

**НАЦИОНАЛЬНАЯ АКАДЕМИЯ НАУК УКРАИНЫ
ИНСТИТУТ КИБЕРНЕТИКИ ИМ. В. М. ГЛУШКОВА**

**МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РЕСПУБЛИКИ МОЛДОВА
АКАДЕМИЯ ТРАНСПОРТА, ИНФОРМАТИКИ И КОММУНИКАЦИЙ**

П. И. СТЕЦЮК

**МЕТОДЫ ЭЛЛИПСОИДОВ
И r -АЛГОРИТМЫ**

КИШИНЭУ * ЭВРИКА* 2014

Серия научных публикаций
«Недифференцируемая оптимизация и ее приложения»,
посвященная академику Н. З. ШОРУ

Редакционная коллегия серии:
акад. НАН Украины И. В. СЕРГИЕНКО,
д-р техн. наук Д. И. СОЛОМОН,
д-р физ-мат. наук П. И. СТЕЦЮК

CZU 519.7/.8(075.8)
С 79

Descrierea CIP a Camerei Naționale a Cărții

Стецюк, П. И.

Методы эллипсоидов и r -алгоритмы / П. И. Стецюк; Нац. акад. наук Украины, Ин-т кибернетики им. В. М. Глушкова, Акад. транспорта, информатики и коммуникаций. – Кишинэу: Эврика, 2014 (Tipografia AŞM). – 488 p.

300 ex.

ISBN 978-9975-4448-5-9.

519.7/.8(075.8)

С 79

Книга содержит результаты по развитию идей метода эллипсоидов, построению ускоренных модификаций методов эллипсоидов и исследованию их связи с r -алгоритмами и другими методами минимизации негладких функций. Значительное внимание уделяется специальным видам одноранговых линейных операторов и обоснованию на их основе субградиентных методов с преобразованием пространства.

Книга рассчитана на специалистов в области математического программирования и его приложений, а также студентов и аспирантов соответствующих специальностей.

Рекомендовано к печати Ученым советом Института кибернетики им. В. М. Глушкова НАН Украины (протокол № 3 от 25.02.14) и Сенатом Академии транспорта, информатики и коммуникаций Республики Молдова (протокол № 4 от 06.03.14)

Рецензенты: **А. М. Гупал**, член-корреспондент НАН Украины,
П. С. Кнопов, член-корреспондент НАН Украины

М–208–96
ISBN 978-9975-4448-5-9.

© Изд-во Эврика, 2014
© Стецюк П.И., 2014



НАУМ ЗУСЕЛЕВИЧ ШОР (1937 – 2006) – один из основоположников направления недифференцируемой оптимизации в математическом программировании. Автор 10 монографий и более 200 статей.

Родился 1 января 1937 г. в г. Киеве. После окончания в 1958 г. Киевского государственного университета им. Т. Г. Шевченко работал в Институте кибернетики им. В. М. Глушкова НАН Украины (с 1983 по 1990 г. – заведующим лабораторией, с 1990 по 2006 г. – заведующим отделом). Профессор (1989), член-корреспондент (1990), академик (1997) Национальной академии наук Украины. Лауреат Государственных премий СССР (1973), СССР (1981), Украины (1993, 1999), премий НАН Украины имени В. М. Глушкова (1987), имени В. С. Михалевича (1997).

Работы Н. З. Шора положили начало численным методам оптимизации недифференцируемых функций и существенно повлияли на развитие линейного, нелинейного, дискретного и стохастического программирования. Разработанные им методы получили мировую известность и способствовали становлению Института кибернетики им. В. М. Глушкова НАН Украины как одного из мировых лидеров в области теории и практики методов оптимизации.

Серия научных публикаций
«Недифференцируемая оптимизация и ее приложения»,
посвященная Науму Зуселевичу Шору

Идеи и методы Н.З. Шора оказали большое влияние на развитие многих направлений математического программирования. Пионер субградиентного метода, он умело сочетал теорию и практику – придал субградиентным методам более высокую скорость сходимости и сделал их эффективным средством минимизации негладких функций, его техника лагранжевой релаксации для квадратичных невыпуклых задач оказалась очень эффективной при решении комбинаторных проблем. Книги Н.З. Шора стали настольными для ведущих отечественных и зарубежных специалистов в области математического программирования. Яркой характеристикой этому может служить письмо профессора С. Бойда (Stephen Boyd) из Стэнфордского университета.

Уважаемый профессор Шор!

Мы никогда не встречались, но ваши работы оказали на меня огромное влияние. Я начал с вашей небольшой книги по субградиентным методам (1985), я тогда был еще аспирантом. А сейчас я читаю вашу новую книгу о недифференцируемой оптимизации (1998) и просто наслаждаюсь ею.

Я высылаю Вам три написанных мною книги. Первая – о проектировании линейных контроллеров с помощью выпуклой оптимизации, вторая – о матричных неравенствах, а третья – учебник по выпуклой оптимизации. [...]

Надеюсь, Вы увидите ваше сильное влияние во всех этих книгах.

С уважением,

Стефан П. Бойд, 15.04.2005

В письме упоминаются две¹ англоязычные монографии Н.З. Шора. В первой книге значительное внимание уделяется использованию субградиентных методов для решения задач выпуклого программирования большой размерности, во второй – для решения многоэкстремальных квадратичных задач (с приложениями в задачах полиномиальной оптимизации и комбинаторных булевых задачах).

¹ SHOR N.Z. (1985) *Minimization Methods for Non-Differentiable Functions*. – Berlin; New York: Springer-Verlag.

SHOR N.Z. (1998) *Nondifferentiable optimization and polynomial problems*. – Boston; Dordrecht; London: Kluwer Academic Publishers.

На развитие идей Н.З. Шора были направлены два совместных проекта Института кибернетики им. В.М. Глушкова НАН Украины и Академии транспорта, информатики и коммуникаций Республики Молдова. Первый проект закончился и был связан с опубликованием избранных трудов Н.З. Шора. Его результатом были три книги:

Шор Н.З. Методы недифференцируемой оптимизации и сложные экстремальные задачи: Сб. избр. тр. – Кишинэу: Эврика, 2008. – 272 с.

Шор Н.З. Методы минимизации негладких функций и матричные задачи оптимизации: Сб. избр. тр. – Кишинэу: Эврика, 2009. – 240 с.

Шор Н.З. Алгоритмы последовательной и негладкой оптимизации: Сб. избр. тр. – Кишинэу: Эврика, 2012. – 270 с.

Второй проект связан с изданием серии книг под общим названием «Недифференцируемая оптимизация и ее приложения», где каждая отдельная книга посвящена современным теоретическим и практическим достижениям тех или иных направлений, связанных с научным наследием академика Н.З. Шора. Первая книга этой серии – монография Соломона Д.И. „*Дробное программирование и недифференцируемая оптимизация*“ была опубликована в 2010 году. Она посвящена методам решения задач дробного программирования с помощью субградиентных алгоритмов в сочетании со схемами декомпозиции.

Книга Стецюка П.И. „Методы эллипсоидов и r -алгоритмы“ продолжает эту серию. В ней представлены исследования автора по развитию методов эллипсоидов, которые являются частным случаем предложенных Н.З. Шором алгоритмов с преобразованием пространства. Эти методы базируются на внешней аппроксимации множества экстремумов эллипсоидами с монотонным уменьшением их объема и имеют „красивую“ геометрическую интерпретацию. В книге приведены ускоренные модификации методов эллипсоидов и показана их тесная связь с r -алгоритмами и другими методами минимизации негладких функций.

Следующей в серии «Недифференцируемая оптимизация и ее приложения» планируется публикация: ЖУРБЕНКО Н.Г. Субградиентные методы минимизации негладких функций. В настоящее время готовятся книги по использованию методов негладкой оптимизации в стохастическом программировании (Годонога А.Ф.) и проблемах раскраски плоских графов (Донец Г.А.).

Надеемся, что данная серия научных публикаций окажется полезной для специалистов в области математического программирования и они найдут в ней ряд оригинальных идей для построения эффективных алгоритмов решения экстремальных задач.

Редакционная коллегия

Предисловие

„Теория всего класса алгоритмов с растяжением пространства далека от совершенства. Нам кажется достаточно реалистичной целью – построение такого алгоритма, который по своей практической эффективности не уступал бы r -алгоритму и был столь же хорошо обоснован, как метод эллипсоидов.“ Несмотря на то, что этой цитате² уже больше тридцати лет, она остается актуальной и сегодня.

Это объясняется тем, что идея внешней аппроксимации множества экстремумов эллипсоидами с монотонным уменьшением их объема до сих пор не имплементирована в эффективные по скорости алгоритмы с растяжением пространства. Цель этой книги – показать, что идея описанных эллипсоидов далеко не исчерпала своих возможностей и к ней следует возобновить интерес, который в значительной мере иссяк после того, как полиномиальный алгоритм Хачияна для задачи линейного программирования на основе метода эллипсоидов не удалось подкрепить конкретными вычислительными результатами.

Настоящая книга содержит результаты по развитию идей метода эллипсоидов, построению ускоренных модификаций методов эллипсоидов и исследованию их связи с r -алгоритмами и другими методами минимизации негладких функций. Большинство результатов получены автором самостоятельно, некоторые в соавторстве, а ряд результатов принадлежат Н.З. Шору и его ученику В.И. Гершовичу. Последние характеризуют основные достижения киевской школы оптимизации на момент самых активных исследований по методам эллипсоидов, которые пришлось на конец 70-х – начало 80-х годов прошлого века.

Материал размещен по тематической направленности в семи разделах книги. В первом разделе, который называется "Методы эллипсоидов и их обобщения" помещены статьи

- 1.1 Шор Н.З. Метод отсечения с растяжением пространства для решения задач выпуклого программирования // Кибернетика, 1977, №1.
- 1.2 Шор Н.З., Гершович В.И. Об одном семействе алгоритмов для решения задач выпуклого программирования // Кибернетика, 1979, №4.
- 1.3 Гершович В.И., Шор Н.З., Метод эллипсоидов, его обобщения и приложения // Кибернетика, 1982, №5.
- 1.4 Шор Н.З. Использование операций растяжения пространства в задачах минимизации выпуклых функций // Кибернетика, 1970, №1.

²Ее можно найти на странице 48 книги, в заключении к статье В.И. Гершовича и Н.З. Шора 1982 года.

В первой статье метод эллипсоидов изложен как алгоритм с растяжением пространства для решения задачи выпуклого программирования. В статьях 1.2 и 1.3 рассмотрены алгоритмы решения задач выпуклого программирования, основанные на использовании нескольких отсечений гиперплоскостями и аппроксимации области локализации экстремума эллипсоидами. В статье 1.3 даны приложения этих методов к проблемам комбинаторной оптимизации и математического программирования.

Последняя статья раздела непосредственно с методом эллипсоидов не связана, однако здесь для обоснования сходимости одного из алгоритмов с растяжением пространства используется техника, очень близкая к той которую позже Н.З. Шор использовал при обосновании сходимости метода эллипсоидов. На эту работу 1970 года иногда ошибочно ссылаются как на метод эллипсоидов. В этом есть определенная доля истины, которая состоит в том, что Н.З. Шор в 1970 году был всего в одном шаге от открытия метода эллипсоидов. В этом легко убедиться, если сравнить доказательство теоремы о сходимости метода эллипсоидов из статьи 1.1 и доказательство теоремы 2 из статьи 1.4.

Второй раздел называется „Приближенный метод эллипсоидов“ и в него включены следующие статьи³

- 2.1 СТЕЦЮК (2003) Приближенный метод эллипсоидов [63].
- 2.2 СТЕЦЮК (2003) Об одном эллипсоиде для внешней аппроксимации n -мерного полушара [64].
- 2.3 СТЕЦЮК (2003) Модификация метода эллипсоидов [62].
- 2.4 STETSYUK, NURMINSKI (2010) On a modification of ellipsoid method [107].
- 2.5 СТЕЦЮК, Колесник, Березовский (2003) Об одном методе нахождения L_p -решения системы линейных уравнений [66].
- 2.6 СТЕЦЮК, Буханцов (2002) К ускорению метода эллипсоидов с помощью использования шарового слоя [58].

Статьи 2.1 – 2.4 посвящены различным аспектам модификации метода эллипсоидов с такой же асимптотической скоростью сходимости по объему, как в известном методе Юдина-Немировского-Шора. Для одномерных задач на каждой итерации гарантируется коэффициент уменьшения объема, который равен $2 - \sqrt{2} \approx 0.5858$. В статье 2.5 эта модификация используется для нахождения L_p -решений переопределенной системы линейных алгебраических уравнений при двусторонних ограничениях на компоненты решения. Показано, что она находит L_p -решение

³Здесь и далее приводятся сокращенные ссылки на включенные в книгу статьи, опуская даже инициалы авторов. Подробную ссылку можно найти по ее номеру в размещенном в конце книги списке работ автора.

системы линейных алгебраических уравнений за конечное число итераций, зависящее от числа неизвестных компонент.

Еще одна модификация метода эллипсоидов описана в статье 2.6. В этой модификации ускорение достигается за счет использования эллипсоида минимального объема, содержащего шаровой слой. Идея модификации состоит в том, что на текущей итерации используется отсекающая гиперплоскость из предыдущей итерации и, если она позволяет уменьшить объем аппроксимирующего эллипсоида, то такое уменьшение реализуется, если же нет, то реализуется шаг метода эллипсоидов. При минимизации существенно овражных функций скорость сходимости этой модификации оказывается в несколько раз лучше, чем скорость сходимости метода эллипсоидов Юдина-Немировского-Шора.

Алгоритмы второго раздела используют оператор растяжения пространства. С его помощью на каждой итерации специальный эллипсоид преобразуется в шар за одно растяжение пространства. В последующих разделах центральную роль играет специальный эллипсоид, преобразование которого в шар требует двух последовательных растяжений пространства в ортогональных направлениях. Алгоритмы, построенные на его основе, идейно ближе к r -алгоритмам Шора-Журбенко, чем алгоритмы второго раздела. Дело в том, что для преобразования специального эллипсоида в шар используется антиовражный прием, близкий к тому, который имеет место в r -алгоритмах. Однако, здесь растяжение пространства реализуется в направлении разности двух нормированных субградиентов, и близким к направлению разности двух субградиентов оно будет только тогда, когда нормы субградиентов близки.

Под названием „ r -алгоритмы и эллипсоиды“ в третьем разделе собраны следующие работы

- 3.1 Стецюк (1995) К вопросу сходимости r -алгоритмов [19].
- 3.2 Стецюк (1995) К обоснованию сходимости алгоритмов с растяжением пространства [20].
- 3.3 Стецюк (1996) r -алгоритмы и эллипсоиды [22].
- 3.4 Стецюк (1996) Метод центров тяжести простых тел [27].
- 3.5 Стецюк (1999) К методам эллипсоидов [37].
- 3.6 Стецюк (2001) Об ускорении сходимости методов эллипсоидов [47].

В статье 3.1 построен пример выпуклой кусочно-линейной функции, для которой возможно «зацикливание» $r_\mu(\alpha)$ -алгоритма. В статье 3.2 анализируются основные причины, которые препятствуют обоснованию сходимости r -алгоритмов, и обсуждается подход, основанный на

аппроксимации множества экстремумов эллипсоидами. В статье 3.3 построен эллипсоид минимального объема для пересечения шара и двух полупространств, граничные гиперплоскости к которым проходят через центр шара. На его основе для задачи минимизации выпуклой функции построены два алгоритма фейеровского типа с преобразованием пространства, идейно близкие к r -алгоритмам. Численные эксперименты подтверждают эффективность этих алгоритмов.

В статье 3.4 для задачи выпуклого программирования рассмотрена общая конструкция метода центров тяжести с использованием линейных преобразований пространства. Она основывается на внешней аппроксимации множества экстремумов простыми геометрическими телами с гарантированным уменьшением их объема на каждом шаге метода и позволяет создавать экономные по сложности обработки информации алгоритмы. В статье 3.5 дана геометрическая интерпретация методов описанных эллипсоидов, содержащих пересечение шара и гиперплоскости. Они позволяют на основе оператора растяжения пространства создавать монотонные по объему алгоритмы для нахождения стационарных точек специальных задач. В статье 3.6 кратко описан оптимальный эллипсоид из статьи 3.3 и построенные на его основе ускоренные алгоритмы фейеровского типа.

Работы четвертого раздела объединены под названием „Ортогонализирующие линейные операторы“. В раздел включены такие работы

- 4.1 СТЕЦЮК (1997) Ортогонализирующие линейные операторы в выпуклом программировании (Часть I) [28].
- 4.2 СТЕЦЮК (1997) Ортогонализирующие линейные операторы в выпуклом программировании (Часть II) [29].
- 4.3 СТЕЦЮК (1996) Классические фейеровские методы с преобразованием пространства [23].
- 4.4 СТЕЦЮК (1997) Об одном способе ускорения метода ортогонального субградиентного спуска [30].
- 4.5 СТЕЦЮК (2000) Об одном методе для нахождения допустимой точки выпуклого неравенства [40].
- 4.6 СТЕЦЮК (2001) К методам решения плохообусловленных систем линейных уравнений [46].

Центральными здесь являются статьи 4.1 и 4.2, где рассмотрены два одноранговых линейных оператора преобразования пространства. На их основе построены субградиентные методы фееровского типа для нахождения минимума выпуклой функции при известном оптимальном значении функции. Приведены результаты численных экспериментов. В статье 4.3 кратко описаны основные из этих методов.

В статье 4.4 получены условия, которые при минимизации овражных кусочно-линейных выпуклых функций позволяют ускорить метод ортогонального субградиентного спуска. Статьи 4.5 и 4.6 связаны с приложениями самого эффективного алгоритма из статьи 4.1. С его помощью построен метод фейеровского типа с преобразованием пространства для нахождения допустимой точки выпуклого неравенства и итерационные методы для решения системы линейных алгебраических уравнений. Эффективность методов подтверждена численными экспериментами при решении плохо обусловленных систем линейных уравнений.

Пятый раздел называется „Одноранговый эллипсоидальный оператор в фейеровских процессах“ и в него включены работы

- 5.1 Стецюк (2009) Субградиентные методы переменной метрики, использующие шаг Агмона-Моцкина и одноранговый эллипсоидальный оператор [102].
- 5.2 Стецюк (2011) Метод `amsg2p` для овражных выпуклых функций [120].
- 5.3 Стецюк (2011) Релаксационный субградиентный метод минимизации овражных выпуклых функций [121].
- 5.4 Стецюк (2012) Ускоренные по Шору модификации субградиентного метода Поляка [137].
- 5.5 Стецюк (2012) Ускорение субградиентного метода Поляка [139].
- 5.6 Стецюк (2012) Shor's acceleration for Polyak's subgradient method [142].
- 5.7 Стецюк (2012) Ускоренные модификации субградиентного метода Поляка для овражных выпуклых функций [147].
- 5.8 Стецюк, Кошлай (2012) Ускоренный субградиентный метод Поляка [149].
- 5.9 Стецюк (2013) Ускоренные по Шору модификации метода Поляка для овражных функций [161].

Статьи пятого раздела связаны с различными аспектами двух субградиентных методов с преобразованием пространства для нахождения точки минимума выпуклой функции при известном минимальном значении. В методах используется шаг Поляка (известен также как шаг Агмона-Моцкина-Шенберга) в направлении нормированного антисубградиента и гарантируется монотонное уменьшение расстояния до точки минимума в последовательно преобразованных пространствах переменных. Преобразование пространства реализуется с помощью однорангового оператора и направлено на уменьшение степени овражности поверхностей уровня выпуклых функций, подобно тому, как это сделано в r -алгоритмах Шора.

Отсюда и частое употребление термина „ускоренный по Шору“ для этих двух методов. Первый метод преобразует пространство, используя два последовательных субградиента, если угол между ними тупой. Второй метод использует еще и агрегатный вектор, который является выпуклой комбинацией вычисленных ранее субградиентов. В том случае, когда с помощью агрегатного вектора нельзя построить тупой угол, этот вектор обновляется. Даны сравнительные результаты вычислительных экспериментов для второго метода и r -алгоритма.

Шестой раздел называется „ r -Алгоритмы и квазиньютоновские методы“ и связан не столько с развитием методов описанных эллипсоидов, сколько с исследованием новых видов линейных операторов для квазиньютоновских методов. В него включены такие работы

- 6.1 СТЕЦЮК (1996) Квазиньютоновские методы и r -алгоритмы [24].
- 6.2 СТЕЦЮК (1998) Линейные операторы в квазиньютоновских методах [34].
- 6.3 СТЕЦЮК (2011) Субградиентные методы с преобразованием пространства для минимизации овражных выпуклых функций [123].
- 6.4 СТЕЦЮК (2013) r -Алгоритмы: теория и практика [166].

В работе 6.1 обсуждается ряд вопросов, связанных с квазиньютоновскими методами и r -алгоритмами. Построен метод с преобразованием пространства для минимизации гладких выпуклых функций. Показана его эффективность для минимизации сильно овражных квадратичных функций и обсуждается перенос аналогичных идей на негладкий случай. В статье 6.2 анализируются линейные операторы преобразования пространства для двух известных квазиньютоновских методов: Давидона – Флетчера – Пауэлла и Бройдена – Флетчера – Шенно. Для этих методов дана геометрическая интерпретация градиентного шага в преобразованном пространстве переменных.

В статье 6.3 приведена octave-функция `ralgb5`, которая реализует r -алгоритм с адаптивной регулировкой шага и постоянным коэффициентом растяжения пространства. Эта статья появилась благодаря доктору физ.-мат. наук С.П. Шарому (Институт Вычислительных Технологий СО РАН, Новосибирск), который использует r -алгоритмы для решения задач интервальной математики. Тезисы 6.4 включают краткий анализ имеющихся теоретических и практических результатов по r -алгоритмам. Это было предметом доклада автора на XI Международной научно-практической конференции „Математичне та програмне забезпечення інтелектуальних систем“, которая состоялась 20–22 ноября 2013 года в Днепропетровском национальном университете имени Олеся Гончара.

Седьмой раздел называется „Юбилейная статья“. В него включены две такие работы:

7.1 СТЕЦЮК (1995) Об одной схеме методов отсечений [21].

7.2 СЕРГИЕНКО, СТЕЦЮК (2012) О трех научных идеях Н.З. Шора [134].

В первой работе обсуждается общая схема метода центров тяжести простых тел для задачи выпуклого программирования. Она базируется на локализации множества минимумов монотонно уменьшающимися по объему эллипсоидами, симплексами и гиперпараллелепипедами и их симметризации с помощью линейных преобразований пространства. Главной в этом разделе является статья, которая посвящена 75-летию со дня рождения Наума Зуселевича Шора. Инициатива этой статьи принадлежит академику НАН Украины Ивану Васильевичу Сергиенко. В статье акцент сделан на трех центральных идеях Н.З. Шора: обобщенном градиентном спуске (1962), использовании линейных неортогональных преобразований пространства для улучшения обусловленности овражных функций (1969), двойственном подходе к получению и уточнению оценок целевой функции в невыпуклых квадратичных задачах (1985). Приведены применения этих идей в методах и алгоритмах, разработанных в Институте кибернетики им. В.М. Глушкова НАН Украины.

Метод эллипсоидов является одним из „красивых“ объяснений второй идеи Н.З. Шора для алгоритмов с растяжением пространства в направлении субградиента. Хочется надеяться, что не менее „красивое“ объяснение будет найдено и для r -алгоритмов. И если теоретическое обоснование таких методов будет использовать внешнюю аппроксимацию множества экстремумов эллипсоидами с монотонным уменьшением их объема, то это и будет осуществлением мечты Наума Зуселевича. Работы в этом направлении продолжаются.

Завершает книгу список научных трудов автора за 1982–2013 годы.

Автор признателен сотрудникам отдела методов негладкой оптимизации Института кибернетики им. В.М. Глушкова НАН Украины за помощь при подготовке книги и надеется, что она окажется полезной для специалистов по вычислительным методам оптимизации.

П.И. Стецюк
Киев, февраль 2014 г.

Р А З Д Е Л I

Метод эллипсоидов и его обобщения

В раздел включены статьи:

- 1.1. Метод отсечения с растяжением пространства для решения
задач выпуклого программирования (*Шор, 1977*) 14
- 1.2. Об одном семействе алгоритмов для решения задач выпуклого ...
программирования (*Шор, Гершович, 1979*) 19
- 1.3. Метод эллипсоидов, его обобщения и приложения
(*Гершович, Шор, 1982*) 30
- 1.4. Использование операции растяжения пространства в задачах
минимизации выпуклых функций (*Шор, 1970*) 51

Метод отсечения с растяжением пространства для решения задач выпуклого программирования

Н. З. Шор

Кибернетика. – 1977. – № 1. – С. 94–95.

Предложен и исследован алгоритм решения задачи выпуклого программирования, гарантирующий уменьшение объема области, в которой локализуется оптимум, со скоростью геометрической прогрессии, причем знаменатель этой прогрессии зависит только от размерности задачи.

В настоящей работе описан простой алгоритм решения задачи выпуклого программирования, гарантирующий уменьшение объема области, в которой локализуется оптимум, со скоростью геометрической прогрессии, причем знаменатель прогрессии зависит лишь от размерности задачи. Этот алгоритм относится к классу алгоритмов обобщенного градиентного спуска с растяжением пространства в направлении градиента (ОГСРП) [1]–[2]. С другой стороны, его можно рассматривать как метод отсечения, в котором операция растяжения пространства используется для симметризации области, в которой локализован оптимум.

Пусть мы имеем задачу выпуклого программирования:

$$\min f_0(x) \quad (1)$$

при ограничениях

$$f_i(x) \leq 0, \quad i = 1, \dots, m, \quad x \in E_n. \quad (2)$$

$f_\nu(x)$ – выпуклые функции, определенные на E_n ; $g_\nu(x)$ – соответствующие субградиенты, причем имеется априорная информация, что оптимальная точка x^* существует (она не обязательно единственная) и находится в шаре радиуса R с центром в точке x_0 (формально к системе ограничений (2) можно добавить ограничение $\|x - x_0\| \leq R$).

Рассмотрим следующий итеративный алгоритм (при $n > 1$).

Перед первым шагом имеем $x_0 \in E_n$, $B_0 = I$ – единичная матрица, $h_0 = \frac{R}{n+1}$. Пусть проделано k шагов и мы получили $x_k \in E_n$; B_k – матрица $n \times n$, $h_k > 0$.

(k + 1)-й шаг. Вычисляем:

$$1) \quad g(x_k) = \begin{cases} g_0(x_k), & \text{если } \max_{1 \leq i \leq m} f_i(x_k) \leq 0, \\ g_{i^*}(x_k), & \text{если } \max_{1 \leq i \leq m} f_i(x_k) = f_{i^*}(x_k) > 0. \end{cases} \quad (3)$$

Если $g(x_k) = 0$, то x_k – оптимальная точка; заметим, что $(g(x_k), x_k - x^*) \geq 0$;

$$2) \quad \xi = \frac{B_k^* g(x_k)}{\|B_k^* g(x_k)\|}; \quad (4)$$

$$3) \quad x_{k+1} = x_k - h_k \cdot B_k \cdot \xi_k; \quad (5)$$

$$4) \quad B_{k+1} = B_k \cdot R_\beta(\xi_k), \quad \beta = \sqrt{\frac{n-1}{n+1}}; \quad (6)$$

$R_\beta(\xi_k)$ – оператор растяжения пространства в направлении ξ_k с коэффициентом β [3];

$$5) \quad h_{k+1} = h_k \cdot r, \quad r = \frac{n}{\sqrt{n^2 - 1}}. \quad (7)$$

Лемма. Последовательность, $\{x_k\}_{k=0}^\infty$ генерируемая алгоритмом (3)–(7), удовлетворяет неравенству

$$\|A_k(x_k - x^*)\| \leq h_k \cdot (n + 1), \quad A_k = B_k^{-1}, \quad k = 0, 1, 2, \dots \quad (8)$$

Доказательство. Доказательство проведем индукцией по k . Для $k = 0$ неравенство (8) переходит в $\|x_0 - x^*\| \leq R$ и выполняется по предположению.

Пусть (8) выполняется для $k = \bar{k}$. Докажем его выполнение для $k = \bar{k} + 1$. Пусть для краткости $A_k(x_k - x^*) = z_k$, $\alpha = 1/\beta$,

$$\begin{aligned} \|z_{\bar{k}+1}\|^2 &= \|R_\alpha(\xi_{\bar{k}})(z_{\bar{k}} - h_{\bar{k}} \xi_{\bar{k}})\|^2 = \\ &= \|z_{\bar{k}} - h_{\bar{k}} \xi_{\bar{k}} + (\alpha - 1)[(z_{\bar{k}} - h_{\bar{k}} \xi_{\bar{k}}, \xi_{\bar{k}}) \xi_{\bar{k}}]\|^2 = \\ &= \|z_{\bar{k}} + [-\alpha h_{\bar{k}} + (\alpha - 1)(z_{\bar{k}}, \xi_{\bar{k}})] \xi_{\bar{k}}\|^2 = \\ &= \|z_{\bar{k}}\|^2 - 2h_{\bar{k}}(z_{\bar{k}}, \xi_{\bar{k}})\alpha^2 + (\alpha^2 - 1)(z_{\bar{k}}, \xi_{\bar{k}})^2 + \alpha^2 h_{\bar{k}}^2 = \\ &= \|z_{\bar{k}}\|^2 - 2h_{\bar{k}}(z_{\bar{k}}, \xi_{\bar{k}}) \cdot \frac{n+1}{n-1} + \frac{2}{n-1}(z_{\bar{k}}, \xi_{\bar{k}})^2 + \frac{n+1}{n-1}h_{\bar{k}}^2. \end{aligned} \quad (9)$$

Заметим, что

$$(z_k^-, \xi_k^-) = \frac{1}{\|\tilde{g}(x_k^-)\|} \left(A_k(x_k^- - x^*), B_k^* g(x_k) \right) = \frac{1}{\|\tilde{g}(x_k^-)\|} \left(x_k^- - x^*, g(x_k^-) \right) \geq 0.$$

Здесь $\tilde{g}(x_k^-) = B_k^* g(x_k^-)$.

Отсюда

$$\frac{2}{n-1} (z_k^-, \xi_k^-)^2 - 2 h_k^- \frac{n+1}{n-1} (z_k^-, \xi_k^-) \leq \frac{2}{n-1} \left[(z_k^-, \xi_k^-)^2 - (z_k^-, \xi_k^-)^2 \right] = 0.$$

Таким образом,

$$\begin{aligned} \|z_{k+1}^-\|^2 &\leq \|z_k^-\|^2 + \frac{n+1}{n-1} h_k^2 \leq (n+1)^2 h_k^2 + \frac{n+1}{n-1} h_k^2 = \\ &= (n+1)^2 \cdot \frac{n^2}{n^2-1} h_k^2 = (n+1)^2 \cdot h_{k+1}^2, \end{aligned}$$

откуда следует справедливость (8).

