# ПАРАЛЛЕЛЬНЫЙ МЕТОД ГЛОБАЛЬНОЙ ОПТИМИЗАЦИИ ЛИПШИЦЕВЫХ ФУНКЦИЙ С ЭФФЕКТИВНЫМ ОЦЕНИВАНИЕМ КОНСТАНТЫ ЛИПШИЦА


Работу выполнил

Студент Горбунов Алексей Степанович

Группа М210

Научный руководитель

Доцент, д.ф.-м.н. Посыпкин Михаил Анатольевич

# Постановка задачи

Задачей глобальной оптимизации называется поиск оптимального решения (экстремума) целевой функции на заданном регионе поиска. В данной работе мною рассматривается задача глобальной оптимизации с интервальными ограничениями:

$$f(x) \to opt, x_i \in [a_i, b_i], i = 1, \dots, n$$

Ограничения задачи, определяемые данной формулой, задают гиперинтервал (n-мерный параллелепипед) $S = [a_1, b_1] \times \cdots \times [a_n, b_n]$.

Одним из основных предположений о целевой функции для решения задачи глобальной оптимизации является предположение об ограниченности относительных изменений целевой функции. В данном случае говорят о решении задачи липшицевой глобальной оптимизации, а целевая функция является липшицевой, поскольку удовлетворяет условию

$$|f(x) - f(y)| \le L\|x - y\|, x, y \in R^n, f: R^n \to R$$

# Цель работы

Разработать метод липшицевой глобальной оптимизации, позволяющего ускорить нахождение решения рассматриваемой задачи за счет применения параллельных вычислений:

- выбор базового подхода для построения алгоритма глобальной оптимизации;
- разработка эффективного и точного способа оценивания константы Липшица для целевой функции на некотором гиперинтервале;
- разработка программной реализации алгоритма, с применением параллельных вычислений;
- Оценка эффективности разработанного метода с различными функциями глобальной оптимизации.

# Актуальность работы

Потребность в решении сложных задач глобальной оптимизации постоянно возрастает. Это связано, в первую очередь, с появлением всё большего числа прикладных задач в самых разных сферах жизни.

Примеры задач глобальной оптимизации:

- экономические задачи о составлении оптимальных графиков;
- задачи о распределении ресурсов и инвестиций;
- задачи нахождения оптимальных параметров материалов и конструкций при проектировании сложных технических объектов;
- задачи оптимального использования и распределения энергетических ресурсов.

Актуальной становится задача эффективного использования вычислительных ресурсов. Одним из основных подходов является организация параллельных вычислений. Таким образом, возможность многопоточного вычисления определяет вектор развития современных высокоэффективных алгоритмов глобальной оптимизации.

# Подходы к решению задачи глобальной оптимизации

Неадаптивные:

- Поиск на равномерной сетке
- Случайный поиск

Адаптивные:

- Метод ветвей и границ
- Градиентные методы
- Методы имитации отжига
- Байесовские методы
- Методы поиска с запретами
- Генетические алгоритмы

# Способы оценивания константы Липшица

- Заданное значение константы Липшица
- Набор возможных значений
- Адаптивное глобальное оценивание
- Адаптивное локальное оценивание

