

Федеральное государственное автономное образовательное учреждение
высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»
Физтех-школа Прикладной Математики и Информатики
Кафедра интеллектуальных систем

Направление подготовки / специальность: 03.03.01 Прикладные математика и физика
(бакалавриат)

Направленность (профиль) подготовки: Компьютерные технологии и
интеллектуальный анализ данных

АДАПТИВНОСТЬ ГРАДИЕНТНЫХ МЕТОДОВ

(бакалаврская работа)

Студент:

Плетнев Никита Вячеславович

(подпись студента)

Научный руководитель:

Гасников Александр Владимирович,
д-р физ.-мат. наук, доц.

(подпись научного руководителя)

Консультант (при наличии):

(подпись консультанта)

Москва 2019

Оглавление

1	Аннотация	2
2	Введение	3
3	Определения и предположения	4
4	Обзор литературы	5
5	Исходный алгоритм	6
6	Адаптивность по константе сильной выпуклости	7
7	Адаптивность по константе Липшица	11
8	Эксперименты	15
9	Заключение	23
10	Ссылки	24

1 Аннотация

Работа посвящена построению более эффективных методов выпуклой оптимизации первого порядка, то есть использующих только значения функции и ее производных. Предлагаются адаптивный по константе сильной выпуклости алгоритм ACGM, основанный на рестартах быстрого градиентного метода OGM-G с обновлением оценки константы сильной выпуклости, и адаптивный по константе Липшица градиента метод ALGM, в котором применение рестартов OGM-G дополнено подбором константы Липшица с проверкой условий выпуклости, используемых в методе универсального градиентного спуска. При этом устраняются недостатки исходного метода, связанные с необходимостью знания данных констант для определения числа шагов и длины шага. Доказываются оценки для сложности построенных алгоритмов. Для проверки полученных результатов проводятся эксперименты на модельной функции.

Ключевые слова: выпуклая оптимизация, методы первого порядка, быстрые градиентные методы, адаптивность по константе сильной выпуклости, адаптивность по константе Липшица градиента.

2 Введение

Работа посвящена методам оптимизации первого порядка, то есть методам, использующим лишь значения функции и ее градиента.

Задачи оптимизации функций высокой размерности имеют многообразные приложения, например, в машинном обучении, управлении, экономике и энергетике. Методы первого порядка пользуются большой популярностью, потому что их реализация обладает относительно невысокой вычислительной сложностью: требует вычисления только значения функции, ее градиента и простейших векторных операций.

В настоящее время активно развиваются быстрые градиентные методы, основанные на следующей идее: задается число операций, строятся оптимальные для данного числа операций последовательности коэффициентов, которые используются для получения последовательности точек. Такой подход реализован в статье [1] (метод OGM-G), а общее описание можно найти в пособии [2].

Проблема данного подхода заключается в том, что требуемое для достижения заданного результата, например, уменьшения нормы градиента вдвое, число итераций неизвестно. Поэтому для эффективного применения подобных методов необходимо оценивать это число.

В пособии [2] предлагается способ оценки, но он требует знания константы сильной выпуклости μ . Также там указана предложенная Ю. Е. Нестеровым идея применения быстрого градиентного метода с оцениванием данного параметра и обновлением его значения при каждом рестарте.

Предлагается применить тот же подход к оцениванию параметра L .

Другим недостатком данных методов является фиксированный шаг $\frac{1}{L}$. Его наличие приводит к замедлению сходимости, хоть и гарантирует ее наличие. И этот параметр — константа Липшица для градиента — неизвестен, как и константа сильной выпуклости. Идея по подбору данного параметра реализована в алгоритме универсального градиентного спуска Ю. Е. Нестерова (параграф 5 пособия [2]).

Основным содержанием работы являются реализация и экспериментальная проверка данных идей. Структура работы: в разделе 3 вводятся используемые определения и обозначения, раздел 4 содержит обзор текущего состояния литературы по теме, в разделе 5 описывается исходный алгоритм OGM-G; раздел 6 посвящен построению алгоритма ACGM, адаптивного по константе сильной выпуклости, а в разделе 7 строится ALGM, адаптивный по константе Липшица

градиента. В разделе 8 проводится экспериментальная проверка полученных результатов, а раздел 9 — заключительный, в нем собраны основные выводы.

3 Определения и предположения

Решается задача безусловной минимизации:

$$\min_{x \in \mathbb{R}^d} f(x). \quad (1)$$

Предполагается, что решение

$$x^* = \arg \min_{x \in \mathbb{R}^d} f(x) \quad (2)$$

существует, а градиент функции $f(x)$ удовлетворяет условию Липшица с константой $L > 0$:

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\| \quad \forall x, y \in \mathbb{R}^d. \quad (3)$$

Также считается, что функция $f(x)$ является сильно выпуклой с неизвестной нам константой $\mu > 0$:

$$\frac{\mu}{2}\|x - x^*\|^2 \leq f(x) - f(x^*) \leq \frac{1}{2\mu}\|\nabla f(x)\|^2. \quad (4)$$

Первое неравенство напрямую следует из определения, второе доказывается в соответствии с [2]:

$$\begin{aligned} f(x^*) &= \min_y f(y) \geq \min_y \left(f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu}{2}\|x - y\|^2 \right) = \\ &= f(x) - \frac{1}{2\mu}\|\nabla f(x)\|^2. \end{aligned}$$

В качестве невязки нахождения точки экстремума функции используется норма градиента. Критерий останова выглядит так:

$$\|\nabla f(x)\| \leq \varepsilon. \quad (5)$$

Траекторией метода оптимизации называется последовательность порождаемых им точек.

Рестартом называется перезапуск метода с использованием результата предыдущего запуска в качестве начального значения.

Константа, с которой функция удовлетворяет определению сильной выпуклости в окрестности некоторой части траектории, превосходящая μ , обозначается μ^{loc} . Аналогично определяется L^{loc} .

Алгоритм называется адаптивным по некоторому параметру, если его применение не требует никаких предположений о значении данного параметра.

4 Обзор литературы

Использованная в работе статья [1] посвящена построению оптимальной последовательности коэффициентов для быстрого градиентного метода. Построенный в ней метод OGM-G является оптимальным среди методов с фиксированным числом шагов, в статье доказаны оценки для его скорости сходимости. Изложению результатов [1] в части оценок, касающихся целей работы, посвящен раздел 5.

В пособии [2] излагается современное состояние быстрых градиентных методов. Среди прочего, в нем содержатся идеи адаптивного подбора неизвестных констант; на этих идеях построено основное содержание работы — разделы 6 и 7. В частности, в параграфе 5 пособия изложен придуманный Ю. Е. Нестеровым метод адаптивного подбора константы Липшица, известный как универсальный градиентный спуск. На основе данного метода в работе построен быстрый алгоритм, адаптивный как по константе сильной выпуклости, так и по константе Липшица для градиента.

В статье [3] разрабатываются численные методы оптимизации энтропии. Там применен подход, похожий на использованный в работе, в котором подбираемый параметр изменяется в одно и то же число раз β , и суммарное количество шагов оценивается с помощью суммы геометрической прогрессии. После этого из соображений минимизации данной оценки выбирается β . Тот же способ определения оптимального коэффициента для подбора используется в работе.

Пособие [4] посвящено изложению основ теории оптимизации и простых методов, которые служат основой для современных алгоритмов.

Статья [5] посвящена решению близкой задачи — построению адаптивного по константе сильной выпуклости метода стохастического градиентного спуска. Однако в ней цель полностью не достигнута, так как полученный алгоритм является эффективным лишь с точностью до логарифмического множителя.