Лемма доказана.

Множество точек x , удовлетворяющих неравенству

$$\|A_k(x_k - x)\| \leq (n+1) h_k = R \cdot \left(\frac{n}{\sqrt{n^2-1}} \right)^k,$$

представляет собой эллипсоид Φ_k , объем которого $v(\Phi_k)$ равен

$$\frac{v_0 R^n \left(\frac{n}{\sqrt{n^2-1}} \right)^{nk}}{\det A_k},$$

где v_0 — объем единичного n -мерного шара. Получаем

$$\begin{aligned} \frac{v(\Phi_{k+1})}{v(\Phi_k)} &= \frac{\left(\frac{n}{\sqrt{n^2-1}} \right)^n \cdot \det A_k}{\det A_{k+1}} = \frac{\left(\frac{n}{\sqrt{n^2-1}} \right)^n \cdot \det A_k}{\det R_\alpha(\xi_k) \cdot \det A_k} = \\ &= \frac{1}{\alpha} \left(\frac{n}{\sqrt{n^2-1}} \right)^n = \sqrt{\frac{n-1}{n+1}} \left(\frac{n}{\sqrt{n^2-1}} \right)^n = q_n < 1. \end{aligned}$$

Таким образом, объем эллипсоида, в котором локализуется оптимальная точка x^* в соответствии с неравенством (8), убывает со скоростью геометрической прогрессии со знаменателем q_n .

К алгоритму (3)–(7) можно прийти, исходя из схемы последовательных отсечений¹. Рассмотрим, шар $S_0 = \{x : \|x - x_0\| \leq R\}$. Проведем гиперплоскость $(g(x_0), x - x_0) = 0$. Область локализации x^* сужается до полусферы $\bar{S} = S_0 \cap \Pi_0$, где $\Pi_0 = \{x : (g(x_0), x - x_0) \leq 0\}$. Опишем вокруг \bar{S} эллипсоид минимального объема. Элементарные геометрические построения показывают, что центр этого эллипсоида будет равен

$$x_0 - \frac{R}{n+1} \cdot \frac{g(x_0)}{\|g(x_0)\|} = x_0 - \frac{R}{n+1} \xi_0,$$

т. е. будет совпадать с x_1 . Если теперь произвести растяжение пространства в направлении $\xi_0 = \frac{g(x_0)}{\|g(x_0)\|}$, то в растянутом пространстве образом указанного эллипсоида будет шар с радиусом $R \cdot \frac{n}{\sqrt{n^2 - 1}}$ с центром в точке $R_\alpha(\xi_0)x_1$. Теперь можно повторить процесс отсечения в растянутом пространстве и т. д. Легко увидеть, что при этом получается алгоритм вида (3)–(7).

Следует отметить, что геометрическая скорость убывания объема области, в которой локализован оптимум, для общего класса задач выпуклого программирования была показана, насколько известно автору, впервые для интересных в теоретическом отношении алгоритмов центрированных сечений и их модификаций, предложенных А. Левиным в [4].

Однако эти алгоритмы требуют для размерности $n > 2$ весьма громоздких процедур на каждом шаге, что делает их практически мало пригодными, по-видимому, уже для $n = 4$.

Литература

1. Шор Н. З. Использование операции растяжения пространства в задачах минимизации выпуклых функций // Кибернетика. – 1979. – № 1. – С. 6–12.
2. Шор Н. З. Обобщенные градиентные методы минимизации негладких функций и их применение к задачам математического программирования (Обзор) // Экономика и мат. методы. – 1976. – 12, вып. 2. – С. 337–356.

¹Такая схема была рассмотрена в [3]

3. Юдин Д. Б., НЕМИРОВСКИЙ А. С. Информационная сложность и эффективные методы решения выпуклых экстремальных задач // Экономика и мат. методы. — 1976. — **12**, вып. 2. — С. 357–369.
4. ЛЕВИН А. Ю. Об одном алгоритме минимизации выпуклых функции // Докл. АН СССР, 1965. — **160**, № 6. — С. 1244–1247.

Об одном семействе алгоритмов для решения задач выпуклого программирования

Н. З. Шор, В. И. Гершович

Кибернетика. — 1979. — № 4. — С. 62–67.

Предлагаются алгоритмы для решения задач выпуклого программирования, комбинирующие схемы отсечений и линейные преобразования в пространстве переменных.

Введение

В работе [1] показано, что одна из модификаций метода обобщенного градиентного спуска с растяжением пространства в направлении градиента (ОГСРП) фактически реализует модифицированный метод центрированных сечений (ММЦС), рассмотренный в [2]. Соответствующий алгоритм применительно к задаче выпуклого программирования гарантирует уменьшение объема области, в которой локализуется минимум, со скоростью геометрической прогрессии. Этот алгоритм является «универсальным» по отношению к классу выпуклых функций в том смысле, что знаменатель q_n указанной геометрической прогрессии зависит лишь от n — размерности пространства переменных, а не от структуры минимизируемой функции. К сожалению, при увеличении размерности пространства q_n стремится к единице, тогда как теоретические исследования позволяют надеяться на возможность построения эффективно реализуемых алгоритмов, обеспечивающих на каждом шаге коэффициент уменьшения объема области локализации минимума $1 - C$, где $C > 0$ не зависит от размерности пространства [2, 3]. Вместе с тем, r -алгоритмы [4], основанные на обобщенном градиентном спуске с растяжением пространства в направлении разности двух последовательных градиентов, при решении большого числа разнообразных практических задач обеспечивают скорость сходимости по функционалу, не уступающую оценкам скорости сходимости теоретически оптимальных алгоритмов, рассмотренных в [2]. Каждому из последовательных субградиентов в r -алгоритме можно сопоставить отсекающую гиперплоскость, однако в явном виде в r -алгоритме идея отсечения не

заложена. Естественным, поэтому, представляется подход, комбинирующий схемы отсечения и преобразования пространства с целью получения алгоритма, не уступающего по эффективности r -алгоритму и более строго обоснованного с точки зрения гарантированной скорости сходимости.

Данная работа предлагает определенное продвижение в указанном направлении.

В отличие от ММЦС (или, что то же самое, от специального варианта ОГСРП), использующих на каждом шаге лишь один субградиент, будем строить методы, которые позволяют использовать информацию о нескольких субградиентах. Определенная попытка использовать такую информацию сделана в [5].

Кратко остановимся на простых геометрических идеях, лежащих в основе методов, комбинирующих схемы отсечения с преобразованиями пространства. Пусть в n -мерном пространстве E^n область G , в которой локализован минимум, представляет собой выпуклую фигуру, отсекаемую в шаре некоторой совокупностью плоскостей. Найдем эллипсоид минимального объема, описанный вокруг G . Отношение объема эллипсоида к объему шара q представляет собой степень уменьшения объема области локализации минимума. Чтобы получить вновь исходную ситуацию и замкнуть тем самым процесс, нужно произвести растяжения пространства в направлениях главных осей эллипсоида, необходимые для преобразования его в сферу.

Такой подход можно рассматривать как схему отсечений, в которой операция растяжения используется для симметризации области локализации минимума. ММЦС и методика, изложенная в работе [5], представляют частные случаи этого подхода. ММЦС соответствует отсечение лишь одной диаметральной плоскостью, и, как указывалось выше, при больших n метод не дает хорошего выигрыша по объему. По-видимому, и прямое применение методики [5], состоящее в описывании эллипсоида вокруг шарового слоя, не даст желаемого эффекта, так как существенный выигрыш по объему возможен лишь для слоев высотой порядка R/\sqrt{n} (R – радиус шара), получение которых мало вероятно.

Описывающий эллипсоид

Согласно изложенному, нам потребуется умение описывать оптимальные (или близкие к ним) эллипсоиды вокруг полученных областей локализации минимума. Проведем соответствующие геометрические по-

строения. Области, в которых локализован минимум, будем представлять фигурами трех типов:

- 1) сегмент — часть шара, отсеченная от него плоскостью;
- 2) слой — часть шара, заключенная между двумя параллельными плоскостями;
- 3) s -пирамида — часть шара, ограниченная совокупностью плоскостей, попарно расположенных под нетупыми углами и проходящими через некоторую внутреннюю точку этого шара (вершину пирамиды), при этом соответствующая совокупность нормалей к плоскостям составляет систему линейно независимых векторов.

Все фигуры удовлетворяют тому условию, что они принадлежат некоторому полушарию шара S , на основе которого образованы (условие $C1$).

Рассмотрим способ описывания эллипсоида вокруг фигуры отдельно для каждого вида фигур. При этом благодаря простоте соответствующих геометрических задач формулы для сегмента и слоя представим с очень краткими пояснениями, а на формуле для пирамиды остановимся более подробно.

В дальнейшем, не ограничивая общности, будем считать, что шар S , на основе которого образованы требующие описывания фигуры, n -мерный радиусом $R = 1$ и объемом v_0 .

Случай сегмента. Пусть сегмент отсекается от шара плоскостью, отстоящей от центра шара на расстоянии d . Искомый оптимальный эллипсоид есть эллипсоид вращения вокруг оси симметрии сегмента. Центр эллипсоида находится на оси вращения внутри сегмента на расстоянии $\frac{R-d}{n+1}$ от плоскости его основания. Величина полуоси, связанной с осью вращения, $a = (R-d)\frac{n}{n+1}$. На остальных $(n-1)$ осях отсекаются отрезки $b = \sqrt{R^2 - d^2} \cdot \frac{n}{\sqrt{n^2 - 1}}$. Коэффициент растяжения для преобразования в сферу

$$\alpha = \frac{b}{a} = \sqrt{\frac{R+d}{R-d} \cdot \frac{n+1}{n-1}}.$$

Объем эллипсоида

$$V = v_0 a b^{n-1} = v_0 b^n \frac{a}{b} = v_0 \left[\frac{R^2 - d^2}{\left(1 + \frac{1}{n}\right) \left(1 - \frac{1}{n}\right)} \right]^{n/2} \cdot \left(\frac{R-d}{R+d} \cdot \frac{n-1}{n+1} \right)^{1/2}.$$

Подставляя для простоты $R = 1$, имеем

$$V = v_0 \left[\frac{1 - d^2}{\left(1 + \frac{1}{n}\right) \left(1 - \frac{1}{n}\right)} \right]^{n/2} \cdot \left(\frac{1 - d}{1 + d} \cdot \frac{n - 1}{n + 1} \right)^{1/2}.$$

Ясно, что существенный выигрыш по объему (выигрыш, практически не ухудшающийся с увеличением n) может быть получен лишь при больших d , начиная с $d \sim 1/\sqrt{n}$, т. е. для сегментов, достаточно удаленных от центра шара.

Случай слоя. Пусть слой, удовлетворяющий условию $C1$, высекается из шара двумя параллельными плоскостями, отстоящими на расстоянии $H - d$ одна от другой, при этом большее основание слоя удалено от центра шара на d . Искомый оптимальный описывающий эллипсоид является эллипсоидом вращения вокруг оси симметрии слоя. Центр эллипсоида находится на оси вращения на расстоянии t от большего основания слоя. Величина t — меньший корень квадратного уравнения:

$$(n + 1)(H + d)t^2 - \left[2(R^2 - d^2) + n(H^2 - d^2) \right]t + (R^2 - d^2)(H - d) = 0. \quad (1)$$

Величина полуоси, связанной с осью вращения,

$$a = \left[\frac{(H + d)t^2 - 2(R^2 - d^2)t + (R^2 - d^2)(H - d)}{H + d} \right]^{1/2}. \quad (2)$$

На остальных $(n - 1)$ главных осях эллипсоида отсекаются отрезки

$$b = \left[\frac{(H + d)t^2 - 2(R^2 - d^2)t + (R^2 - d^2)(H - d)}{H - d - 2t} \right]^{1/2}. \quad (3)$$

Коэффициент растяжения для преобразования в сферу

$$\alpha = \frac{b}{a} = \left(\frac{H + d}{H - d - 2t} \right)^{1/2}.$$

Подставляя в (2) и (3) $R = 1$ и полагая для простоты $d = 0$ (при $d > 0$ выигрыш по объему может лишь улучшиться), получаем

$$a = \left(\frac{Ht^2 - 2t + H}{H} \right)^{1/2}, \quad b = \left(\frac{Ht^2 - 2t + H}{H - 2t} \right)^{1/2} \quad (4)$$

и уравнение для определения t :

$$(n + 1)Ht^2 - (2 + nH^2)t + H = 0. \quad (5)$$

Выражение для объема эллипсоида имеет вид

$$V = v_0 a b^{n-1} = v_0 \frac{a}{b} b^n = v_0 \left(1 - \frac{2t}{H}\right)^{1/2} \cdot \left(1 + \frac{Ht^2}{H-2t}\right)^{n/2}. \quad (6)$$

Несложное исследование зависимости V от H с учетом (5) показывает, что существенный выигрыш по объему получается лишь при достаточно малых H (начиная с $H \sim 1/\sqrt{n}$), т. е. для достаточно тонких слоев. Так, при $H = \frac{1}{\sqrt{n}}$ из (5) имеем $t = \frac{3-\sqrt{5}}{2} \cdot \frac{1}{\sqrt{n}}$. Тогда

$$V \approx v_0 (\sqrt{5} - 2)^{1/2} \left[1 + \frac{(3-\sqrt{5})^2}{4(\sqrt{5}-2)} \cdot \frac{1}{n} \right]^{1/2} \approx 0,66 v_0.$$

Случай s -пирамиды. Пусть s -пирамида Π высекается из n -мерного единичного шара S совокупностью из k гиперплоскостей с нормальными соответственно g^1, g^2, \dots, g^k . Обозначим через L_k подпространство, натянутое на g^1, g^2, \dots, g^k (предполагаем, что они линейно независимы). Под s -пирамидой будем понимать пересечение s -пирамиды с подпространством, натянутым на совокупность нормалей к ее граням. Таким образом, каждой s -пирамиде ставится в соответствие s -пирамида, и наоборот. Обозначим через Π s -пирамиду, соответствующую s -пирамиде Π . По определению $\Pi = \Pi \cap L_k$. Очевидно, все векторы из Π можно представить в виде $x = v + \sum_{i=1}^k \lambda_i p^i$, где вектор v — вершина пирамиды; $\lambda_i \geq 0$, $\{p^i, i = 1, 2, \dots, k\}$ — набор k -мерных векторов, имеющих простой геометрический смысл ребер s -пирамиды Π . При этом длину каждого вектора p^i можно выбрать таким образом, что, если его начало поместить в вершину пирамиды, то конец окажется на сферической поверхности шара S . С s -пирамидой естественным образом связан симплекс, натянутый на ее ребра. Соответствующий симплекс для Π обозначим через M .

Введем в E^n систему координат с началом в вершине s -пирамиды Π и ортонормированным базисом l^1, l^2, \dots, l^n , такую, что первые k базисных векторов будут определять подпространство, совпадающее с L_k . Рассмотрим эллипсоиды, описанные вокруг Π , с центром в начале координат. Среди них будем искать минимальный по объему. В дальнейшем будем оперировать с s -пирамидами, у которых ребра соответствующих s -пирамид по длине не превышают радиуса шара (условие $C2$). Ясно, что в искомом эллипсоиде, построенном для s -пирамиды Π , удо-

влетворяющей условию $C2$, $(n-k)$ полуосей (связанных с подпространством, ортогональным к L_k) определяются независимо от остальных. Для всех них $h = \sqrt{1 - \rho^2}$, где ρ – расстояние между центром шара и вершиной s -пирамиды Π . Остальные k полуосей совпадают с полуосями минимального по объему k -мерного эллипсоида с центром в начале координат, описанного вокруг симплекса M , связанного с Π . Для построения такого эллипсоида воспользуемся тем, что отношение объемов тел в E^n есть инвариант невырожденного линейного преобразования. Поэтому, так как все k -мерные симплексы аффинно подобны, достаточно построить оптимальный эллипсоид для одного из них. В качестве последнего возьмем симплекс M_0 с ребрами l^1, l^2, \dots, l^k . Легко видеть, что оптимальным эллипсоидом (среди тех, которые имеют центр в начале координат!), описывающим M_0 , является k -мерная сфера единичного радиуса. Существует невырожденное линейное преобразование A , переводящее M в M_0 . При этом выполняется $Ap^i = l^i$, $i = 1, 2, \dots, k$, или в матричной записи

$$AP = E, \quad (7)$$

где P – матрица векторов столбцов p^i ; E – матрица векторов столбцов l^i , $i = 1, 2, \dots, k$. Если применить к M_0 преобразование $B = A^{-1}$ (см. (7)), то M_0 перейдет в M , а k -мерная единичная сфера – в оптимальный (вновь среди имеющих центром начало координат) эллипсоид, описанный вокруг M . Этот эллипсоид – k -мерное сечение искомого эллипсоида, описанного вокруг s -пирамиды Π . Очевидно, объем последнего $v_0 h^{n-k} \|\det B\|$. Так как Π образована на основе единичной сферы объемом v_0 , выигрыш по объему при переходе от этой сферы к эллипсоиду, описанному вокруг Π будет

$$q = \frac{v_0 h^{n-k} \|\det B\|}{v_0} = h^{n-k} \|\det P\|. \quad (8)$$

Но $\|\det P\|$ численно равен объему k -мерного параллелепипеда, натянутого на ребра p^1, p^2, \dots, p^k . Поэтому для s -пирамид, ребра которых, не превышают радиуса, из (8) можно заключить, что хороший выигрыш по объему получается в случае, когда ребра p^1, p^2, \dots, p^k лежат под углами, заметно отличными от $\pi/2$. Отметим, что при $k = 2$ $\det P$ равен векторному произведению p^1 и p^2 , т. е. $q \leq \sin \varphi$, где φ – угол между p^1 и p^2 . Таким образом, избегаем жесткой зависимости q от размерности пространства (как в ММЦС) и получаем возможность учитывать специфику конкретной функции.

Выше было рассмотрено построение описанного вокруг s -пирамиды Π эллипсоида с центром в фиксированной точке (вершине), имеюще-

го минимальный объем. Ясно, что минимальный по объему эллипсоид, описанный вокруг Π , может иметь центр, не совпадающий с вершиной пирамиды. Приступим к построению такого эллипсоида.

Рассмотрим такую s -пирамиду, что соответствующая ей s -пирамида Π_0 имеет своими ребрами векторы l^1, l^2, \dots, l^k . Очевидно, симплекс, соответствующий s -пирамиде Π_0 , совпадает с используемым ранее M_0 . Аналогично указанному, для построения оптимального эллипсоида, описанного вокруг Π (обозначим его Φ), достаточно описать оптимальный эллипсоид вокруг Π_0 (обозначим его Φ_0) и применить к нему линейное преобразование B .

Задача построения оптимального эллипсоида для Π_0 решается аналитически, если соображения симметрии позволяют заранее указать положение его центра и направление всех главных осей.

Таким образом, справедливы следующие утверждения:

- 1) центр эллипсоида находится на медиатриссе симплекса M_0 , вдоль которой проходит одна выделенная главная ось (длину соответствующей полуоси обозначим a);
- 2) группа из $(k - 1)$ ортогональных к медиатриссе осей образует вместе с ней ортогональную систему в L_k (всем осям соответствуют одинаковые по величине полуоси b);
- 3) группа из $(n - k)$ осей дополняет указанные k осей до ортогональной системы в E^n (им соответствуют полуоси длиной c);
- 4) искомый эллипсоид проходит через концы векторов l^1, l^2, \dots, l^k . Тогда после несложного геометрического анализа получаем

$$a = \left(\frac{1 - \frac{k}{n+1}}{1 + \frac{1}{n}} \right)^{1/2}; \quad (9)$$

$$b = \left(\frac{1}{1 + \frac{1}{n}} \right)^{1/2}; \quad (10)$$

$$c = \left(\frac{1 + \frac{1}{n-k}}{1 + \frac{1}{n}} \right)^{1/2} \cdot h. \quad (11)$$

Центр эллипсоида Φ_0 лежит на медиатриссе симплекса M_0 , отсекая от нее $k/(n+1)$ часть, считая от вершины. И так как длина медиатриссы симплекса равна $1/\sqrt{k}$, то центр эллипсоида отсекает $\sqrt{k}/(n+1)$ часть

радиуса сферы, содержащего медиатриссу. Объем эллипсоида

$$V = v_0 a b^{k-1} c^{n-k} = v_0 \left[\frac{\left(1 - \frac{k}{n+1}\right) \left(1 + \frac{1}{n-k}\right)^{n-k}}{\left(1 + \frac{1}{n}\right)^n} \right]^{1/2} \cdot h^{n-k}.$$

В результате применения к Φ_0 преобразования B и получения эллипсоида, описанного вокруг Π , имеем выигрыш по объему:

$$q = \left[\frac{\left(1 - \frac{k}{n+1}\right) \left(1 + \frac{1}{n-k}\right)^{n-k}}{\left(1 + \frac{1}{n}\right)^n} \right]^{1/2} \cdot h^{n-k} |\det P|. \quad (12)$$

Отметим одно обстоятельство. Безусловно, необходимо выполнение условия

$$\Pi \subseteq B\Phi_0. \quad (13)$$

Если ребра p^1, p^2, \dots, p^k лежат под углами, заметно отличными от $\pi/2$, условие (13) выполняется. Если ребра ортогональны (или почти ортогональны), т. е. Π с точностью до поворота совпадает с Π_0 , (13) перестает выполняться, так как включение $\Pi_0 \subseteq \Phi_0$ не является справедливым. Дело в том, что принадлежащая Π_0 точка Q , в которой медиатрисса пересекает сферическую поверхность шара S , отстоит от центра Φ_0 на $a' = 1 - \frac{\sqrt{k}}{n+1}$, тогда как связанная с медиатриссой полуось

$$a = \left(\frac{1 - \frac{k}{n+1}}{1 + \frac{1}{n}} \right)^{1/2} < a'.$$

Рассмотрим эллипсоид Φ'_0 , отличающийся от Φ_0 лишь тем, что в нем вместо полуоси a взята полуось a' . Очевидно, $\Pi_0 \subseteq \Phi'_0$ и вместо условия (13) имеем $\Pi_0 \subseteq B\Phi'_0$. Выигрыш по объему при использовании Φ'_0 будет

$$q' = \left(1 - \frac{\sqrt{k}}{n+1} \right) \left[\frac{\left(1 + \frac{1}{n-k}\right)^{n-k}}{\left(1 + \frac{1}{n}\right)^{n-1}} \right]^{1/2} \cdot h^{n-k} |\det P|. \quad (14)$$

Таким образом, сделан переход от Φ_0 к Φ'_0 , поскольку утверждения 1–4, на основании которых построен Φ_0 , не гарантируют описывания всей s -пирамиды Π_0 , а только тела $\Pi_0 \setminus (\Pi_0 \setminus M_0)$. Добавив дополнительное требование прохождения искомого эллипсоида через точку Q

можно получить эллипсоид, описанный вокруг P_0 . Следует отметить, что формулы для полуосей такого эллипсоида несколько более сложные по сравнению с (9)–(11) и здесь не приводятся.

Общая схема алгоритма

Из (12) и (14) видно, что с ростом k выигрыш по объему улучшается (как и должно быть при увеличении количества граней пирамиды). Однако желание точнее аппроксимировать область локализации минимума, т. е. использовать пирамиду с большим числом граней, приводит к увеличению вычислительной работы при построении соответствующего описывающего эллипсоида. Поэтому остановимся на том варианте алгоритма, при котором $k = 2$, т. е. используются две отсекающие плоскости. При этом основной вклад в уменьшение объема области локализации минимума будет вносить (см. (12) и (14)) член $h^{n-k} |\det P|$. Выбор $k = 2$ вызван еще тем соображением, что, r -алгоритму, как указывалось выше, соответствует некоторая схема отсечений с двумя субградиентами. Для получения более сильного отсечения целесообразно использовать свойство выпуклости функции. Пусть выпуклая функция $f(x)$ в точке x_0 принимает значение $f(x_0)$ и обладает субградиентом $g(x_0)$. Область локализации минимумов функции $f(x)$ лежит в полупространстве $H_0 = \{x : (g(x_0), x - x_0) \leq 0\} = \{x : (g(x_0), x) \leq f_0, f_0 = (g(x_0), x_0)\}$. Если известно, что $f(x)$ может принимать значение $f^* < f(x_0)$ (например, f^* – полученное ранее рекордное значение или априорно известное значение минимума), то H_0 можно сузить до полупространства

$$\begin{aligned} H_1 &= \left\{ x : \left(g(x_0), x - x_0 + \frac{(f(x_0) - f^*)}{\|g(x_0)r\|} \cdot \frac{g(x_0)}{\|g(x_0)\|} \right) \leq 0 \right\} = \\ &= \left\{ x : (g(x_0), x) \leq f_0 - (f(x_0) - f^*), \quad f_0 = (g(x_0), x_0) \right\}. \end{aligned}$$

Такое отсечение с использованием информации о рекордном значении функции будем называть суперотсечением. Ниже под отсекающей плоскостью будем понимать суперотсечение. Перейдем теперь непосредственно к алгоритму.

Пусть имеем задачу безусловной минимизации выпуклой функции $f(x)$, $x \in E^n$, а $g(x)$, соответствующий субградиент, и есть априорная информация, что оптимальная точка x^* существует (не обязательно единственная) и находится в шаре радиусом R_0 с центром в точке x_0 .

Перед первым шагом имеем $x \in E^n$, $B_0 = I$ – единичная матрица порядка $n \times n$, число $R_0 > 0$. При этом $x^* \in S_0$, где $S_0 = \{x : \|x - x_0\| \leq R_0\}$. Пусть проделано k шагов и получено $x_k \in E^n$, матрица $B_k = A_k^{-1}$ порядка $n \times n$, радиус R_k . Матрица A_k содержит информацию о всех преобразованиях пространства, проделанных за предыдущие k шагов. При этом можно утверждать, что $y^* \in S_k$, где $S_k = \{y : \|y - y_k\| \leq R_k\}$, $y_k = A_k x_k$, $y^* = A_k x^*$. Следующий, $(k+1)$ -й, шаг состоит из нескольких этапов:

- 1) вычисляем в точке y_k функцию $\varphi(y_k) = f(B_k y_k) = f(x_k)$, градиент $\tilde{g}_k^1 = B_k^* g(x_k)$ и получаем первую отсекающую плоскость P_k^1 в преобразованном пространстве;
- 2) выбираем вторую отсекающую плоскость P_k^2 и описываем оптимальный эллипсоид Φ_k вокруг фигуры, аппроксимирующей область локализаций минимума;
- 3) получаем x_{k+1} как прообраз центра Φ_k в исходном пространстве;
- 4) производим растяжения пространства, необходимые для преобразования Φ_k в сферу S_{k+1} радиусом R_{k+1} . Получаем результирующее преобразование S_{k+1} .

Этап 2 особенно ответственный, поэтому остановимся на нем подробнее. Стратегии, применяемые на этом этапе, могут быть самыми разнообразными. В частности, для выбора P_k^2 можно использовать некоторый набор отсекающих плоскостей, полученных на предыдущих шагах. В зависимости от применяемой стратегии получаются различные алгоритмы. Опишем один из возможных вариантов такой стратегии. В качестве P_k^2 выберем плоскость, связанную с субградиентом \tilde{g}_k^2 функции $\varphi(y)$ в некоторой точке \bar{y}_k , принадлежащей лучу

$$L = \left\{ y : y = y_k - t \tilde{g}_k^1, \quad t > 0, \quad \|y - y_k\| \leq R_k \right\}.$$

Выбор точки \bar{y}_k можно рассматривать как линейный поиск в направлении \tilde{g}_k^1 . При этом нет необходимости проводить такой поиск точно. Будем двигаться от y_k по точкам из L с некоторым шагом (по-видимому, целесообразно выбирать шаг $\sim \frac{R_k}{\sqrt{n}}$) и вычислять в каждой точке функцию $\varphi(y)$ и субградиент \tilde{g} , получая некоторую отсекающую плоскость P . Если в очередной точке $(\tilde{g}, \tilde{g}_k^1) > 0$, значение функции меньше, чем в

предыдущей точке. В этом случае область локализации минимума можно аппроксимировать одним из сегментов, отсекаемых от S_k плоскостями соответственно P_k^1 и P . Если в очередной точке из L оказывается выполненным условие $(\tilde{g}, \tilde{g}_k^1) \leq 0$, то получено как бы «встречное» отсечение и область локализации минимума аппроксимируется слоем или, что более вероятно при больших n , s -пирамидой. Теперь из всех полученных при поиске вдоль L аппроксимаций области локализации минимума выбираем оптимальную с точки зрения минимума объема описывающего эллипсоида. Заметим, что по построению используемые для аппроксимации области локализации минимума фигуры удовлетворяют условию $C1$. Наконец, если для некоторой полученной s -пирамиды Π не выполняется условие $C2$, то вместо Π всегда можно использовать другую s -пирамиду Π' , для которой $C2$ выполняется и соответствующий описанный эллипсоид содержит Π . Вершина Π' есть основание перпендикуляра, опущенного из y_k на Π , а ребра Π' таковы, что их концы совпадают с концами ребер Π .

В заключение отметим, что о точности решения по функционалу позволяет судить значение величины $\|\tilde{g}_k\| R_k$. Неравенство $\|\tilde{g}_k\| R_k < \varepsilon$, можно использовать в качестве условия прекращения счета.

Для экспериментального исследования предложенного алгоритма в настоящее время разрабатывается программа для ЕС ЭВМ.

Литература

1. Шор Н. З. Метод отсечения с растяжением пространства для решения задач выпуклого программирования // Кибернетика. – 1977. – № 1. – С. 94–96.
2. Юдин Д. Б., Немировский А. С. Информационная сложность и эффективные методы решения выпуклых экстремальных задач // М.: Экономика и мат. методы. – 1976. – 12, вып. 2. – С. 357–369.
3. Левин А. Ю. Об одном алгоритме минимизации выпуклых функций // Докл. АН СССР. – 1965. – 160, № 6. – С. 1244–1247.
4. Шор Н. З., Журбенко Н. Г. Метод минимизации, использующий операцию растяжения в направлении разности двух последовательных градиентов // Кибернетика. – 1971. – № 3. – С. 51–59.
5. Гулинский О. В., Поляк Б. Т. Об одном алгоритме выпуклого программирования. – В сб.: «Системы программного обеспечения решения задач оптимального планирования». Тез. докл. V Всесоюз. симпоз. – М., 1978.

Метод эллипсоидов, его обобщения и приложения

В. И. Гершович, Н. З. Шор

Кибернетика. — 1982. — № 5. — С. 61–69.

