

Интенсификация сокращения области поиска в методе неравномерных покрытий

А.А Абдилдаева, М.А. Посыпкин, Е.М. Чечеткина

Аннотация— В данной работе развивается метод неравномерных покрытий, относящийся к классу детерминированных методов глобальной оптимизации. Предлагается подход, основанный на решении вспомогательной оптимизационной задачи, позволяющий увеличить объем покрывающих множеств, и тем самым, ускорить метод неравномерных покрытий. Приводится описание реализации предложенного подхода в рамках программного комплекса для решения задач глобальной оптимизации и результаты вычислительных экспериментов.

Ключевые слова — детерминированные методы оптимизации, глобальная оптимизация, метод неравномерных покрытий, программные комплексы оптимизации.

1. ВВЕДЕНИЕ

Методы решения задач оптимизации можно условно разделить на две категории: эвристические [1] и детерминированные [2, 3]. Эвристики позволяют найти некоторое приближение к глобальному экстремуму функции, но не дают какой-либо информации о близости найденного значения к истинному значению минимума. Такие методы широко применяются на практике, когда размерность задачи велика или отсутствует аналитическое представление целевых функций и ограничений задачи.

Детерминированные методы, напротив, гарантируют точность найденных решений, т.е. предоставляют оценку отклонения найденного решения от оптимума. Платой за гарантию оптимальности является высокая вычислительная сложность: детерминированные оптимизационные, как правило, работают существенно медленнее эвристических методов. Это делает особенно актуальной задачу повышения производительности детерминированных методов.

Данная работа посвящена повышению

Работа выполнена в рамках грантового финансирования по проекту "Математическое моделирование, разработка, исследование и реализация методов решения задач динамической оптимизации большой размерности на современной высокопроизводительной вычислительной технике" номер гос. регистрации проекта - 0115РК00554 и проекта РФФИ 13-01-12110 офи_м.

А.А. Абдилдаева работает в Институт информационных и вычислительных технологий, Казахстан (e-mail: abass_81@mail.ru)

М.А.Посыпкин работает в ИППИ РАН, ВЦ РАН (e-mail: mpopypkin@gmail.com)

Е.М. Чечеткина работает в РХТУ им. Д.И. Менделеева (e-mail: elena-chechetkina@rambler.ru)

производительности метода неравномерных покрытий, предложенного в 1971 году Ю.Г. Евтушенко [4]. Метод основан на построении так называемых покрывающих множеств, которые постепенно заполняют все допустимое множество. В работе предлагается алгоритм, позволяющий увеличить площадь покрывающих множеств, повышая тем самым скорость работы метода.

В пункте II кратко описывается метод неравномерных покрытий. Предлагаемый новый алгоритм построения покрывающих множеств и их дополнений рассматривается в разделе III. Принципы программной реализации предложенных алгоритмов рассмотрены в разделе IV. Раздел V посвящен экспериментальному исследованию разработанных подходов.

II. МЕТОД НЕРАВНОМЕРНЫХ ПОКРЫТИЙ

Рассматривается задача глобальной оптимизации в следующей постановке:

$$f(x) \rightarrow \min, x \in X \quad (1)$$

Требуется найти оптимальное решение задачи (1), т.е. такое $x_* \in X$, что $f(x_*) \leq f(x)$ для всех $x \in X$. На практике точное решение задачи (1) возможно только для очень простых целевых функций и допустимых множеств. В более сложных случаях ищется приближенное решение: ε -оптимальным решением задачи (1) называется такое $x_\varepsilon \in X$, что $f(x_\varepsilon) \leq f(x_*) + \varepsilon$.

Метод неравномерных покрытий (МНП) базируется на понятии Лебегова множества, определяемого следующим образом:

$$L(f, Z, \lambda) = \{x \in Z : f(x) \geq \lambda\},$$

где $f : R^n \rightarrow R$ - произвольная функция, $Z \subseteq R^n$ - некоторое подмножество R^n , а λ - действительное число.

Другим важнейшим понятием, используемым в МНП, является понятие миноранты. Функция $\mu : R^n \rightarrow R$ называется минорантой для функции $f : R^n \rightarrow R$ на множестве $Z \subseteq R^n$, если неравенство $f(x) \geq \mu(x)$ справедливо для всех $x \in Z$.