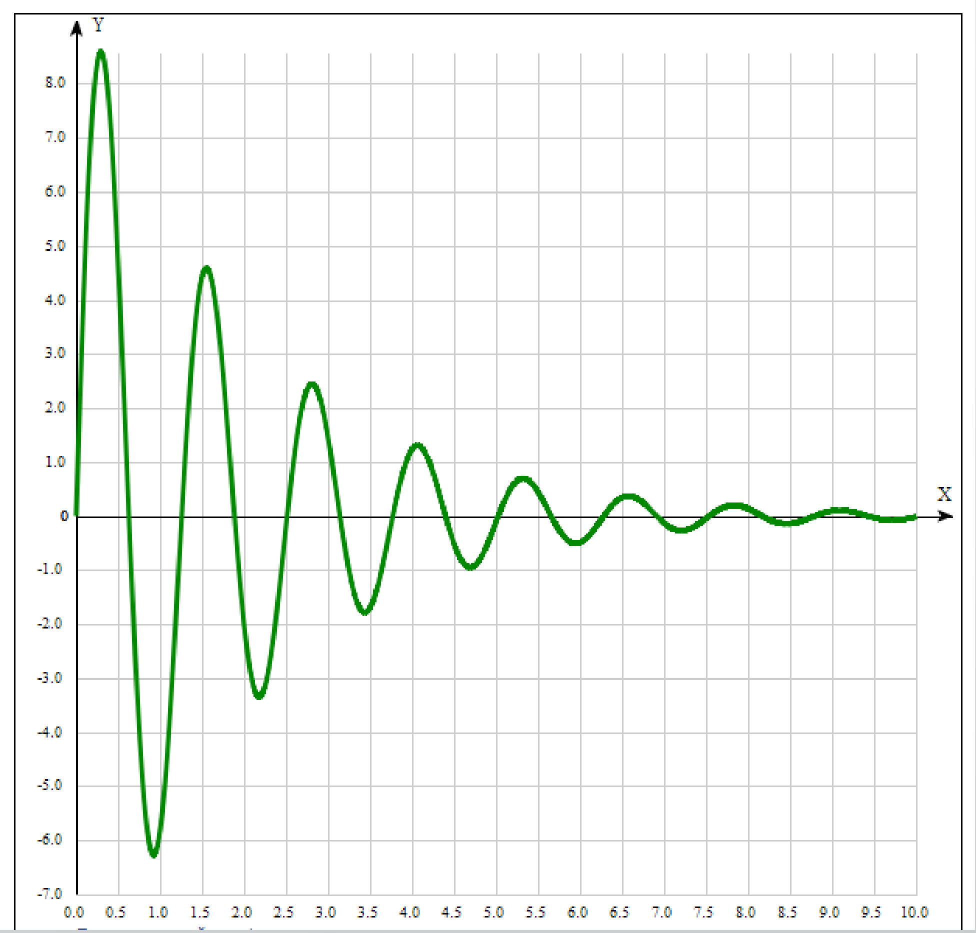



График функции $f(x) = \frac{10\sin(5x)}{\sqrt[2]{e^x}}$

# Метод ветвей и границ при решении задачи глобальной оптимизации

Работа метода начинается с одного гиперинтервала - исходного региона поиска. Для него определим верхнюю и нижнюю оценки минимума функции. После проверим гиперинтервал на соответствие условию отсева. Если условие выполнено, гиперинтервал исключается. Если условие не выполнено, разделим исходный гиперинтервал пополам.

Минимальную из всех верхних оценок минимума, полученных на данной итерации, а также и на всех предыдущих итерациях, назовем текущим рекордом $f^*$.

Условие отсева для метода ветвей и границ:

$$f_l \geq f^* - \varepsilon,$$

где $f_l$ – нижняя оценка минимума функции на данном гиперинтервале, а $\varepsilon$ – требуемая точность нахождения минимума.

# Нахождение верхней и нижней оценок минимума функции

Можно применить поиск на равномерной сетке.

Вычислив значения целевой функции во всех узлах сетки, можем вычислить верхнюю и нижнюю оценки минимума функции на исследуемом гиперинтервале.

Формула для вычисления верхней оценки минимума целевой функции:

$$f_u = \min_{x \in \mathrm{X}} f(x)$$

Допустив условие липшицевости целевой функции, можно найти нижнюю границу значений функции на гиперинтервале, используя значение константы Липшица.

$$f_l = f_u - L * \delta$$

где $\delta = \sum_{i=1}^{N} \frac{b_i - a_i}{2}$, где $N$ – размерность задачи, а $b_i, a_i$ соответствующие границы исходного региона поиска $S$.

# Оценка константы Липшица

Формула для нахождения значения константы Липшица на гиперинтервале будет следующей:

$$L = \max_{x',x'' \in P} \frac{|f(x') - f(x'')|}{||x' - x''||}$$

где $P$ – заданный гиперинтервал.