Дается обзор методов решения задач выпуклого программирования, основанных на использовании отсечений гиперплоскостями и аппроксимации области локализации экстремума эллипсоидами, а также приложений этих методов к проблемам комбинаторной оптимизации и математического программирования.

Введение

Конструктивное доказательство полиномиальной сложности задачи линейного программирования, полученное в [5, 6], вызвало большой интерес в связи с использованной в этих работах методикой. Л. Г. Хачиян, рассматривая проблему решения системы линейных неравенств (к которой сводится задача линейного программирования), использовал модифицированный метод центров тяжести (ММЦТ). Этот метод, предложенный Д. Б. Юдиным, А. С. Немировским [15] и независимо разработанный Н. З. Шором [11], применим для решения общей задачи выпуклого программирования. При решении системы линейных неравенств соответствующей минимизируемой выпуклой функцией является «невязка» системы. ММЦТ был назван в западной литературе «алгоритмом эллипсоидов». После публикации [15] появился лавинообразный поток зарубежных работ, в которых рассматривались вопросы экспериментального сравнения работоспособности ММЦТ (метода эллипсоидов) и симплекс-метода, а также усовершенствования ММЦТ (так как он явно проигрывал при сравнении). Вульф специально выпустил по этим работам обширную библиографию [18]. Хороший обзор по этой тематике и смежным вопросам подготовил Шредер [20]. Следует отметить, что еще до появления [5] авторы данной статьи предложили в [14] семейство алгоритмов для решения задач выпуклого программирования, обобщающих ММЦТ и обладающих возможностями значительно повысить его эффективность. При их разработке важным явилось понимание того факта, что они, так же как и ММЦТ, являются алгоритмами градиентного типа с растяжением пространства.

Алгоритмы градиентного типа с растяжением пространства

Методы градиентного типа с растяжением пространства были разработаны для ускорения сходимости субградиентных процессов. Медленная сходимость этих процессов обусловлена тем, что, как правило, верхняя граница углов между направлением на точку минимума и направлением антисубградиента близка к $\pi/2$ или равна этому числу. В такой ситуации расстояние до точки минимума убывает, если шаг мал по сравнению с ним, при этом величина убывания намного меньше, чем длина шага, а значит, и скорость уменьшения шага не может быть слишком большой, если нужно гарантировать сходимость к минимуму.

С помощью неортогональных преобразований пространства аргументов можно изменить углы между направлением антисубградиента, вычисленным в точке, и направлением из этой точки на точку минимума. В квазиньютоновских алгоритмах с переменной метрикой, в основе которых лежит идея квадратичной аппроксимации данной функции, изменение метрики направлено на то, чтобы получить в преобразованном пространстве близкие к сферическим поверхности уровня в окрестности минимума. Формально это сводится к построению приближения к матрице, обратной к гессиану в точке минимума. Для минимизации негладких функций аналогичный подход в принципе невозможен. Например, для кусочно-линейных функций гессиан почти везде равен нулю и квадратичная аппроксимация бессмысленна. Для недифференцируемой оптимизации Н. З. Шор предложил использовать в качестве элементарных операторов преобразования пространства операторы растяжения пространства в определенных направлениях.

Отметим, что оператор растяжения пространства в направлении ξ ($\xi \in E_n$; $\|\xi\| = 1$) с коэффициентом $\beta > 0$ обозначается $R_\beta(\xi)$ и действует на произвольный вектор $x \in E_n$ по формуле (см. [13])

$$R_\beta(\xi)x = x + (\beta - 1)(x, \xi)\xi.$$

В матричной форме $R_\beta(\xi)$ представляется как

$$R_\beta(\xi) = I + (\beta - 1)\xi\xi^*,$$

где I — единичный оператор (знак «*» сверху здесь и ниже означает транспонирование), т. е. в виде вещественной симметричной матрицы.

Используя произведение таких операторов, можно перейти от сферической метрики к произвольной эллиптической метрике. Растяже-

ние пространства в определенном направлении с большим коэффициентом приближенно соответствует проектированию градиента на подпространство, ортогональное этому направлению. Можно рассматривать семейство алгоритмов, основанных на построении последовательности операторов преобразования пространства в виде произведения операторов растяжения пространства и выборе направления спуска, соответствующего направлению антиградиента в пространстве с измененной метрикой. Среди них подробно были исследованы алгоритмы двух типов: с растяжением пространства в направлении субградиента; с растяжением пространства в направлении разности двух последовательных субградиентов [7]–[9].

Опишем принципиальную схему алгоритмов градиентного типа с растяжением пространства в направлении градиента. Для минимизации функции $f(x)$, $g_f(x)$ – градиент (обобщенный градиент) функции $f(x)$ в точке x .

Заданы: $x_0 \in E_n$, $B_0 = A_0^{-1} = I$ (единичная матрица $n \times n$).

После k шагов имеем: $x_k \in E_n$, $B_k = A_k^{-1}$; A_k – матрица $n \times n$ преобразования пространства после k шагов.

($k + 1$)-й шаг.

1. Вычисляем $g_f(x_k)$ (если $g_f(x_k) = 0$, процесс останавливается).

2. Определяем

$$\tilde{g}_k = g_{\varphi_k}(y_k) = B_k^* g_f(x_k), \quad (1)$$

где $\varphi_k(y) = f(B_k y)$; $y_k = A_k x_k$; \tilde{g}_k – обобщенный градиент функции $\varphi_k(y)$, определенной в «растянутом» пространстве.

$$3. \xi_k = \tilde{g}_k / \|\tilde{g}_k\|; \quad x_{k+1} = x_k - h_k B_k \xi_k. \quad (2)$$

Формуле (2) соответствует движение по антиградиенту в растянутом пространстве: $A_k x_{k+1} = y_k - h_k \xi_k$.

$$4. B_{k+1} = A_{k+1}^{-1} = B_k R_{\beta_k}(\xi_k); \quad \beta_k = \alpha_k^{-1}. \quad (3)$$

Формуле (3) соответствует растяжение «растянутого» пространства в направлении ξ_k :

$$A_{k+1} = R_{\alpha_k}(\xi_k) A_k, \quad \alpha_k > 1.$$

5. $(k + 1) \rightarrow (k + 2)$. Переход к п. 1.

Первые проведенные Н. З. Шором и В. И. Билецким в 1969 г. вычислительные эксперименты по описанному алгоритму дали обнадеживающие результаты. Основная сложность при конструировании «работоспособного» алгоритма состояла в разумном выборе коэффициентов

растяжения пространства α_k и соответствующей стратегии изменения шаговых множителей h_k . Так как добиться монотонности спуска для семейства алгоритмов с растяжением пространства в направлении градиента в негладком случае не удастся, то пришлось довольствоваться априорным заданием способа изменения шаговых множителей при движении в направлении антисубградиента в растянутом пространстве. Как показывают простые рассуждения для одномерного случая, при использовании алгоритма с постоянным шаговым множителем в растянутом пространстве для получения сходимости необходимо выбирать коэффициент растяжения пространства $\alpha_k \leq 2$.

Вычислительные эксперименты показали, что выбор $\alpha_k = 2$ и $h_k = \text{const}$ для многих примеров выпуклых овражных функций дает хорошие результаты. К сожалению, такой простой способ не всегда приводит к цели. При построении других вариантов алгоритмов, которые удалось теоретически обосновать, был выбран шаговый множитель и коэффициенты растяжения пространства таким образом, чтобы последовательность расстояний до точки минимума в метрике соответствующих «растянутых» пространств не возрастала. Этот принцип гарантирует сходимость со скоростью геометрической прогрессии по функционалу.

Для реализации указанного принципа необходима некоторая априорная информация о функции $f(x)$ – значение функции в точке минимума f^* и так называемые константы роста M и N .

Теорема 1. [7, 8, 10]. Пусть $f(x)$ – почти дифференцируемая (локально липшицева) функция – рассматривается в некоторой сферической окрестности $S_d(x^*)$ точки локального минимума x^*

$$S_d(x^*) = \{x : \|x - x^*\| \leq d\}$$

и в тех точках, где функция дифференцируема, ее производная по направлению $\mu(x) = x - x^*$ удовлетворяет неравенству

$$[f(x) - f^*]N \leq f'_{\mu(x)}(x) \leq M[f(x) - f^*]; \quad M \geq N > 0. \quad (4)$$

Тогда, если в алгоритме (1)–(3) принять:

$$x_0 \in S_d(x^*), \quad (5)$$

$$h_k = \frac{2M}{M+N} \cdot \frac{f(x_k) - f^*}{\|\tilde{g}_k\|}, \quad (6)$$

$$\alpha_k = \alpha = \frac{M + N}{M - N}, \quad k = 1, 2, \dots, \quad (7)$$

то

$$\|A_k(x_k - x^*)\| \leq d. \quad (8)$$

Отметим, что неравенство (8) эквивалентно $(G(x_k - x^*), x_k - x^*) \leq d^2$, где $G = A_k^* A_k$ – положительно определенная матрица, причем

$$\det G = (\det A_k)^2 = \left(\prod_{i=0}^{k-1} \det R_{\alpha_i}(\xi_i) \right)^2 = \left(\prod_{i=0}^{k-1} \alpha_i \right)^2.$$

Оно дает для x^* локализацию в форме эллипсоида S_k с центром в точке x_k и с произведением длин полуосей, равным

$$\frac{d^n}{\det A_k^*} = d^n \det B_k,$$

причем отношение объемов эллипсоидов $\frac{\nu(S_k)}{\nu(S_{k+1})} = \frac{M + N}{M - N}$.

Таким образом, семейство алгоритмов (1)–(3) содержит в себе, как частный случай, алгоритмы, получившие после работы [5], название **методов эллипсоидов**.

Легко доказывается следующая лемма.

Лемма [8]. Пусть в процессе реализации алгоритма (1)–(3) последовательность $\{x_k\}_{k=0}^{\infty}$ ограничена, $1 + \delta \leq \alpha_k \leq \alpha^*$. Тогда существует подпоследовательность $x_{k_1}, x_{k_2}, \dots, x_{k_p}, \dots$, $k_i < k_{i+1}$, и $c > 0$ такие,

$$\text{что } \|\tilde{g}_{k_p}\| \leq c \left(\prod_{j=0}^{k_p} \beta_j \right)^{1/n}.$$

Из этой леммы, в условиях теоремы 1, вытекает, что справедливо следующее утверждение [8]: существуют константа $c > 0$ и подпоследовательность $\{k_p\}_{p=1}^{\infty}$, $k_p < k_{p+1}$, такие, что $f(x_{k_p}) - f^* \leq \bar{c} \alpha^{-k_p/n}$. В [12] для этого случая доказано более точное утверждение.

Теорема 2. В предположениях предыдущей леммы при $\alpha_k = \alpha > 1$

$$\nu_k = \min_{0 \leq r \leq k} \|\tilde{g}_r\| \leq \frac{d \sqrt{k(\alpha^2 - 1)}}{\sqrt{\alpha^{2k/n} - 1}}, \quad d = \max_{x \in S_d(x^*)} \|g_f(x)\|.$$

Из этой теоремы вытекает, что для случая алгоритма с растяжением пространства в направлении градиента с известными M , N и f^*

справедлива следующая оценка:

$$\min_{0 \leq r \leq k} [f(x_k) - f^*] \leq \frac{\bar{c} \sqrt{k(1 - \beta^2)}}{\sqrt{1 - \beta^{2k/n}}} \beta^{k/n-2}, \quad \text{где} \quad \beta = \frac{M - N}{M + N}.$$

Таким образом, скорость сходимости по функционалу зависит от β .

Отметим, что для квадратичной положительно определенной функции в неравенстве (4) можно выбирать $M = N = 2$, а для кусочно-линейной функции, надграфик которой представляет конус с вершиной в точке $\{x^*, f^*\}$, $M = N = 1$. Для этих случаев $\beta = 0$, алгоритм вырождается и сходится за число шагов, не превышающее n .

В случае, если имеем дело с невырожденной системой n линейных уравнений с n неизвестными $(a_i, x) + b_i$, $i = 1, \dots, n$, можем заменить ее решение нахождением минимума

$$f(x) = \max_i |(a_i, x) + b_i|.$$

Беря $f^* = 0$, $\beta_k = 0$ и применяя (1)–(3) — (5)–(7), получаем алгоритм, соответствующий известной конечной процедуре решения линейных алгебраических систем — методу ортогонализации градиентов.

Аналогичным приемом можно воспользоваться при решении систем нелинейных уравнений $f_i(x) = 0$, $i = 1, \dots, n$. Для $f(x) = \max_i |f_i(x)|$ можно показать, что если x^* (решение системы) — регулярная точка (т. е. $f_i(x)$ непрерывно-дифференцируемы в этой точке и якобиан системы $j(x^*)$ отличен от нуля), то для любого $\delta > 0$ найдется достаточно малая окрестность $S_\delta(x^*)$ такая, что константы M и N в (4) можно выбирать соответственно $M = 1 + \delta$, $N = 1 - \delta$, при этом

$$\beta = \frac{M - N}{M + N} = \delta.$$

Как показано в [13], если применять предельный вариант алгоритма с $\beta = 0$ и восстановлением после каждых n итераций (большой цикл), то при обычных предположениях гладкости и регулярности можно для решения систем нелинейных уравнений показать квадратичную скорость сходимости (относительно больших циклов).

Использование алгоритма в форме, указанной в теореме 1, затруднено в общем случае двумя обстоятельствами:

- а) неизвестны M и N ;
- б) неизвестно f^* .

Правда, в случае выпуклости $f(x)$ всегда можно брать $N = 1$, а M для большинства практических задач при не слишком плохом начальном приближении – в пределах от 3 до 10.

Допустим, что M и N выбраны правильно. Тогда к f^* можно последовательно приближаться, используя процесс, близкий к дихотомии, на основании следующих признаков.

Теорема 3. [8]. Пусть выпуклая функция $f(x)$ удовлетворяет условиям: существует $M \geq 1$ такое, что если $\varphi(\alpha) = f[(1-\alpha)x_1 + \alpha x_2]$, $0 \leq \alpha \leq 1$, строго убывает по α , то выполняется неравенство

$$f'_{x_1-x_2}(x_1) \leq M[f(x_1) - f(x_2)] \quad \text{и} \quad \lim_{\|x\| \rightarrow +\infty} f(x) = +\infty.$$

Тогда, если при использовании алгоритма (1)–(3) $\alpha_k = \frac{M+1}{M-1}$, $h_k = \frac{2M}{M+1} \cdot \frac{f(x_k) - \bar{f}}{\|\bar{g}_k\|}$, \bar{f} выбрано большим или равным \bar{f}^* , то последовательность $\{h_k\}_{k=1}^{\infty}$ является ограниченной и для произвольного $\varepsilon > 0$ найдется \bar{k} такое, что $f(x_{\bar{k}}) \leq \bar{f} + \varepsilon$ (счет прерывается, если на некотором шаге $f(x_k) \leq \bar{f}$); если \bar{f} выбрано меньшим f^* , то $\{h_k\}_{k=1}^{\infty}$ является неограниченной.

Естественно, что при неизвестном f^* трудоемкость алгоритма для получения определенной точности минимизации по функционалу значительно возрастает.

Опыт использования алгоритмов с растяжением пространства в направлении градиента показал возможность существенного ускорения субградиентных процессов с использованием линейных операторов, изменяющих метрику пространства. Трудности, связанные с выбором шаговых множителей при незнании f^* , стимулировали поиск новых методов негладкой оптимизации с переменной метрикой, в которых выбор шагового множителя связан с приближенным поиском минимума по направлению. В 1970 г. Н. З. Шором в докторской диссертации было предложено и вместе с Н. Г. Журбенко экспериментально исследовано семейство алгоритмов градиентного типа с растяжением пространства в направлении разности двух последовательных градиентов [9] (r -алгоритмы).

Разработанные на основе r -алгоритмов практические программы обеспечивают при минимизации n -мерных выпуклых функций сходи-

мость по функционалу со скоростью геометрической прогрессии, причем относительная точность по функционалу за n итераций улучшается, как правило, в три–десять раз.

Успехи в практическом применении r -алгоритмов стимулировали поиск методов с растяжением пространства, работающих столь же эффективно, но лучше обоснованных с теоретической точки зрения. Хотя на этом пути не были достигнуты окончательные результаты, но то, что уже получено, оказало большое влияние не только на развитие методов негладкой оптимизации, но и на более общие вопросы теории сложности алгоритмов. Ниже речь пойдет о методе эллипсоидов.

Метод эллипсоидов, его модификации и теоретические приложения

К методу эллипсоидов можно прийти, анализируя процессы субградиентного типа с растяжением пространства в направлении градиента с постоянным $\alpha_k = \alpha > 1$ и с последовательностью шаговых множителей $\{h_k\}_{k=0}^{\infty}$, меняющихся по формуле геометрической прогрессии: $h_{k+1} = h_k \varrho$.

Оказалось, что при реализации такого рода процессов для выпуклых функций нельзя, вообще говоря, рассчитывать, что расстояние от текущей точки до точки минимума в растянутом пространстве будет ограниченным, как в теореме 1, однако при определенном выборе α и $\varrho > 1$ можно добиться, что объем сферы в этом пространстве с радиусом, равным расстоянию от точки минимума до текущей точки, будет расти медленнее, чем геометрическая прогрессия со знаменателем α . Это обеспечивает сходимость «рекордов» по функционалу со скоростью геометрической прогрессии.

Итак, для решения задачи минимизации выпуклой функции $f(x)$, $x \in E_n$ был применен метод (1)–(3) с такими параметрами:

$$\alpha_k = \alpha = \sqrt{\frac{n+1}{n-1}}; \quad n \geq 2; \quad h_0 \geq \frac{\|x_0 - x^*\|}{n+1}; \quad h_{k+1} = h_k \varrho; \quad \varrho = \frac{n}{\sqrt{n^2-1}}.$$

Основной результат состоит в следующем.

Теорема 4. [11]. *Минимизирующая последовательность $\{x_k\}_{k=0}^{\infty}$, генерируемая указанным алгоритмом, удовлетворяет условию*

$$\|A_k(x_k - x^*)\| \leq h_k, \quad k = 0, 1, 2, \dots,$$

откуда следует, что x^* после k итераций локализовано в эллипсоиде S_k объема $\nu(S_k) = \nu_0 q_n^k$, где

$$q_n = \sqrt{\frac{n-1}{n+1}} \left(\frac{n}{\sqrt{n^2-1}} \right)^n \approx 1 - \frac{1}{2n}.$$

При подготовке рукописи статьи [11] к печати на одном из семинаров автору было указано, что аналогичный алгоритм в другой форме и из других соображений был разработан А. С. Немировским и Д. Б. Юдиным и опубликован под названием модифицированный метод центров тяжести (ММЦТ) в [15].

Д. Б. Юдин и А. С. Немировский, исследуя информационную сложность субградиентных алгоритмов выпуклого программирования, пришли к выводу, что с точки зрения информационного критерия (т. е. числа «порций» информации о субградиентах и значениях функции в определенных точках, необходимых для получения заданной относительной точности), известный алгоритм А. Ю. Левина [4] – метод центров тяжести (МЦТ) – является близким к оптимальному.

Каждая итерация МЦТ состоит в нахождении центра тяжести $x_{Ц}$ многогранной выпуклой области локализации минимума, вычислении субградиента $g(x_{Ц})$ в точке $x_{Ц}$ и отсечении с помощью гиперплоскости, ортогональной $g(x_{Ц})$. После отсечения объем области локализации уменьшается на величину, составляющую не менее $\left(\frac{n}{n+1}\right)^n$ часть первоначального объема. Таким образом, МЦТ гарантирует уменьшение объема области локализации со скоростью геометрической прогрессии со знаменателем

$$q_n = 1 - \left(\frac{n}{n+1} \right)^n < 1 - \frac{1}{\varepsilon}.$$

Однако сложность вычисления центра тяжести уже при $n \geq 3$ делает МЦТ практически непригодным.

В [15] было предложено в качестве области локализации минимума выбрать центрально-симметричное тело – эллипсоид, а после отсечения – «упрощать» полученное тело, описывая вокруг него эллипсоид минимального объема. Так как с помощью линейного преобразования пространства эллипсоид может быть превращен в сферу, то все операции можно проводить по отношению к сфере, переходя от одной системы координат к другой.

Итак, пусть имеется область локализации x^* – сфера с центром в точке x_0 радиуса R . Вычислив в точке x_0 субградиент $g(x_0)$ и построив соответствующую отсекающую гиперплоскость, получим полусферу.

Найдем параметры оптимального по объему эллипсоида, описанного вокруг полусферы. Из соображений симметрии следует, что его центр находится на луче, проведенном из точки x_0 в направлении антисубградиента, малая полуось по направлению совпадает с $g(x_0)$, а по длине равна $\frac{nR}{n+1}$, все остальные полуоси имеют в $\sqrt{\frac{n+1}{n-1}}$ большие длины. Прделав операцию растяжения пространства в направлении субградиента $g(x_0)$ с коэффициентом растяжения $\alpha = \sqrt{\frac{n+1}{n-1}}$, получим в растянутом пространстве в качестве области локализации минимума сферу радиуса $\frac{nR}{\sqrt{n^2-1}}$. Цикл вычислений замыкается.

Таким образом, метод эллипсоидов можно рассматривать как частный случай методов растяжения пространства в направлении градиента. Он требует на каждой итерации, кроме вычисления субградиентов, еще ряда векторно-матричных преобразований с числом вычислительных операций порядка $O(n^2)$, т. е. легко реализуем. Однако скорость уменьшения объема области локализации минимума в ММЦТ примерно в n раз медленнее, чем в МЦТ и во столько же раз медленнее скорость сходимости «рекордов» по функционалу.

Приблизительно в n раз ММЦТ уступает при проведении практических расчетов и r -алгоритму.

При использовании ММЦТ для решения задач выпуклого программирования легко видеть, что он мало чувствителен по отношению к увеличению числа ограничений. Отличие по сравнению с безусловной минимизацией состоит в том, что если в текущей точке не выполняется некоторое ограничение, то вместо субградиента от целевой функции берется субградиент от этого ограничения, во всем остальном алгоритм остается без изменения.

Несмотря на относительно медленную сходимость в теоретическом отношении, метод эллипсоидов оказался весьма полезным. Схема метода проста и прозрачна, скорость сходимости «рекордов» по функционалу геометрическая, причем знаменатель соответствующей геометрической прогрессии зависит лишь от размерности пространства (а не от конкретной минимизируемой функции). Л. Г. Хачиян первым открыл теоретические «способности» метода эллипсоидов, связанные с генерацией полиномиальных алгоритмов решения проблем оптимизации.

В течение многих лет в теории сложности алгоритмов оставалась нерешенной следующая проблема: задана произвольная система линейных неравенств с целыми коэффициентами, нужно построить алгоритм,

работающий полиномиальное время по отношению к объему информации о системе и дающий ответ на вопрос, имеет ли она решение.

Из существования такого алгоритма уже легко следует существование полиномиального алгоритма, дающего точное решение задачи линейного программирования с целыми (рациональными) коэффициентами. Попытки решения этой проблемы на базе процедур типа симплекс-метода не приводили к успеху, так как число итераций в таких методах может расти экспоненциально относительно размерности задачи.

Л. Г. Хачиян пошел по другому пути [5, 6]. Он получил простые выражения для \bar{R} – радиуса сферы с центром в начале координат, в которой гарантированно содержится решение системы, если оно существует, а также оценку снизу $\bar{\Delta}$ для минимального значения максимума невязки системы, если она несовместна. Затем свел решение к задаче минимизации функции максимума невязки (минимизация выпуклой кусочно-линейной функции) методом эллипсоидов, при этом в качестве начальной бралась сфера с центром в начале координат радиуса \bar{R} и итерации алгоритма проводились до тех пор, пока не получалось решение системы, либо не была достигнута определенная точность по функционалу порядка $\bar{\Delta}$. Число итераций метода эллипсоидов не превышает $O(n^2 L)$, где L – информационная сложность системы, откуда и следовал вывод о полиномиальности алгоритма.

Метод эллипсоидов позволил в скором времени получить ряд новых результатов. Так, в работе [16] был построен полиномиальный алгоритм для решения задач выпуклого квадратичного программирования с целыми коэффициентами, в [19] – полиномиальные алгоритмы для решения ряда задач анализа свойств многогранных выпуклых множеств P , задаваемых системой линейных неравенств с рациональными коэффициентами, в частности задачи выделения несущественных ограничений, определения минимальной порождающей системы вершин и образующих многогранного множества и др.

В работе [20] указан полученный с использованием этого метода полиномиальный алгоритм решения линейной проблемы дополненности для случая неотрицательно определенной матрицы M : для данной матрицы M размерности $n \times n$ и вектора $q \in E_n$, найти вектор $z \in E_n$ такой, что $Mz + q \geq 0$, $z \geq 0$; $(z, Mz + q) = 0$. Эту задачу можно рассматривать как частный случай задачи выпуклого квадратичного программирования, для которой имеется полиномиальный алгоритм.

Наиболее интересные результаты, связанные с использованием метода эллипсоидов для анализа комбинаторных проблем, получены в [21, 24]. В работе [21] рассматривается задача максимизации линейной

функции на n -мерном выпуклом теле K , задаваемом параметрами r, R (внутренний и внешний радиусы). С этой задачей связываются (слабая) оптимизационная задача: при заданном c и $\varepsilon > 0$ найти вектор $y \in E_n$ такой, что

- 1) $d(y, K) \leq \varepsilon$ ($d(y, K)$ – евклидово расстояние),
- 2) для любого $x \in K$ $(c, x) \leq (c, y) + \varepsilon$,

а также (слабая) задача отделения: для данного вектора $y \in E_n$ и $\varepsilon > 0$ нужно показать, что $d(y, K) \leq \varepsilon$, либо найти вектор $c \in E_n$, $\|c\| = 1$, такой, что для любого $x \in K$ $(c, x) \leq (c, y) + \varepsilon$.

Там же доказана следующая теорема.

Теорема. Пусть K – класс выпуклых тел, определяемых параметрами r и R , и задано $\varepsilon > 0$. Если существует полиномиальный алгоритм для слабой задачи отделения, то метод эллипсоидов решает слабую оптимизационную задачу не более чем за $O\left(n^2 \log \frac{R^2}{r\varepsilon} \|c\|\right)$ шагов. Требуемая точность решения задачи отделения имеет порядок $O\left(\frac{R}{n} \left(\frac{R^2}{r\varepsilon} \|c\|\right)^{-n^2}\right)$, и вычисления могут вестись с точностью, не превышающей $O\left(n^2 \log \frac{R^2}{r\varepsilon} \|c\| \log \frac{\sqrt{n}}{R^2}\right)$ двоичных знаков.

В этой же работе показана полиномиальная эквивалентность для заданного класса выпуклых тел задачи (слабой) оптимизации и (слабого) отделения.

Для рациональных выпуклых тел справедлива следующая теорема:

Теорема. Если (слабая) задача делимости полиномиально разрешима для класса рациональных выпуклых тел P , т. е. тел, все вершины которых описываются рациональными координатами, то тогда решение точных задач делимости и оптимизации может быть достигнуто за время, являющееся полиномом от n , $\log \|c\|$ и $\log T$, где T – максимальное по модулю значение числителей и знаменателей компонент вершин P .

Используя указанные результаты, Грётшель, Ловас и Схрейвер [21] показали, что большинство комбинаторных проблем, для которых были известны полиномиальные алгоритмы, могут быть решены за полиномиальное время с использованием метода эллипсоидов путем перехода к соответствующей задаче отделения. Среди них такие задачи, как минимальный разрез, оптимизация линейной функции на пересечении двух

матроидов, максимальное взвешенное паросочетание, задача о китайском почтальоне, минимизация субмодулярной функции на структуре подмножеств и др. С использованием теоремы об эквивалентности получены новые результаты о NP -полноте некоторых проблем.

Наиболее изящные математические результаты получены для оценки Ловаса [23, 24] числа внутренней устойчивости графа, которая одновременно является оценкой и для шенноновской информационной емкости графа.

Ортонормированным представлением графа G с n вершинами назовем систему (v_1, \dots, v_n) единичных векторов в пространстве E_n таких, что если i и j – несмежные вершины, то $(v_i, v_j) = 0$. Значением ортонормированного представления $U = \{u_1, \dots, u_n\}$ называется число $\varphi(U)$, определяемое по формуле

$$\varphi(U) = \min_{\|c\|=1} \max_i \frac{1}{(c, u_i)^2}, \quad c \in E_n.$$

Показано [23], что $\nu(G) = \min_U \varphi(U)$ является оценкой сверху для чисел $\alpha(G)$ – внутренней устойчивости и $\theta(G)$ – шенноновской емкости графа G и не превышает $\alpha^*(G)$ – дробного числа внутренней устойчивости, получающегося при решении соответствующей задачи линейного программирования, число ограничений которой может расти, вообще говоря, экспоненциально по отношению к числу вершин графа.

В [23] задача нахождения числа Ловаса $\nu(G)$ сведена к следующей задаче выпуклого программирования: рассматривается класс неотрицательно определенных симметричных матриц $B(G) = \{b_{ij}\}_{i,j=1}^n$

таких, что $\sum_{i=1}^n b_{ij} = 1$, $b_{ij} = 0$, если i и j – смежные вершины. Най-

ти $B^*(G)$, на которой достигается $\max \sum_{i,j=1}^n b_{ij} = \nu(G)$ при указанных ограничениях.

Используя метод эллипсоидов, Грётшель, Ловас и Схрейвер построили полиномиальный относительно n и $\log \frac{1}{\varepsilon}$ (ε – точность определения $\nu(G)$) алгоритм решения указанной задачи, при этом был применен интересный прием построения отсечений, основанный на критерии Сильвестра неотрицательной определенности матрицы B . На основе оценки Ловаса $\nu(G)$ в [24] предложен полиномиальный алгоритм нахождения $\alpha(G)$ и максимального независимого множества для так называемых совершенных графов, у которых числа $\alpha(G)$, $\theta(G)$, $\nu(G)$, $\alpha^*(G)$ совпадают, причем это свойство справедливо для их произвольных подграфов.

Другой способ нахождения $\nu(G)$ связан также с задачей выпуклого программирования.

Рассматривается класс симметричных матриц $A(G) = \{a_{ij}\}_{i,j=1}^n$, у которых $a_{ij} = 1$, $i = 1, \dots, n$; $a_{ij} = 0$, если i и j – несмежные вершины. Нужно найти

$$\nu(G) = \min_{\{a_{ij}\}} \max_i \lambda_i[A(G)], \quad (9)$$

где $\lambda_i[A]$ – собственные числа матрицы A .