В статье [6] также строится метод, адаптивный по константе сильной вы-

пуклости, но оценка сложности полученного алгоритма тоже содержит логарифмический множитель.

5 Исходный алгоритм

5.1 OGM-G

В качестве базового алгоритма взят ускоренный градиентный метод с фиксированным шагом OGM-G. В [1] показана его оптимальность в классе методов с заданным числом шагов фиксированной длины.

Вход: $f \in \mathcal{F}_L(\mathbb{R}^d)$, $\mathbf{x}_0 = \mathbf{y}_0 \in \mathbb{R}^d$, $N \geq 1$.

Для $i = 0 \dots N - 1$:

$$\mathbf{y}_{i+1} = \mathbf{x}_i - \frac{1}{L} \nabla f(\mathbf{x}_i);$$

$$\mathbf{x}_{i+1} = \mathbf{y}_{i+1} + \beta_i(\mathbf{y}_{i+1} - \mathbf{y}_i) + \gamma_i(\mathbf{y}_{i+1} - \mathbf{x}_i).$$

Коэффициенты β_i, γ_i вычисляются по формулам:

$$\beta_i = \frac{(\theta_i - 1)(2\theta_{i+1} - 1)}{\theta_i(2\theta_i - 1)}; \gamma_i = \frac{2\theta_{i+1} - 1}{2\theta_i - 1},$$

где последовательность $\{\theta_i\}_{i=0}^N$ строится следующим образом:

$$\theta_i = \begin{cases} \frac{1 + \sqrt{1 + 8\theta_1^2}}{2}, & i = 0; \\ \frac{1 + \sqrt{1 + 4\theta_{i+1}^2}}{2}, & 1 \leq i < N; \\ 1, & i = N. \end{cases}$$

5.2 Оценки

По теореме 2 из [1], при применении OGM-G

$$\|\nabla f(x^N)\|^2 \leq \frac{4L(f(x^0) - f(x^*))}{N^2}. \quad (6)$$

Оттуда же,

$$f(x^N) - f(x^*) \leq \frac{L\|x^0 - x^*\|^2}{N^2}. \quad (7)$$

Из (4) и (6):

$$\|\nabla f(x^N)\|^2 \leq \frac{4L}{N^2} \frac{1}{2\mu} \|\nabla f(x^0)\|^2, \quad (8)$$

или

$$\|\nabla f(x^N)\| \leq \sqrt{\frac{2L}{\mu N^2}} \|\nabla f(x^0)\|. \quad (9)$$

Таким образом, выполнение N итераций гарантирует уменьшение нормы градиента $f(x)$ как минимум вдвое, где

$$N = 2\sqrt{2\frac{L}{\mu}} \quad (10)$$

Согласно лемме 4 статьи [1], полученная оценка является неулучшаемой в худшем случае.

5.3 Недостатки метода

Полученная оценка показывает, что использование OGM-G неявно предполагает, помимо наличия известной константы Липшица, знание константы сильной выпуклости.

Практически во всех реальных случаях применения методов оптимизации ни одно из этих предположений не выполняется: свойства функции заранее неизвестны, а вычисление данных параметров требует нахождения минимума и максимума собственных значений матрицы Гессе, что значительно сложнее, чем исходная задача оптимизации.

Указанные соображения делают оптимальный теоретически метод неприменимым на практике. Решению данной проблемы посвящен следующий раздел.

6 Адаптивность по константе сильной выпуклости

6.1 ACGM

Ю. Е. Нестеровым в пособии [2] предложен способ построения адаптивного по μ алгоритма, основанного на рестартах OGM-G.

В этом разделе OGM-G используется в качестве «черного ящика», получающего на вход функцию f , начальную точку \mathbf{x}_0 , константу Липшица L и константу сильной выпуклости μ . Число итераций N , используемое методом, вычисляется по формуле (10). В дальнейшем применение OGM-G как шага в алгоритмах будет обозначаться как $OGMG(f, \mathbf{x}_0, L, \mu)$.

ACGM — Adaptive by constant of strong Convexity Gradient Method — решает проблему неизвестности μ , инициализируя ее произвольным значением с последующим изменением.

На каждом шаге предполагаемое значение μ умножается на одно и то же $\beta > 1$.

ACGM

Вход: $f \in \mathcal{F}_L(\mathbb{R}^d)$, $\mathbf{x}_0 \in \mathbb{R}^d$, L , μ_0 , β , ε .

Пока не выполнено условие остановки (5):

k — номер шага.

1 $\mu_k := \beta \mu_{k-1}$;

2 $\mathbf{x}_k = OGMG(f, \mathbf{x}_{k-1}, L, \mu_k)$;

3 Если выполнено условие $\|\nabla f(x_k)\| \leq \frac{1}{2} \|\nabla f(x_{k-1})\|$, то перейти к следующему шагу;

4 Иначе $\mu_k := \frac{\mu_k}{\beta}$ и вернуться к пункту 2. Если при этом выполнено условие $\|\nabla f(x_k)\| < \|\nabla f(x_{k-1})\|$, то x_{k-1} заменяется на x_k .

6.2 Оценки

В результате применения ACGM очередное уменьшение вдвое нормы градиента будет выполнено за

$$2\sqrt{2\frac{L}{\mu_k^{init}}} + 2\sqrt{2\frac{L}{\mu_k^{init}/\beta}} + \dots + 2\sqrt{2\frac{L}{\mu_k^{init}/\beta^m}} = \sqrt{8\frac{L}{\mu_k}} \sum_{i=0}^m \frac{1}{\sqrt{\beta^i}} \lesssim \frac{\sqrt{8\beta}}{\sqrt{\beta}-1} \sqrt{\frac{L}{\mu_k}}$$

итераций метода OGM-G, где m — количество повторений цикла на шаге k , а индекс *init* указывает на то, что в формуле используется не конечное значение переменной, а то, которым она была инициализирована.

При этом μ_k отличается не более чем в β раз от μ^{loc} . Использование значения μ , подходящего для всего пространства, могло бы повысить количество операций.

Действительно, если последовательные n точек траектории ACGM лежат в области, в которой $f(\mathbf{x})$ сильно выпукла с константой $\mu^{loc} \geq \beta^s \mu$, то в данных точках ACGM применяется с $\mu_0 \geq \frac{\mu^{loc}}{\beta} \geq \beta^{s-1} \mu$. Тогда количество обращений к вычислению градиента для каждого уменьшения его нормы вдвое оказывается не более $2\sqrt{2\frac{L}{\beta^{s-1}\mu}}$ — то есть, в $\beta^{\frac{s-1}{2}}$ раз меньше, чем при $\mu_k \equiv \mu$.

Суммарное количество итераций при работе ACGM с использованием критерия останова (5) оценивается следующим образом. Требуется выполнить $K = \log_2 \frac{\|\nabla f(x^0)\|}{\varepsilon}$ шагов. Каждый шаг содержит $O\left(\sqrt{\frac{L}{\mu}}\right)$ итераций, поэтому

алгоритм завершит работу, выполнив $O\left(\sqrt{\frac{L}{\mu}} \log_2 \frac{\|\nabla f(x^0)\|}{\varepsilon}\right)$ итераций — то есть, вычислений $f(x)$ и $\nabla f(x)$.

