $\bar{x} = (3.747692, 7.171420, 2.362317)$, где $f(\bar{x}) = 3.747692$. Вычисления заняли 33.703 с на компьютере Pentium IV 2.53 GHz.

Для применения метода неравномерных покрытий в качестве множества P был выбран параллелепипед $-10 \le x^{(i)} \le 10$, $i = 1, 2, 3$, заведомо содержащий допустимое множество рассматриваемой задачи. Использовалась миноранта (25).

На фиг. 3 показана проекция допустимой области задачи на плоскость $x^{(3)} = 5$. Она представляет собой пересечение внутренней части эллипса, определяемой ограничением (33), и внешней части другого эллипса, соответствующей ограничению (34). Это пересечение состоит из заштрихованной области и точки x_*, где эллипсы касаются. На фиг. 3(II) заштрихована δ-допустимая область X^δ при $\delta > 0$. Эта область содержит точку минимума x_* вместе с некоторой окрестностью. При $\delta < 0$ множество X^δ представляет собой односвязное множество и изолированная точка x_* ему не принадлежит.

Таблица 1. Результаты расчетов при различных значениях δ

δ	Рекорд x_r	$\varphi(x_r)$	Время расчетов (с)	Число итераций
−0.01	(3.722, 7.276, 7.452)	−0.00204	0.61	10602
−0.0001	(3.721, 7.15, 2.331)	−0.00001	24.57	506351
0.0001	(0.997, 4.007, 4.999)	0.00007	7.94	165547
0.01	(0.965, 4.071, 4.994)	0.00993	0.16	2671

Таблица 2. Время расчетов (с) для различных серий полиномов

Серия \ Метод		O0	O1	O2	O3	O3+	BR	LG
1	AVR	5.399	0.73	0.49	0.08	0.05	1.07	5.42
	MAX	15.12	1.77	1.15	0.12	0.07	1.36	6.02
	MIN	0.66	0.33	0.11	0.07	0.04	0.65	3.43
2	AVR	T	11.24	8.52	0.54	0.29	4.86	E
	MAX	T	28.92	39.17	0.67	0.38	6.56	E
	MIN	T	4.03	2.09	0.46	0.24	3.68	E
3	AVR	T	T	T	2.2	1.38	A	E
	MAX	T	T	T	2.49	1.75	A	E
	MIN	T	T	T	1.9	1.15	A	E
4	AVR	T	107.31	44.15	0.94	0.54	3.51	23.63
	MAX	T	350.07	120.49	1.46	0.77	3.89	27.71
	MIN	T	22.03	10.49	0.76	0.41	3.02	20.54
5	AVR	T	T	T	11.72	5.85	A	E
	MAX	T	T	T	14.11	7.14	A	E
	MIN	T	T	T	9.93	4.7	A	E

Эксперименты проводились на персональном компьютере с процессором Intel Core 2 Quad, 2.33 GHz. Использовалось одно ядро процессора, полагалось $\varepsilon = |\delta|$. Результаты экспериментов приведены в табл. 1, значение $f(x_r)$ совпадает с первой компонентой вектора x_r, приведенного во втором столбце.

При $\delta_1 = 0$, $\delta_2 = \delta > 0$ получены приближенные решения, близкие к оптимальному решению x_*. Найденные при $\delta_1 = \delta < 0$, $\delta_2 = 0$ точки допустимы и близки к точке $\bar{x}$, при этом полученное значение целевой функции меньше, чем найденное в [15], в третьей значащей цифре.

Данная задача была использована для тестирования возможностей нахождения целочисленного решения. Для этого на все переменные было наложено условие целочисленности. Расчеты проводились в трех различных вариантах при $\varepsilon = \delta = 0$. В первом использовалась базовая схема алгоритма COVERNLP, применялись стандартные липшицевы миноранты (23). Во втором варианте использовались миноранты (25). В первых двух вариантах покрывающее множество формировалось только с использованием правила (22). В третьем варианте дополнительно применялась техника определения покрывающих множеств, рассмотренная в разд. 5. Глобальный минимум был найден за 585, 121, 55 итераций в первом, втором и третьем вариантах соответственно. Таким образом, в данной задаче введение условия целочисленности существенно ускоряет расчеты. Поэтому при решении сложных задач представляется целесообразным вводить это условие искусственно для нахождения приближенного решения исходной непрерывной задачи с целью ускорения расчетов и получения близких к оптимуму рекордных значений.

7.2. Тестирование на случайным образом заданных полиномах

Эксперименты проводились для задач безусловной оптимизации, в которых в качестве целевой функции были выбраны полиномы нескольких переменных, коэффициенты которых генерировались случайным образом. Рассматривались полиномы вида

$$f(x) = \sum_{i=1}^{n} a(x^{(i)})^m + \sum_{d\in D} a_d (x^{(i_1)})^{d_1} \dots (x^{(i_n)})^{d_n}, \tag{35}$$

где n – размерность вектора x, m – четная степень полинома (35), $a > 0$ – заданное фиксированное число, $D = \{(d_1, \dots, d_n) : d_i \in \mathbb{Z}_+, \Sigma_{i=1}^n d_i \le m-1\}$ – множество кортежей степеней мономов, числа a_d равномерно распределены в интервале $[0, a]$.