Это типичная задача негладкой оптимизации относительно переменных $\{a_{ij}\}$, соответствующих ребрам графа G . Н. З. Шор совместно со студентом-дипломником Даниловичем И. С. провел численные эксперименты по решению задач типа (9) с использованием r -алгоритма. Рассматривались графы с числом вершин 5, 6, 24, 40, 62 и соответственно ребер – 5, 7, 66, 100, 180 и были получены достаточно удовлетворительные результаты. Так, для графа C_5 за 56 итераций было получено 6 верных знаков после запятой, для графа с 6 вершинами за 35 итераций – 7 верных знаков, с 24 вершинами за 150 итераций – 4 верных знака, с 40 вершинами за 100 итераций – 2 верных знака, с 62 вершинами за 60 итераций – решение с точностью 0,1.

Следует отметить, что, как указано в [24], метод эллипсоидов работает гораздо медленнее, так что получение $\nu(G)$ для графа с числом вершин 10–15 уже вызывает значительные вычислительные трудности.

Для улучшения сходимости этого метода было разработано большое число модификаций. Достаточно подробный обзор западных публикаций в этом направлении имеется в [20]. Здесь остановимся на отечественных работах.

Один из путей, позволяющих сблизить информационную эффективность алгоритмов ММЦТ и МЦТ, состоит в рассмотрении более сложных, чем полусфера, локализующих тел, которые могут получиться за счет дополнительной информации о функции и после двух и более последовательных отсечений. Чем больше таких отсечений рассматривается, тем более сложной, вообще говоря, становится задача аппроксимации локализующего тела описанным эллипсоидом, но тем больший выигрыш получается по критерию уменьшения объема этого эллипсоида.

Пусть фиксирована точка $x_0 \in E_n$, не являющаяся точкой минимума выпуклой функции $f(x)$. Основное неравенство, из которого следуют отсечения, имеет вид $f(x) \geq f(x_0) + (g(x_0), x - x_0)$. Полагая $x = \bar{x}$, полу-

чаем, что искомые точки минимума расположены в полупространстве

$$H(x_0, f(\bar{x})) = \left\{ x : (-g(x_0), x - x_0) \geq f(x_0) - f(\bar{x}) \right\}.$$

Соответствующая отсекающая гиперплоскость $P(x_0, \bar{f})$ определяется уравнением

$$\left(g(x_0), x - x_0 + \frac{f(x_0) - \bar{f}}{\|g(x_0)\|} \cdot \frac{g(x_0)}{\|g(x_0)\|} \right) = 0, \quad \bar{f} = f(\bar{x}).$$

Она отличается от используемых в ММЦТ гиперплоскостей при $f(x_0) > f(\bar{x})$ параллельным сдвигом в сторону антисубградиента на

$$\Delta = \frac{f(x_0) - \bar{f}}{\|g(x_0)\|}.$$

Если известно значение функции в точке минимума f^* , то наилучшее отсечение такого типа получается при $\bar{f} = f^*$. Если f^* неизвестно, то в качестве \bar{f} можно использовать гипотетическое значение f^* , при этом если нет уверенности, что $\bar{f} \leq f^*$, то соответствующее отсечение не является обоснованным и в алгоритме может использоваться только на пробных шагах.

Рассмотрим «стандартную» ситуацию: задан n -мерный шар $S_0 = \{x : \|x - x_0\| \leq r_0\}$, включающий точку минимума, и константа для сдвигов f^* . Предлагается следующая процедура построения локализующего тела T .

Вычисляем значение $f(x_0)$ и $g(x_0)$ и получаем $P(x_0, f^*)$ – первую отсекающую гиперплоскость. Вычисляя $f(\bar{x})$ и $g(\bar{x})$ в определенной точке отрезка $L_0 = \{x : x = x_0 - tg(x_0), \quad t > 0, \quad x \in S_0\}$, получаем $P(\bar{x}, f^*)$ – вторую отсекающую гиперплоскость. Выбор T осуществляется в зависимости от ситуации:

- 1) $(g(\bar{x}), g(x_0)) > 0$ – получено «попутное» отсечение; в качестве T берется тот из сегментов $H(x_0, f^*) \cap S_0$ и $H(\bar{x}, f^*) \cap S_0$, который более удален от точки x_0 ;
- 2) $(g(\bar{x}), g(x_0)) \leq 0$ – получено «встречное» отсечение.

Полагаем $W = H(x_0, f^*) \cap H(\bar{x}, f^*)$ и $V = P(x_0, f^*) \cap P(\bar{x}, f^*)$ и выбираем точку $v_0 \in V$, лежащую в плоскости, проведенной через x_0 коллинеарно $g(x_0)$ и $g(\bar{x})$.

Если $v_0 \notin \text{int } S_0$, то в качестве T берется «слой» (фигура, получаемая путем пересечения сферы с телом, отсекаемым двумя параллельными гиперплоскостями) минимальной толщины, содержащий $W \cap S_0$, если $v_0 \in \text{int } S_0$ – то так называемая s -пирамида $\pi = W \cap S_0$.

Таким образом, выбор точки $x \in L_0$ и $g(x)$ однозначно определяют локализирующее тело: сегмент, слой или s -пирамиду. Для перечисленных фигур в [1, 2, 14] получены формулы, позволяющие строить для них минимальные по объему описанные эллипсоиды $\Phi(T)$. За счет однократного (или в случае s -пирамиды – двукратного) применения оператора растяжения пространства локализирующий эллипсоид становится сферой и переходим к новому циклу вычислений. Поиск точки \bar{x} , необходимой для второго отсечения, предлагается вести из определенных соображений с шагом порядка r_0/\sqrt{n} в соответствии с эвристическими критериями [14].

Конечно, можно использовать и большее, чем два, число отсечений для построения T , но при этом алгоритм значительно усложняется.

Эффективность алгоритма зависит также от стратегии выбора f^s . Если f^* известно (например, при нахождении решения системы выпуклых неравенств), то проблем нет: $f^s = f^*$. Если выбрано $f^s \geq f^*$, то после определенного числа итераций можно получить x_k такое, что $f(x_k) \leq f^s$, в противном случае $\liminf_{k \rightarrow \infty} f(x_k) = f^s$, при этом последовательность эллипсоидов содержит на каждом шаге точку минимума. Если $f^s < f^*$, то происходят «сверхотсечения», и после конечного числа шагов получим пустое локализирующее тело. Таким образом, имеются признаки, позволяющие строить последовательность $\{f^s\}_l$ сходящуюся к f^* .

Алгоритм с двойным отсечением и адаптивным подбором f^s описан в [2, 14]. Численные эксперименты показали определенное преимущество этого метода по сравнению с ММЦТ, возрастающее при увеличении размерности. Так, при $n = 20$ указанный метод с неизвестным f^* для достижения сопоставимой точности требовал примерно в три раза меньше вычислений функций и градиентов, чем ММЦТ, при известном – объем вычислений уменьшался в четыре – пять раз по сравнению с вариантом алгоритма при известном f^* .

Дополнительный резерв ускорения методов эллипсоидов состоит в использовании констант роста функции M и N (см. неравенство (4)). Знание констант и f^* (N всегда можно выбрать равным единице) позволяет получить быстро сходящийся вариант этого метода [3], причем

показатель уменьшения объема на каждой итерации

$$q < \frac{M - N}{M + N} \leq \frac{M - 1}{M + 1}.$$

Различные упомянутые выше модификации метода эллипсоидов вкладываются в общую принципиальную схему.

Пусть x^* — точка минимума выпуклой функции $f(x)$ и задан исходный шар $S_0 = \{x : \|x - x_0\| \leq r_0\}$ такой, что $x^* \in S_0$. Перед началом $(k + 1)$ -го шага ($k = 0, 1, \dots$) имеем точку $x_k \in E_n$, неособенную $n \times n$ матрицу B_k и число $r_k > 0$, задающие в E_n эллипсоид

$$S_k = \{x : \|B_k^{-1} \times (x - x_k)\| \leq r_k\}$$

такой, что $x^* \in S_k$ ($B_0 = I$).

(k+1)-й шаг.

1. Уточнение локализации. В соответствии с некоторой процедурой (процедурой уточнения локализации) выбираем совокупность пар $\{x_k^l, f_{kl}^s\}$ (f_{kl}^s — значение для сдвигов). Каждая пара из этой совокупности определяет отсекающую гиперплоскость

$$P(x_k^l, f_{kl}^s) = \{x : (g(x_k^l), x - x_k^l) = f_{kl}^s - f(x_k^l)\},$$

задающую полупространство вида

$$H(x_k^l, f_{kl}^s) = \{x : (g(x_k^l), x - x_k^l) \leq f_{kl}^s - f(x_k^l)\}$$

такое, что $x^* \in H(x_k^l, f_{kl}^s)$. Тело $Z_k = S_k \cap \left(\bigcap_{l=0}^{l_k} H(x_k^l, f_{kl}^s) \right)$ содержит x^* .

Основное свойство, которому должна удовлетворять указанная процедура, состоит в существовании эллипсоида $S(Z_k)$, $Z_k \subseteq S(Z_k)$, такого, что существует $Q < 1$, при котором выполняется неравенство

$$q(Z_k) = \frac{v(S(Z_k))}{v(S_k)} \leq Q$$

(здесь $v(D)$ — n -мерный объем тела D).

2. Пересчет параметров. Полагаем $S_{k+1} = S(Z_k)$, т. е. вычисляем точку $x_{k+1} \in E_n$, неособенную матрицу B_{k+1} и число r_{k+1} такие, что

$$S_{k+1} = \{x : \|B_{k+1}^{-1}(x - x_{k+1})\| \leq r_{k+1}\} = S(Z_k).$$

Рассмотрим последовательность рекордов $\{f_k^r\}_{k=0}^\infty$, порождаемых в процессе работы алгоритма:

$$f_k^r = \min \left\{ f_{k-1}^r \min \left\{ f(x_k^l), \quad l = 0, \dots, l_k \right\} \right\}.$$

Обобщая результаты Л. Г. Хачияна [6] и Б. Корте и Р. Шрадера [22], можно доказать следующую теорему.

Теорема 5. Пусть $x^* \in S_0$; $\|g(x)\| \leq L$ при $x \in S_0$,

$$f_{kl}^s = \min \left\{ f_{k-1}^r \min \left\{ f(x_k^j), \quad j = 0, 1, \dots, l \right\} \right\}, \quad k = 0, 1, \dots$$

Тогда последовательности матриц $\{B_k\}_{k=0}^\infty$, чисел $\{r_k\}_{k=0}^\infty$ и $\{f_k^r\}_{k=0}^\infty$, генерируемых алгоритмом эллипсоидов, удовлетворяют неравенствам

$$f_k^r - f(x^*) \leq 2Lr_{k+1} \sqrt{\lambda(H_{k+1})}, \quad (10)$$

где $\lambda(H_{k+1})$ — минимальное собственное значение матрицы $H_{k+1} = B_{k+1}B_{k+1}^*$.

Следствие. В условиях теоремы справедливы неравенства:

$$f_k^r - f(x^*) \leq 2Lr_0 \left(\prod_{l=0}^k q(Z_l) \right)^{1/n} \leq 2Lr_0 Q^{k/n}, \quad k = 0, 1, \dots \quad (11)$$

Неравенство (10) дает значительно более точную оценку величины $f_k^r - f(x^*)$, чем (11), и позволяет раньше идентифицировать момент достижения удовлетворительной точности решения. Однако для этого необходимо иметь величины собственных значений $\lambda(H_{k+1})$.

При определенном способе вычислений эти величины оказываются доступными. Пусть $H_k = B_k B_k^*$. Симметричная положительно определенная матрица H_k может быть представлена совокупностью своих собственных векторов и значений. Оказалось, что существует простой с вычислительной точки зрения переход от такого представления матрицы H_k к представлению матрицы H_{k+1} (подробно см. [17]).

Конкретные алгоритмы — представители класса алгоритмов эллипсоидов — получаются из приведенной принципиальной схемы, если на ее первой стадии фиксировать конкретную процедуру уточнения локализации. Алгоритмы эллипсоидов, использующие в качестве такой процедуры описанную процедуру построения локализуемого тела, предложены в [1]–[3], [14].

Совершенствование алгоритмов эллипсоидов продолжается быстрыми темпами, идет накопление опыта их практического использования. Но и в настоящее время они могут оказаться полезными при решении задач выпуклого программирования небольшой размерности (до 20–30 переменных) с большим числом ограничений. Кроме того, они могут применяться в сочетании с r -алгоритмами на проверочном этапе расчетов для подтверждения (или отрицания) того, что g -алгоритм обеспечил определенную точность по функционалу, и в случае отрицания — для доводки решения до гарантированной точности.

Заключение

После взрыва энтузиазма, вызванного результатом Л. Г. Хачияна [5], методы эллипсоидов оказались в центре внимания многих оптимизаторов. Через некоторое время стали признавать лишь их теоретическую ценность, ставя под сомнение практическую эффективность. И во многом это справедливо, если говорить о методе ММЦТ и его незначительных модификациях.

В данной работе авторы пытались показать, что методы эллипсоидов являются лишь весьма частным случаем семейства алгоритмов градиентного типа, использующих операцию растяжения пространства. Другие представители этого семейства, как например, r -алгоритмы, являются эффективным практическим средством решения многих сложных задач математического программирования, сводящихся к недифференцируемой оптимизации. Теория всего класса алгоритмов с растяжением пространства еще далека от совершенства. Нам кажется достаточно реалистической целью построение такого алгоритма, который по своей практической эффективности не уступал бы r -алгоритму и был столь же хорошо обоснован, как метод эллипсоидов.

Литература

1. ГЕРШОВИЧ В. И. Об одном методе отсечений, использующем линейное преобразование пространства // Теория оптимальных решений. — 1979. — С. 15–23.
2. ГЕРШОВИЧ В. И. Об одном методе минимизации, использующем линейное преобразование пространства // Теория оптимальных решений. — 1980. — С. 38–45.

3. ГЕРШОВИЧ В. И. Об одном алгоритме эллипсоидов. — О некоторых алгоритмах негладкой оптимизации и дискретного программирования. — К.: ИК АН УССР, 1981. — С. 8–13.
4. ЛЕВИН А. Ю. Об одном алгоритме минимизации выпуклых функций // Докл. АН СССР. — 1965. — **160**, № 6. — С. 1244–1247.
5. ХАЧИЯН Л. Г. Полиномиальный алгоритм в линейном программировании // Докл. АН СССР. — 1979. — **244**, № 5. — С. 1093–1096.
6. ХАЧИЯН Л. Г. Полиномиальный алгоритм в линейном программировании // Вычисл. математика и мат. физика. — 1979. — **20**, № 2. — С. 51–68.
7. ШОР Н. З. Использование операций растяжения пространства в задачах минимизации выпуклых функций // Кибернетика. — 1970. — № 1. — С. 2–12.
8. ШОР Н. З. О скорости сходимости метода обобщенного градиентного спуска с растяжением пространства // Там же. — 1970. — № 2. — С. 80–85.
9. ШОР Н. З., ЖУРБЕНКО Н. Г. Метод минимизации, использующий операцию растяжения пространства в направлении разности двух последовательных градиентов // Там же. — 1971. — № 3. — С. 51–59.
10. ШОР Н. З. О методе минимизации почти-дифференцируемых функций // Там же. — 1972. — № 4. — С. 65–70.
11. ШОР Н. З. Метод отсечения с растяжением пространства для решения задач выпуклого программирования // Там же. — 1977. — № 1. — С. 94–95.
12. ШОР Н. З. Новые направления в развитии методов негладкой оптимизации // Там же. — 1977. — № 6. — С. 87–91.
13. ШОР Н. З. Методы минимизации недифференцируемых функций и их приложения — К.: Наук. думка, 1979. — 199 с.
14. ШОР Н. З., ГЕРШОВИЧ В. И. Об одном семействе алгоритмов для решения задач выпуклого программирования // Кибернетика. — 1979. — № 4. — С. 62–67.

15. Юдин Д. Б., НЕМИРОВСКИЙ А. С. Информационная сложность и эффективные методы решения выпуклых экстремальных задач // Экономика и мат. методы. – 1976. – **12**, вып. 2, – С. 357–369.
16. КОЗЛОВ М. К., ТАРАСОВ С. П., ХАЧИЯН Л. Г. Полиномиальная разрешимость выпуклого квадратичного программирования // Докл. АН СССР. – 1979. – **248**, № 5. – С. 1049–1051.
17. ШОР Н. З., ГЕРШОВИЧ В. И. Об одной модификации алгоритмов градиентного типа с растяжением пространства для решения задач большой размерности // Кибернетика. – 1981. – № 5. – С. 67–70.
18. WOLFE P. A bibliography for the ellipsoid algorithm (Rep.) // IBM Res. Center. – RC 8237. – New York, 1980. – 6 p.
19. BACHEM A., GRÖTSCHEL M. Characterizations of adjacency of faces of polyhedra // Mat. Progr. Study. – 1981. – N 14. – P. 1–22.
20. SHRADER R. Ellipsoid methods (Rep.) // Institut für öconometrie und operations research. – N 81174-OR. – Bonn, 1980. – 62 p.
21. GRÖTSCHEL M., LOVASZ L., SCHRIJVER A. The ellipsoid method and its consequences in combinatorial optimization // Combinatorica. – 1981. – **1**, N 2. – P. 169–197.
22. KORTE B., SHRADER R. A note on convergence proofs for Shor–Khachian methods // Lect. Notes control and Inform. Sci. – 1980. – N 30. – P. 51–57.
23. LOVASZ L. On the Shannon capacity of a graph // IEEE Trans. Inform. Theory. – **25**, N 1. – P. 1–7.
24. GRÖTSCHEL M., LOVASZ L., SHRIJVER A. Polynomial algorithms for perfect graphs: (Rep.) // Institut für öconometrie und operations research. – WP 81176-OR. – Bonn, 1981. – 40 p.

Использование операции растяжения пространства в задачах минимизации выпуклых функций

Н. З. Шор

Кибернетика. — 1970. — № 1. — С. 6–12.

Предлагается новый метод минимизации выпуклых функций, основанный на сочетании обобщенного градиентного спуска и операции растяжения пространства. Доказываются теоремы о сходимости метода.

Введение

Рассмотрим задачу минимизаций выпуклой функции $f(x)$, определенной на всем n -мерном евклидовом пространстве E_n . При этом от функции $f(x)$ не будем требовать непрерывности производных по направлению. В этих условиях, как известно, обычные градиентные методы или методы покоординатного спуска не работают, и приходится использовать метод обобщенного градиентного спуска [2], который сходится при определенной регулировке шага для произвольной выпуклой функции $f(x)$, имеющей минимум. Однако, если использовать универсальный способ регулировки шага, т. е. требовать, чтобы $\sum_{k=1}^{\infty} h_k = \infty$ и $h_k \rightarrow 0$, где h_k — длина приращения вектора-аргумента на k -ом шаге, то скорость сходимости получается весьма медленной. В работе [3] показано, что при определенных ограничениях, накладываемых на $f(x)$, при помощи специального способа регулировки шага можно добиться сходимости обобщенного, градиентного спуска со скоростью геометрической прогрессии. Точнее, если выпуклая функция $f(x)$ достигает своего минимума в точке x^* и мы выбрали начальное приближение x_0 и начальный шаг h_0 таким образом, что существует $\sigma \geq \sqrt{2}$ такое, что:

$$1) \quad \|x_0 - x^*\| \leq h_0 \sigma;$$

2) для произвольной пары x, y такой, что

$$f(x) = f(y) \neq f(x^*); \quad \|x - x^*\| \leq h_0 \sigma; \quad \|y - y^*\| \leq h_0 \sigma,$$

выполняется неравенство $\frac{\|x - x^*\|}{\|y - y^*\|} \leq \sigma$, то, выбирая шаг в методе обобщенного градиентного спуска по формуле $h_{k+1} = h_k \frac{\sqrt{\sigma^2 - 1}}{\sigma}$, получим сходящуюся последовательность $\{x_k\}$, $k = 0, 1, 2, \dots$; при этом $\|x_k - x^*\| \leq h_k \sigma$. На практике σ может оказаться довольно большим числом, при этом знаменатель геометрической прогрессии оказывается близким к единице и скорость сходимости медленной. Более того, для некоторых $f(x)$ вообще невозможно подобрать $\sigma < \infty$, удовлетворяющее выше приведенным условиям. Такие функции будем называть существенно овражными.

Следует отметить, что существенно овражные функции встречаются на практике весьма часто, особенно в нелинейных задачах минимаксного типа.

Например, функция

$$f(x_1, x_2) = \max [x_1^2 + x_2^2, (x_1 - 1)^2 + x_2^2]$$

является существенно овражной. Из выше сказанного видно, какое большое значение имеет ускорение сходимости метода обобщенного градиентного спуска.

Ниже будет изложен метод минимизации выпуклых функций, представляющий собой комбинацию метода обобщенного градиентного спуска и преобразования пространства аргументов. Как показали теоретические исследования и численные эксперименты, этот метод, лишь незначительно отличаясь по сложности от обобщенного градиентного спуска, обладает большими преимуществами по сравнению с ним в скорости сходимости.

2. Вычисление обобщенного градиента функции при линейном преобразовании пространства аргументов

Обобщенным градиентом выпуклой функции $f(x)$ в точке x_0 называется вектор $\hat{g}_f(x_0)$, удовлетворяющий следующему неравенству для произвольного $x \in E_n$:

$$f(x) - f(x_0) \geq (\hat{g}_f(x_0), x - x_0). \quad (1)$$

Пусть произведено неособенное линейное преобразование пространства E_n , определяемое оператором A :

$$y = Ax; \quad x = A^{-1}y.$$

Рассмотрим функцию

$$\varphi(y) = f(A^{-1}y).$$

Функция $\varphi(y)$ является выпуклой. В самом деле, для произвольных $\alpha, \beta \geq 0, \quad \alpha + \beta = 1$

$$\begin{aligned} \varphi(\alpha y_1 + \beta y_2) &= f(\alpha A^{-1}y_1 + \beta A^{-1}y_2) \leq \\ &\leq \alpha f(A^{-1}y_1) + \beta f(A^{-1}y_2) = \alpha \varphi(y_1) + \beta \varphi(y_2). \end{aligned}$$

Вычислим обобщенный градиент функции $\varphi(y)$ в точке $y_0 = Ax_0$.

$$\begin{aligned} \varphi(y) - \varphi(y_0) &= f(A^{-1}y) - f(A^{-1}y_0) \geq \\ &\geq (\hat{g}_f(x_0), A^{-1}(y - y_0)) = ((A^*)^{-1}\hat{g}_f(x_0), y - y_0). \end{aligned}$$

A^* — оператор, сопряженный A .

Таким образом,

$$\varphi(y) - \varphi(y_0) \geq ((A^*)^{-1}\hat{g}_f(x_0), y - y_0). \quad (2)$$

Сравнивая формулы (2) и (1), получаем, что $(A^*)^{-1}\hat{g}_f(x_0)$ является обобщенным градиентом функции $\varphi(y)$ в точке $y_0 = Ax_0$.

$$\hat{g}_\varphi(y_0) = (A^*)^{-1}\hat{g}_f(x_0); \quad x_0 = A^{-1}y_0. \quad (3)$$

Так как обобщенный градиент в заданной точке определяется неоднозначно, то равенство (3) следует понимать следующим образом: между множествами обобщенных градиентов $\{\hat{g}_\varphi(y_0)\}$ и $\{\hat{g}_f(x_0)\}$ можно установить такое взаимно однозначное соответствие, что будут выполняться равенства (3). Формулы (3) позволяют вычислять обобщенный градиент функции $\varphi(x)$.

3. Операторы растяжения пространства

Пусть задан вектор $\xi \in E_n, \|\xi\| = 1$, и число $\alpha > 0$. Каждый вектор $x \in E_n$ однозначно представим в следующем виде:

$$x = \gamma_\xi(x) \cdot \xi + d_\xi(x), \quad (\xi, d_\xi(x)) = 0. \quad (4)$$

Определение 1. Оператором растяжения в направлении ξ с коэффициентом α назовем оператор $R_\alpha(\xi)$, действующий следующим образом на вектор x , представленный в форме (4):

$$R_\alpha(\xi)x = \alpha\gamma_\xi(x) \cdot \xi + d_\xi(x). \quad (5)$$

Из этого определения сразу следует:

- 1) $R_{\alpha\beta}(\xi) = R_\alpha(\xi) \cdot R_\beta(\xi)$;
- 2) $R_\alpha(\xi) \cdot R_{1/\alpha}(\xi) = R_1(\xi) = E$.

Замечание. Оператор растяжения с $\alpha < 1$ мы будем иногда называть оператором сжатия.

Пусть координаты вектора ξ в некоторой ортонормированной системе координат равны $\xi^{(1)}, \dots, \xi^{(n)}$. Тогда в этой системе координат преобразованию $R_\alpha(\xi)$ соответствует матрица $R_\alpha(\xi)$ с элементами $\{r_{ij}\}$, вычисляемыми по следующим формулам:

$$r_{ij} = \begin{cases} (\alpha - 1) \xi^{(i)} \xi^{(j)} & \text{для } i \neq j; \\ (\alpha - 1) (\xi^{(i)})^2 + 1 & \text{для } i = j. \end{cases} \quad (6)$$

Заметим, что умножение матрицы $R_\alpha(\xi)$ на вектор требует порядка cn операций умножения, сложения и вычитания, а умножение $R_\alpha(\xi)$ на матрицу требует порядка cn^2 арифметических операций.

Определение 2. Коэффициентом объемного расширения подпространства $R_m \subseteq E_n$ под воздействием оператора A назовем \sqrt{G} , где G – определитель Грама, построенный на векторах $\sigma_1, \dots, \sigma_m$, $\sigma_i = Ae_i$, $i = 1, \dots, m$, $\{e_i\}$ – ортонормированный базис подпространства R_m . Этот коэффициент будем обозначать $\vartheta_{R_m}(A)$.

Из свойств определителя Грама следуют утверждения:

$$1) \quad \vartheta_{R_m}(A_1 A_2) = \vartheta_{A_2 R_m}(A_1) \vartheta_{R_m}(A_2); \quad (7)$$

$$2) \quad \vartheta_{E_n}(R_\alpha(\xi)) = \alpha; \quad (8)$$

$$3) \quad \vartheta_{R_m}(R_\alpha(\xi)) = \sqrt{1 + (\alpha^2 - 1) \|\psi\|^2}, \quad (9)$$

где ψ – проекция вектора ξ на подпространство R_m ;

- 4) пусть R_{n-m} – ортогональное дополнение пространства R_m в E_n , тогда

$$\vartheta_{R_m}(R_\alpha(\xi)) \cdot \vartheta_{R_{n-m}}(R_\alpha(\xi)) \geq \alpha; \quad (10)$$

из (9) вытекает важное следствие –

5) если $\alpha \geq 1$, то $\vartheta_{R_m}(R_\alpha(\xi)) \geq 1$ для произвольного R_m .

4. Обобщенный градиентный спуск с растяжением пространства

Ниже мы рассмотрим класс алгоритмов минимизации выпуклых функций, на каждом шаге которых движение в направлении обобщенного антиградиента будет сочетаться с операцией растяжения пространства аргументов в направлении обобщенного градиента. Алгоритмы этого класса мы будем называть алгоритмами обобщенного градиентного спуска с растяжением пространства (сокращенно ОГСРП-алгоритмы).

I. Для заданной выпуклой функции $f(x)$ имеется алгоритм вычисления $\widehat{g}_f(x)$ в произвольной точке $x \in E_n$; заданы операторы вычисления последовательностей положительных чисел $\{h_k\}$ и $\{\alpha_k\}$, $k = 1, 2, \dots$

II. 0 – й шаг алгоритма.

Выбираем начальное приближение $x = x_0$ и матрицу $B_0 = A_0^{-1} = E$.

III. $(k+1)$ – й шаг алгоритма. $k = 0, 1, \dots$

Вычисляем:

$$\begin{aligned} 1) & \widehat{g}_f(x_k); \\ 2) & \widehat{g}_{\varphi_k}(y_k) = B_k^* \widehat{g}_f(x_k). \end{aligned} \quad (11)$$

Примечание. $(\varphi_k(y) = f(B_k y); \quad y_k = A_k x_k).$

$$3) \quad \xi_{k+1} = \frac{\widehat{g}_{\varphi_k}(y_k)}{\|\widehat{g}_{\varphi_k}(y_k)\|}; \quad (12)$$

$$4) \quad h_{k+1};$$

$$5) \quad \alpha_{k+1};$$

$$6) \quad x_{k+1} = x_k - B_k h_{k+1} \xi_{k+1}; \quad (13)$$

$$7) \quad R_{\alpha_{k+1}}^{-1}(\xi_{k+1}) = R_{1/\alpha_{k+1}}(\xi_{k+1});$$

$$8) \quad B_{k+1} = A_{k+1}^{-1} = B_k \cdot R_{1/\alpha_{k+1}}(\xi_{k+1}). \quad (14)$$

Различные варианты алгоритмов ОГСРП будут отличаться друг от друга способами вычисления последовательностей $\{\alpha_k\}$ и $\{h_k\}$.

Теорема 1. Пусть в процессе выполнения ОГСРП выполняются следующие условия для $k = 0, 1, 2, \dots$:

$$\inf_{x^* \in M^*} \|A_k(x_k - x^*)\| \leq r; \quad (15)$$

$$\alpha_k \geq 1 + \varepsilon, \quad (16)$$

где r и ε некоторые положительные константы, M^* — множества точек минимума $f(x)$, которое предполагается ограниченным. Тогда

$$\lim_{k \rightarrow \infty} f(x_k) = f(x^*), \quad x^* \in M^*.$$

Доказательство. Для доказательства введем некоторые определения. Пусть задано множество операторов $\mathcal{M} = \{A_\gamma\}$. Подпространство $R_m \subseteq E_n$ назовем особым по отношению к множеству \mathcal{M} , если $\sup \vartheta_{R_m}(A_\gamma) < \infty$.

В дальнейшем под $\vartheta_x(A)$ мы будем понимать $\vartheta_{R_1}(A)$, где R_1 порождено вектором $x \neq 0$. Докажем лемму.