Как показано выше, полученная оценка по порядку величины может быть уточнена. Если $\mu_k \geq \frac{\mu_k^{loc}}{\beta}$, то каждый шаг ASCGM содержит не более $\frac{\sqrt{8\beta}}{\sqrt{\beta}-1} \sqrt{\frac{L}{\mu_k}} \leq \frac{\sqrt{8\beta}}{\sqrt{\beta}-1} \sqrt{\frac{L}{\mu_k^{loc}}}$ итераций, соответственно общее количество итераций не превосходит

$$\frac{\sqrt{8\beta}}{\sqrt{\beta}-1} \sum_{k=0}^{K-1} \sqrt{\frac{L}{\mu_k^{loc}}}.$$

Данное вычисление основано на идее оценки из [3].

Минимизация зависящего от β коэффициента дает $\beta = 4$.

Это значение является оптимальным лишь с точки зрения худшего случая, когда $\mu_k = \frac{\mu_k^{loc}}{\beta}$. В реальных случаях, поскольку данное равенство является лишь теоретически возможным предельным случаем, значение коэффициента может оказаться меньше данного, но в любом случае оно превосходит $\inf_{\beta>1} \frac{\sqrt{\beta}}{\sqrt{\beta}-1} = 1$

Таким образом, доказаны теоремы о сходимости построенного метода.

Теорема 1 Алгоритм ASCGM с оптимальным $\beta = 4$ и $\mu_0 > \max_k \mu_k^{loc}$ достигает точки \mathbf{x} , удовлетворяющей критерию останова (5), за не более чем $C \sum_{k=0}^{K-1} \sqrt{\frac{L}{\mu_k^{loc}}}$ вычислений градиента, где $K = \log_2 \frac{\|\nabla f(\mathbf{x}^0)\|}{\varepsilon}$, $C = \frac{\sqrt{8\beta}}{\sqrt{\beta}-1} = 8\sqrt{2}$.

Данная теорема имеет лишь теоретический смысл, поскольку оценка μ_k^{loc} крайне затруднительна. Следующая теорема содержит менее точную, но более удобно применимую оценку.

Так как $\mu_k^{loc} \geq \mu$ при всех k , каждое слагаемое в сумме из теоремы 1 не превосходит $\sqrt{\frac{L}{\mu}}$, откуда сразу следует

Теорема 2 Алгоритм ASCGM с оптимальным $\beta = 4$ и $\mu_0 > \max_k \mu_k^{loc}$ достигает точки \mathbf{x} , удовлетворяющей критерию останова (5), за не более чем $CK \sqrt{\frac{L}{\mu}}$ вычислений градиента, где $K = \log_2 \frac{\|\nabla f(\mathbf{x}^0)\|}{\varepsilon}$, $C = \frac{\sqrt{8\beta}}{\sqrt{\beta}-1} = 8\sqrt{2}$.

6.3 К выбору оптимального μ_0 . Случай $\mu_0 < \mu^{loc}$

Для упрощения вычислений, пусть $\mu_0 = \frac{\mu^{loc}}{4^k}$. Тогда по формуле (10) на первом шаге ASCGM будет выполнено $M = 2\sqrt{\frac{2L \cdot 4^k}{\mu^{loc}}} = 2^k N$ итераций. При этом, согласно (9), норма градиента умножится не более, чем на $\sqrt{\frac{2L}{\mu^{loc} M^2}} = \frac{1}{2^{k+1}}$.

Для достижения такого результата требуется $k+1$ рестартов, то есть $(k+1)N$ итераций при использовании OGM-G.

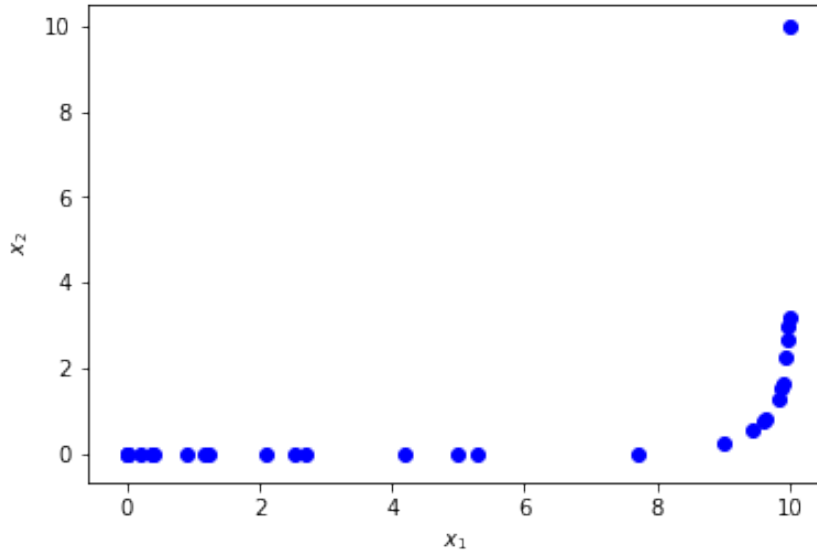
При использовании ASCGM i -ый рестарт выполняется с $\mu = \mu_0 \beta^i$, т. е. потребует $\frac{N}{2^i}$ итераций. Суммарное количество не превосходит $2N$.

Поскольку $2^k > 2$, применение заниженного значения константы сильной выпуклости приводит к значительному увеличению количества итераций.

Объединяя все сказанное о величине начального предполагаемого значения константы сильной выпуклости, оптимальным будет такой выбор μ_0 , что $\mu_0 > \mu_k^{loc}$ при всех k . Таким является, например $\mu_0 = L$.

6.4 Иллюстрации

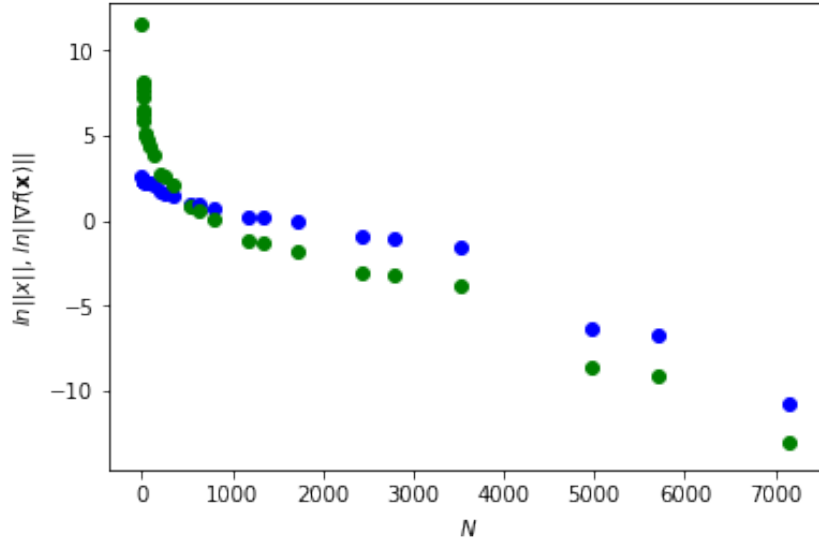
Для иллюстрации работы ASCGM используется функция $f(x_1, x_2)$, свойства которой будут описаны в разделе 8 «Эксперименты».



Траектория ASCGM(f , (10,10), L , L , 4)

Синим цветом отмечен натуральный логарифм $\|\mathbf{x}\|$, зеленым — $\|\nabla f(\mathbf{x})\|$.

Для изображения выбраны логарифмы, потому что норма стремится к нулю, и точки становятся неотделимы от оси координат и друг от друга.