Следующая теорема [5] дает достаточное условие ε -оптимальности для задачи (1). Пусть $N_k = \{x_1, \dots, x_k\}$ - совокупность точек допустимого множества X , X_1, \dots, X_k - совокупность его подмножеств. На каждом

множестве X_i зададим миноранту $\mu_i(x)$, такую что $f(x) \geq \mu_i(x)$ при $x \in X_i$.

Теорема. Если существует такая последовательность множеств L_1, \dots, L_k , что выполняются включения

$$L_i \subseteq L(\mu_i, X_i, f(x_i) - \varepsilon), \quad i = 1, \dots, k, \quad (2)$$

$$\text{и } X \subseteq \bigcup_{i=1}^k L_i, \quad (3)$$

то точка $x_r \in \arg \min_{x \in N_k} f(x)$ будет ε -оптимальным решением задачи (1), т. е. будет выполнено неравенство $f(x_r) \leq f(x_*) + \varepsilon$.

В процессе своей работы МНП строит последовательность точек N_k и совокупность покрывающих множеств L_1, \dots, L_k . Метод завершает свою работу, как только достигается выполнение условия покрытия (3).

III. ПОСТРОЕНИЕ ПОКРЫВАЮЩИХ МНОЖЕСТВ

Важная вспомогательная задача, решаемая в рамках МНП, состоит в построении покрывающих множеств, удовлетворяющих условию (2). Для простоты и эффективности реализации в качестве множеств X_i используются параллелепипеды. Соответственно, множества L_i выбираются таким образом, чтобы дополнение $X_i \setminus L_i$ являлось параллелепипедом.

В [5] показано, что для различных вариантов Липшицевых минорант множество $L(\mu_i, X_i, f(x_i) - \varepsilon)$ представляет собой пересечение шара и параллелепипеда, полупространства и параллелепипеда или дополнение параллелепипеда до шара. Далее в этой же работе предложен алгоритм нахождения такого параллелепипеда $L_i \subseteq L(\mu_i, X_i, f(x_i) - \varepsilon)$, что его дополнение $X_i \setminus L_i$ до параллелепипеда X_i также было бы параллелепипедом. Пример такого построения приведен на Рис. 1.

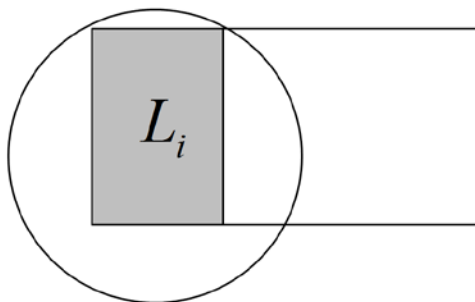


Рис. 1. Построение покрывающего множества

Такой параллелепипед не всегда удается построить.

Например, в случае, проиллюстрированном на Рис. 2, не существует такого параллелепипеда, находящегося внутри шара, чтобы его дополнение до параллелепипеда X_i являлось параллелепипедом. При этом, дополнение $X_i \setminus B_i$ можно представить в виде совокупности двух параллелепипедов.

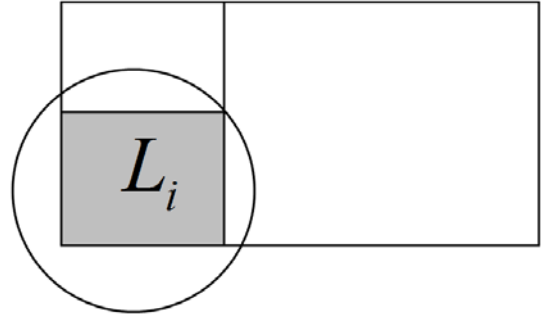


Рис. 2. Дополнение покрывающего множества, состоящее из двух параллелепипедов

В [6] показано, что дополнение любого параллелепипеда до другого параллелепипеда представимо в виде совокупности параллелепипедов и дан алгоритм построения этой совокупности. Приведем псевдокод этого алгоритма.