Лемма. Пусть заданы 2 линейно независимых вектора x, y , оператор A и коэффициент растяжения $\vartheta_x(A) = \frac{\|A_x\|}{\|x\|}$ и $\vartheta_y(A) = \frac{\|A_y\|}{\|y\|}$. Тогда справедливо следующее неравенство для произвольного вектора $z = ax + by$, $z \neq 0$:

$$\vartheta_z(A) \leq \frac{\sqrt{2} \max(\vartheta_x(A), \vartheta_y(A))}{\sqrt{1 - |\cos(\widehat{x, y})|}} \quad (17)$$

Доказательство.

$$\begin{aligned} \vartheta_z(A) &= \frac{\|A(ax + by)\|}{\|ax + by\|} \leq \frac{|a| \cdot \|Ax\| + |b| \cdot \|Ay\|}{\sqrt{a^2\|x\|^2 + 2ab(x, y) + b^2\|y\|^2}} \leq \\ &\leq \frac{[\|x\| \cdot |a| + |b| \cdot \|y\|] \cdot \max(\vartheta_x(A), \vartheta_y(A))}{\sqrt{a^2\|x\|^2 + 2ab\|x\| \cdot \|y\| \cos(\widehat{x, y}) + b^2\|y\|^2}}. \end{aligned} \quad (18)$$

Найдем минимум выражения $a^2\|x\|^2 + 2ab\|x\| \cdot \|y\| \cos \varphi + b^2\|y\|^2$ по

$a, b, \|x\|, \|y\|$ при ограничении $|a| \cdot \|x\| + |b| \cdot \|y\| = c$, где $c > 0$. Легко видеть, что этот минимум достигается при $|a| \cdot \|x\| = |b| \cdot \|y\| = \frac{c}{2}$ и при $ab \cos \varphi \leq 0$ равен $\frac{c^2}{2} (1 - |\cos \varphi|)$. Отсюда и из (18) получаем:

$$\vartheta_z(A) \leq \frac{c \max(\vartheta_x(A), \vartheta_y(A))}{\sqrt{\frac{c^2}{2} (1 - |\cos(\widehat{x, y})|)}} = \frac{\sqrt{2} \max(\vartheta_x(A), \vartheta_y(A))}{\sqrt{1 - |\cos(\widehat{x, y})|}}.$$

Доказательство леммы завершено.

Следствие. Совокупность особых векторов по отношению к множеству операторов \mathcal{M} образует подпространство.

Доказательство. Выберем максимальное число линейно независимых особых векторов. Тогда из предыдущей леммы следует, что и произвольная линейная комбинация этих вектор будет особым вектором, а это значит, что совокупность особых векторов образует подпространство.

Теперь перейдем к доказательству теоремы. Так как A_k является произведением операторов растяжения (см. (14)), то из (16), (7), (9) следует, что последовательность $\vartheta_{R_m}(A)$ для произвольного подпространства $R_m \subseteq E_n$ будет неубывающей по k , причем для всех $k = 0, 1, 2, \dots$ $\vartheta_{R_m}(A_k) \geq 1$.

Множество минимумов M^* функции $f(x)$ является замкнутым, поэтому в формуле (15) инфимум достигается в некоторой точке, которую мы будем обозначать $x^*(x_k)$.

Так как $\|A_k(x_k - x^*(x_k))\| = \vartheta_{x_k - x^*(x_k)}(A_k) \times T\|x_k - x^*(x_k)\|$ и $\vartheta_{x_k - x^*(x_k)}(A_k) \geq 1$, то из (15) следует:

$$\|x_k - x^*(x_k)\| \leq r. \quad (19)$$

Пусть \overline{R}_m — пространство особых векторов по отношению к множеству $\{A_k\}$, $k = 0, 1, \dots$; \overline{R}_{n-m} — ортогональное дополнение \overline{R}_m в E_n .

Из (19) и ограниченности M^* следует, что последовательность $\{x_k\}$ имеет предельные точки. Пусть $\overline{x} \in M^*$ — некоторая предельная точка, $\{x_{k_i}\}$ — некоторая последовательность, сходящаяся к этой точке. Покажем, что $\overline{x} - x^*(\overline{x}) \in \overline{R}_m$.

Рассмотрим разложения $\bar{x} - x^*(\bar{x}) = \bar{y} + \bar{z}$; $x_{k_i} - x^*(x_k) = y_{k_i} + z_{k_i}$; где \bar{y} , $y_{k_i} \in \bar{R}_m$; \bar{z} , $z_{k_i} \in \bar{R}_{n-m}$.

Легко показать, что $\lim_{k \rightarrow \infty} \min_{z \in \bar{R}_{n-m}} \vartheta_z(A_k) = \infty$. В самом деле, рассмотрим пересечение множеств: $\mathcal{M}_d = \bigcap_{k=0}^{\infty} \left\{ z : z \in \bar{R}_{n-m}, \vartheta_z(A_k) \leq d \right\}$, где $d > 0$. Если это пересечение не пусто, то найдется $z \neq 0$, $z \in \bar{R}_{n-m}$, что $\vartheta_z(A_k) \leq d$ для $k = 0, 1, 2, \dots$. Но тогда $z \in \bar{R}_m$. Получили противоречие. Следовательно,

$$\lim_{k \rightarrow \infty} \min_{z \in \bar{R}_{n-m}} \vartheta_z(A_k) = \infty. \quad (20)$$

Если $\bar{z} \neq 0$, то, начиная с некоторого i , будет выполняться неравенство $\|z_{k_i}\| \geq \delta > 0$, так как $\lim_{i \rightarrow \infty} z_{k_i} = \bar{z}$. Но тогда

$$\left\| A_{k_i} (x_{k_i} - x^*(x_{k_i})) \right\| \geq \vartheta_{z_{k_i}}(A_{k_i}) \times \delta - \|A_{k_i} y_{k_i}\|.$$

Так как $\left\{ \|A_{k_i} y_{k_i}\| \right\}$ ограничена, то из (20) получаем

$$\left\| A_{k_i} (x_{k_i} - x^*(x_{k_i})) \right\| \rightarrow \infty \quad \text{при } t \rightarrow \infty,$$

но это противоречит (15). Таким образом, $\bar{z} = 0$, т. е.

$$\bar{x} - x^*(\bar{x}) \in \bar{R}_m.$$

По определению \bar{R}_m существует конечный предел последовательности

$$\vartheta_{\bar{x} - x^*(\bar{x})}(A_k), \quad k = 0, 1, \dots$$

Но тогда

$$\lim_{k \rightarrow \infty} \frac{\vartheta_{\bar{x} - x^*(\bar{x})}(A_{k+1})}{\vartheta_{\bar{x} - x^*(\bar{x})}(A_k)} = 1. \quad (21)$$

В силу (9), (16), (21) проекция вектора ξ_k на направление $A_k(\bar{x} - x^*(\bar{x}))$ должна стремиться к 0, откуда обобщенная производная от функции $f(x)$ по направлению $\bar{x} - x^*(\bar{x})$ в точке x_k должна стремиться к 0 при $k \rightarrow \infty$.

Замечание 1. Под обобщенной производной от функции $f(x)$ по направлению η в заданной точке x_0 понимается проекция обобщенного вектора градиента в этой точке на указанное направление. Мы будем для обобщенной производной по направлению применять обозначение

$f'_\eta(x_0)$. Для производной по направлению от выпуклой функции $f(x)$ справедливо неравенство

$$|f'_{x-x_0}(x)| \cdot \|x - x_0\| \geq f(x) - f(x_0). \quad (22)$$

Продолжим доказательство теоремы.

Из неравенства (22) получаем:

$$|f'_{\bar{x}-x^*(\bar{x})}(x_{k_i})| \cdot \|\bar{x} - x^*(\bar{x})\| \geq f(x_{k_i}) - f(x_{k_i} - \bar{x} + x^*(\bar{x})). \quad (23)$$

Перейдем к пределу при $i \rightarrow \infty$.

Получим из (23): $0 \geq f(\bar{x}) - f(x^*(\bar{x}))$. Значит, $f(\bar{x}) = f(x^*(\bar{x}))$, т. е. $\bar{x} \in M^*$.

Таким образом, все предельные точки последовательности $\{x_k\}$ принадлежат области минимумов функции $f(x)$, что и доказывает теорему 1.

В теореме 1 мы постулировали определенные свойства последовательности $\{A_k x_k\}$. В теореме 2 при некоторых предположениях, накладываемых на $f(x)$, мы сформулируем способ выбора последовательностей $\{h_k\}$ и $\{\alpha_k\}$, при котором требуемые в теореме 1 свойства $\{A_k x_k\}$ выполняются.

Теорема 2. Пусть множество M^* точек минимума выпуклой функции $f(x)$ ограничено и $f(x)$ принимает на M^* значение m^* .

Введем обозначения: $\varrho(x) = \min_{x^* \in M^*} \|x - x^*\|$; $x^*(x)$ — точка, для которой $\|x - x^*(x)\| = \varrho(x)$. Пусть для всех x , удовлетворяющих условию $\varrho(x) \leq c$, $x \in M^*$ и $x^* \in M^*$ выполняются следующие неравенства:

$$N[f(x) - m^*] \leq |f'_{x-x^*}(x)| \cdot \|x - x^{**}\| \leq M[f(x) - m^*], \quad (24)$$

где x^{**} — ближайшая к x точка минимума, лежащая на луче

$$y = x + t(x - x^*), \quad t \geq 0.$$

Тогда, если при применении ОГСРП

$$1) \quad \varrho(x_0) \leq c; \quad (25)$$

$$2) \quad h_{k+1} = \frac{2MN}{M+N} \cdot \frac{[f(x_k) - m^*]}{\|\bar{g}_{\varphi_k}(y_k)\|}; \quad (26)$$

$$3) \quad 1 + \varepsilon \leq \alpha_{k+1} \leq \frac{M + N}{M - N}, \quad (27)$$

то будут выполнены условия теоремы 1. (Здесь M , N , c , ε – положительные константы.)

Доказательство. Для доказательства достаточно показать, что последовательность $\min_{x^* \in M^*} \|A_k(x_k - x^*)\|$ является невозрастающей.

Используя (14), (13), получаем :

$$\begin{aligned} & \min_{x^* \in M^*} \|A_k(x_k - x^*)\|^2 = \\ &= \min_{x^* \in M^*} \|R_{\alpha_k}(\xi_k) \cdot A_{k-1}(x_k - x^*)\|^2 = \\ &= \min_{x^* \in M^*} \|R_{\alpha_k}(\xi_k) \cdot A_{k-1}(x_{k-1} - x^*) - R_{\alpha_k}(\xi_k)h_k\xi_k\|^2 \leq \\ &\leq \|R_{\alpha_k}(\xi_k) \cdot A_{k-1}(x_{k-1} - x_{k-1}^*) - R_{\alpha_k}(\xi_k)h_k\xi_k\|^2, \end{aligned} \quad (28)$$

где x_{k-1}^* – значение x^* , при котором достигается минимум в выражении:

$$\min_{x^* \in M^*} \|A_{k-1}(x_{k-1} - x^*)\|. \quad (29)$$

Введем обозначение:

$$A_{k-1}(x_{k-1} - x_{k-1}^*) = z_{k-1}. \quad (30)$$

Из (28), (9) получаем :

$$\begin{aligned} & \min_{x^* \in M^*} \|A_k(x_k - x^*)\|^2 \leq \|R_{\alpha_k}(\xi_k)z_{k-1} - R_{\alpha_k}(\xi_k) \times h_k\xi_k\|^2 = \\ &= \|R_{\alpha_k}(\xi_k)z_{k-1}\|^2 - 2h \left(R_{\alpha_k}(\xi_k)z_{k-1}, R_{\alpha_k}h_k\xi_k \right) + \alpha_k^2 h_k^2 = \\ &= \left[1 + (\alpha_k^2 - 1) \frac{(z_{k-1}, \xi_k)^2}{\|z_{k-1}\|^2} \right] \cdot \|z_{k-1}\|^2 - 2\alpha_k^2 h_k (z_{k-1}, \xi_k) + \alpha_k^2 h_k^2 = \\ &= \|z_{k-1}\|^2 + \alpha_k^2 \left[(z_{k-1}, \xi_k) - h_k \right]^2 - (z_{k-1}, \xi_k)^2. \end{aligned} \quad (31)$$

Оценим (z_{k-1}, ξ_k) , используя (11), (12) :

$$(z_{k-1}, \xi_k) = \left(A_{k-1}(x_{k-1} - x_{k-1}^*), \xi_k \right) = \left(x_{k-1} - x_{k-1}^*, A_{k-1}^* \xi_k \right) =$$

$$\begin{aligned}
&= \left(x_{k-1} - x_{k-1}^*, \frac{\widehat{g}_f(x_{k-1})}{\|\widehat{g}_{\varphi_{k-1}}(y_{k-1})\|} \right) = \\
&= \frac{f'_{x_{k-1}-x_{k-1}^*}(x_{k-1}) \cdot \|x_{k-1} - x_{k-1}^*\|}{\|\widehat{g}_{\varphi_{k-1}}(y_{k-1})\|} \quad (32)
\end{aligned}$$

(здесь мы воспользовались определением обобщенной производной по направлению (см. замечание 1)).

Рассмотрим два случая.

I. $(z_{k-1}, \xi_k) \leq h_k$; из (24), (26), (32) получаем :

$$\begin{aligned}
h_k &= \frac{[f(x_{k-1}) - m^*]}{\|\widehat{g}_{\varphi_{k-1}}(y_{k-1})\|} \cdot \frac{2MN}{M+N} \leq \\
&\leq \frac{f'_{x_{k-1}-x_{k-1}^*}(x_{k-1}) \cdot \|x_{k-1} - x_{k-1}^*\| \cdot 2N}{\|\widehat{g}_{\varphi_{k-1}}(y_{k-1})\| \cdot (M+N)} = (z_{k-1}, \xi_k) \cdot \frac{2N}{M+N}.
\end{aligned}$$

Получаем

$$\begin{aligned}
&\alpha_k^2 \left[(z_{k-1}, \xi_k) - h_k \right]^2 - (z_{k-1}, \xi_k)^2 \leq \\
&\leq \alpha_k^2 \left[(z_{k-1}, \xi_k) - (z_{k-1}, \xi_k) \cdot \frac{2N}{M+N} \right]^2 - (z_{k-1}, \xi_k)^2 \leq \\
&\leq \left[\left(\frac{M+N}{M-N} \right)^2 \cdot \left(1 - \frac{2N}{M+N} \right)^2 - 1 \right] \times (z_{k-1}, \xi_k)^2 = 0.
\end{aligned}$$

II. $(z_{k-1}, \xi_k) \geq h_k$; из (24), (26), (32) получаем :

$$\begin{aligned}
h_k &\geq \frac{f'_{x_{k-1}-x_{k-1}^*}(x_{k-1}) \cdot \|x_{k-1} - x_{k-1}^*\| \times 2M}{\|\widehat{g}_{\varphi_{k-1}}(y_{k-1})\| \cdot (M+N)} = \\
&= (z_{k-1}, \xi_k) \cdot \frac{2M}{M+N}.
\end{aligned}$$

Отсюда

$$\alpha_k^2 \left[(z_{k-1}, \xi_k) - h_k \right]^2 - (z_{k-1}, \xi_k)^2 \leq$$

$$\begin{aligned}
&\leq \alpha_k^2 \left[(z_{k-1}, \xi_k) - (z_{k-1}, \xi_k) \cdot \frac{2M}{M+N} \right]^2 - (z_{k-1}, \xi_k)^2 \leq \\
&\leq \left[\left(\frac{M+N}{M-N} \right)^2 \cdot \left(1 - \frac{2M}{M+N} \right)^2 - 1 \right] \cdot (z_{k-1}, \xi_k)^2 = 0.
\end{aligned}$$

Таким образом, в обоих случаях:

$$\alpha_k^2 \cdot \left[(z_{k-1}, \xi_k) - h_k \right]^2 - (z_{k-1}, \xi_k)^2 \leq 0. \quad (33)$$

Подставляя (33) в (31) получаем :

$$\min_{x^* \in M^*} \|A_k(x_k - x_k^*)\|^2 \leq \|z_{k-1}\|^2 = \min_{x^* \in M^*} \|A_{k-1}(x_{k-1} - x_{k-1}^*)\|^2.$$

Отсюда сразу следует теорема 2. Если функция $f(x)$ имеет единственный минимум в точке x^* , то в качестве следствия из теоремы 2 получаем теорему 3.

Теорема 3. Пусть минимум выпуклой функции $f(x)$ достигается в единственной точке x^* и $f(x^*) = m^*$, а для всех x , удовлетворяющих условию $\varrho(x) \leq c$, выполняются следующие неравенства:

$$N \cdot [f(x) - m^*] \leq |f'_{x-x^*}(x)| \cdot \|x - x^*\| \leq M [f(x) - m^*].$$

Тогда, если при применении ОГСРП

- (1) $\varrho(x_0) \leq c$,
- (2) $h_{k+1} = \frac{2MN}{M+N} \cdot \frac{[f(x) - m^*]}{\|\hat{g}_{\varphi_k}(y_k)\|}$,
- (3) $1 + \varepsilon \leq \alpha_{k+1} \leq \frac{M+N}{M-N}$,

то будут выполнены условия теоремы 1 (M, N, c, ε – положительные константы).

5. Обсуждение результатов

Теоремы 2 и 3 в сочетании с теоремой 1 дают нам возможность строить сходящиеся процедуры ОГСРП. Что можно сказать о скорости сходимости? Если выполняются условия теоремы 3, то $\|A_k(x_k - x^*)\| \leq c$ для

$k = 0, 1, \dots$. В силу (7), (8) коэффициент объемного расширения

$$\vartheta_{E_n}(A_k) = \prod_{i=1}^k \alpha_i.$$

Если обозначить объем n -мерного шара $\|y\| \leq c$ через ϑ_c , то вектор $x_k - x^*$ будет локализован в объеме $\vartheta_c / \vartheta_{E_n}(A_k) = \frac{\vartheta_c}{\prod_{i=1}^k \alpha_i}$. Если

$\alpha_i = \frac{M+N}{M-N}$, $i = 1, 2, \dots$, то объем области локализации $x_k - x^*$ уменьшается не медленнее, чем геометрическая прогрессия со знаменателем $\frac{M-N}{M+N}$. Следует отметить, что в силу (22) $N \geq 1$, т. е. при небольших M этот знаменатель существенно отличен от 1. Заметим также, что константы M и N не связаны непосредственно с «коэффициентом овражности» σ (см. § 1) и инвариантны по отношению к линейным преобразованиям пространства E_n .

Если $f(x) = (Kx, x)$ — положительно определённая квадратичная форма, то $M=N=2$. В этом случае α_k может быть сколько угодно большим положительным числом. В пределе при $\alpha_k \rightarrow \infty$ ОГСРП переходит в один из методов ортогонализации для решения систем линейных уравнений [1], который сходится за n шагов.

Теоремы 2 и 3 можно применять непосредственно, когда заранее известно значение $f(x)$ в точках минимума (m^*).

Это имеет место, если нам нужно решать систему из равенств и выпуклых неравенств:

$$\begin{cases} F_i(x) = 0; & i = 1, \dots, r; \\ H_j(x) \leq 0; & j = 1, \dots, p. \end{cases} \quad (34)$$

Решение этой системы можно свести к задаче минимизации

$$f(x) = \max \left[0, \max_i F_i^2, \max_j H_j(x) \right].$$

Если решение системы (34) существует, то $m^* = \min f(x) = 0$. В общем случае m^* неизвестно. В настоящее время нами исследуются алгоритмы, в которых m^* подбирается в процессе расчетов. Кроме того, исследуются алгоритмы ОГСРП, в которых $\{\alpha_k\}$ и $\{h_k\}$ строятся по схемам, отличным от приведенных в данной статье. В настоящее время в ИК АН УССР В.И.Билецким разработана стандартная программа

для ЭВМ М-220, с помощью которой проведено большое число экспериментальных расчетов по различным модификациям ОГСРП. Результаты расчетов показывают, что скорость сходимости ОГСРП практически намного выше, чем при использовании обобщенного градиентного спуска без растяжения пространства. Подробные данные о результатах экспериментальных расчетов содержатся в статье [4].

В заключение следует отметить, что операцию растяжения пространства удобно сочетать с различными методами градиентного типа. Можно надеяться, что для многих задач применение операции растяжения пространства повлечет существенное ускорение сходимости процесса минимизации.

Литература

1. Фаддеев Д. К., Фаддеева В. Н. Вычислительные методы линейной алгебры. Физматгиз. — М., 1960.
2. Шор Н. З. Обобщенный градиентный спуск // Труды первой зимней школы по математическому программированию (г. Дрогобыч). — М., 1969. — Вып. III. — С. 578–585.
3. Шор Н. З. О скорости сходимости обобщенного градиентного спуска // Кибернетика. — 1970. — №3. — С. 80–85.
4. Шор Н. З., Билецкий В. И. Метод растяжения пространства для ускорения сходимости в задачах овражного типа // Теория оптимальных решений. — 1969, вып. 2. — С. 3–18.

Р А З Д Е Л II

Приближенный метод эллипсоидов

В раздел включены статьи:

2.1. Приближенный метод эллипсоидов (<i>Стецюк, 2003</i>)	66
2.2. Об одном эллипсоиде для внешней аппроксимации n -мерного полушара (<i>Стецюк, 2003</i>)	74
2.3. Модификация метода эллипсоидов (<i>Стецюк, 2003</i>)	83
2.4. On a modification of ellipsoid method (<i>Stetsyuk, Nurminski, 2010</i>) . .	86
2.5. Об одном методе нахождения L_p -решения системы линейных уравнений (<i>Стецюк, Колесник, Березовский, 2003</i>)	92
2.6. К ускорению метода эллипсоидов с помощью использования шарового слоя (<i>Стецюк, Буханцов, 2002</i>)	100

Приближенный метод эллипсоидов

П. И. Стецюк

Кибернетика и системный анализ. — 2003. — № 3. — С. 141–146.

Предложен вариант метода эллипсоидов с такой же асимптотической скоростью сходимости по объему, как в известном методе Юдина-Немировского-Шора. Для одномерных задач он гарантирует на каждой итерации коэффициент уменьшения объема, который равен $2 - \sqrt{2} \approx 0.5858$.

В известном методе эллипсоидов (МЭ), предложенном в работах [1, 2], используется тот факт, что эллипсоид минимального объема, описанный вокруг полушара в n -мерном евклидовом пространстве E^n , имеет объем, меньший, чем объем самого шара. Коэффициент уменьшения объема равен

$$q_n = \sqrt{\frac{n-1}{n+1}} \left(\frac{n}{\sqrt{n^2-1}} \right)^n < \exp -\frac{1}{2n} < 1$$

и при больших n хорошо приближается асимптотической формулой

$$q_n \approx 1 - \frac{1}{2n}.$$

Итерация МЭ состоит в переходе к очередному эллипсоиду с постоянной скоростью уменьшения объема, которая характеризуется коэффициентом q_n .

Однако МЭ неприменим для одномерного случая (т. е. при $n = 1$), хотя очевидно, что основываясь на схеме последовательных отсечений, аналог такого метода в одномерном случае легко построить по правилу дихотомии. Этот недостаток МЭ связан с тем, что большая полуось эллипсоида минимального объема, описанного вокруг полушара радиуса r , равна $r \frac{n}{\sqrt{n^2-1}}$ и не имеет содержательного смысла при $n = 1$.

Указанный недостаток МЭ легко исправить, если построить метод, где полушар в E^n аппроксимируется не эллипсоидом минимального объема, а очень близким к нему эллипсоидом. Для полушара радиуса r в E^n ($n \geq 2$) это можно сделать с помощью эллипсоида, меньшая полуось которого равна

$$a = \left(1 - \frac{1}{n} \left(\sqrt{1 + \frac{1}{n^2}} - \frac{1}{n} \right) \right) r,$$

а $(n-1)$ больших полуосей одинаковы и равны

$$b = r \sqrt{1 + \frac{1}{n^2}}.$$

Центр эллипсоида находится внутри полушара и на расстоянии

$$t = \frac{r}{n} \left(\sqrt{1 + \frac{1}{n^2}} - \frac{1}{n} \right)$$

от центра шара в направлении меньшей полуоси.

Рассматриваемый вариант метода эллипсоидов условимся называть приближенным методом эллипсоидов (ПМЭ), поскольку указанный эллипсоид имеет объем несколько больше, чем тот, который используется в МЭ. Однако, это различие оказывается столь несущественным, что трудоемкости ПМЭ и МЭ по числу итераций фактически совпадают (см. таблицу). Кроме того, ПМЭ имеет такую же асимптотическую скорость сходимости по объему, как и МЭ. В отличие от последнего ПМЭ применим для одномерных задач, где коэффициент уменьшения объема на каждой итерации составляет $2 - \sqrt{2} \approx 0.5858$.

При описании ПМЭ будем основываться на такой же постановке задачи, которая используется для МЭ (см. [2]).

Пусть на E^n задано векторное поле $g(x)$, не обязательно непрерывное, $g(x) \in E^n$, $x \in E^n$. Требуется найти такую точку x^* , что

$$(g(x), x - x^*) \geq 0 \text{ при всех } x \in E^n.$$

Предполагается, что $g(x) \neq 0$ при $x \neq x^*$. Кроме того, задана априорная информация о локализации x^* , т. е. $x^* \in S(x_0, r_0)$, где $S(x_0, r_0)$ — замкнутый шар радиуса r_0 с центром в точке x_0 .

В [2] даны сведения к этой постановке следующих задач: задача минимизации выпуклой функции на шаре, общая задача выпуклого программирования, задача о седловой точке выпукло-вогнутых функций. Кроме того, ПМЭ может быть использован для поиска точек из некоторого стационарного множества, если мы умеем в произвольной точке из E^n убеждаться, принадлежит эта точка этому множеству или строить отсекающую это множество гиперплоскость.

Пусть $R_\alpha(\xi)$ — оператор растяжения пространства с коэффициентом $\alpha > 1$ в направлении $\xi \in E^n$, $\|\xi\| = 1$ [3]. Здесь $\|\cdot\|$ — евклидова норма. В матричном виде он имеет представление

$$R_\alpha(\xi) = I_n + (\alpha - 1) \xi \xi^T,$$

где $(\cdot)^T$ – символ транспонирования, I_n – единичная матрица размера $n \times n$. Обратным к оператору растяжения пространства является оператор «сжатия» пространства

$$R_\beta(\xi) = I_n + (\beta - 1) \xi \xi^T, \quad \beta = 1/\alpha.$$

В ПМЭ будет использоваться один и тот же коэффициент растяжения пространства

$$\alpha = \left(\sqrt{1 + \frac{1}{n^2}} + \frac{1}{n} \right), \quad (1)$$

который следует из соотношения большей (b) и меньшей (a) полуосей эллипсоида. Ему соответствует коэффициент "сжатия"

$$\beta = \frac{1}{\alpha} = \left(\sqrt{1 + \frac{1}{n^2}} - \frac{1}{n} \right). \quad (2)$$

ПМЭ реализуется следующей итеративной процедурой.

Инициализация. Фиксируется стартовая точка $x_0 \in E^n$ (здесь $n \geq 1$) и начальный радиус r_0 , такие, что

$$\|x_0 - x^*\| \leq r_0. \quad (3)$$

Вычислим β по формуле (2). Введем в рассмотрение матрицу B размера $n \times n$ и положим $B_0 := I_n$. Перейдем к первой итерации со значениями x_0 , r_0 и B_0 .

Пусть на k -й итерации найдены значения $x_k \in E^n$, r_k , B_k . Переход к $(k+1)$ -й итерации состоит в выполнении следующей последовательности действий.

Шаг 1. Вычислим $g(x_k)$. Если $g(x_k) = 0$, то ОСТАНОВ « $x_k = x^*$ ». Иначе переходим к шагу 2.

Шаг 2. Положим

$$\xi_k := \frac{B_k^T g(x_k)}{\|B_k^T g(x_k)\|}.$$

Шаг 3. Вычислим очередную точку

$$x_{k+1} := x_k - h_k B_k \xi_k, \quad \text{где} \quad h_k = \frac{r_k}{n} \beta.$$

Шаг 4. Вычислим

$$B_{k+1} := B_k R_\beta(\xi_k) \quad \text{и} \quad r_{k+1} := r_k \sqrt{1 + \frac{1}{n^2}}.$$

Шаг 5. Переходим к $(k + 1)$ -й итерации со значениями x_{k+1} , r_{k+1} и B_{k+1} .

Заметим, что согласно [2, 3] изложенный выше ПМЭ следует рассматривать как метод в B -форме, т. е. на каждой итерации метода хранится и корректируется матрица обратного преобразования пространства (матрица B), которая за исключением нулевой и первой итерации не обязательно будет симметричной. В то же время есть и другая форма этого метода, так называемая H -форма, которая связана с хранением и коррекцией симметричной матрицы H , где $H = BB^T$. Так, например, в работах [1, 4] МЭ представлен именно в H -форме.

Аналогично как и для МЭ в B -форме [2], для ПМЭ справедлива следующая теорема.

Теорема 1. *Генерируемая ПМЭ последовательность точек $\{x_k\}_{k=0}^{\infty}$ удовлетворяет неравенству*

$$\|A_k(x_k - x^*)\| \leq r_k, \quad k = 0, 1, 2, \dots,$$

где $A_k = B_k^{-1}$.

Доказательство теоремы 1 приводить не будем. Отметим лишь, что оно построено по типу того доказательства, которое для МЭ дано Н. З. Шором в работе [2]. Это не представляет особых проблем, так как ПМЭ отличается от МЭ лишь иным выбором h_k , r_k и β .

Множество точек x , удовлетворяющих неравенству

$$\|A_k(x_k - x)\| \leq r_k$$

представляет собой локализирующий точку x^* эллипсоид Φ_k , объем которого равен

$$\text{vol}(\Phi_k) = \frac{v_0 r_k^n}{\det A_k},$$

где v_0 — объем единичного n -мерного шара.

Вопрос о скорости сходимости ПМЭ сводится к оценке скорости уменьшения объема эллипсоида Φ_k , локализирующего x^* . Ответ на этот вопрос дает следующая теорема.

Теорема 2. *Коэффициент Q_n уменьшения объема на каждой итерации ПМЭ есть величина постоянная и равная*

$$Q_n = \left(1 + \frac{1}{n^2}\right)^{n/2} \left(\sqrt{1 + \frac{1}{n^2}} - \frac{1}{n}\right) < 1.$$

Доказательство. Для A_{k+1} имеем

$$A_{k+1} = B_{k+1}^{-1} = (B_k R_\beta(\xi_k))^{-1} = R_\alpha(\xi_k) B_k^{-1} = R_\alpha(\xi_k) A_k$$

и, кроме того, имеем $\det R_\alpha(\xi_k) = \alpha$ — для оператора растяжения пространства (см. [3]). Учитывая это, найдем коэффициент уменьшения объема

$$\begin{aligned} Q_n &= \frac{\text{vol}(\Phi_{k+1})}{\text{vol}(\Phi_k)} = \frac{v_0 r_{k+1}^n \det A_k}{v_0 r_k^n \det A_{k+1}} = \left(\frac{r_{k+1}}{r_k} \right)^n \frac{\det A_k}{\det R_\alpha(\xi_k) \det A_k} = \\ &= \left(\frac{r_{k+1}}{r_k} \right)^n \frac{1}{\alpha} = \left(\sqrt{1 + \frac{1}{n^2}} \right)^n \beta = \left(1 + \frac{1}{n^2} \right)^{n/2} \left(\sqrt{1 + \frac{1}{n^2}} - \frac{1}{n} \right), \end{aligned}$$

что и требовалось доказать.