Зависимость нормы переменной и градиента от номера итерации

Графики демонстрируют сходимость метода.

7 Адаптивность по константе Липшица

7.1 Универсальный градиентный спуск

Вход: $f \in \mathcal{F}(\mathcal{Q})$, $\mathbf{x}_0 \in \mathcal{Q}$, L_0 , ε .

Пока не выполнено условие остановки (5):

k — номер шага.

- 1 $L_{k+1} := \frac{L_k}{2}$;

- 2 $\mathbf{x}_{k+1} = \arg \min_{\mathbf{x} \in \mathcal{Q}} \{f(\mathbf{x}_k) + \langle \nabla f(\mathbf{x}_k), \mathbf{x} - \mathbf{x}_k \rangle + L_{k+1} V(\mathbf{x}, \mathbf{x}_k)\}$;

- 3 Если выполнено условие

$$f(\mathbf{x}_{k+1}) \leq f(\mathbf{x}_k) + \langle \nabla f(\mathbf{x}_k), \mathbf{x}_{k+1} - \mathbf{x}_k \rangle + L_{k+1} V(\mathbf{x}_{k+1}, \mathbf{x}_k) + \frac{\varepsilon}{2},$$

то перейти к следующему шагу;

- 4 Иначе $L_{k+1} := 2L_{k+1}$ и вернуться к пункту 2.

В замечании 2.1 пособия [2] показано, что в качестве $V(x, y)$ подходит функция $V(x, y) = \frac{1}{2} \|x - y\|^2$.

Поскольку рассматривается задача безусловной оптимизации, $\mathcal{Q} = \mathbb{R}^d$. Градиент минимизируемого выражения равен $\nabla f(\mathbf{x}_k) + L_{k+1}(\mathbf{x} - \mathbf{x}_k)$, поэтому формула шага 2 преобразуется к виду $\mathbf{x}_{k+1} = \mathbf{x}_k - \frac{1}{L_{k+1}} \nabla f(\mathbf{x}_k)$, а условие перехода к следующему шагу — к виду $f(\mathbf{x}_{k+1}) \leq f(\mathbf{x}_k) - \frac{1}{2L_{k+1}} \|\nabla f(\mathbf{x}_k)\|^2 + \frac{\varepsilon}{2}$.

7.2 OGM-GL

Метод универсального градиентного спуска решает проблему неизвестности константы Липшица, но обладает недостатками простейшего градиентного метода, так как для функций с большим числом обусловленности матрицы Гессе направление градиента значительно отличается от направления на экстремум. Поэтому применение универсального градиентного метода на практике неэффективно.

Предлагается следующий вариант OGM-G, адаптивный по константе Липшица, основанный на подборе L аналогично тому, как это делается в универсальном градиентном методе, с проверкой условия пункта 3 при каждом вычислении \mathbf{y}_{i+1} .

Если условие нарушено, то вычисление последовательностей \mathbf{x}_i и \mathbf{y}_i начинается сначала с тем же количеством шагов и увеличенным значением L .

OGM-GL

Вход: $f \in \mathcal{F}(\mathbb{R}^d)$, $\mathbf{x}_0 = \mathbf{y}_0 \in \mathbb{R}^d$, L_0 , $N \geq 1$, ε .

Коэффициенты β_i, γ_i вычисляются по формулам:

$$\beta_i = \frac{(\theta_i - 1)(2\theta_{i+1} - 1)}{\theta_i(2\theta_i - 1)}; \gamma_i = \frac{2\theta_{i+1} - 1}{2\theta_i - 1},$$

где последовательность $\{\theta_i\}_{i=0}^N$ строится следующим образом:

$$\theta_i = \begin{cases} \frac{1 + \sqrt{1 + 8\theta_1^2}}{2}, & i = 0; \\ \frac{1 + \sqrt{1 + 4\theta_{i+1}^2}}{2}, & 1 \leq i < N; \\ 1, & i = N. \end{cases}$$

k — номер попытки выполнения цикла

$$L_{k+1} := \frac{L_k}{2}$$

Для $i = 0 \dots N - 1$:

$$\mathbf{y}_{i+1} = \mathbf{x}_i - \frac{1}{L_{k+1}} \nabla f(\mathbf{x}_i);$$

Если $f(\mathbf{y}_{i+1}) > f(\mathbf{x}_i) - \frac{1}{2L_{k+1}} \|\nabla f(\mathbf{x}_i)\|^2 + \frac{\varepsilon}{2}$, то $L_{k+1} := 2L_{k+1}$ и вернуться к началу цикла.

$$\mathbf{x}_{i+1} = \mathbf{y}_{i+1} + \beta_i(\mathbf{y}_{i+1} - \mathbf{y}_i) + \gamma_i(\mathbf{y}_{i+1} - \mathbf{x}_i).$$

Выход: \mathbf{x}_N , L_{end} .

7.3 ALGM

Алгоритм построен на том же принципе, что и ACGM, только вместо OGM-G используется адаптивный по L OGM-GL. Однако, поскольку при работе OGM-GL сохраняется N , определяемое через отношение $\frac{L}{\mu}$, согласованность вычислений при изменении L требует изменения μ , выполняемого в пункте 3.

ALGM

Вход: $f \in \mathcal{F}(\mathbb{R}^d)$, $\mathbf{x}_0 \in \mathbb{R}^d$, $L_0, \mu_0, \beta, \varepsilon$.

Пока не выполнено условие остановки (5):

k — номер шага.

$$1 \quad \mu_k := \beta \mu_{k-1};$$

$$2 \quad \mathbf{x}_k, L_k = OGMGL \left(f, \mathbf{x}_{k-1}, L_{k-1}, \left\lceil \sqrt{\frac{8L_{k-1}}{\mu_k}} \right\rceil, \varepsilon, \right);$$

$$3 \quad \mu_k := \mu_k \cdot \frac{L_k}{L_{k-1}}$$

4 Если выполнено условие $\|\nabla f(x_k)\| \leq \frac{1}{2} \|\nabla f(x_{k-1})\|$, то перейти к следующему шагу;

5 Иначе $\mu_k := \frac{\mu_k}{\beta}$ и вернуться к пункту 2. Если при этом выполнено условие $\|\nabla f(x_k)\| < \|\nabla f(x_{k-1})\|$, то x_{k-1} заменяется на x_k .

Выход: \mathbf{x}_k .

7.4 Оценки

Алгоритм подбора L гарантирует, согласно комментарию к алгоритму универсального градиентного спуска в пособии [2], выполнение условия $L_{k+1} \geq \frac{L^{loc}}{2}$, то есть $L^{loc} \leq 2L_{k+1}$. Поскольку при $L' < L''$ $\mathcal{F}_{L'}(\mathbb{R}^d) \subset \mathcal{F}_{L''}(\mathbb{R}^d)$, для OGM-GL выполнены оценки сходимости, доказанные для OGM-G, откуда следует возможность применения OGM-GL как составной части для алгоритма, подобного ACGM.

Один запуск OGM-GL требует не более $2N$ вычислений функции и N вычислений градиента на каждое увеличение L_k , а суммарное количество вычислений функции и градиента за один запуск составляет $O \left(N \left(\log_2 \frac{L_{end}}{L_{init}/2} + 1 \right) \right)$, то есть $O \left(\sqrt{\frac{L}{\mu}} \log_2 \frac{4L_k^{end}}{L_k^{init}} \right)$, т. к. $L_{loc} \leq L$, $\mu_{loc} \geq L$.