Алгоритм COMPLEMENT

Входные параметры:

n – размерность пространства

$B = [a, b]$ – исходный параллелепипед

$B' = [a', b']$ – удаляемый параллелепипед

Выходные параметры:

S – список параллелепипедов, составляющих дополнение $B \setminus B'$

ЦИКЛ по i от 1 до n

ЕСЛИ $b_i < a'_i$ или $b'_i < a_i$, ТО $S = \emptyset$, ВЫХОД

ЕСЛИ $a'_i \geq a_i$, ТО добавить к списку

S параллелепипед $[a_1, b_1] \times \dots \times [a_i, a'_i] \times \dots \times [a_n, b_n]$ и

положить $a_i = a'_i$.

КОНЕЦ ЦИКЛА

Учитывая этот факт, множество L_i удается построить всегда. Естественно предположить, что в целях ускорения работы метода целесообразно максимизировать объем отбрасываемого множества L_i .

Возникает следующая вспомогательная задача:

$$L_i = \max Vol(P \cap X_i), \quad (4)$$

$$P \in \Pi_n, P \subseteq L(\mu_i, X_i, f(x_i) - \varepsilon),$$

где через $Vol(\cdot)$ обозначен объем фигуры, Π_n – множество всех параллелепипедов размерности n .

Рассмотрим задачу (4) для наиболее распространенного случая, когда Лебегово множество

$L(\mu_i, X_i, f(x_i) - \varepsilon)$ представляет собой внутренность шара. В этом случае для решения задачи (4) используется следующий стохастический алгоритм.

Алгоритм MAXPAR

Входные параметры:

n - размерность параллелепипеда,

$X_i = [a, b]$ - текущий параллелепипед,

Лебегово множество – шар с центром в точке s и радиусом r .

Выходные параметры:

Параллелепипед L_i

$s := r, v := 0$

ЦИКЛ по k от 1 до \maxit

ЦИКЛ по j от 1 до n

$t := \text{rand}(0, s)$

$s := \sqrt{s^2 - t^2}$

$\hat{a}_j = \max(a_j, c_j - t)$

$\hat{b}_j = \min(b_j, c_j + t)$

ЕСЛИ $\hat{a}_j \geq \hat{b}_j$ ТО ПЕРЕЙТИ К СЛЕДУЮЩЕМУ k
КОНЕЦ ЦИКЛА

ЕСЛИ $\text{Vol}([\hat{a}, \hat{b}]) > v$ ТО $v := u, L_i := [\hat{a}, \hat{b}]$

КОНЕЦ ЦИКЛА

Количество итераций внешнего цикла, во-многом, определяет качество решения задачи (4): чем больше величина \maxit – тем больше объем построенного параллелепипеда. Вместе с тем, увеличение \maxit приводит к дополнительным вычислительным расходам, что может негативно сказаться на эффективности алгоритма. Эксперименты показывают, что выбор $\maxit = 10 \cdot 2^n$ дает неплохие результаты.

В случае, когда на часть переменных задачи наложено условие целочисленности, алгоритм MAXPAR несколько меняется. Покрывающий параллелепипед может быть расширен до целых границ в направлениях, соответствующих целочисленным переменным.

Алгоритм MAXPARINT

Входные параметры:

n - размерность параллелепипеда,

J - список целочисленных переменных,

$X_i = [a, b]$ - текущий параллелепипед,

Лебегово множество – шар с центром в точке s и радиусом r .

Выходные параметры:

Параллелепипед L_i

$s := r, v := 0$

ЦИКЛ по k от 1 до \maxit

ЦИКЛ по j от 1 до n

$t := \text{rand}(0, s)$

$s := \sqrt{s^2 - t^2}$

ЕСЛИ $j \in J$ ТО

$\hat{a}_j = \max(a_j, \lfloor c_j - t \rfloor)$

$\hat{b}_j = \min(b_j, \lceil c_j + t \rceil)$

ИНАЧЕ

$\hat{a}_j = \max(a_j, c_j - t)$

$\hat{b}_j = \min(b_j, c_j + t)$

ЕСЛИ $\hat{a}_j \geq \hat{b}_j$ ТО ПЕРЕЙТИ К СЛЕДУЮЩЕМУ k
КОНЕЦ ЦИКЛА

ЕСЛИ $\text{Vol}([\hat{a}, \hat{b}]) > v$ ТО $v := u, L_i := [\hat{a}, \hat{b}]$

КОНЕЦ ЦИКЛА

Общая схема работы метода выглядит следующим образом. Текущий список параллелепипедов инициализируется исходным параллелепипедом. Затем, в главном внешнем цикле из текущего списка извлекается один из параллелепипедов, для которого с использованием верхних и нижних оценок строятся Лебеговские множества. Далее с помощью процедуры MAXPARINT строятся покрывающие параллелепипеды, и вычисляются их дополнения с помощью процедуры COMPLEMENT. Список параллелепипедов, полученный в результате, добавляется к текущему списку. Итерации главного цикла выполняются до тех пор, пока текущий список не становится пустым.