Рассмотрим два следствия из теоремы 2.

Следствие 2. При больших n коэффициент Q_n аппроксимируется формулой

$$Q_n \approx 1 - \frac{1}{2n}.$$

Доказательство. Воспользуемся следующими неравенствами

$$\begin{aligned} \left(1 + \frac{1}{n^2} \right)^{n/2} &= \left(\left(1 + \frac{1}{n^2} \right)^{n^2} \right)^{1/2n} \leq e^{1/2n}, \\ \sqrt{1 + \frac{1}{n^2}} &\leq \sqrt{1 + \frac{1}{n^2} + \frac{1}{4n^4}} = \sqrt{\left(1 + \frac{1}{2n^2} \right)^2} = 1 + \frac{1}{2n^2}. \end{aligned}$$

Тогда при больших n имеем

$$\begin{aligned} Q_n \leq e^{1/2n} \left(1 + \frac{1}{2n^2} - \frac{1}{n} \right) &\approx \left(1 + \frac{1}{2n} + o\left(\frac{1}{n}\right) \right) \left(1 - \frac{1}{n} + o\left(\frac{1}{n}\right) \right) \approx \\ &\approx 1 + \frac{1}{2n} - \frac{1}{n} + o\left(\frac{1}{n}\right) \approx 1 - \frac{1}{2n}. \end{aligned}$$

Следствие 2. Коэффициент Q_n при $n = 1$ равен

$$Q_1 = 2 - \sqrt{2} \approx 0.5858.$$

Следствие 2 получается непосредственной подстановкой в формулу для Q_n значения $n = 1$, что дает

$$Q_1 = (1 + 1)^{1/2} (\sqrt{1 + 1} - 1) = \sqrt{2} (\sqrt{2} - 1) = 2 - \sqrt{2}.$$

Таблица 1. Коэффициенты q_n и Q_n для некоторых n

n	q_n	Q_n	Q_n/q_n	K_1	K_2
2	0.769800359	0.772542486	1.003562127	177	179
3	0.843750000	0.844163252	1.000489780	407	408
4	0.881318877	0.881423365	1.000118559	730	730
5	0.904224537	0.904260020	1.000039241	1144	1144
6	0.919685526	0.919700112	1.000015860	1651	1651
7	0.930834735	0.930841589	1.000007363	2249	2250
8	0.939259212	0.939262768	1.000003785	2940	2940
9	0.945850803	0.945852793	1.000002104	3723	3723
10	0.951149840	0.951151023	1.000001244	4598	4598
11	0.955503012	0.955503751	1.000000773	5565	5565
12	0.959143072	0.959143552	1.000000501	6624	6624
13	0.962232129	0.962232452	1.000000336	7776	7776
14	0.964886581	0.964886805	1.000000232	9019	9019
15	0.967192176	0.967192335	1.000000164	10354	10355

Следовательно, ПМЭ имеет такую же асимптотическую оценку скорости сходимости по объему, как и известный МЭ (следствие 1). Кроме того, ПМЭ применим для одномерного случая (следствие 2), где дает неплохое по сравнению с идеальным методом ($Q_1^* = 0.5$) уменьшение объема на итерации. Здесь под идеальным методом для одномерного случая подразумевается метод последовательных отсечений, основанный на правиле дихотомии.

Однако, эти два следствия характеризуют только предельные случаи ПМЭ. В целом же при произвольных $n \geq 2$ имеем $Q_n > q_n$. Что же можно сказать о других значениях n , особенно в пределах первого десятка, где использование МЭ достаточно приемливо. Не может ли ПМЭ потребовать значительно большего числа итераций, чем МЭ. Оказывается, различие при $n \sim 10$ очень незначительное и ПМЭ для достаточно сильного уменьшения объема требует практически такое же число итераций, как и МЭ. Об этом можно судить из таблицы, где для $n = 2, 3, \dots, 15$ даны значения q_n и Q_n , которые вычислялись по формулам:

$$q_n = \exp \left\{ 0.5 \times \ln \left(\frac{n-1}{n+1} \right) + n \times \ln \left(\frac{n}{\sqrt{n \times n-1}} \right) \right\},$$

$$Q_n = \exp \left\{ \ln \left(\sqrt{1 + 1/n/n} - 1/n \right) + 0.5 \times n \times \ln (1 + 1/n/n) \right\}.$$

Расчеты выполнены на Фортране с использованием двойной точности вычислений. При вычислении экспоненты, логарифма и квадратного корня использовались стандартные функции DEXP, DLOG и DSQRT.

Из табл. 1 видно, что отношение $\frac{Q_n}{q_n}$ достаточно близко к единице при $n = 2$ и по мере увеличения n стремится к единице. Для того, чтобы оценить количество итераций, которое требуется обоим методам для достижения достаточно сильного уменьшения объема, в таблице даны два столбца K_1 и K_2 . Здесь K_1 и K_2 задают для МЭ и ПМЭ то число итераций, которое необходимо сделать, чтобы для данного n гарантировать уменьшение объема, равное 10^{-10n} ; K_1 и K_2 вычислялись по формулам:

$$K_1 = \frac{-10n \ln 10}{\ln q_n} + 1; \quad K_2 = \frac{-10n \ln 10}{\ln Q_n} + 1,$$

чтобы гарантировать

$$(q_n)^{K_1} \leq 10^{-10n}; \quad (Q_n)^{K_2} \leq 10^{-10n}.$$

Из таблицы видим, что число итераций ПМЭ и число итераций МЭ практически одно и то же при всех n для указанного уменьшения объема, что означает, что для решения задач при $n \sim 10$ мы с равным успехом можем использовать, как первый так и второй методы. Например, при минимизации выпуклой функции $f(x)$, $x \in E^n$ такое уменьшение объема гарантирует улучшение рекорда по значению функции от

оптимального на 10 порядков. Следовательно, чтобы при удачном (не сильно завышенном) выборе r_0 в (3) найти минимум функции $f(x)$ с достаточно хорошей точностью ($\sim 10^{-12} - 10^{-10}$) можно использовать как МЭ так и ПМЭ.

В заключение отметим, что приведенный в работе метод не является единственным в рамках методов эллипсоидов, которые в асимптотике не отличаются от известного МЭ и имеют содержательный смысл при $n = 1$. Таких методов можно указать много, но не все они имеют достаточно «красивые» пределы при $n = 1$. Исключением есть только метод, который хотя и чуть хуже по оценке скорости сходимости, чем ПМЭ, но зато в одномерном случае дает уменьшение объема, равное «золотому сечению».

Литература

1. Юдин Д. Б., НЕМИРОВСКИЙ А. С. Информационная сложность и эффективные методы решения выпуклых экстремальных задач // Экономика и мат. методы. — 1976, вып.2. — С. 357–359.
2. ШОР Н. З. Метод отсечения с растяжением пространства для решения задач выпуклого программирования // Кибернетика. — 1977. — №1. — С. 94–95.
3. ШОР Н. З. Методы минимизации недифференцируемых функций и их приложения. — К.: Наук. думка, 1979. — 199с.
4. GRÖTSCHEL M., LOVASZ L., SHRIJVER A. Geometric Algorithms and Combinatorial Optimization. — Springer-Verlag, Berlin. — 1988. — 362 p.

Об одном эллипсоиде для внешней аппроксимации n -мерного полушара

П. И. Стецюк

Компьютерная математика. — 2003, вып. 2. — С. 144–151.

Предложен эллипсоид, содержащий полушар в n -мерном евклидовом пространстве, объем которого незначительно превышает объем эллипсоида минимального объема, содержащего этот же полушар. Показано, что на основе предложенного эллипсоида можно создать модификацию метода эллипсоидов, которая в отличие от известного метода эллипсоидов будет применима также и при решении одномерных задач. При $n \geq 2$ эта модификация будет иметь почти такую же скорость сходимости, как и метод эллипсоидов.

Введение.

Метод эллипсоидов предложен Д. Б. Юдиным и А. С. Немировским в [1], и независимо Н. З. Шором в [2]. Он может быть использован для решения задач поиска точки из некоторого стационарного выпуклого множества $X^* \subset E^n$, если в каждой точке $x \in E^n \setminus X^*$, можно построить гиперплоскость, отсекающую эту точку от множества X^* . Здесь E^n — n -мерное евклидово пространство. В указанный класс задач попадает задача безусловной минимизации выпуклой (гладкой или негладкой) функции. В [2] показано как применять метод эллипсоидов для решения следующих задач математического программирования: задача минимизации выпуклой функции на шаре, общая задача выпуклого программирования, задача о седловой точке выпукло-вогнутых функций. Скорость сходимости метода эллипсоидов зависит только от размерности пространства n ($n \geq 2$) и не зависит от самих свойств функций для указанных задач. Это замечательное свойство метода эллипсоидов, которое не омрачается даже его медленной скоростью сходимости при сравнительно небольших значениях n .

Однако, метод эллипсоидов не применим в одномерном случае (т. е. при $n = 1$). Дело в том, что аналогом метода эллипсоидов в одномерном случае считается метод дихотомии, хотя последний и не является частным случаем метода эллипсоидов при $n = 1$. Поэтому

при решении задач с помощью метода эллипсоидов для $n \geq 1$ следует учитывать то обстоятельство, что при $n \geq 2$ нужно применять метод эллипсоидов, а если $n = 1$ – переключиться на метод дихотомии. Такая схема решения задач фактически предполагает использование двух методов разного типа.

В данной работе рассмотрен новый эллипсоид для внешней аппроксимации полушара в E^n , что позволяет создать модификацию метода эллипсоидов, которая при $n \geq 2$ достаточно близка к методу эллипсоидов, но в то же время применима и в одномерном случае. Другими словами, при решении задач эта модификация будет корректной и в том случае, когда $n = 1$, и не потребует привлечения метода дихотомии, как это требуется при использовании метода эллипсоидов.

1. Эллипсоид минимального объема для внешней аппроксимации n -мерного полушара

Эллипсоид минимального объема, содержащий полушар радиуса r в E^n , имеет сплюснутую форму в направлении нормали к гиперплоскости, которая проходит через центр шара и определяет сам полушар. Параметры эллипсоида минимального объема даны на рис. 1, где a – длина меньшей полуоси эллипсоида, b – длина большей полуоси эллипсоида (количество таких полуосей будет равно $n - 1$), h – расстояние от центра шара до центра эллипсоида в направлении меньшей его полуоси.

Объем минимального эллипсоида меньше, чем объем шара радиуса r в E^n , и коэффициент q_n уменьшения объема можем записать в виде

$$q_n = \left(\frac{a}{r}\right) \left(\frac{b}{r}\right)^{n-1} = \frac{n}{n+1} \left(\frac{n}{\sqrt{n^2-1}}\right)^{n-1} < 1.$$

В работе [3] показано, что

$$q_n < \exp \left\{ -\frac{1}{2n} \right\} < 1,$$

и, следовательно, при больших n коэффициент уменьшения объема хорошо аппроксимируется асимптотической формулой

$$q_n \approx 1 - \frac{1}{2n}.$$

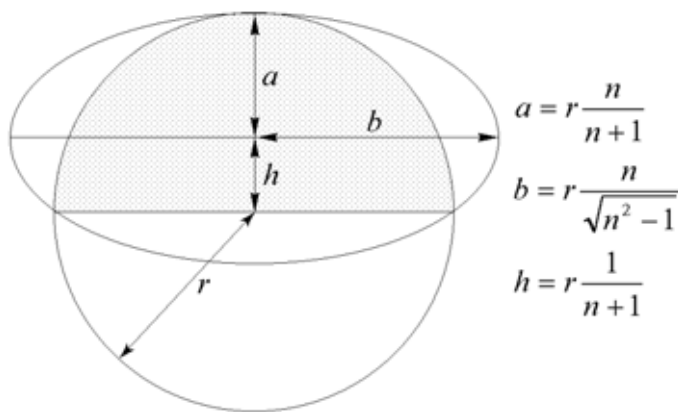


Рис. 1. Эллипсоид минимального объема, содержащий полушар в E^n .

На идее использования эллипсоида минимального объема базируется известный метод эллипсоидов. Итерация этого метода состоит в переходе от текущего эллипсоида к следующему эллипсоиду с постоянным коэффициентом уменьшения их объемов, равным q_n . Поэтому, чтобы в 10 раз уменьшить объем эллипсоида, локализирующего решение задачи, требуется сделать K итераций, где

$$K = -\frac{\ln 10}{\ln q_n} \approx (2 \ln 10)n \approx 4.6n.$$

При безусловной минимизации выпуклой функции такое уменьшение объема означает, что методу эллипсоидов потребуется сделать $4.6n^2$ итераций для того, чтобы на порядок улучшить отклонение найденного рекордного значения минимизируемой функции от оптимального ее значения. Это число итераций достаточно велико при n порядка нескольких десятков и становится огромным при n порядка нескольких сотен. Чтобы при $n = 10$ найти точку минимума выпуклой функции с относительной точностью по значению функции, равной 10^{-10} , методу эллипсоидов достаточно осуществить 4600 итераций. Для современных персональных компьютеров это требует небольших (порядка одной или нескольких секунд) вычислительных затрат. Учитывая, что метод эллипсоидов имеет более широкую область приложений, чем только минимизация выпуклых функций, его можно достаточно эффективно использовать при решении задач с числом переменных $n \leq 10$.

Причина того, что метод эллипсоидов не применим в одномерном случае, достаточно проста. Она состоит в том, что длина большей полуоси эллипсоида минимального объема (см. рис. 1) определяется по формуле

$$b = r \frac{n}{\sqrt{n^2 - 1}},$$

которая не имеет смысла, когда $n = 1$. В силу этого, реализация метода эллипсоидов с помощью формул, задающих параметры эллипсоида минимального объема, не может быть автоматически перенесена на одномерный случай. Здесь требуется некоторая другая процедура, какой в одномерном случае становится метод дихотомии. Тогда коэффициент уменьшения объема в расчете и на одномерный случай можно задать формулой из [4, стр. 83].

$$q(n) = \begin{cases} q_n, & \text{если } n \geq 2; \\ 1/2, & \text{если } n = 1. \end{cases}$$

из которой следует, что решить задачу для всех n нельзя с помощью метода, основанного на эллипсоиде минимального объема. Таким образом, если $n \geq 2$ нужно использовать метод эллипсоидов, а при $n = 1$ – метод дихотомии.

Итак, решение задачи с помощью метода эллипсоидов выглядит несколько искусственным, так как мы сознательно предполагаем возможность использования двух различных методов. Рассмотренная ниже модификация метода эллипсоидов позволит упростить решение задач, сделав его единообразным при любом n , в том числе при $n = 1$.

2. Новый эллипсоид для внешней аппроксимации n -мерного полушара

Рассмотрим новый эллипсоид, который содержит полушар радиусом r в E^n . Его параметры даны на рис. 2:

a' – длина меньшей полуоси нового эллипсоида,

b' – длина его большей полуоси (количество таких полуосей будет равно $n - 1$),

h' – расстояние от центра шара до центра нового эллипсоида в направлении короткой полуоси.

Эти параметры близки к параметрам эллипсоида минимального объема. Так, например, при больших значениях n

$$a' \approx a \left(1 - \frac{1}{n^2}\right); \quad b' \approx b \left(1 - \frac{1}{2n^4}\right); \quad h' \approx h \left(1 + \frac{1}{n}\right).$$

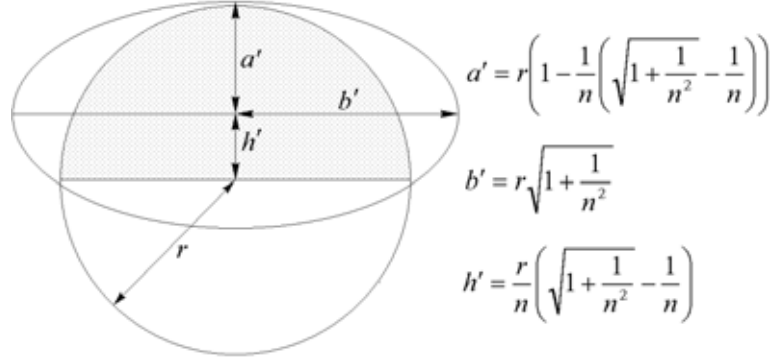


Рис. 2. Новый эллипсоид, содержащий полушар в E^n .

Объем нового эллипсоида меньше, чем объем шара, и коэффициент уменьшения объема равен

$$Q_n = \left(\frac{a'}{r}\right) \left(\frac{b'}{r}\right)^{n-1} = \left(1 + \frac{1}{n^2}\right)^{n/2} \left(\sqrt{1 + \frac{1}{n^2}} - \frac{1}{n}\right) < 1.$$

На его основе можно построить сходящийся со скоростью геометрической прогрессии итерационный метод (по типу метода эллипсоидов), который условимся называть модификацией метода эллипсоидов.

Итерация этой модификации будет состоять в переходе от текущего эллипсоида, локализирующего решение задачи, к очередному эллипсоиду, локализирующему это же решение, объем которого будет меньше, чем объем текущего эллипсоида в Q_n раз. Операция, позволяющая замкнуть цикл вычислений при переходе от одного эллипсоида к другому, связана с преобразованием нового эллипсоида в шар.

Для преобразования нового эллипсоида в шар достаточно растянуть его в направлении короткой полуоси с коэффициентом

$$\alpha = \frac{b'}{a'} = \sqrt{1 + \frac{1}{n^2}} + \frac{1}{n}.$$

Эту процедуру выполняет оператор растяжения пространства [5], который в матричной форме имеет следующий вид:

$$R_\alpha(\xi) = I_n + (\alpha - 1)\xi\xi^T, \quad \xi \in E^n, \quad \|\xi\| = 1,$$

где $(\cdot)^T$ – транспонирование, $\|\cdot\|$ – евклидова норма, I_n – единичная матрица порядка n .

Оператор растяжения пространства, в основном, служит для геометрической иллюстрации модификации метода эллипсоидов и используется при доказательстве ее сходимости. Такая же роль оператора растяжения пространства и в методе эллипсоидов. При конкретной реализации модификации метода эллипсоидов используется оператор (обратный к оператору растяжения пространства), который имеет следующую форму [5]:

$$R_\beta(\xi) = I_n + (\beta - 1)\xi\xi^T, \quad \xi \in E^n, \quad \|\xi\| = 1,$$

где

$$\beta = \frac{1}{\alpha} = \sqrt{1 + \frac{1}{n^2}} - \frac{1}{n}.$$

Отметим, что в отличие от эллипсоида минимального объема новый эллипсоид может быть использован также и при $n = 1$, т. е. в одномерном случае. В этом случае длины полуосей нового эллипсоида будут следующими:

$$b' = r\sqrt{2}, \quad a' = (2 - \sqrt{2})r,$$

следовательно, они пределены корректно (для эллипсоида минимального объема это было не так).

Применяя построенную модификацию метода эллипсоидов в одномерном случае, на каждой итерации будет гарантироваться уменьшение объема длины интервала, локализующего решение задачи в Q_1 раз, где

$$Q_1 = \frac{a'}{r} = 2 - \sqrt{2} \approx 0.5858.$$

Заметим, что указанное уменьшение длины интервала достаточно близко к тому, которое обеспечивается при использовании метода дихотомии (обозначим его $Q_1^* = 0.5$), и составляет не более, чем

$$\frac{Q_1 - Q_1^*}{Q_1^*} = \frac{0.0858}{0.5} \approx 0.17 \approx 17\%,$$

Поэтомe вместо пяти итераций метода дихотомии потребуется выполнить не более шести итераций модификации метода эллипсоидов. Для

одномерного поиска 30 – 40 итераций метода дихотомии обеспечивают обычно очень высокую точность решения. Следовательно, для достижения такой же точности при использовании модификации метода эллипсоидов потребуется не более 36 – 48 итераций.

При $n \geq 2$ рассматриваемая модификация будет близка к методу эллипсоидов по числу итераций. Это обусловлено тем, что параметры нового эллипсоида незначительно отличаются от параметров эллипсоида минимального объема и в силу этого $Q_n \approx q_n$. Приведем два аргумента в пользу данного выше утверждения.

Во-первых, при больших n коэффициент Q_n аппроксимируется с помощью такой же асимптотической формулы $Q_n \approx 1 - \frac{1}{2n}$, как и коэффициент уменьшения объема q_n в методе эллипсоидов. Этот факт следует из того, что при больших n верно

$$Q_n = \left(1 + \frac{1}{n^2}\right)^{n/2} \left(\sqrt{1 + \frac{1}{n^2}} - \frac{1}{n}\right) \approx \left(1 + \frac{1}{2n}\right) \left(1 - \frac{1}{n}\right) \approx 1 - \frac{1}{2n}.$$

Следовательно, модификация метода эллипсоидов будет иметь такую же асимптотическую оценку скорости сходимости, как и метод эллипсоидов.

Однако, наш первый аргумент связан с характеристикой только предельного случая, когда n – достаточно большое. В общем случае, для произвольных $n \geq 2$ имеем $Q_n > q_n$.

Второй аргумент связан со сравнением скорости сходимости метода эллипсоидов и предложенной модификации для небольших значений n ($2 \leq n \leq 10$). В таблице для указанных размерностей приведено число итераций, которое требуется обоим методам, чтобы найти решение задачи с относительной точностью, равной 10^{-10} .

Здесь использованы следующие обозначения: K_1 , K_2 – число итераций для метода эллипсоидов и рассматриваемой модификации (соответственно), которые вычислены по формулам:

$$K_1 = \frac{-10n \ln 10}{\ln q_n} + 1; \quad K_2 = \frac{-10n \ln 10}{\ln Q_n} + 1.$$

Это гарантирует точность по уменьшению объема

$$(q_n)^{K_1} \leq 10^{-10n}; \quad (Q_n)^{K_2} \leq 10^{-10n},$$

которой достаточно для решения задач с относительной точностью 10^{-10} .

Таблица 1.

n	q_n	Q_n	Q_n/q_n	K_1	K_2
2	0.7698004	0.7725425	1.0035621	177	179
3	0.8437500	0.8441633	1.0004898	407	408
4	0.8813189	0.8814234	1.0001186	730	730
5	0.9042245	0.9042600	1.0000392	1144	1144
6	0.9196855	0.9197001	1.0000159	1651	1651
7	0.9308347	0.9308416	1.0000074	2249	2250
8	0.9392592	0.9392628	1.0000038	2940	2940
9	0.9458508	0.9458528	1.0000021	3723	3723
10	0.9511498	0.9511510	1.0000012	4598	4598

В таблице даны также значения q_n и Q_n . Очевидность их незначительного отличия для обоих методов подчеркивает отношение Q_n/q_n .

Из таблицы видно, что при всех значениях n число итераций для модификации метода эллипсоидов практически то же самое, что и число итераций для метода эллипсоидов. Поэтому можно утверждать, что и в области значений $n \sim 10$ всеравно какой из обоих методов использовать для решения задач.

Заключение. Проведенный анализ вычислительной эффективности модификации метода эллипсоидов подтверждает, что новый эллипсоид целесообразно использовать для внешней аппроксимации n -мерного полушара взамен эллипсоида минимального объема. При $n \geq 2$ его использование гарантирует практически такую же скорость сходимости, как и использование эллипсоида минимального объема. Учитывая, что построенная на его основе модификация метода эллипсоидов позволяет решать задачи и в одномерном случае, то очевидно, что использование последней целесообразнее, чем применение комбинации методов эллипсоидов и дихотомии.

Литература

1. Юдин Д. Б., НЕМИРОВСКИЙ А. С. Информационная сложность и эффективные методы решения выпуклых экстремальных задач // Экономика и мат. методы. – 1976, вып.2. – С. 357–359.
2. ШОР Н. З. Метод отсечения с растяжением пространства для решения задач выпуклого программирования // Кибернетика. – 1977. – №1. – С. 94–95.
3. GRÖTSCHEL M., LOVASZ L., A. SHRIJVER A. Geometric Algorithms and Combinatorial Optimization. – Springer-Verlag, Berlin. – 1988. – 362 p.
4. НЕМИРОВСКИЙ А. С., ЮДИН Д. Б. Сложность задач и эффективность методов оптимизации. – М.: Наука, 1979, – 383 с.
5. ШОР Н. З. Методы минимизации недифференцируемых функций и их приложения. – К.: Наук. думка, 1979. – 199с.

Модификация метода эллипсоидов ¹

П. И. Стецюк

*Информационный бюллетень Ассоциации математического
программирования. №10. Научное издание. – Екатеринбург: УрО
РАН, 2003. – С. 216–217.*

Рассмотренная ниже модификация метода эллипсоидов (ММЭ) предназначена для решения следующей задачи.

Пусть на E^n задано векторное поле $g(x)$, необязательно непрерывное, $g(x) \in E^n$, $x \in E^n$ ($n \geq 1$). Требуется найти такую точку x^* , что $(g(x), x - x^*) \geq 0$ при всех $x \in E^n$. Предполагается, что $g(x) \neq 0$ при $x \neq x^*$.

ММЭ реализуется следующей итеративной процедурой.

Инициализация. Фиксируется стартовая точка $x_0 \in E^n$ и начальный радиус r_0 , такой, что $\|x_0 - x^*\| \leq r_0$. Вычислим β по формуле $\beta = \left(\sqrt{1 + \frac{1}{n^2}} - \frac{1}{n} \right)$. Введем в рассмотрение матрицу B размера $n \times n$ и положим $B_0 := I_n$, где I_n – единичная матрица размером $n \times n$. Перейдем к первой итерации со значениями x_0 , r_0 и B_0 .

Пусть на k -й итерации найдены значения $x_k \in E^n$, r_k , B_k . Переход к $(k+1)$ -й итерации состоит в выполнении следующей последовательности действий.

Шаг 1. Вычислим $g(x_k)$. Если $g(x_k) = 0$, то ОСТАНОВ « $x^* = x_k$ ». Иначе переходим к шагу 2.

Шаг 2. Положим

$$\xi_k := \frac{B_k^T g(x_k)}{\|B_k^T g(x_k)\|}.$$

Шаг 3. Вычислим очередную точку

$$x_{k+1} := x_k - h_k B_k \xi_k, \quad \text{где} \quad h_k = \frac{r_k}{n} \beta.$$

Шаг 4. Вычислим

$$B_{k+1} := B_k R_\beta(\xi_k) \quad \text{и} \quad r_{k+1} := r_k \sqrt{1 + \frac{1}{n^2}},$$

¹Работа выполнена при частичной финансовой поддержке Украинского научно-технологического центра (грант N 1625)

где $R_\beta(\xi) = I_n + (\beta - 1)\xi\xi^T$ – оператор «сжатия» пространства с коэффициентом β в направлении $\xi \in E^n$ ($\|\xi\| = 1$).

Шаг 5. Переходим к $(k + 1)$ -й итерации со значениями x_{k+1} , r_{k+1} , B_{k+1} .

Факт сходимости ММЭ и его скорость сходимости устанавливается посредством следующих теорем.

Теорема 1. Пусть $A_k = B_k^{-1}$. Тогда, генерируемая ММЭ последовательность точек $\{x_k\}_{k=0}^\infty$ удовлетворяет неравенству

$$\|A_k(x_k - x^*)\| \leq r_k, \quad k = 0, 1, 2, \dots$$

Множество точек x , удовлетворяющих $\|A_k(x_k - x)\| \leq r_k$, представляет собой локализующий точку x^* эллипсоид Φ_k , объем которого $\text{vol}(\Phi_k) = v_0 r_k^n / \det A_k$. Здесь v_0 – объем единичного n -мерного шара.

Вопрос о скорости сходимости ММЭ сводится к оценке скорости уменьшения объема эллипсоида Φ_k , локализующего x^* . Ответ на этот вопрос дает следующая теорема.

Теорема 2. Коэффициент Q_n уменьшения объема на каждой итерации ММЭ есть величина постоянная и равная

$$\begin{aligned} Q_n = \frac{\text{vol}(\Phi_{k+1})}{\text{vol}(\Phi_k)} &= \left(1 + \frac{1}{n^2}\right)^{n/2} \left(\sqrt{1 + \frac{1}{n^2}} - \frac{1}{n}\right) < \\ &< \exp\left\{-\frac{1}{2n} + \frac{1}{2n^2}\right\} < 1. \end{aligned}$$

Из теоремы 2 следуют такие основные свойства ММЭ:

- (i) при больших n коэффициент Q_n аппроксимируется такой же асимптотической формулой $Q_n \approx 1 - \frac{1}{2n}$, как и коэффициент уменьшения объема в известном методе эллипсоидов [1,2];
- (ii) если $n = 1$, то $Q_1 = 2 - \sqrt{2} \approx 0.5858$. Заметим, что метод эллипсоидов [1,2] не работает в одномерном случае.

При всех n ММЭ требует для решения задач практически такого же числа итераций, как и метод эллипсоидов. Последнее обусловлено тем, что в ММЭ используется эллипсоид, очень близкий к тому, который положен в основу метода эллипсоидов.

Следовательно, ММЭ можно использовать с таким же успехом, как и известный метод эллипсоидов. Однако в отличие от последнего он применим к одномерному случаю и есть близким к методу дихотомии.

Литература

1. Юдин Д. Б., Немировский А. С. Информационная сложность и эффективные методы решения выпуклых экстремальных задач // Экономика и мат. методы. – 1976. – Вып.2. – С. 357–359.
2. Шор Н. З. Метод отсечения с растяжением пространства для решения задач выпуклого программирования // Кибернетика. – 1977. – №1. – С. 94–95.

On a modification of ellipsoid method.

Petro Stetsyuk and Evgeni Nurminski

Математическое моделирование, оптимизация и информационные технологии. – Кишинэу: Эврика, 2010. – С. 37–42.

A modification of ellipsoid method (EM) is considered. This modification (MEM) is asymptotically equivalent to the original EM for large-scale problems but applicable to lower dimensions down to one-dimensional case. MEM preserves close to EM volume reduction factors for all dimensions and gives close to optimal 0.5 accuracy improving multiplier for Id. Keywords: one-dimensional optimization, ellipsoid method, volume reduction factor.