Числовой множитель составляет $\sqrt{8}$ для количества вычислений градиента и $2\sqrt{8}$ для количества вычислений значения функции.

Верхняя оценка количества итераций для каждого уменьшения нормы градиента вдвое определяется аналогично вычислению из раздела 6.2 (расчет для количества обращений к градиенту функции; j — номер перезапуска OGM-GL в пределах одного шага ALGM):

$$\begin{aligned} \sum_{j=0}^J \sqrt{8 \frac{L}{\mu_k^{init}/\beta^j}} \left(2 + \log_2 \frac{L_{kj}}{L_{k,j-1}} \right) &\leq \sqrt{8 \frac{L}{\mu_k}} \sum_{j=0}^J \frac{1}{\sqrt{\beta^j}} \left(2 + \left(\log_2 \frac{L_k}{L_{k-1}} \right)_+ \right) \lesssim \\ &\lesssim \frac{\sqrt{8\beta}}{\sqrt{\beta}-1} \sqrt{\frac{L}{\mu_k}} \left(2 + \left(\log_2 \frac{L_k}{L_{k-1}} \right)_+ \right) \leq \frac{\sqrt{8\beta}}{\sqrt{\beta}-1} \sqrt{\frac{L}{\mu}} \left(2 + \left(\log_2 \frac{L_k}{L_{k-1}} \right)_+ \right) \end{aligned}$$

Положительная срезка логарифма появляется для того, чтобы оценка выполнялась даже в том случае, если $L_k = \frac{L_{k-1}}{2}$. Алгоритм построен так, что за запуск OGM-GL константа Липшица может уменьшиться не более, чем в два раза.

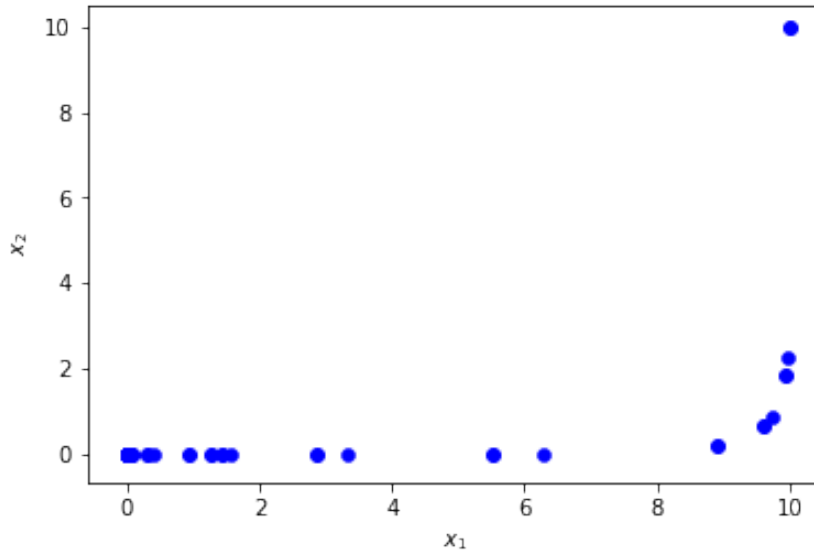
Количество шагов ALGM не превышает $K = \log_2 \frac{\|\nabla f(\mathbf{x}^0)\|}{\varepsilon}$. Поскольку для $t \geq -1$ выполнено свойство $(t)_+ \leq t + 1$, суммарное количество вычислений градиента не превосходит $\frac{\sqrt{8\beta}}{\sqrt{\beta}-1} \sqrt{\frac{L}{\mu}} \left(3K + \log_2 \frac{L}{L_0} \right)$; оценка количества вычислений функции отличается только числовым множителем и превышает полученное значение вдвое.

Как и в разделе 6.2, числовой множитель минимален при $\beta = 4$. Таким образом, получена

Теорема 3 Алгоритм ALGM с оптимальным $\beta = 4$ достигает точки \mathbf{x} , удовлетворяющей критерию останова (5), за не более чем $C \sqrt{\frac{L}{\mu}} \left(3K + \log_2 \frac{L}{L_0} \right)$ вычислений градиента и $2C \sqrt{\frac{L}{\mu}} \left(3K + \log_2 \frac{L}{L_0} \right)$ вычислений функции, где $K = \log_2 \frac{\|\nabla f(\mathbf{x}^0)\|}{\varepsilon}$, $C = \frac{\sqrt{8\beta}}{\sqrt{\beta}-1} = 8\sqrt{2}$.

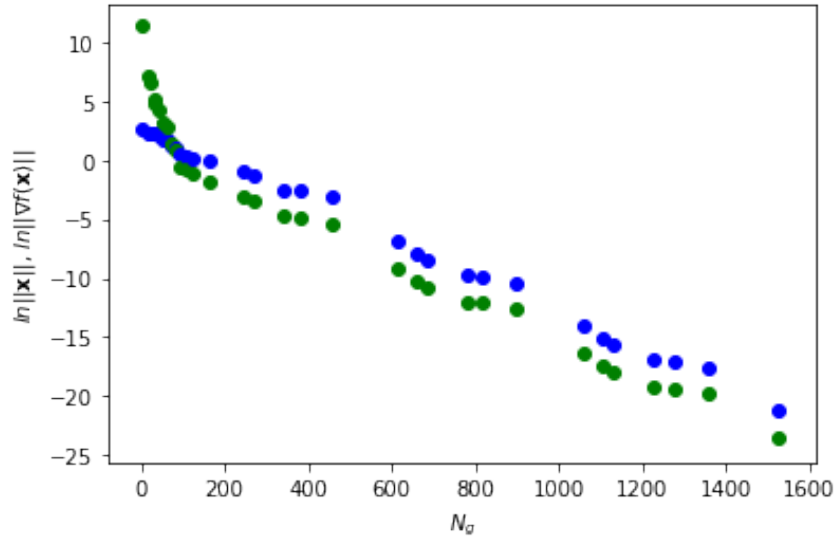
Иллюстрации

Для иллюстрации работы ASCGM используется функция $f(x_1, x_2)$, свойства которой будут описаны в разделе 8 «Эксперименты».



Траектория $\text{ALGM}(f, (10, 10), 1, 1, 4, 10^{-10})$

Синим цветом отмечен натуральный логарифм $\|\mathbf{x}\|$, зеленым — $\|\nabla f(\mathbf{x})\|$.
Для изображения выбраны логарифмы, потому что норма стремится к нулю, и точки становятся неотделимы от оси координат и друг от друга.



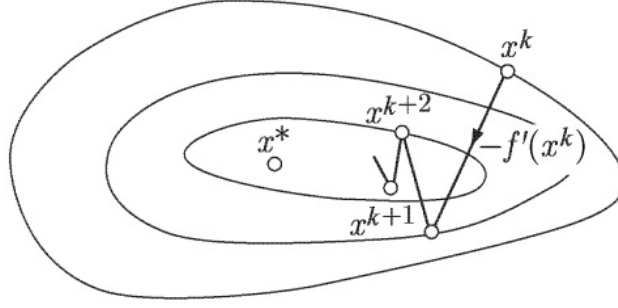
Зависимость нормы переменной и градиента от номера итерации

8 Эксперименты

8.1 Выбор функции

Известно, что недостатком градиентных методов является медленная сходимость на так называемых «овражных» функциях, то есть функциях с большим

значением $\frac{\lambda_{max}}{\lambda_{min}}$ (λ — собственные значения матрицы Гессе).



Траектория градиентного метода при оптимизации «овражной» функции

Использован рисунок из пособия [4].