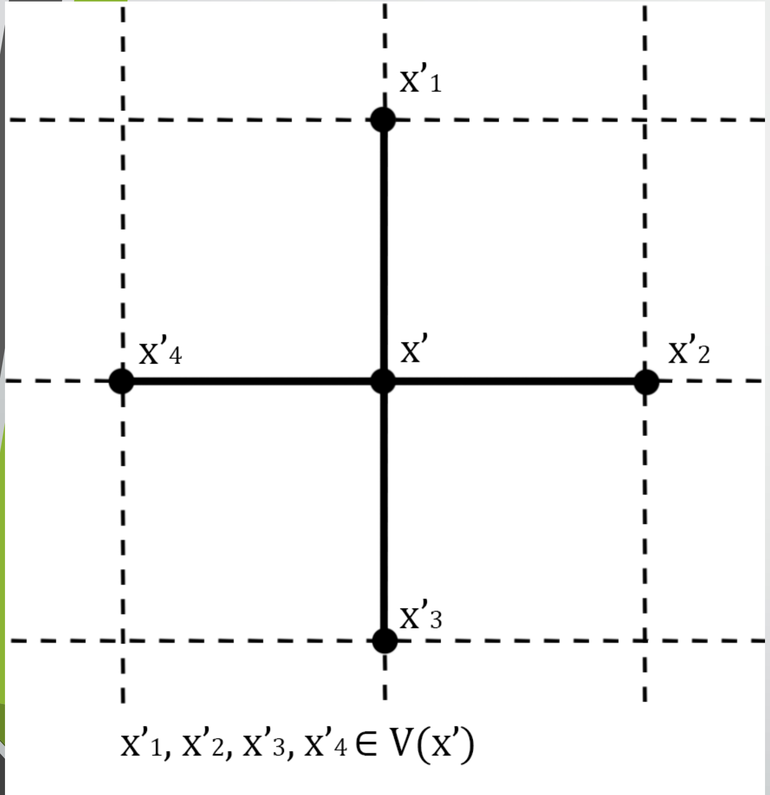
Обозначим множество узлов сетки с числом узлов на одно измерение $n$ как $N(n)$.

$$L^* = \max_{x',x'' \in N(n)} \frac{|f(x') - f(x'')|}{||x' - x''||}$$

Точность оценки зависит от гранулярности разбиения. Чтобы компенсировать возможную ошибку вводится коэффициент надежности