The modification of ellipsoid method (MEM) consists in the following. On k -th iteration a new ellipsoid Φ_{k+1} with the volume $\text{vol}(\Phi_{k+1})$ is constructed in such a way that $\text{vol}(\Phi_{k+1}) < \text{vol}(\Phi_k)$ with a volume reduction factor

$$Q_n = \frac{\text{vol}(\Phi_{k+1})}{\text{vol}(\Phi_k)} = \left(1 + \frac{1}{n^2}\right)^{n/2} \left(\sqrt{1 + \frac{1}{n^2}} - \frac{1}{n}\right) < 1,$$

where n is a space dimension. This modification has following properties:

- volume reduction factor Q_n is approximated for large n by the same asymptotic formula $Q_n \approx 1 - \frac{1}{2n}$ as in the known Yudin-Nemirovskii-Shor ellipsoid method (EM);
- $Q_1 = 2 - \sqrt{2} \approx 0.5858$, which is close to the optimal 0.5 for one-dimensional dichotomy method.

Ellipsoid Method (EM)

Ellipsoid method (EM) was proposed by D.B.Yudin and A.S.Nemirovskii [1] and by N.Z.Shor [2]. It uses the minimal volume ellipsoid circumscribed around a half-ball in n -dimensional Euclidean space E^n (see Fig.1). This ellipsoid has smaller volume than the volume of a ball, the volume reduction factor being equal to

$$q_n = \left(\frac{a}{r}\right) \left(\frac{b}{r}\right)^{n-1} = \sqrt{\frac{n-1}{n+1}} \left(\frac{n}{\sqrt{n^2-1}}\right)^n < \exp\left\{-\frac{1}{2n}\right\} < 1.$$

For large n it is closely approximated by the asymptotic formula $q_n \approx 1 - \frac{1}{2n}$. EM iteration consists in subsequent construction of the ellipsoid with the constant volume reduction factor q_n .

A disadvantage of EM is that it does not work in the one-dimensional case. The reason is that the larger semi-axis b of the minimal volume ellipsoid (see Fig.1) is determined by the formula, which makes no sense while $n = 1$.

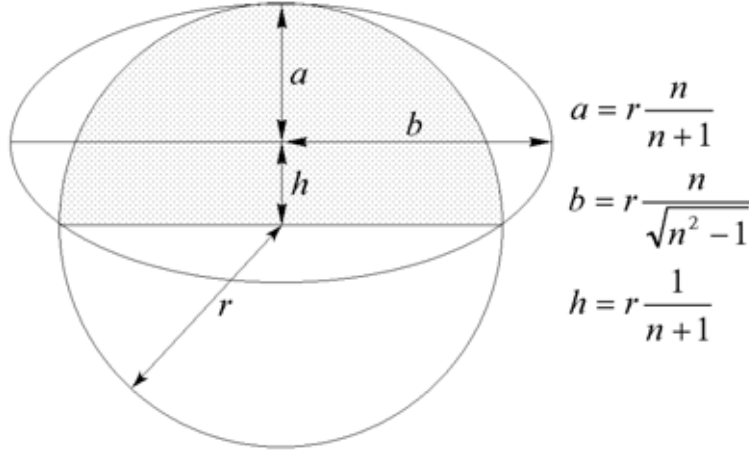


Figure 1. Minimal volume ellipsoid containing a half-ball in E^n .

A Modification of Ellipsoid Method

The main idea of our modification of ellipsoid method (MEM) consists in using another ellipsoid instead of minimal volume one [3]. Parameters of the new ellipsoid are shown in the Fig. 2. They are very close to those used in EM.

The volume of this ellipsoid is also less than the volume of the ball, and the volume reduction factor is equal to

$$Q_n = \left(\frac{a}{r}\right) \left(\frac{b}{r}\right)^{n-1} = \left(1 + \frac{1}{n^2}\right)^{n/2} \left(\sqrt{1 + \frac{1}{n^2}} - \frac{1}{n}\right) < 1.$$

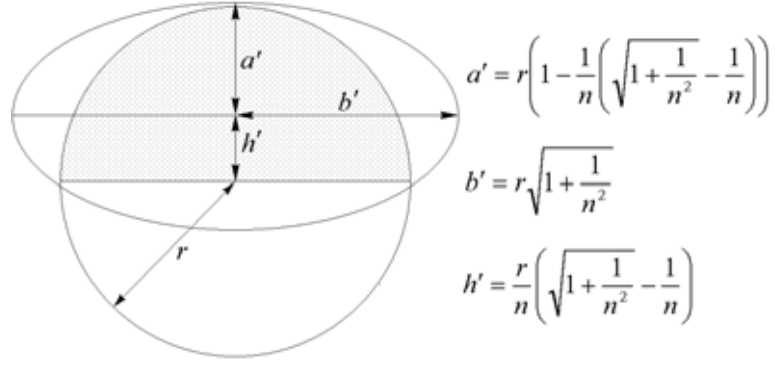


Figure 2. The new ellipsoid containing a half-ball.

To transform resulting ellipsoid into a new ball it is sufficient to stretch it in the direction of the smaller axis with a coefficient

$$\alpha = \frac{b}{a} = \left(\sqrt{1 + \frac{1}{n^2}} + \frac{1}{n} \right).$$

It can be done using the following operator of a space dilation (see [4])

$$R_\alpha(\xi) = I_n + (\alpha - 1)\xi\xi^T, \quad \xi \in E^n, \quad \|\xi\| = 1,$$

where $(\cdot)^T$ means a transposition, $\|\cdot\|$ is an Euclidean norm, I_n is $n \times n$ identity matrix.

Algorithmic details of MEM

MEM can be used to solving the following problem.

Let a vector field $g(x)$ (not necessary continuous) maps E^n into itself: $g : E^n \rightarrow E^n$. Find a point x^ such that $(g(x), x - x^*) \geq 0$ for all $x \in E^n$.*

It is assumed that this problem has a unique solution.

MEM can be implemented by the following iterative procedure.

Initialization. The iteration process starts with values x_0 , r_0 , B_0 which are set as follows.

- Fix $x_0 \in E^n$ and an initial radius r_0 such that $\|x_0 - x^*\| \leq r_0$.
- Calculate β by formula $\beta = \frac{1}{\alpha} = \left(\sqrt{1 + \frac{1}{n^2}} - \frac{1}{n} \right)$.
- Set $B_0 := I_n$ and iteration counter $k = 0$.

Let on the k -th iteration the values $x_k \in E^n$, r_k , B_k were found. The $(k+1)$ -th iteration consists in the following sequence of steps.

Step 1. Calculate $g(x_k)$. If $g(x_k) = 0$ then $x^* = x_k$ and stop.

Step 2. Determine $\xi_k := \frac{B_k^T g(x_k)}{\|B_k^T g(x_k)\|}$.

Step 3. Calculate $x_{k+1} = x_k - h_k B_k \xi_k$, where $h_k = \frac{r_k}{n} \beta$.

Step 4. Calculate $B_{k+1} := B_k R_\beta(\xi_k)$ and $r_{k+1} := r_k \sqrt{1 + \frac{1}{n^2}}$.

Step 5. Proceed to the $k+1$ -th iteration with values x_{k+1} , r_{k+1} and B_{k+1} .

Convergence prorperties of MEM

Theorem 1. *A sequence of points $\{x_k\}_{k=0}^\infty$ generated by MEM satisfies the inequality $\|A_k(x_k - x^*)\| \leq r_k$, $k = 0, 1, 2, \dots$ with $A_k = B_k^{-1}$.*

A set of points x satisfying inequality $\|A_k(x_k - x)\| \leq r_k$ is an ellipsoid Φ_k which contains the point x^* and has the volume $\text{vol}(\Phi_k) = \frac{\nu_0 r_k^n}{\det A_k}$. Here ν_0 is a volume of n -dimensional unit ball.

The rate of convergence of MEM is given by the theorem.

Theorem 2. *The volume reduction factor Q_n is a constant of the form*

$$\begin{aligned} Q_n &= \frac{\text{vol}(\Phi_{k+1})}{\text{vol}(\Phi_k)} = \left(1 + \frac{1}{n^2}\right)^{n/2} \left(\sqrt{1 + \frac{1}{n^2}} - \frac{1}{n}\right) < \\ &< \exp \left\{ -\frac{1}{2n} + \frac{1}{2n^2} \right\} < 1, \end{aligned}$$

which does not depend on an iteration number k .

There are two corollaries from the 2.

Corollary 1. *For a large n , the factor Q_n is approximated by the asymptotic formula $Q_n \approx 1 - \frac{1}{2n}$.*

Corollary 2. *If $n = 1$ then the factor $Q_1 = 2 - \sqrt{2} \approx 0.5858$ which is close to the optimal 0.5 for one-dimensional dichotomy method.*

Therefore, the proposed modification of ellipsoid methods has the same asymptotic estimation of volume convergence rate as the known Yudin-Nemirovskii-Shor ellipsoid method (1). Moreover, it can be successfully applied also in one-dimensional case (2).

Comparative Analysis of MEM and EM

Corollaries 1 and 2 characterize limit cases of MEM. Generally, for arbitrary $n \geq 2$ we have $Q_n > q_n$. However, it turns out that the number of iterations for MEM almost coincides with one for EM, as it can be seen in the Table for $n = 2, 3, \dots, 10$.

Table 1.

n	q_n	Q_n	Q_n/q_n	K_1	K_2
2	0.7698004	0.7725425	1.0035621	177	179
3	0.8437500	0.8441633	1.0004898	407	408
4	0.8813189	0.8814234	1.0001186	730	730
5	0.9042245	0.9042600	1.0000392	1144	1144
6	0.9196855	0.9197001	1.0000159	1651	1651
7	0.9308347	0.9308416	1.0000074	2249	2250
8	0.9392592	0.9392628	1.0000038	2940	2940
9	0.9458508	0.9458528	1.0000021	3723	3723
10	0.9511498	0.9511510	1.0000012	4598	4598

Here values q_n , Q_n , their ratios Q_n/q_n , iteration numbers for EM and for MEM required to solve the problem with a relative accuracy equal to 10^{-10} are given. Last two columns represent numbers of iterations for EM and MEM sufficient to achieve 10^{-10n} reduction in volume.

It can be seen that iteration numbers for EM and MEM quickly became the same even for relatively small n . It means that MEM can be safely used instead of EM in automatic solvers when dimensionality of the problem

is not known in advance and can be as low as I . The possible area of applications is small-set coordinate descent methods considered now as a way to approach huge-scale optimization problems [5].

This research is supported by the joint Ukrainian-Russian grant F28.1/005 (FRSF, Ukraine)- 09-01-90413 (RFBR, Russia).

Bibliography

1. YUDIN D. B. AND NEMIROVSKII A. S. Informational complexity and efficient methods for the solution of convex extremal problems // *Matekon* **13 (3)**, 1976. – P. 25–45).
2. SHOR N. Z. Cut-off method with space dilation in convex programming problems // *Cybernetics* **13 (1)**. 1977, – P. 94–96).
3. STETSYUK P. I. An Approximate Method of Ellipsoids // *Cybernetics and Systems Analysis*, **39(3)**, 2003. – P. 435–439.
4. SHOR N. Z. *Minimization Methods for Non-Differentiable Functions*. – Berlin: Springer-Verlag, 1985. – 178 p.
5. NESTEROV Y. Efficiency of coordinate descent methods on huge-scale optimization problems CORE Discussion Paper // 2010/2, (January, 2010), 23 pp.

Об одном методе нахождения L_p -решения системы линейных уравнений

П. И. Стецюк, Ю. С. Колесник, О. А. Березовский
Теория оптимальных решений. – 2003. – № 2. – С. 83–90.

Рассматривается задача нахождения L_p -решений переопределенной системы линейных алгебраических уравнений при интервальных ограничениях на компоненты решения. Предложен метод для нахождения L_p -решений, основанный на использовании модификации метода эллипсоидов. Показано, что этот метод находит L_p -решение системы линейных алгебраических уравнений за конечное число итераций, зависящее от числа неизвестных компонент в L_p -решении.

Постановка задачи. Пусть имеется следующая система линейных алгебраических уравнений:

$$Ax \approx b, \quad (1)$$

$$l \leq x \leq u, \quad (2)$$

где $A - n \times m$ вещественная матрица; $b - m$ -мерный вещественный вектор; $l, u - n$ -мерные векторы, такие, что для всех $i = 1, \dots, n$ $u_i \geq l_i$; $x - n$ -мерный вектор неизвестных параметров. Требуется найти такой вектор x^* , который удовлетворяет ограничениям (2) и «наилучшим образом» (обозначено знаком « \approx ») выполняет соотношение (1).

Понятие «наилучшим образом», как правило, принято понимать как наилучшее решение системы (1)–(2) в так называемой L_p -норме, т. е. когда норма некоторого вектора $y = (y_1, \dots, y_n)$ определена следующим образом: $\|y\|_p = \left(\sum_{i=1}^n |y_i|^p \right)^{1/p}$, где $p \geq 1$. Случай $p = \infty$ определяется как $\|y\|_\infty = \max_{i=1, \dots, n} |y_i|$. Случай $p = 2$ соответствует стандартной евклидовой норме.

Нахождению наилучшего L_p -решения системы (1)–(2) может быть поставлена в соответствие следующая задача выпуклого программирования: найти

$$\min_{x \in R^n} \left\{ f_p(x) = \|Ax - b\|_p \right\} \quad (3)$$

при ограничениях:

$$l \leq x \leq u, \quad (4)$$

где $p \in R^1$ — скалярный параметр, такой что $p \geq 1$, который гарантирует выпуклость функции $f_p(x)$.

Подобные задачи часто встречаются в самых разных областях прикладной математики, например: при обработке результатов наблюдений, построении и анализе различного рода моделей (физических, биологических, экономических, социальных и др.), при поиске компромиссных решений в моделях с противоречивыми данными и т. д.

Когда для переопределенной системы линейных уравнений ранг матрицы A равен n , то задача (3)–(4) имеет единственное решение (даже, когда ни одно из ограничений (4) не активно). В общем случае, сложно что-то утверждать о единственности x_p^* (все зависит как от границ на переменные x_i , $i = 1, \dots, n$, так и от свойств матрицы A). Однако, при создании метода для нахождения x_p^* , будем использовать только тот факт, что задача (3)–(4) всегда имеет решение, и то, что $f_p^* = f_p(x_p^*) \geq 0$. В случае неоднозначности x_p^* будем находить только одно из решений, соответствующее оптимальному значению функции f_p^* .

О методах решения задачи. В частных случаях (т. е. при некоторых конкретных значениях p) задача (3)–(4) может быть решена достаточно эффективно посредством стандартных численных процедур оптимизации. Так, например, когда $p = 1$, $p = +\infty$, задача может быть сведена к задаче линейного программирования и решена симплекс-методом, либо методом внутренних точек. Когда $p = 2$, то она может быть сведена к задаче квадратичного программирования и решена посредством численных методов, например, из [1].

В общем случае, т. е. при произвольном значении параметра p ($p \geq 1$), задача (3)–(4) есть общей задачей выпуклого программирования при простейших (интервальных) ограничениях на переменные. При этом минимизируемая целевая функция может быть как гладкой, что имеет место при $p = 2$, так и негладкой, что имеет место при $p = 1$ и при $p = \infty$. Поэтому для ее решения требуется разработка численных методов решения, основанных на алгоритмах выпуклой недифференцируемой минимизации. Заметим, что разработка таких методов напрашивается даже тогда, когда речь идет только о наиболее распространенных значениях p (т. е. $p = 1, 2, \infty$). В этом случае одним и тем же методом можно решать задачу для каждого из указанных значений p , что практичнее, чем привлечение для решения этих задач существенно различных методов, какими есть методы линейного программирования и методы квадратичного программирования.

Конкретный метод решения задачи (3)–(4) будет определяться выбранным алгоритмом негладкой оптимизации. Так, например, такой метод можно построить на основе r -алгоритмов [2], как эффективного средства решения задач негладкой оптимизации. Подобная методика применялась в [3] и базировалась на использовании модификации $r(\alpha)$ -алгоритма. Разработанный в [3] метод предназначен для решения задачи безусловной минимизации выпуклой функции $f_p(x)$ и соответствует случаю, когда в задаче (3)–(4) отсутствуют интервальные ограничения вида (4). Однако, учет интервальных ограничений для r -алгоритмов не представляет особых проблем. Это можно сделать как посредством использования негладкой штрафной функции максимума из нарушенных ограничений, либо посредством «четного» периодического продолжения целевой функции (заданной на отрезке) на все пространство посредством замены переменных [4].

Однако, на практике часто требуется локализовать решение x_p^* с определенной (иногда достаточно «грубой») точностью. Семейство r -алгоритмов таким свойством не обладает. Это наводит на мысль построить более «простой» метод для решения задачи (3)–(4), но с возможностью оценки области локализации x_p^* на каждой итерации этого метода.

Это легко сделать, если в качестве алгоритма негладкой оптимизации использовать модификацию метода эллипсоидов [5]. Заметим, что с таким же успехом можно использовать и классический вариант метода эллипсоидов [2], так как скорость сходимости обоих методов приблизительно одинакова. Однако, преимущество модификации в том, что ее итерация единообразно описывается и для одномерного случая (т. е. когда $n = 1$), что позволит использовать метод и x_p^* при решении одномерных задач вида (3)–(4).

При обосновании сходимости такого метода не возникает особых проблем. Тем более, что интервальные ограничения на переменные (4) играют существенную роль при использовании любого метода по типу метода эллипсоидов. Они дают возможность ограничить область поиска точки минимума функции $f_p(x)$ и выполнить начальные условия для применения метода эллипсоидов, которые состоят в том, что точку x_p^* требуется искать в шаре некоторого заданного радиуса. Такой шар легко построить, описав вокруг параллелепипеда, заданного ограничениями (4), либо шар, либо эллипсоид минимального объема. Кроме того, в методах эллипсоидов при построении отсекающих гиперплоскостей важно именно нормированное направление субградиента, что упрощает расчет при произвольном значении параметра p .

Однако, пойдя на использование модификации метода эллипсоидов, мы избавимся от проблем, связанных с обоснованием сходимости метода решения задачи (3)–(4). Но проблема медленной скорости сходимости, сопутствующая методам эллипсоидов, останется, и чем больше n , тем более медленную скорость сходимости будет обеспечивать построенный метод. Поэтому, заранее отметим, что рассматриваемый далее метод будет применим для эффективного решения задач (3)–(4) при небольших значениях n (порядка десятка переменных). Величина m роли не играет и метод может быть применим при достаточно больших ее значениях (порядка сотен тысяч).

Метод нахождения x_p^* на основе модификации метода эллипсоидов. Модификация метода эллипсоидов [5] предназначена для решения следующей задачи.

Пусть на E^n задано векторное поле $g(x)$, необязательно непрерывное, $g(x) \in E^n$, $x \in E^n$, $n \geq 1$. Требуется найти такую точку x^* , что $(g(x), x - x^*) \geq 0$ при всех $x \in E^n$. Предполагается, что $g(x) \neq 0$ при $x \neq x^*$.

Кроме того, априорная информация, которая требуется модификации метода эллипсоидов, связана с выбором начальной стартовой точки x_0 и такого радиуса шара r_0 с центром в x_0 , чтобы в этом шаре находилась искомая точка x^* .

Удовлетворить эти требования для задачи (3)–(4) не представляет особых проблем. Так, первую часть этих требований можно удовлетворить, используя следующую лемму.

Лемма 1. Пусть $t^* = \max\{t_{i^*}, t_{j^*}\}$, где $t_{i^*} = \max_{i=1, \dots, n} \{x_i - u_i\}$ и $t_{j^*} = \max_{j=1, \dots, n} \{l_j - x_j\}$. Обозначим: i^* - значение i ($1 \leq i \leq n$), на котором достигается t_{i^*} ; j^* - значение j ($1 \leq j \leq n$), на котором достигается t_{j^*} ; $\partial f_p(x)$ - субградиент функции $f_p(x)$; e_k - k -й орт в E^n , $1 \leq k \leq n$.

Тогда, вектор

$$g_p(x) = \begin{cases} \partial f_p(x), & \text{если } t^* \leq 0, \\ e_{i^*}, & \text{если } t^* > 0 \text{ и } t^* > t_{i^*}, \\ -e_{j^*}, & \text{если } t^* > 0 \text{ и } t^* \leq t_{j^*}. \end{cases}$$

удовлетворяет свойству

$$(g_p(x), x - x^*) \geq 0 \quad \text{для всех } x \in E^n. \quad (5)$$

Доказательство леммы 1 приводить не будем. Отметим лишь, что она имеет следующий содержательный смысл. Если точка x находится внутри допустимой области, заданной ограничениями (4), то в качестве $g(x)$ выбирается субградиент функции $f_p(x)$ в этой точке, а если точка находится вне допустимой области, то выбирается субградиент к максимально нарушенному ограничению вида (4). Учитывая выпуклость функции и выпуклость ограничений (4), очевидно, что такой выбор $g(x)$ гарантирует выполнение свойства (5).

В соответствии с правилом вычисления $g_p(x)$ в лемме 1 построим формулу для вычисления «обобщенного» значения функции задачи (3)-(4):

$$F_p(x) = \begin{cases} +\infty, & \text{если } t^* > 0; \\ f_p(x), & \text{если } t^* \leq 0. \end{cases}$$

Значение $F_p(x)$ будем использовать при построении одного из критериев останова в методе. Этот критерий обусловлен тем, что, если $f_p^* = 0$, то вычисление субградиента в точке x_p^* некорректно при большинстве значений p , что связано с операцией деления на величину пропорциональную корню из f_p^* (т. е. будет иметь место деление на ноль).

Вторую часть требований, связанную с априорной информацией о локализации x_p^* , легко обеспечить, выбрав в качестве центра шара центр параллелепипеда, заданного интервальными ограничениями на переменные (4), и установив радиус шара таким, чтобы этот шар содержал параллелепипед и имел минимальный объем. Это обеспечивает следующая лемма.

Лемма 2. Пусть

$$x_0 = \left\{ \frac{u_1 + l_1}{2}, \dots, \frac{u_n + l_n}{2} \right\} \quad \text{и} \quad r_0 = \frac{1}{2} \sqrt{\sum_{i=1}^n (u_i - l_i)^2}.$$

Тогда параллелепипед $P(x) = \{x : l_i \leq x_i \leq u_i, \quad i = 1, \dots, n\}$ содержится в шаре $S(x_0, r_0) = \{x : \|x - x_0\| \leq r_0\}$.

Заметим, что это самый простой и очевидный выбор x_0 и r_0 для модификации метода эллипсоидов. Более сложный выбор можно осуществить, описав вокруг параллелепипеда эллипсоид минимального объема. Однако, в этом случае усложняется доказательство метода, если некоторые переменные фиксированы (т. е. для некоторых i $l_i = u_i$). Дело в

том, что эллипсоид минимального объема приводит к проектированию на орты, соответствующие этим равенствам, и обратная матрица, задающая этот эллипсоид, является вырожденной. Такая схема выбора x_0 и r_0 возможна, хотя мы ее рассматривать не будем. Заметим лишь, что она имеет некоторые преимущества перед леммой 2, так как приводит к уменьшению размерности решаемой задачи.

Учитывая вышеизложенное, метод для нахождения x_p^* примет следующий вид. Здесь p будем считать входным параметром метода ($p \geq 1$), а ε_f определяет абсолютную точность, с которой требуется найти значение f_p^* .

Инициализация. Установим стартовую точку $x_0 = (u + l)/2$ и начальный радиус $r_0 = \frac{1}{2} \sqrt{\sum_{i=1}^n (u_i - l_i)^2}$. Вычислим β по формуле

$$\beta = \left(\sqrt{1 + \frac{1}{n^2}} - \frac{1}{n} \right).$$

Введем в рассмотрение матрицу B размера $n \times n$ и положим $B_0 := I_n$, где I_n – единичная матрица размером $n \times n$. Перейдем к первой итерации со значениями x_0 , r_0 и B_0 .

Пусть на k -й итерации найдены значения $x_k \in E^n$, r_k , B_k . Переход к $(k + 1)$ -й итерации состоит в выполнении такой последовательности действий.

Шаг 1. Вычислим $F_p(x_k)$.

Если $F_p(x_k) = 0$, то «ОСТАНОВ» $x_p^* = x_k$. Если критерий останова не сработал, то вычислим $g_p(x_k)$.

Если $\|B_k^T g_p(x_k)\| r_k \leq \varepsilon_f$, то «ОСТАНОВ» $x_p^* = x_k$. Иначе переходим к шагу 2.

Шаг 2. Положим

$$\xi_k := \frac{B_k^T g_p(x_k)}{\|B_k^T g_p(x_k)\|}.$$

Шаг 3. Вычислим очередную точку

$$x_{k+1} := x_k - h_k B_k \xi_k, \quad \text{где} \quad h_k = \frac{r_k}{n} \beta.$$

Шаг 4. Вычислим

$$B_{k+1} := B_k R_\beta(\xi_k) \quad \text{и} \quad r_{k+1} := r_k \sqrt{1 + \frac{1}{n^2}},$$

где $R_\beta(\xi) = I_n + (\beta - 1)\xi\xi^T$ – оператор «сжатия» пространства с коэффициентом β в направлении $\xi \in E^n$ ($\|\xi\| = 1$), I_n – единичная матрица размером $n \times n$.

Шаг 5. Переходим к $(k+1)$ -й итерации со значениями x_{k+1} , r_{k+1} , B_{k+1} .

Факт сходимости вышеприведенного метода обеспечивает следующая теорема, доказательство которой не будем приводить в силу его громоздкости.

Теорема 1. *Если не сработал ни один из критериев останова (см. шаг 1.), то генерируемая методом последовательность точек $\{x_k\}_{k=0}^\infty$ удовлетворяет неравенству*

$$\|B_k^{-1}(x_k - x_p^*)\| \leq r_k, \quad k = 0, 1, 2, \dots$$

Скорость сходимости метода можно установить посредством следующих рассуждений.

Множество точек x , удовлетворяющих $\|B_k^{-1}(x_k - x)\| \leq r_k$, представляет собой локализирующий точку x_p^* эллипсоид Φ_k объемом

$$\text{vol}(\Phi_k) = v_0 r_k^n / \det(B_k^{-1}).$$

Здесь v_0 – объем единичного n -мерного шара. Следовательно, вопрос о скорости сходимости метода нахождения x_p^* сводится к оценке на каждом шаге этого метода скорости уменьшения объема эллипсоида Φ_k , локализирующего x_p^* .

Теорема 2. *Коэффициент q уменьшения объема эллипсоида, локализирующего x_p^* , на каждой итерации k метода есть величина постоянная и равная*

$$q = \frac{\text{vol}(\Phi_{k+1})}{\text{vol}(\Phi_k)} = \left(1 + \frac{1}{n^2}\right)^{n/2} \times \beta < \exp\left\{-\frac{1}{2n} + \frac{1}{2n^2}\right\} < 1.$$

Из теоремы 2 следует: для того, чтобы уменьшить объем эллипсоида, локализирующего x_p^* , в 10 раз, требуется выполнить K итераций, где

$$K = -\frac{\ln 10}{\ln q} \approx (2 \ln 10)n \approx 4.6n.$$

Так, например, для задачи (3)–(4) такое уменьшение объема означает следующее: чтобы на порядок улучшить отклонение найденного рекордного значения минимизируемой функции $f_p(x)$ от ее оптимального значения f_p^* , потребуется сделать $4.6n^2$ итераций. Таким образом, когда в

задаче (3)–(4) число переменных $n \leq 10$, то для ее решения вышеприведенный метод будет достаточно эффективен. Его эффективность при этих значениях n можно оценить из таблицы, где приведено число итераций itn для нахождения x_p^* с точностью 10^{-10} по значению функции.

n	2	3	4	5	6	7	8	9	10
itn	179	408	730	1144	1651	2250	2940	3723	4598

Для современных персональных компьютеров при $n \sim 10$ и $m \sim 3n$ требует небольших вычислительных затрат по времени (порядка секунды). Следовательно, его можно успешно применять для нахождения x_p^* , когда число переменных $n \leq 10$. Значение m существенно не влияет на скорость сходимости метода. Однако от него зависит трудоемкость вычислений значения функции $f_p(x)$ и ее субградиента, который при $m \sim 1000$ будут вносить более весомый вклад в трудоемкость метода, чем алгоритмические операции (шаги 2–4).

В заключение, отметим, что вышеприведенный метод свободен от параметров регулировки шага, настройка всех его параметров делается автоматически. Кроме того, за конечное число итераций он гарантирует нахождение x_p^* с требуемой точностью (на каждой итерации можно указать область локализации точки минимума).

Литература

1. Пшеничный Б. Н., Данилин Ю. М. Численные методы в экстремальных задачах. – М.: Наука, 1975. – 319 с.
2. Шор Н. З. Методы минимизации недифференцируемых функций и их приложения. – К.: Наук. думка, 1979. – 199с.
3. Стецюк П. И., Колесник Ю. Г. К вопросу выбора метода аппроксимации результатов измерений // Интеллектуальные информационно-аналитические системы и комплексы. – К.: Ин-т кибернетики им. В.М.Глушкова НАН Украины, 2000. – С. 62–67.
4. Шор Н. З., Стеценко С. И. Квадратичные экстремальные задачи и недифференцируемая оптимизация. – К.: Наук. думка, 1989. – 208с.
5. Стецюк П. И. Приближенный метод эллипсоидов // Кибернетика и системный анализ. – 2003. – № 3. – С. 141–146.

К ускорению метода эллипсоидов с помощью использования шарового слоя.

П. И. Стецюк, Д. М. Буханцов

Теория оптимальных решений. — 2002. — № 2. — С. 63–70.

Предложена модификация метода эллипсоидов, ускоренная за счет использования эллипсоида минимального объема, содержащего шаровой слой. Идея модификации состоит в том, что на текущей итерации используется информация об отсекающей гиперплоскости из предыдущей итерации и, если эта информация позволяет уменьшить объем аппроксимирующего эллипсоида, то такое уменьшение реализуется, если же нет, то реализуется шаг известного метода эллипсоидов. Приведены результаты численных экспериментов, которые показывают, что при минимизации существенно овражных функций скорость сходимости этой модификации оказывается в несколько раз лучше, чем скорость сходимости метода эллипсоидов.

В известном методе эллипсоидов (МЭ), предложенном в работах [1, 2], используется тот факт, что эллипсоид минимального объема, описанный вокруг полушара в n -мерном евклидовом пространстве E^n , имеет объем, меньший, чем объем самого шара. Итерация МЭ состоит в переходе к очередному эллипсоиду с постоянной скоростью уменьшения объема, которая характеризуется коэффициентом уменьшения объема q_n :

$$q_n = \sqrt{\frac{n-1}{n+1}} \left(\frac{n}{\sqrt{n^2-1}} \right)^n < \exp \left\{ -\frac{1}{2n} \right\} < 1.$$

При больших n коэффициент q_n приближается формулой $q_n \approx 1 - 1/2n$ и близок к единице, что обуславливает медленную сходимость МЭ.