Поэтому в качестве тестовой функции взята нелинейная функция:

$$f(x_1, x_2) = a\left(\frac{x_1^2}{2} + \frac{x_1^4}{4}\right) + b\left(\frac{x_2^2}{2} + \frac{x_2^4}{4}\right),$$

где $a = 0.1, b = 100$. Матрица Гессе имеет вид

$$\begin{vmatrix} a(1 + 3x_1^2) & 0 \\ 0 & b(1 + 3x_2^2) \end{vmatrix}$$

Соответственно, в каждой области $\mu^{loc} = \min\{a(1 + 3x_{1min}^2), b(1 + 3x_{2min}^2)\}$, $L^{loc} = \max\{a(1 + 3x_{1max}^2), b(1 + 3x_{2max}^2)\}$, где наименьшие и наибольшие значения координат берутся для данной области.

Рассмотрение такой функции в качестве функции с липшицевым градиентом возможно, поскольку при работе рассматриваемых методов соответствующее L^{loc} только уменьшается, а значит, в окрестности траектории метода условие Липшица выполняется.

Минимум данной функции достигается в точке $\mathbf{x}^* = (0, 0)$, в ее окрестности $\frac{L}{\mu} = \frac{b}{a} = 1000$.

8.2 Проверка ASGM

Проверка теоремы 2

Выполнены запуски ASGM для f с начальной точкой $\mathbf{x}_0 = (10, 10)$ ($L = 30100$) и $\beta \in \{2, 3, 4, \dots, 16\}$, $\mu_0 \in \{0.1, 1, 10, 100, 1000, 10000, 100000\}$, $L = 30100, 100000\}$,

$\varepsilon \in \{\frac{\nabla f(\mathbf{x}_0)}{10}, \frac{\nabla f(\mathbf{x}_0)}{100}, \dots, \frac{\nabla f(\mathbf{x}_0)}{10^{11}}\}$, зафиксировано количество итераций. Для каждого набора параметров рассчитана верхняя оценка количества итераций из теоремы 2. Проверено, что во всех рассмотренных случаях эта оценка не превышает.

В таблице 1 представлены измеренные количества итераций при $\beta = 2$ и est — оценки по теореме 2.

Таблица 1: $f, \beta = 2$

$\varepsilon \backslash \mu_0$	0.1	1	10	10^2	10^4	L	10^5	est	$\frac{max}{est}$
$\frac{\nabla f(\mathbf{x}_0)}{10}$	1098	347	110	35	4	2	2	24889	0.044
$\frac{\nabla f(\mathbf{x}_0)}{10^2}$	1098	347	110	35	4	13	14	49779	0.022
$\frac{\nabla f(\mathbf{x}_0)}{10^3}$	1098	347	110	60	47	78	66	74669	0.015
$\frac{\nabla f(\mathbf{x}_0)}{10^4}$	1098	347	188	357	382	375	348	99559	0.011
$\frac{\nabla f(\mathbf{x}_0)}{10^5}$	1098	593	674	1001	954	1086	1000	124449	0.009
$\frac{\nabla f(\mathbf{x}_0)}{10^7}$	4297	4365	5548	5224	5891	5822	5271	174229	0.034
$\frac{\nabla f(\mathbf{x}_0)}{10^8}$	6171	7717	8546	8232	8927	8296	8314	199119	0.045
$\frac{\nabla f(\mathbf{x}_0)}{10^9}$	8045	10087	11544	10129	11963	11069	10233	224009	0.053
$\frac{\nabla f(\mathbf{x}_0)}{10^{11}}$	8045	11069	11544	10129	11963	11069	10233	273789	0.044

В таблице 2 представлены измеренные количества итераций при $\beta = 4$ и est — оценки по теореме 2.

Таблица 2: $f, \beta = 4$

$\varepsilon \backslash \mu_0$	0.1	1	10	10^2	10^4	L	10^5	est	$\frac{max}{est}$
$\frac{\nabla f(\mathbf{x}_0)}{10}$	776	246	78	25	3	2	2	20619	0.038
$\frac{\nabla f(\mathbf{x}_0)}{10^2}$	776	246	78	25	3	14	15	41239	0.019
$\frac{\nabla f(\mathbf{x}_0)}{10^3}$	776	246	78	63	43	76	80	61858	0.013
$\frac{\nabla f(\mathbf{x}_0)}{10^4}$	776	246	410	361	469	350	380	82478	0.009
$\frac{\nabla f(\mathbf{x}_0)}{10^5}$	776	246	955	854	1021	1168	878	103097	0.011
$\frac{\nabla f(\mathbf{x}_0)}{10^7}$	1164	4298	4217	5373	4324	4975	5451	144336	0.038
$\frac{\nabla f(\mathbf{x}_0)}{10^8}$	4074	7244	6080	5373	6210	4975	5451	164956	0.044
$\frac{\nabla f(\mathbf{x}_0)}{10^9}$	4074	10190	7943	7730	8096	7149	7836	185575	0.055
$\frac{\nabla f(\mathbf{x}_0)}{10^{11}}$	4850	10190	7943	7730	8096	7149	7836	226814	0.045

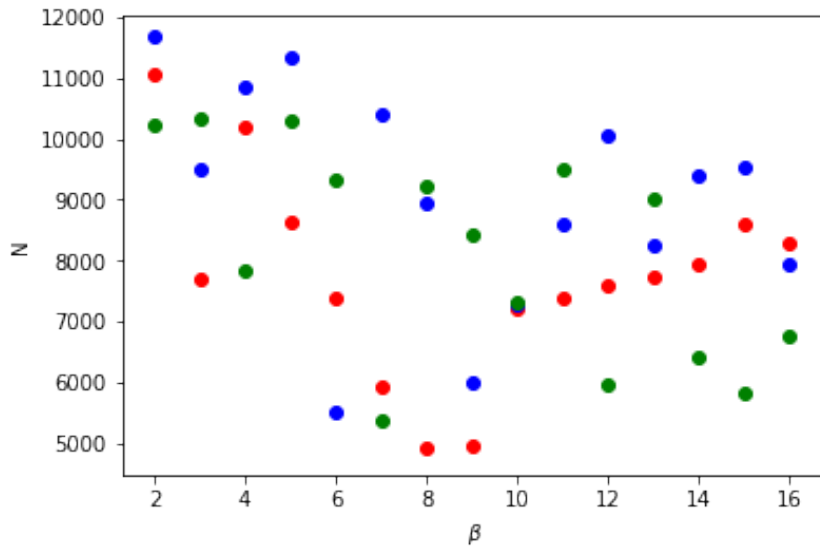
В таблице 3 представлены измеренные количества итераций при $\beta = 8$ и est — оценки по теореме 2.