IV. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ

Разработанные алгоритмы построения покрывающих множеств реализованы в рамках программного комплекса BNB-Solver [7] на языке C++. Реализация базируется на классах CutFactory и CutApplicator. Класс CutFactory (Листинг 1) задает интерфейс для генерации Лебеговских множеств.

```
template <class FT> class CutFactory {
public:
/**
 * Получение набора Лебеговских множеств
 *
 * box - текущий параллелепипед
 * cuts - вектор Лебеговских множеств
 */
virtual void getCuts(const Box<FT>& box,
std::vector< Cut<FT> >& cuts) = 0;
};
```

Листинг 1. Интерфейс для генерации Лебеговских множеств

Единственный метод `getCuts()` этого класса создает набор Лебеговских множеств по заданному параллелепипеду. Так как Лебеговские множества задаются неравенствами, их можно рассматривать как вариант отсечений, что оправдывает употребление слова “cut” (отсечение).

Класс CutApplicator (Листинг 2) задает интерфейс для классов, реализующих построение покрывающих

множеств по совокупности Лебеговских и определение дополнения Лебеговского множества до текущего параллелепипеда.

```
template <class FT> class CutApplicator {
public:
/**
 * Определяет покрывающее множество
 * и его дополнение
 *
 * Параметры:
 * cuts - список Лебеговских множеств
 * box - текущий параллелепипед
 * v - список параллелепипедов,
 * составляющих дополнение
 * покрывающего множества до
 * текущего параллелепипеда
 */
virtual void ApplyCut(const std::vector<
Cut<FT> > &cuts, const Box<FT> &box, std::vector<
Box<FT> > &v) = 0;
};
```

Листинг 2. Интерфейс для определения покрывающего множества и его дополнения

Метод `ApplyCut` класса `CutApplicator` получает на вход список Лебеговских множеств и текущий параллелепипед. Далее реализация должна определить покрывающие множества, включающиеся в данные Лебеговские и вычислить дополнения покрывающих множеств до текущего параллелепипеда. Дополнение также должно быть представлено как список параллелепипедов.

В текущей реализации от базового класса `CutApplicator` наследуются два класса `BasicCutApplicator` и `SmartCutApplicator` (Рис. 3). Первый реализует традиционный механизм, предложенный в [4] и состоящий в том, что покрывающее множество полагается равным текущему параллелепипеду, если он целиком включается в Лебеговское множество или равным пустому множеству в противном случае. Класс `SmartCutApplicator` реализует один из двух механизмов построения покрывающих множеств, описанных в предыдущем разделе. Какой из механизмов используется, определяется соответствующей опцией.

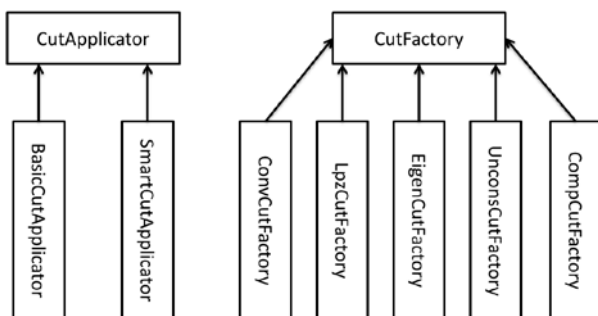


Рис.3. Иерархия классов для работы с Лебеговскими множествами

Класс `CutFactory` наследуется пятью классами (Рис. 3). Класс `ConvCutFactory` создает Лебеговское множество, совпадающее с текущим параллелепипедом, если нижняя оценка на спектр Гессиаана положительная и точка минимума целевой функции на этом множестве находится локальным методом. Класс `LpzCutFactory`

генерирует Лебеговское множество в виде шара на основе Липшицевой миноранты первого порядка. Класс `EigenCutFactory` формирует Лебеговское множество в виде внутренности шара, внешности шара или полупространства в зависимости от значений границ спектра Гессиаана. Дополнительные возможности для формирования Лебеговских множеств дает случай безусловной оптимизации, рассмотренный в [8]. Этот случай реализован в классе `UnconsCutFactory`. И, наконец, класс `ComprCutFactory` позволяет создавать генератор Лебеговских множеств на основе композиции нескольких генераторов. Эти генераторы применяются последовательно к текущему параллелепипеду.