Несложно показать, что при четном m полиномы указанного вида имеют глобальный оптимум, который лежит в пределах параллелепипеда $[-M, M]^n$, где $M = |D|$ – мощность множества кортежей D. Указанный параллелепипед выбирался в качестве допустимого множества X.

Эксперименты проводились для 5 серий, каждая из которых содержала 10 тестовых задач, везде $a = 10$.

Серия 1: $n = 3$, $m = 4$.

Серия 2: $n = 3$, $m = 6$.

Серия 3: $n = 3$, $m = 8$.

Серия 4: $n = 4$, $m = 4$.

Серия 5: $n = 4$, $m = 6$.

В табл. 2 приводятся минимальное (MIN), максимальное (MAX) и среднее (AVR) время расчетов в секундах для каждой из серий при решении методами, использующими следующие варианты:

O0 – стандартную липшицеву миноранту (23),

O1 – миноранту (24),

O2 – миноранту (25),

O3 – миноранту (30),

O3+ – миноранту (30) в сочетании с техникой сокращения области поиска (п. 5).

В вариантах O0–O3 покрывающее множество определялось по правилу (7). В колонках BR и LG приводятся значения времени решения тестовых примеров, полученные с помощью программ поиска глобального экстремума BARON (см. [9]) и LINDOGLOBAL (см. [10]) из пакета GAMS (см. [11]).

В табл. 2 используются следующие обозначения: T – вычисления продолжались слишком долго и были остановлены; A – программа завершила счет аварийно; E – результат расчетов неверен.

Анализ результатов, приведенных в табл. 2, показывает, что миноранты, введенные в разд. 4, и методы построения покрывающих множеств, рассмотренные в разд. 5, дают заметное ускорение расчетов по сравнению со стандартной техникой. Наилучшие результаты были получены при использовании миноранты (30) в сочетании с сокращением области поиска методами, приведенными в разд. 5. При этом решение задачи осуществляется гораздо быстрее по сравнению с программами BARON и LINDOGLOBAL, которые в ряде случаев не находили решение из-за аварийного завершения или выдавали заведомо не оптимальное решение.

СПИСОК ЛИТЕРАТУРЫ

1. *Евтушенко Ю.Г.* Численный метод поиска глобального экстремума функций (перебор на неравномерной сетке) // Ж. вычисл. матем. и матем. физ. 1971. Т. 11. № 6. С. 1390–1403.
2. *Евтушенко Ю.Г., Ратькин В.А.* Метод половинных делений для глобальной оптимизации функции многих переменных // Техн. кибернетика. 1987. № 1. С. 119–127.
3. *Evtushenko Yu.G., Potapov M.A., Korotkikh V.V.* Numerical methods for global optimization. Recent advances in global optimization. Princeton Univ. Press, 1992. P. 274–297.
4. *Ratchek H., Rokne J.* New computer methods for global optimization. Chichester: Ellis Horwood, 1988.

5. *Евтушенко Ю.Г., Малкова В.У., Станевичюс А.А.* Параллельный поиск глобального экстремума функций многих переменных // Ж. вычисл. матем. и матем. физ. 2009. Т. 49. 2. С. 255–269.
6. *Evtushenko Y., Posypkin M., Sigal I.* A framework for parallel large-scale global optimization // Comput. Sci. - Res. and Development. 2009. V. 23. № 3. P. 211–215.
7. *Евтушенко Ю.Г., Потапов М.А.* Методы решения многокритериальных задач // Докл. АН СССР. 1986. Т. 291. № 1. С. 25–39.
8. *Посыпкин М.А.* Решение задач глобальной оптимизации в среде распределенных вычислений // Программные продукты и системы. 2010. № 1. С. 23–29.
9. *Tawarmalani M., Sahinidis N.V.* A polyhedral branch-and-cut approach to global optimization // Math. Program. 2005. V. 103. № 2. P. 225–249.
10. http://www.lindo.com/
11. http://www.gams.com/
12. *Evans J.P., Gould F.J.* Stability in nonlinear programming // Operat. Res. 1970. V. 18. P. 107–118.
13. *Голиков А.И., Коткин Г.Г.* Характеристика множества оптимальных оценок задачи многокритериальной оптимизации // Ж. вычисл. матем. и матем. физ. 1988. Т. 28. № 10. С. 1461–1474.
14. *Тыртышников Е.Е.* Матричный анализ и линейная алгебра. М.: Физматлит, 2007.
15. *Tuy H.* D(C)-optimization and robust global optimization // J. Global Optimizat. 2010. V. 47. P. 485–501.