Таблица 3: $f, \beta = 8$

$\varepsilon \backslash \mu_0$	0.1	1	10	10^2	10^4	L	10^5	est	$\frac{max}{est}$
$\frac{\nabla f(\mathbf{x}_0)}{10}$	549	174	55	18	2	2	2	22554	0.024
$\frac{\nabla f(\mathbf{x}_0)}{10^2}$	549	174	55	18	10	18	17	45108	0.012
$\frac{\nabla f(\mathbf{x}_0)}{10^3}$	549	174	75	93	90	52	84	67663	0.008
$\frac{\nabla f(\mathbf{x}_0)}{10^4}$	549	236	361	300	202	416	320	90217	0.006
$\frac{\nabla f(\mathbf{x}_0)}{10^5}$	549	258	1011	882	669	1174	601	112771	0.010
$\frac{\nabla f(\mathbf{x}_0)}{10^7}$	3587	3038	2848	2525	3504	3317	3929	157880	0.025
$\frac{\nabla f(\mathbf{x}_0)}{10^8}$	5688	3038	4529	5533	6801	5278	6177	180435	0.038
$\frac{\nabla f(\mathbf{x}_0)}{10^9}$	5688	4917	6210	8541	10413	5278	9220	202989	0.051
$\frac{\nabla f(\mathbf{x}_0)}{10^{11}}$	6237	4917	6210	8541	10413	5278	9220	248098	0.042

Экспериментальные данные показывают, что теорема 2 выполняется, причем условие остановки выполняется после значительно меньшего числа итераций, чем следует из теоретических оценок. Это объясняется тем, что все оценки в доказательстве теоремы сделаны для наихудших случаев и не достигаются.

Выбор оптимального β



Зависимость числа итераций от параметра β

На графике показана зависимость числа итераций при работе ASCM от

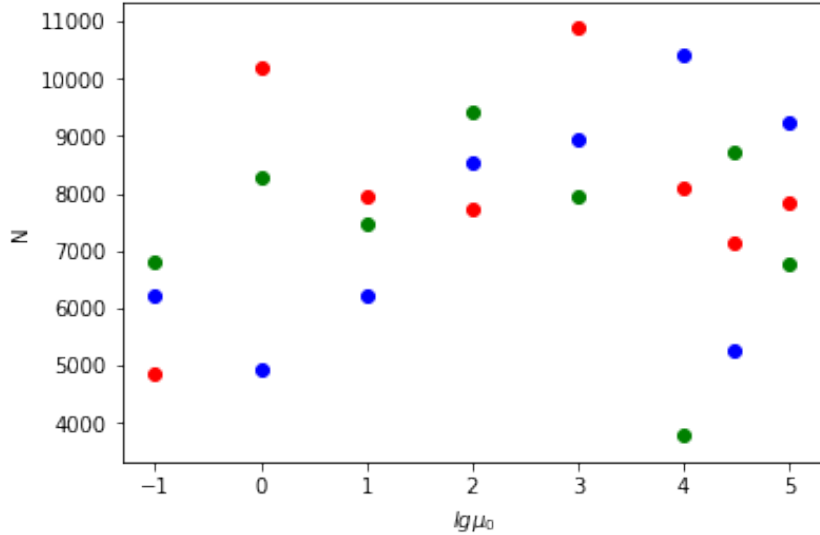
β при $\varepsilon = \frac{\|\nabla f(\mathbf{x}_0)\|}{10^{10}}$ для $\mu_0 = 1$ (красные точки), $\mu_0 = 1000$ (синие точки), $\mu_0 = 100000$ (зеленые точки).

Из графика видно, что четко выраженный минимум не наблюдается ни в одном из случаев; полученное в разделе 6.2 значение β ни в одном из случаев не является оптимальным.

Это связано с тем, что минимизация зависящего от β коэффициента выполнялась в предположении о достижении всех верхних, наихудших оценок. Поскольку данное предположение не выполняется, то и количество итераций оказывается минимальным при других значениях β .

Однако именно $\beta = 4$ дает наименьшее гарантированное значение коэффициента в оценке скорости сходимости.

Выбор оптимального μ_0



Зависимость числа итераций от параметра μ_0

На графике показана зависимость числа итераций при работе ACGM от μ_0 при $\varepsilon = \frac{\|\nabla f(\mathbf{x}_0)\|}{10^{10}}$ для $\beta = 4$ (красные точки), $\beta = 8$ (синие точки), $\beta = 16$ (зеленые точки).

Закономерность в расположении точек не обнаруживается, для каждого из рассмотренных значений β выделяется два минимума: в окрестности L и в окрестности истинного значения μ , что частично подтверждает полученные выводы об оптимальном значении μ_0 .

Сравнение ACGM с OGM-G

OGM-G принимает количество итераций на вход, а ACGM работает до достижения условия останова. Для сравнения эффективности используется модификация OGM-G, которая повторяет выполнение алгоритма с теми же заданными L и μ до выполнения условия останова.

OGM-Gtest

Вход: $f \in \mathcal{F}_L(\mathbb{R}^d)$, $\mathbf{x}_0 \in \mathbb{R}^d$, L , μ_0 .

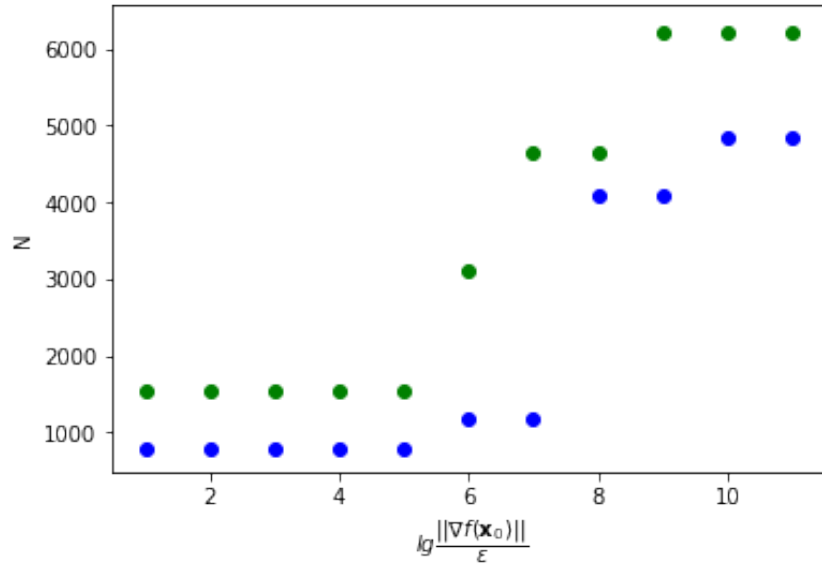
Пока не выполнено условие останова (5): $\mathbf{x}_k = OGMG(f, \mathbf{x}_{k-1}, L, \mu_0)$.

Результаты ACGM ($\beta = 4$) находятся в таблице 3; таблица 4 содержит результаты OGM-Gtest.