Класс `vnCSolver` организует главный цикл метода неравномерных покрытий, используя при этом экземпляры классов, наследующих интерфейсы `CutFactory` и `CutApplicator` для того, чтобы создавать Лебеговские множества и определять их дополнения. Такой достаточно гибкий механизм позволяет легко изменять способы построения и исключения Лебеговских множеств, добавляя новые модули и не меняя ядро реализации.

V. РЕЗУЛЬТАТЫ ВЫЧИСЛИТЕЛЬНЫХ ЭКСПЕРИМЕНТОВ

A. Непрерывные переменные

Для проведения вычислительных экспериментов был выбраны следующие тестовые функции двух переменных [9]:

1. Произведение параметров

$$f(x) = x_1 x_2, -a \leq x_1 \leq a, -a \leq x_2 \leq a,$$

$$f^* = -a^2, x^* = (-a, a).$$

2. Гиперболическая функция

$$f(x) = x_1^2 - x_2^2, -a \leq x_1 \leq a, -a \leq x_2 \leq a,$$

$$f^* = -a^2, x^* = (0, a).$$

3. Функция Алуффи-Пентини

$$f(x) = 0.25 x_1^4 - 0.5 x_1^2 + 0.1 x_1 + 0.5 x_2,$$

$$-a \leq x_1 \leq a, -a \leq x_2 \leq a,$$

$$f^* = -0.35, x^* = (-1.0465, 0).$$

4. Функция «Трехгорбый верблюд»

$$f(x) = 2x_1^2 - 1.05 x_1^4 + \frac{1}{6} x_1^6 + x_1 x_2 - 4x_1^2 + 4x_2^2,$$

$$-a \leq x_1 \leq a, -a \leq x_2 \leq a,$$

$$f^* = 0, x^* = (0, 0).$$

5. Функция «Шестигорбый верблюд»

$$f(x) = 4x_1^2 - 2.1 x_1^4 + \frac{1}{3} x_1^6 + x_1 x_2 - 4x_1^2 + 4x_2^2,$$

$$-a \leq x_1 \leq a, -a \leq x_2 \leq a,$$

$$f^* = -1.0316, x^* = (-0.089842, 0.712656).$$

6. Задача Ньюмаера

$$f(x) = (x_1 - 1)^2 + (x_2 - 1)^2 - x_1 x_2,$$

$$-a \leq x_1 \leq a, -a \leq x_2 \leq a,$$

$$f^* = -2, x^* = (2, 2).$$

7. Функция Розенброка

$$f(x) = 100(x_2 - x_1^2)^2 + (x_1 - 1)^2,$$

$$-a \leq x_1 \leq a, -a \leq x_2 \leq a,$$

$$f^* = -2, x^* = (2, 2).$$

8. Функция Буга

$$f(x) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2,$$

$$-a \leq x_1 \leq a, -a \leq x_2 \leq a,$$

$$f^* = 0, x^* = (1, 3).$$

9. Функция Гольдштейна-Прайса

$$f(x) = (1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2))$$

$$(30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)),$$

$$-a \leq x_1 \leq a, -a \leq x_2 \leq a,$$

$$f^* = 3, x^* = (0, -1).$$

10. Модифицированная функция Розенброка

$$f(x) = 100(x_2 - x_1^2)^2 + (6.4(x_2 - 0.5)^2 - x_1 - 0.6)^2,$$

$$-a \leq x_1 \leq a, -a \leq x_2 \leq a,$$

$$f^* = 0, x^* = (1, 1), (0.3412, 0.1164).$$

11. Функция Белли

$$f(x) = (1.5 - x_1 + x_1x_2)^2 + (2.25 - x_1 + x_1x_2^2)^2$$

$$+ (2.625 - x_1 + x_1x_2^3)^2,$$

$$-a \leq x_1 \leq a, -a \leq x_2 \leq a,$$

$$f^* = 0, x^* = (3, 0.5).$$

12. Функция Зеттля

$$f(x) = (x_1^2 + x_2^2 - 2x_1)^2 + 0.25x_1,$$

$$-a \leq x_1 \leq a, -a \leq x_2 \leq a,$$

$$f^* = -0.0003791, x^* = (-0.0029, 0).$$

Для всех примеров брался исходный параллелепипед $[-20, 20] \times [-20, 20]$.