$$L^{**} = \kappa L^*$$

# Оценка константы Липшица



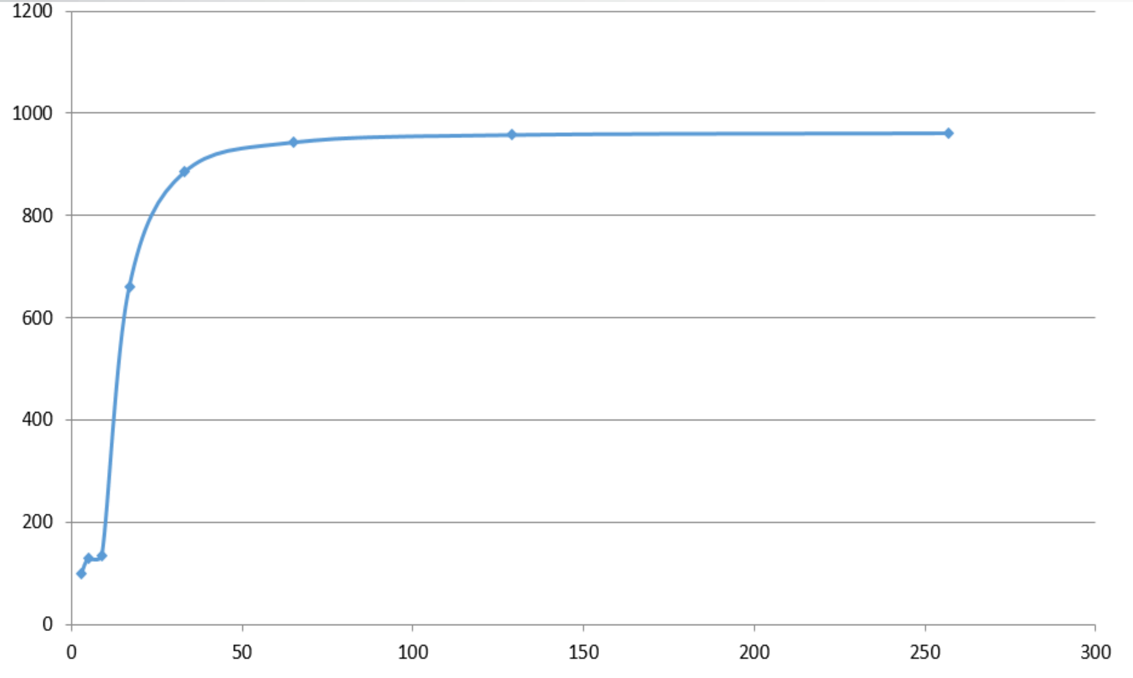

Для сетки $N(n)$ на некотором гиперинтервале верно:

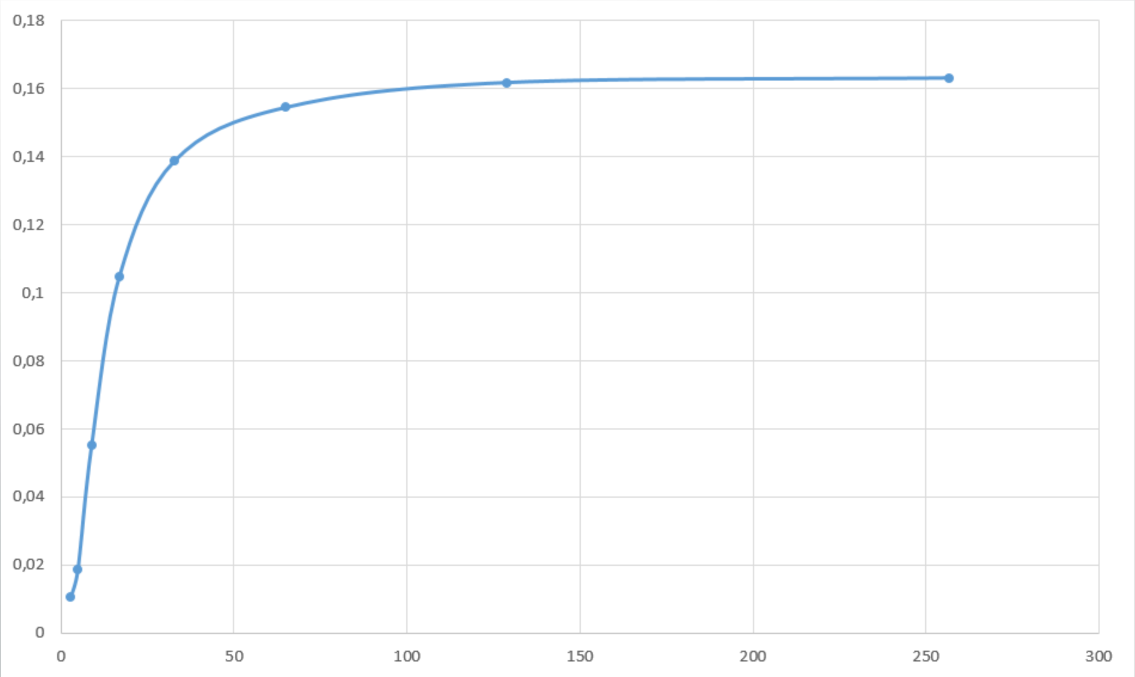
$$L^* = \max_{x',x''\in N(n)} \frac{|f(x') - f(x'')|}{||x' - x''||_1} = \max_{x'\in N(n)} \max_{x''\in V(x')} \frac{|f(x') - f(x'')|}{||x' - x''||_1}$$

где $V(x')$ – множество точек, соседних с $x'$, то есть множество точек, отстоящих от $x'$ лишь по одному измерению ровно на один шаг сетки вдоль этого измерения (соседние узлы сетки).