Таблица 4: f , OGM-G

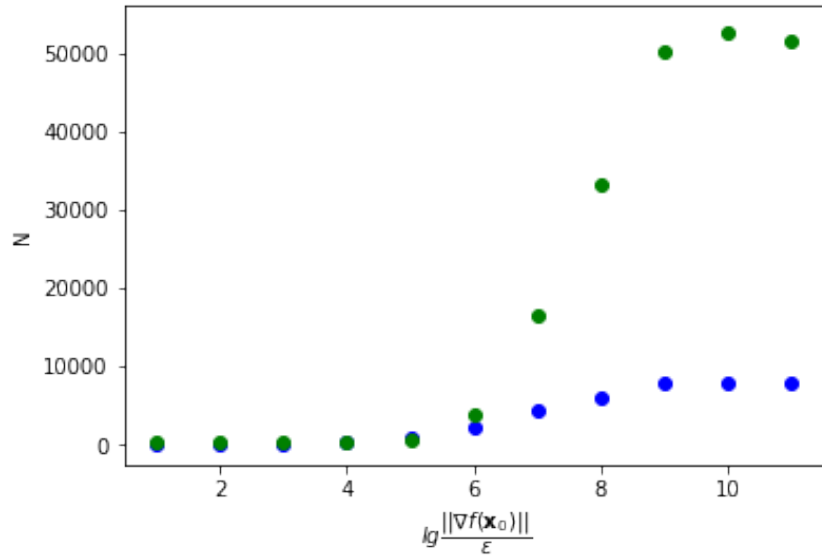
$\varepsilon \setminus \mu_0$	0.1	1	10	10^2	10^4	L	10^5
$\frac{\nabla f(\mathbf{x}_0)}{10}$	1552	491	156	50	5	3	2
$\frac{\nabla f(\mathbf{x}_0)}{10^2}$	1552	491	156	50	5	3	14
$\frac{\nabla f(\mathbf{x}_0)}{10^3}$	1552	491	156	50	55	108	134
$\frac{\nabla f(\mathbf{x}_0)}{10^4}$	1552	491	312	400	2120	2688	3106
$\frac{\nabla f(\mathbf{x}_0)}{10^5}$	1552	491	468	2250	12300	15582	18000
$\frac{\nabla f(\mathbf{x}_0)}{10^6}$	3104	982	3744	11900	65575	83082	95976
$\frac{\nabla f(\mathbf{x}_0)}{10^7}$	4656	982	16380	48700	267600	339040	391660
$\frac{\nabla f(\mathbf{x}_0)}{10^8}$	4656	3437	33228	97150	533500	675910	780810
$\frac{\nabla f(\mathbf{x}_0)}{10^9}$	6208	8347	50076	145850	801080	1014900	1172400
$\frac{\nabla f(\mathbf{x}_0)}{10^{10}}$	6208	9329	52572	153150	841020	1065500	1230900
$\frac{\nabla f(\mathbf{x}_0)}{10^{11}}$	6208	8838	51480	150000	823790	1043700	1205700

Графики показывают количество итераций до останова в зависимости от требуемой точности при использовании ACGM (синие точки) и OGM-Gtest (зеленые точки).



Зависимость числа итераций от ε , $\mu_0 = 0.1$

В данном случае, когда начальное значение μ_0 совпадает с истинным значением константы сильной выпуклости, ACGM требует несколько меньше итераций, чем OGM-G.



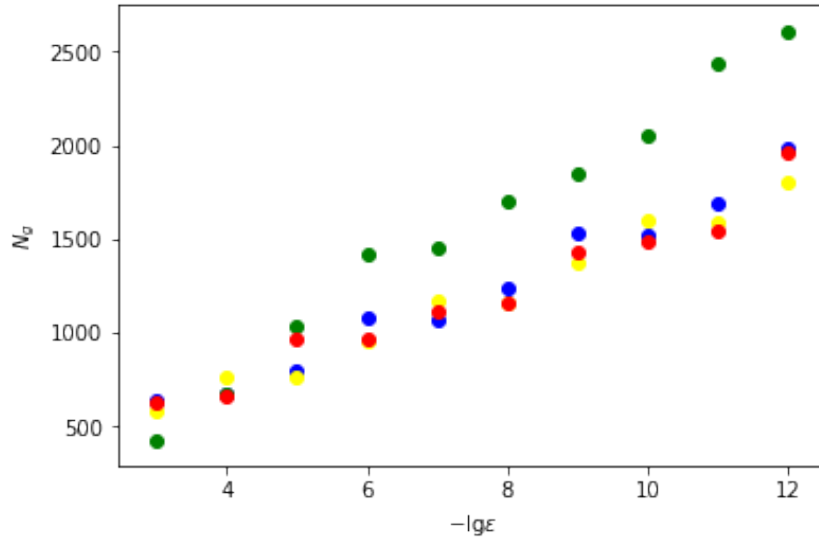
Зависимость числа итераций от ε , $\mu_0 = 10$

Когда начальное значение μ_0 превышает на порядок истинное значение константы сильной выпуклости, ACGM требует значительно меньше итераций, чем OGM-G.

Из таблицы 4 очевидно, что если $\mu_0 \gg \mu$, то OGM-G требует на несколько порядков больше итераций, чем ACGM.

8.3 Проверка ALGM

Проверка сходимости при разных L_0



Зависимость числа итераций от ε , L_0

Синие точки соответствуют $L_0 = 1$, зеленые — $L_0 = 10$, желтые — $L_0 = 100$, красные — $L_0 = 1000$.

График подтверждает линейный характер зависимости числа итераций от логарифма требуемой погрешности.

Проверка теоремы 3

Поскольку $L = 30100$, $\mu = 0.1$, $\nabla f(\mathbf{x}_0) = (101, 101000)$, достигаемая при $\varepsilon = 10^{-3}$ и $L_0 = 1000$ минимальная оценка для значений параметров из данной серии экспериментов по теореме 3 составляет 85800 итераций, что превосходит все полученные значения.

Таким образом, теорема 3 дает значительно завышенные оценки количества итераций. Этот результат является закономерным, поскольку в теореме фигурирует значение L , пригодное на всей траектории метода, а локальное значение L для рассматриваемой модельной функции убывает при приближении к минимуму.

Также проведена проверка при разных истинных значениях L , полученных путем изменения коэффициента b . Во всех случаях количество итераций оказалось меньше предсказанного теоремой 3 более чем в 300 раз.

9 Заключение

Работа посвящена построению адаптивных по константе сильной выпуклости и константе Липшица для градиента методов оптимизации первого порядка.

Построен алгоритм выпуклой оптимизации первого порядка ACGM, адаптивный по константе сильной выпуклости. Доказана теоретически и проверена экспериментально его эффективность по сравнению с базовым алгоритмом OGM-G (статья [1]), не обладающим свойством адаптивности. Доказаны теоремы 1 и 2, гарантирующие, что сложность построенного алгоритма составляет не более $O\left(\sqrt{\frac{L}{\mu}} \log_2 \frac{\|\nabla f(\mathbf{x}^0)\|}{\varepsilon}\right)$ вычислений градиента функции.

Построен алгоритм ALGM, адаптивный по константе Липшица. Доказана теорема 3 о том, что он выполняет $O\left(\sqrt{\frac{L}{\mu}} \left(\log_2 \frac{\|\nabla f(\mathbf{x}^0)\|}{\varepsilon} + \log_2 \frac{L}{L_0}\right)\right)$ вычислений градиента и функции. Проведена экспериментальная проверка полученных результатов.

10 Ссылки

- [1] | Optimizing the Efficiency of First-order Methods for Decreasing the Gradient of Smooth Convex Functions, Donghwan Kim, Jeffrey A. Fessler, 2018, 14 с., arXiv:1803.06600v2;
- [2] | Современные численные методы оптимизации. Метод универсального градиентного спуска. Учебное пособие, Гасников А. В., 2018, 238 с., ISBN 978-5-7417-0667-1;
- [3] | Об эффективных численных методах решения задач энтропийно-линейного программирования, Гасников А. В., Гасникова Е. В., Нестеров Ю. Е., Чернов А. В., Журнал вычислительной математики и математической физики. 2016. Т. 56. № 4;
- [4] | Курс методов оптимизации: Учебное пособие, Сухарев А. Г., Тимохов А. В., Федоров В. В., ФИЗМАТЛИТ, 2005, 368 с., ISBN 5-9221-0559-0;
- [5] | On the Adaptivity of Stochastic Gradient-Based Optimization, Lihua Lei, Michael I. Jordan, 2019, 46 с., arXiv:1904.04480v2;
- [6] | Restarting accelerated gradient methods with a rough strong convexity estimate, Olivier Fercoq, Zheng Qu, 2016, 23 с., arXiv:1609.07358