Проводилось две серии экспериментов. В первой исходное значение рекорда полагалось равным числу, заведомо большему, чем минимум. В этой серии проверялось, насколько быстро поиск продвигается к точкам, близким к глобальному минимуму и эффективность сокращения допустимого множества. Во второй серии начальное значение рекорда полагалось равным значению глобального минимума. Тем самым тестировалась только эффективность доказательства оптимальности при известном минимуме.

Для построения Лебеговских множеств использовался класс `EigenCutFactory`, формирующий Лебеговское множество в виде внутренности шара, внешности шара или полупространства в зависимости от значений границ спектра Гессииана. Средства автоматизации построения границ спектра Гессииана для полиномов, встроенные в используемую библиотеку `BNB-Solver`, позволили существенно упростить процесс тестирования.

Сравнивались две алгоритмические схемы. В первой

схеме, предложенной ранее в [5], находится такой параллелепипед $L_i \subseteq L(\mu_i, X_i, f(x_i) - \varepsilon)$, что его дополнение $X_i \setminus L_i$ до параллелепипеда X_i также является параллелепипедом. Вторая схема, предложенная в данной работе, предполагает, что строится параллелепипед максимального объема, дополнением которого является совокупность параллелепипедов.

В таблице 1 приведено число итераций для первой и второй схем. Для каждой функции приведено два столбца значений: первый соответствует заведомо завышенному рекорду (10^5), второй – предустановленному точному значению минимума. Точность расчетов во всех случаях полагалась равной 10^{-3} .

Таблица 1. Результаты для непрерывной постановки

| № функ. | 1 | | 2 | | 3 | |
|---------|---------|-------|--------|---------|--------|--------|
| | Схема 1 | 25759 | 1 | 1191451 | 55 | 2367 |
| Схема 2 | 23217 | 1 | 195792 | 70 | 1739 | 71 |
| № функ. | 4 | | 5 | | 6 | |
| | Схема 1 | 527 | 527 | 8279 | 371 | 10015 |
| Схема 2 | 512 | 512 | 8114 | 318 | 10015 | 1 |
| № функ. | 7 | | 8 | | 9 | |
| | Схема 1 | 5221 | 5221 | 219161 | 157 | 616435 |
| Схема 2 | 4816 | 4816 | 219161 | 157 | 325477 | 71327 |
| № функ. | 10 | | 11 | | 12 | |
| | Схема 1 | 2793 | 765 | 221465 | 8723 | 531 |
| Схема 2 | 1732 | 694 | 188650 | 7659 | 491 | 459 |

Как показывают проведенные эксперименты, схема 1 в большинстве случаев дает эффект сокращения числа итераций. Но, как правило, сокращение не очень существенное (1.2-2 раза).

В. Все переменные целочисленные

В первой серии экспериментов предполагалось, что все переменные непрерывны, а во второй, что они принимают только целочисленные значения. Для тестирования были выбраны примеры 1, 2, 4, 6, 7, 9, в которых минимум достигается в точке с целыми координатами.

Таблица 2. Результаты для целочисленной постановки

| № функ. | 1 | | 2 | | 4 | |
|---------|---------|----|-----|-----|------|------|
| | Схема 1 | 74 | 1 | 72 | 23 | 245 |
| Схема 2 | 40 | 1 | 38 | 25 | 237 | 237 |
| № функ. | 6 | | 7 | | 9 | |
| | Схема 1 | 15 | 1 | 279 | 279 | 3450 |
| Схема 2 | 15 | 1 | 221 | 221 | 1566 | 1566 |

Приведенные результаты показывают, что применение предлагаемой в работе схемы позволяет заметно, но не радикально сократить число итераций метода неравномерных покрытий.