# Модификация метода для более точного оценивания

Было проведено исследование зависимости значения оценки константы Липшица от величины ячейки сетки. Анализ полученных графиков подтвердил предположение о том, что получаемая последовательность будет неубывающей. Оказалось, что для большинства функций полученный график зависимости имеет асимптотический вид.



# Модификация метода для более точного оценивания

Был разработан метод оценки константы Липшица с помощью трех сеток $G_5, G_9, G_{17}$, разбивающих гиперинтервал по каждому измерению, соответственно, на 4, 8 и 16 частей. Получаемые на этих сетках оценки $L(4), L(8), L(16)$ сравниваются с целью определения возможности их использования для определения близкого к точному значения константы Липшица.

Критерий возможности использования оценок для расчета константы Липшица:

$$[L(8) - L(4) > L(16) - L(8)]\&[L(16) - L(8) < 0.05 * L(16)]\&[L(8) - L(4) < 0.1 * L(8)]$$

Если условие выполнено, найдем близкое к точному значение константы Липшица, используя следующую оценку:

$$L^* = L(16) + \frac{L(16) - L(8)}{2}$$

# Модификация метода для более точного оценивания

Свойства разработанного метода:

1. Монотонность

$$L(n_1) \leq L(n_2) \leq L$$

2. Согласованность

$$|f(x') - f(x'')| \leq L^* \left||x' - x''|\right|, x', x'' \in N(n)$$

3. Однородность

$$\lim_{n \to \infty} L(n) = L$$

4. Возможность организации параллельных вычислений

# Реализация

Разработанный алгоритм был реализован в виде программы на языке C++.

При этом было реализовано два варианта: последовательный и параллельный.

При реализации параллельного алгоритма была применена библиотека OpenMP.

Наличие двух вариантов алгоритма позволяет произвести сравнение быстродействия и оценить эффективность распараллеливания.

# Экспериментальные результаты

| Точность 0.0001 | GKLS размерность задач 2 (100 задач) | GKLS размерность задач 3 (100 задач) |
|---|---|---|
| Время выполнения последовательный, мс | 4891 | 206976 |
| Время выполнения паралельный, мс | 1799 | 56731 |
| Ускорение, раз | 2,72 | 3,65 |

Набор тестовых функций GKLS

| Функция оптимизации, точность 0.0001 | Время последовательного расчета, мс | Время параллельного расчета, мс | Ускорение, раз |
|---|---|---|---|
| Sphere, 3 | 379 | 119 | 3,18 |
| Rosenbrock, 3 | 1099290 | 386898 | 2,84 |
| Beale, 2 | 4069 | 1766 | 2,32 |
| Goldstein Price, 2 | 4513 | 1975 | 2,29 |
| Booth, 2 | 343 | 127 | 2,71 |
| Matyas, 2 | 253 | 89 | 2,84 |
| Himmelblau, 2 | 856 | 271 | 3,16 |
| Camel Three Hump, 2 | 791 | 395 | 2,01 |
| Table 2 Holder Table 2, 2 | 1547 | 723 | 2,14 |
| McCormick, 2 | 174 | 81 | 2,15 |

Некоторые задачи из набора mathexplib

# Заключение

Был разработан параллельный метод глобальной липшицевой оптимизации.

От теоретического описания был осуществлен переход к практической реализации в виде программы на языке программирования C++.

При этом было реализовано две версии алгоритма: последовательная и параллельная.

Результаты экспериментов показывают применимость алгоритма: при испытаниях алгоритм успешно справился с отысканием глобального минимума функций глобальной оптимизации с заданной точностью. При этом параллельная версия алгоритма достаточно эффективна, поскольку позволяет добиться значительного ускорения вычислений.

СПАСИБО ЗА ВНИМАНИЕ!