В отношении приведенных экспериментов следует сделать одно важное замечание. Целью экспериментов было выяснение эффективности предложенного подхода с точки зрения сокращения числа итераций метода неравномерных покрытий. С целью унификации условий экспериментов всегда применялась одна и та же

миноранта, хотя на практике выбор миноранты зависит от решаемой задачи.

потенциалов межатомного взаимодействия // ЖВМиМФ, том 54, № 12, с. 1994–2001, 2014.

VI. ЗАКЛЮЧЕНИЕ

В данной работе рассмотрен подход, позволяющий увеличить производительность метода неравномерных покрытий, основанный на решении вспомогательной оптимизационной задачи. Предложенный подход позволяющий увеличить объем покрываемых множеств, и тем самым, ускорить метод неравномерных покрытий. Приведено описание реализации предложенного подхода в рамках программного комплекса для решения задач глобальной оптимизации и результаты вычислительных экспериментов. Результаты экспериментов показали, что для большинства тестовых примеров удается достичь ускорения метода неравномерных покрытий.

В дальнейшем планируется рассмотреть комбинации различных минорант, применяемых для построения покрываемых множеств. В частности, возможно рассмотрение интервальных минорант [6], вогнутых минорант [3] и прочих подходов к получению нижних оценок целевой функции. Несомненный интерес представляет апробация данного метода для задач с функциональными ограничениями.

Разрабатываемые в данной работе подходы планируется применить при решении практических задач химических технологий [10] и нанотехнологий [11].

БИБЛИОГРАФИЯ

- [1] Кочетов Ю. А., Плясунов А. В. Генетический локальный поиск для задачи о разбиении графа на доли ограниченной мощности // Журнал вычислительной математики и математической физики. – 2012. – Т. 52. – №. 1. – С. 164-176.
- [2] Стронгин Р. Г. Параллельная многоэкстремальная оптимизация с использованием множества разверток // Журнал вычислительной математики и математической физики. – 1991. – Т. 31. – №. 8. – С. 1173-1185.
- [3] Хамисов О. В. Невыпуклая оптимизация с нелинейными опорными функциями // Труды Института математики и механики УрО РАН. – 2013. – Т. 19. – №. 2. – С. 295-306.
- [4] Евтушенко Ю. Г. Численный метод поиска глобального экстремума функций (перебор на неравномерной сетке) // Журнал вычислительной математики и математической физики. – 1971. – Т. 11. – №. 6. – С. 1390-1403.
- [5] Евтушенко Ю. Г., Посыпкин М. А. Применение метода неравномерных покрытий для глобальной оптимизации частично целочисленных нелинейных задач // Журнал вычислительной математики и математической физики. – 2011. – Т. 51. – №. 8. – С. 1376-1389.
- [6] Kearfott R. B. Rigorous global search: continuous problems. – Springer Science & Business Media, 2013. – Т. 13.
- [7] Evtushenko Y., Posypkin M., Sigal I. A framework for parallel large-scale global optimization // Computer Science-Research and Development. – 2009. – Т. 23. – №. 3-4. – С. 211-215.
- [8] Evtushenko Y., Posypkin M. A deterministic approach to global box-constrained optimization // Optimization Letters. – 2013. – Т. 7. – №. 4. – С. 819-829.
- [9] Jamil M., Yang X. S. A literature survey of benchmark functions for global optimisation problems // International Journal of Mathematical Modelling and Numerical Optimisation. – 2013. – Т. 4. – №. 2. – С. 150-194.
- [10] Беспалов А. В., Чечёткина Е. М. Гидродинамические модели и режимы течения различных сред через неупорядоченные и упорядоченные структуры // Труды. – 2009. – Т. 46. – С. 153-157..
- [11] Абгарян К.К., Посыпкин М. А., Применение оптимизационных методов для решения задач параметрической идентификации

Intensification of search domain contraction in non-uniform covering method

A.A. Abdildaeva, M.A. Posypkin, E.M. Chechetkina

Abstract – In this paper we further develop the nonuniform covering method belonging to the class of deterministic global optimization methods. We propose an approach based on solving an auxiliary optimization problem that allows to increase the amount of covering sets thereby accelerating the nonuniform covering method. The paper describes the implementation of the proposed approach in the framework of a program complex for solving global optimization and computational results.

Keywords – deterministic optimization methods, global optimization, the nonuniform covering method, software for optimization