В работах [3], [4] предприняты попытки построить модификации МЭ с более сильными коэффициентами уменьшения объема на шаге. Для этого использовались оптимальные эллипсоиды, описанные вокруг сложных выпуклых тел (сегмент, «шаровой» слой и s -пирамида). Итерация этих методов связана с переходом в центр минимального по объему эллипсоида, описанного вокруг одного из этих выпуклых тел. Эти

модификации МЭ, как правило, характеризовались коэффициентами уменьшения объема, достаточно близкими к q_n .

На итерации методов из [3] для построения «шарового» слоя использовалась информация о нескольких отсекающих гиперплоскостях, которые вычислялись на этой итерации. В то же время «забывалась» информация об отсекающей гиперплоскости с предыдущей итерации, которая уже вычислена ранее. В данной работе построим модификацию МЭ, для которой попытаемся исправить эту ситуацию.

Сделаем это следующим образом. Если на текущей итерации можно построить шаровой слой (в направлении вычисленной на этой итерации отсекающей гиперплоскости), используя информацию об предыдущей отсекающей гиперплоскости, то мы будем переходить в центр минимального по объему эллипсоида, описанного вокруг этого шарового слоя. Если же это не удастся, то будем переходить в центр минимального по объему эллипсоида, содержащего полушар, как в МЭ. Понятно, что для такой модификации МЭ можно гарантировать скорость сходимости не хуже, чем скорость сходимости МЭ. В том же случае, когда на итерациях часто получаем «хорошие» шаровые слои, то скорость сходимости будет значительно лучше, чем у МЭ. При этом на каждой итерации будет вычисляться только одна отсекающая гиперплоскость.

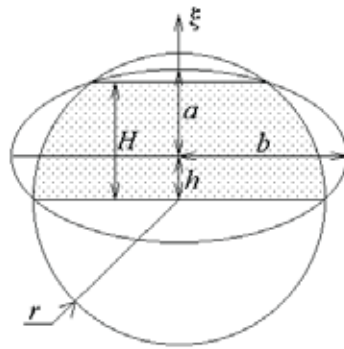


Рис. 1.

В основу этой модификации МЭ положен эллипсоид минимального объема, содержащий «шаровой» слой (см. рис. 1), который получен в результате пересечения шара радиуса r с центром в некоторой точке

$x_k \in E^n$ и множества вида

$$PU(x_k, \xi, H) = \left\{ x : 0 \leq (x - x_k, \xi) \leq H \right\},$$

где $\xi \in E^n$ – вектор единичной длины, H – скаляр такой, что $0 < H \leq r$.

Параметры эллипсоида минимального объема, содержащего указанный «шаровой» слой, определяются по следующим формулам (см. [4]):

$$h(r, H) = \frac{1}{2} \left(H - \sqrt{\frac{n-1}{n+1} H^2 + \left(\frac{2r^2 - H^2}{(n+1)H} \right)^2} + \frac{2r^2 - H^2}{(n+1)H} \right), \quad (1)$$

$$\beta(r, H) = \sqrt{\sqrt{\frac{n-1}{n+1} + \left(\frac{2r^2 - H^2}{(n+1)H^2} \right)^2} - \frac{2r^2 - H^2}{(n+1)H^2}}, \quad (2)$$

$$b(r, H) = \sqrt{r^2 - \left(\frac{H}{2} \right)^2 \frac{(1 - \beta^2(r, H))^2}{\beta^2(r, H)}}. \quad (3)$$

Здесь $b(r, H)$ – длина большой полуоси эллипсоида, $h(r, H)$ – расстояние от центра шара до центра эллипсоида в направлении ξ , которое совпадает с направлением меньшей полуоси, $\beta(r, H)$ – коэффициент, задающий отношение длины меньшей полуоси к длине большой полуоси. Эти параметры однозначно определяют эллипсоид и вместе со способом вычисления H их достаточно для реализации модификации МЭ. Для нахождения параметра H (см. рис. 2) на каждой итерации будем использовать формулы из [4].

При описании модификации МЭ будем использовать оператор $R_\alpha(\xi)$ – оператор растяжения пространства с коэффициентом $\alpha > 1$ в направлении $\xi \in E^n$, $\|\xi\| = 1$ [5]. Здесь $\|\cdot\|$ – евклидова норма. В матричном виде он имеет представление

$$R_\alpha(\xi) = I_n + (\alpha - 1)\xi\xi^T,$$

где $(\cdot)^T$ – символ транспонирования, I_n – единичная матрица размера $n \times n$. Обратным к оператору растяжения пространства является оператор «сжатия» пространства

$$R_\beta(\xi) = I_n + (\beta - 1)\xi\xi^T, \quad \beta = 1/\alpha.$$

Модификация МЭ будет иметь дело с такой же постановкой задачи, которая используется для МЭ (см. [2]).

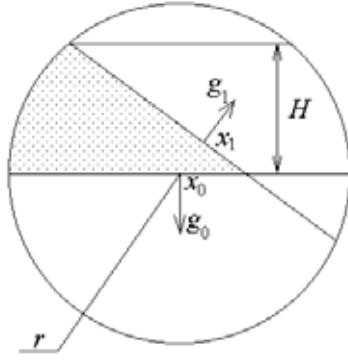


Рис. 2.

Пусть на E^n задано векторное поле $g(x)$, не обязательно непрерывное, $g(x) \in E^n$, $x \in E^n$. Требуется найти такую точку x^* , что $(g(x), x - x^*) \geq 0$ при всех $x \in E^n$. Предполагается, что $g(x) \neq 0$ при $x \neq x^*$. Кроме того, задана априорная информация о локализации x^* , т. е. $x^* \in S(x_0, r_0)$, где $S(x_0, r_0)$ — замкнутый шар радиуса r_0 с центром в точке x_0 .

Модификация МЭ реализуется следующей итеративной процедурой.

Инициализация. Фиксируем стартовую точку $x_0 \in E^n$ и начальный радиус r_0 , такие, что $\|x_0 - x^*\| \leq r_0$.

Введем в рассмотрение матрицу B размера $n \times n$ и положим $B_0 := I_n$.

Первая итерация требует выполнения следующей последовательности действий.

Шаг 1.1. Вычислим $g(x_0)$. Если $g(x_0) = 0$, то ОСТАНОВ « $x_0 = x^*$ ». Иначе переходим к шагу 1.2.

Шаг 1.2. Положим

$$\xi_0 := \frac{g(x_0)}{\|g(x_0)\|}, \quad h_0 = \frac{1}{n+1}r_0, \quad \beta_0 = \sqrt{\frac{n-1}{n+1}}.$$

Шаг 1.3. Вычислим очередную точку

$$x_1 := x_0 - h_0 \xi_0.$$

Шаг 1.4. Вычислим

$$B_1 := B_0 R_{\beta_0}(\xi_0) \quad \text{и} \quad r_1 = r_0 \frac{n}{\sqrt{n^2 - 1}}.$$

Шаг 1.5. Переходим к очередной итерации со значениями $x_1, B_1, r_1, \xi_0, h_0, \beta_0$.

Пусть на k -й итерации найдены значения $x_k \in E^n, B_k, r_k, \xi_{k-1}, h_{k-1}, \beta_{k-1}$. Переход к $(k+1)$ -й итерации состоит в выполнении следующей последовательности действий.

Шаг k.1. Вычислим $g(x_k)$. Если $g(x_k) = 0$, то ОСТАНОВ « $x_k = x^*$ ». Иначе переходим к шагу k.2.

Шаг k.2. Положим

$$\xi_k := \frac{B_k^T g(x_k)}{\|B_k^T g(x_k)\|}.$$

Шаг k.3. Если $h_{k-1}/\beta_{k-1} < -r_k(\xi_k, \xi_{k-1})$ то положим:

$$H_k = -\frac{h_{k-1}}{\beta_{k-1}}(\xi_k, \xi_{k-1}) + \sqrt{r_k^2 - \left(\frac{h_{k-1}}{\beta_{k-1}}\right)^2} \sqrt{1 - (\xi_k, \xi_{k-1})^2}.$$

Иначе $H_k = r_k$.

Шаг k.4. Вычислим $h_k = h(r_k, H_k)$ по формуле (1) и $\beta_k = \beta(r_k, H_k)$ по формуле (2).

Шаг k.5. Вычислим очередную точку

$$x_{k+1} := x_k - h_k B_k \xi_k.$$

Шаг k.6. Вычислим

$$B_{k+1} := B_k R_{\beta_k}(\xi_k).$$

Шаг k.7. Переходим к $(k+1)$ -й итерации со значениями $x_{k+1}, B_{k+1}, r_{k+1}, \xi_k, h_k$ и β_k .

Справедлива следующая теорема.

Теорема 1. Генерируемая модификацией МЭ последовательность точек $\{x_k\}_{k=0}^\infty$ удовлетворяет неравенству

$$\|A_k(x_k - x^*)\| \leq r_k, \quad k = 0, 1, 2, \dots, \quad (4)$$

где $A_k = B_k^{-1}$.

Множество точек x , удовлетворяющих неравенству

$$\|A_k(x_k - x)\| \leq r_k,$$

представляет собой локализирующий точку x^* эллипсоид Φ_k , объем которого равен

$$\text{vol}(\Phi_k) = \frac{v_0 r_k^n}{\det A_k},$$

где v_0 - объем единичного n -мерного шара.

Вопрос о скорости сходимости модификации МЭ сводится к оценке скорости уменьшения объема эллипсоида Φ_k , локализирующего x^* . Ответ на этот вопрос дает следующая теорема.

Теорема 2. Коэффициент Q_k уменьшения объема на k -й итерации модификации МЭ не хуже, чем коэффициент q_n для МЭ, т. е.

$$Q_k \leq q_n = \sqrt{\frac{n-1}{n+1}} \left(\frac{n}{\sqrt{n^2-1}} \right)^n < \exp \left\{ -\frac{1}{2n} \right\} < 1.$$

Теорема 2 утверждает, что скорость сходимости модификации МЭ не хуже, чем скорость сходимости МЭ. В случае, когда на итерации будем иметь значение $H_k \ll r_k$, то коэффициент Q_k будет значительно меньше, чем q_n . Естественно, что такие итерации способны значительно ускорить сходимость модификации МЭ по отношению к МЭ.

Таблица 1.

n	$r_0 = 5$		$r_0 = 10$		$r_0 = 100$		$r_0 = 1000$	
	K_1	K_2	K_1	K_2	K_1	K_2	K_1	K_2
2	108	82	118	76	136	105	145	113
3	254	151	255	162	297	181	326	208
4	429	217	456	247	565	241	614	306
5	694	299	718	328	850	364	911	373
10	2902	884	3081	823	3633	959	4011	1110
15	7625	1517	8195	1607	8825	1765	10094	1878
20	15646	2398	16066	2517	18145	2792	20009	3020
25	27375	3586	28619	3614	31325	4021	33469	4262
30	44495	4941	45431	5000	49391	5513	50000	5799

Таблица 2.

∇	$r_0 = 5$		$r_0 = 10$		$r_0 = 100$		$r_0 = 1000$	
	K_1	K_2	K_1	K_2	K_1	K_2	K_1	K_2
2	121	79	119	83	147	99	152	107
3	276	135	298	123	345	171	350	185
4	505	221	542	239	599	255	688	275
5	874	314	909	324	984	332	1109	404
10	3830	1061	4025	1002	4484	1167	4953	1254
15	9586	2169	9959	2134	10956	2376	11965	2680
20	18666	3786	19221	3924	20903	4283	22899	4483
25	31475	5758	32363	5846	35217	6595	38112	7187
30	48674	8246	49823	8297	50000	9036	50000	10158

Это подтверждают вычислительные эксперименты для двух тестовых примеров, которые связаны с нахождением точки минимума выпуклой функции. В качестве тестовых рассматривались следующие функции:

$$f_1(x_1, \dots, x_n) = \sum_{i=1}^n 2^{i-1} x_i^2 \quad \text{и} \quad f_2(x_1, \dots, x_n) = \sum_{i=1}^n 2^{i-1} |x_i|.$$

Сравнительные результаты экспериментов для стартовой точки $x_0 = (1, \dots, 1)$ и четырех значений $r_0 = \{5, 10, 100, 1000\}$ даны в таблице 1 для функции $f_1(x)$ и в таблице 2 для функции $f_2(x)$ при числе переменных от 2 до 30. Здесь K_1 – число итераций МЭ и K_2 – число итераций модификации МЭ, которые потребовались для нахождения точки минимума при одном и том же критерии останова. Для $f_1(x)$ использовался критерий останова $\|g(x_k)\| \leq 10^{-6}$, а для $f_2(x)$ критерий останова $f(x_k) - f^* \leq 10^{-6}$. Максимальное количество итераций для МЭ было ограничено до 50000, что сработало для обеих функций при $n = 30$, $r_0 \geq 100$. Из таблиц видим, что по числу итераций модификация МЭ обгоняет МЭ, иногда и значительно, до 8 раз.

Литература

1. Юдин Д. Б., НЕМИРОВСКИЙ А. С. Информационная сложность и эффективные методы решения выпуклых экстремальных задач // Экономика и мат. методы. – 1976. – Вып.2. – С. 357–359.
2. ШОР Н. З. Метод отсечения с растяжением пространства для решения задач выпуклого программирования // Кибернетика. – 1977. – №1. – С. 94–95.
3. ШОР Н. З., ГЕРШОВИЧ В. И. Об одном семействе алгоритмов для решения задач выпуклого программирования // Кибернетика. – 1979. – №4. – С.62–67.
4. KONIG H. AND PALLASCHKE D. On Khachian's algorithm and minimal ellipsoids // Numerische Mathematik. – 1981. – **36**. – P.211–223.
5. ШОР Н. З. Методы минимизации недифференцируемых функций и их приложения. – К.: Наук. думка, 1979. – 199с.

Р А З Д Е Л III

***r*-алгоритмы и эллипсоиды**

В раздел включены статьи:

3.1. К вопросу сходимости $r_\mu(\alpha)$ -алгоритма (<i>Стецюк, 1995</i>)	110
3.2. К обоснованию сходимости алгоритмов с растяжением пространства (<i>Стецюк, 1996</i>)	116
3.3. <i>r</i> -алгоритмы и эллипсоиды (<i>Стецюк, 1996</i>)	121
3.4. Метод центров тяжести простых тел (<i>Стецюк, 1996</i>)	150
3.5. К методам эллипсоидов (<i>Стецюк, 1999</i>)	177
3.6. Об ускорении сходимости методов эллипсоидов (<i>Стецюк, 2001</i>)	185

К вопросу сходимости $r_\mu(\alpha)$ -алгоритма

П. И. Стецюк

Кибернетика и системный анализ. — 1995. — № 6. — С. 173–177.

В статье построен пример выпуклой кусочно-линейной функции, для которой возможно «зацикливание» $r_\mu(\alpha)$ -алгоритма. Обсуждаются модификации алгоритма, которые позволяют преодолеть эту ситуацию.

Методы с растяжением пространства в направлении разности двух последовательных субградиентов (или так называемые r -алгоритмы) зарекомендовали себя как наиболее эффективные и практически проверенные средства для решения широкого круга задач негладкой минимизации. Алгоритмы этого типа были предложены Н. З. Шором и Н. Г. Журбенко [1].

В настоящее время наиболее общим результатом о сходимости r -алгоритмов можно считать теорему, доказанную Н.З. Шором в [2]. Для класса почти дифференцируемых кусочно-гладких функций [2] она определяет достаточные условия, при которых «идеализированный» вариант r -алгоритма ($r_\mu(\alpha)$ -алгоритм) сходится к локальному минимуму. Из-за громоздкости не будем приводить эти условия, отметим лишь, что они слишком сильные и не выполняются даже для кусочно-линейных функций. Наиболее типичная ситуация, когда это происходит, связана с нарушением линейной независимости множества почти-градиентов $f(x)$ в точке $x - G_f(x)$.

В этой связи возникает вопрос: можно ли обосновать сходимость $r_\mu(\alpha)$ -алгоритма для общего случая почти дифференцируемой кусочно-гладкой функции? Оказывается, нельзя. Легко построить пример выпуклой кусочно-линейной функции, такой, что $r_\mu(\alpha)$ -алгоритм «зацикливается» в точке, которая не является оптимальной.

Для облегчения дальнейших рассуждений выделим основные операции, характеризующие итерацию $r_\mu(\alpha)$ -алгоритма в преобразованном пространстве: направление движения из текущей точки совпадает с противоположным к вычисленному в ней почти-градиенту и очередная точка выбирается из условия минимума функции по этому направлению; растяжение пространства производится с постоянным коэффициентом и в направлении разности двух последовательных почти-градиентов.

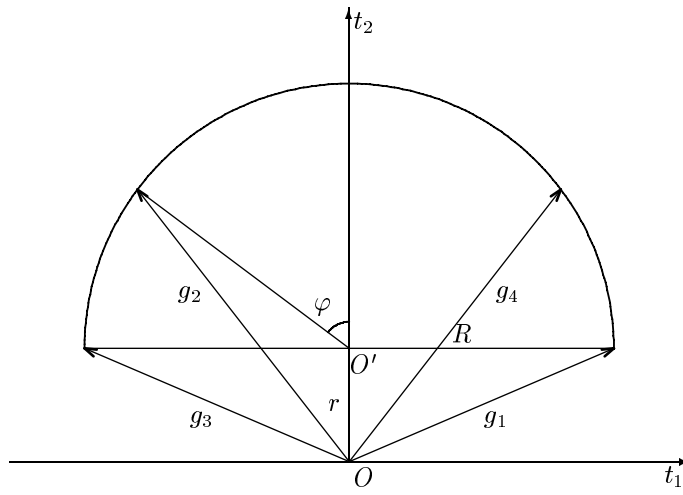


Рис. 1.

Если направление движения не является направлением убывания функции, то очередная точка совпадает с текущей, т. е. итерация алгоритма реализуется с нулевым шагом, а в качестве очередного почти-градиента выбирается тот, который образует с предыдущим неострый угол. Вообще, строго говоря, выбор очередного почти-градиента для $r_\mu(\alpha)$ -алгоритма зависит от параметра $\mu : 0 \leq \mu < 1$, но образующий с предыдущим неострый угол удовлетворяет любому значению параметра μ .

Итак, перейдем к описанию способа построения выпуклой кусочно-линейной функции двух переменных $f(x)$, $x \in E^2$, для которой $r_\mu(\alpha)$ -алгоритм можно «заикливать» в точке z , $z \in E^2$.

Пусть точка z принадлежит четырем кускам (областям гладкости функции $f(x)$), и множество почти-градиентов в этой точке $G_f = \{g_1, g_2, g_3, g_4\}$ такое, как показано на рис. 1: концы векторов лежат на полуокружности радиуса R с центром в точке O' ; расстояние $OO' = r$; почти-градиенты g_1, g_2 и g_4 симметричны относительно оси t_2 .

В терминах обозначений, приведенных на рис. 1, почти-градиенты, которые нас интересуют, следующие:

$$\begin{aligned} g_1 &= (R, r); & g_2 &= (-R \sin \varphi, r + R \cos \varphi); \\ g_3 &= (-R, r); & g_4 &= (R \sin \varphi, r + R \cos \varphi). \end{aligned}$$

Легко видеть, что растягивая пространство аргументов E^2 с одним и тем же коэффициентом α дважды: сначала по направлению $\frac{(g_2 - g_1)}{\|g_2 - g_1\|}$, а затем по $\frac{(g'_3 - g'_2)}{\|g'_3 - g'_2\|}$, где g'_3 и g'_2 – векторы g_3 и g_2 после первого растяжения, мы преобразуем его в двух ортогональных направлениях, что равносильно растяжению всего пространства. Таким образом, после этих растяжений картина почти-градиентов останется такой же, как и на рис. 1, только их длины уменьшатся в α раз.

Потребуем, чтобы почти-градиенты g_1, g_2, g_3 удовлетворяли условиям

$$(g_1, g_2) \leq 0, \quad (1)$$

$$(R_{1/\alpha}(\xi) g_2, R_{1/\alpha}(\xi) g_3) \leq 0, \quad (2)$$

где $\alpha > 1$ – коэффициент растяжения пространства, $R_{1/\alpha}(\xi)$ – оператор растяжения пространства градиентов с коэффициентом $1/\alpha$ в направлении $\xi = \frac{(g_2 - g_1)}{\|g_2 - g_1\|}$. Это значит, что $-g_1$ не является для точки z направлением убывания функции в исходном, а $-R_{1/\alpha}(\xi) g_2$ – в преобразованном пространстве, что дает возможность осуществлять итерации $r_\mu(\alpha)$ -алгоритма с нулевым шагом.

Неравенства (1)–(2) выполняются для любого коэффициента растяжения пространства $\alpha > 1$, если связь между параметрами R, r, φ удовлетворяет соотношению

$$0 \leq \frac{2r}{R} \leq \sin \varphi \leq \frac{(\alpha^2 - 1)}{(\alpha^2 + 1)} - \frac{2r}{R}.$$

Например, для $\alpha = 3$ можно выбрать следующие значения: $\sin \varphi = 0.6$, $R = 10$, $r = 1$. При этом искомая система почти-градиентов будет иметь вид: $g_1 = (10, 1)$; $g_2 = (-6, 9)$; $g_3 = (-10, 1)$; $g_4 = (6, 9)$.

Следовательно, интересующим нас примером может служить выпуклая кусочно-линейная функция

$$f(x) = \max \left\{ \max_{i=1}^4 (-g_i, x - a_1), \max_{i=1}^4 (g_i, x - a_2) \right\}, \quad (3)$$

где $a_1 = (0, -1)$, $a_2 = (0, 1)$.

Минимальное значение функция (3) принимает в точке $x^* = (0, 0)$ и $f^* = -1$. Кроме оптимальной, для нее существуют две точки

$z_1 = (0, -1)$ и $z_2 = (0, 1)$: $f(z_1) = f(z_2) = 0$, в которых нарушено условие линейной независимости семейств почти-градиентов $G_f(z_1)$ и $G_f(z_2)$. Любой из антипочти-градиентов в этих точках не является направлением убывания функции, так как существует антипочти-градиент, образующий с ним тупой угол.

Пусть $\alpha=3$, $x_0=z_2$ – начальная стартовая точка для $r_\mu(\alpha)$ -алгоритма и $g_0 = g_1$ – почти-градиент в ней. Тогда в качестве последовательных почти-градиентов по итерациям выберем последовательность $\{g_1, g_2\}$, $\{g'_2, g'_3\}$, $\{g''_3, g''_4\}$, $\{g'''_4, g'''_1\}$. Здесь «штрихами» обозначены почти-градиенты в преобразованном пространстве: одним – после первого растяжения, двумя – после двух и т. д. Такая последовательность удовлетворяет требованиям $r_\mu(\alpha)$ -алгоритма. При этом шаг на каждой итерации будет нулевым, т. е. сдвига с точки x_0 происходить не будет. Картина же почти-градиентов после четырех итераций останется такой же, как и вначале, только длины всех векторов уменьшатся в $\alpha^2 = 9$ раз.

Если этот цикл повторять, то $r_\mu(\alpha)$ -алгоритм с коэффициентом растяжения пространства $\alpha = 3$ не сможет выбраться из точки $x_0 = (0, 1)$, хотя она не является оптимальной для функции (3). Поэтому, такие точки могут служить «ловушками» для минимизирующей последовательности $r_\mu(\alpha)$ -алгоритма. Следовательно, обосновать его сходимость для общего случая почти дифференцируемых кусочно-гладких функций не представляется возможным.

Основная причина этого состоит в том, что $r_\mu(\alpha)$ -алгоритм не позволяет отличить точку оптимума от точки с линейно-зависимым множеством почти-градиентов, которая может являться «ловушкой».

Пример функции (3) фактически обнажает его основной недостаток, а именно: отсутствие в $r_\mu(\alpha)$ -алгоритме четкого критерия останова. Поэтому зададимся вопросом: каким образом можно его модифицировать, чтобы избежать ситуации «зацикливания»? Для этого в качестве критерия останова выберем необходимое и достаточное условие оптимальности для выпуклых функций — $0 \in \partial f(x)$, где $\partial f(x)$ – субдифференциал, и дополним $r_\mu(\alpha)$ -алгоритм следующим правилом: если в точке минимум по направлению реализуется при нулевом шаге, то для очередной итерации направление движения в преобразованном пространстве будем выбирать из аналога наискорейшего спуска для выпуклых функций.

При этом приведенная в [2] вычислительная схема $r_\mu(\alpha)$ -алгоритма претерпевает незначительные изменения, а именно: если $h_k = 0$, то ищем направление убывания функции из точки $y_k = A_k x_k$. Для этого

в преобразованном пространстве решаем задачу нахождения кратчайшего вектора до выпуклой оболочки почти-градиентов, накопленных в этой точке. В качестве направления движения для очередной итерации выбираем обратное к найденному кратчайшему вектору или заканчиваем работу алгоритма, если кратчайшим является нулевой вектор. Как только найдено направление убывания функции, то все накопленные почти-градиенты забываем. В противном случае в список накопленных почти-градиентов включаем тот, который образует с кратчайшим неострый угол и переходим к следующей итерации. Растяжение пространства будем производить в направлении разности последнего почти-градиента и кратчайшего вектора.

Следует отметить, что поступая таким образом, сохраняем основное предназначение операции растяжения пространства, т. е. более быстрый поиск направления убывания функции. Но при этом возникают некоторые трудности: при вычислении кратчайшего вектора не представляется возможным использовать тот факт, что очередной набор векторов отличается от предыдущего добавлением всего одного вектора; попытка ограничить число хранимых почти-градиентов приводит к проблемам, аналогичным тем, которые обсуждались для $r_\mu(\alpha)$ -алгоритма. От этих недостатков легко избавиться, если растяжение пространства производить не на каждом шаге, а только когда найдено направление убывания функции. В этом случае имеется возможность для решения задачи нахождения кратчайшего вектора применять аппарат, разработанный в [3].

Очевидно, что модифицированный одним из приведенных выше способов $r_\mu(\alpha)$ -алгоритм для почти дифференцируемых кусочно-гладких функций всегда гарантирует выход из любой точки, в которой существует направление убывания функции, и, более того, дает возможность утверждать, что получен оптимум, если кратчайшим до выпуклой оболочки почти-градиентов окажется нулевой вектор. Такая схема привлекательна и в теоретическом плане, так как вопросы сходимости при этом фактически сводятся к выяснению существования в точке направления убывания функции. Но их также, как и $r_\mu(\alpha)$ -алгоритм, можно считать «идеализированными», поскольку для реализации наискорейшего спуска с высокой точностью необходимо значительное количество вычислений значений функции и ее градиента.

Поэтому в практических приложениях одним из наиболее эффективных методов зарекомендовал себя вариант r -алгоритма с адаптивной регулировкой шага ($r(\alpha)$ -алгоритм). В отличие от $r_\mu(\alpha)$ -алгоритма, здесь шаговый множитель h_k выбирается из условия приближенного поиска

минимума по направлению и так, чтобы выполнялось условие $h_k \geq h_k^*$, где h_k^* соответствует минимуму функции по направлению. Этот способ регуляции шага подробно описан в [4] и, в основном, направлен на уменьшение общего числа вычислений значений функции. Кроме того, он позволяет преодолевать точки, которые являются «ловушками» для минимизирующей последовательности $r_\mu(\alpha)$ -алгоритма.

Литература

1. ШОР Н. З., ЖУРБЕНКО Н. Г. Метод минимизации, использующий операцию растяжения пространства в направлении разности двух последовательных градиентов // Кибернетика. — 1971. — № 3. — С. 51–59.
2. ШОР Н. З. Исследование сходимости метода градиентного типа с растяжением пространства в направлении разности двух последовательных субградиентов // Там же. — 1975. — № 4. — С. 48–53.
3. WOLFE P. Finding the nearest point in a polytope // Math. Program. — 1976. — 11, № 2. — P. 128–149.
4. ШОР Н. З. Методы минимизации недифференцируемых функций и их приложений. — К.: Наук. думка, 1979. — 199 с.

К обоснованию сходимости алгоритмов с растяжением пространства

П. И. Стецюк

Теория оптимальных решений. – 1995. – С. 4–8.

Анализируются основные причины, которые препятствуют обоснованию сходимости r -алгоритмов. Обсуждается подход, который позволяет надеяться на обоснование геометрической скорости сходимости для алгоритмов с растяжением пространства.

Для решения широкого круга задач негладкой минимизации одним из наиболее эффективных и практически проверенных средств зарекомендовали себя r -алгоритмы [1, 2]. Но теоретическое обоснование сходимости этого класса алгоритмов проведено недостаточно полно. Отметим известные результаты в этом направлении.

Наиболее общий из них [2] относится к «идеализированному» варианту r -алгоритма ($r_\mu(\alpha)$ -алгоритм) и заключается в том, что для класса почти дифференцируемых кусочно-гладких функций получены достаточные условия, при которых $r_\mu(\alpha)$ -алгоритм сходится к локальному минимуму. Но эти условия слишком сильные и не выполняются даже для кусочно-линейных функций. Приведенный в [3] пример выпуклой кусочно-линейной функции показывает, что доказать сходимость $r_\mu(\alpha)$ -алгоритма для выпуклых функций не представляется возможным.

Ряд результатов имеет место для гладких функций и связан с предельным вариантом r -алгоритма, т. е. $\beta_k = 0$, h_k находится из условия минимума $f(x)$ по направлению. Первый из них [2] заключается в том, что для квадратичной положительно определенной функции $f(x) = (Ax, x)/2$, $x \in E^n$, предельный вариант r -алгоритма сходится к точке минимума не более чем за n шагов, его можно рассматривать как проективный вариант метода сопряженных градиентов. Второй из результатов [2] утверждает, что применительно к задаче минимизации выпуклой непрерывно дифференцируемой функции $f(x)$, $x \in E^n$, предельный вариант r -алгоритма с восстановлением матрицы B_k после каждых n шагов обладает по отношению к большим циклам, состоящим из n шагов r -алгоритма, квадратичной скоростью сходимости при обычных условиях гладкости и регулярности для $f(x)$.

Все перечисленные теоретические результаты относятся к вариантам r -алгоритма, которые используют точный поиск минимума функции по направлению. Реализация этой операции с достаточно высокой точностью требует значительного количества вычислений значений $f(x)$ и ее субградиентов. Поэтому в практических приложениях используется вариант r -алгоритма с адаптивной регулировкой шага ($r(\alpha)$ -алгоритм). В отличие от $r_\mu(\alpha)$ -алгоритма здесь шаговый множитель h_k выбирается из условия приближенного поиска минимума $f(x)$ по направлению так, чтобы выполнялось условие $h_k > h_k^*$, где h_k^* соответствует минимуму функции по направлению. Этот способ регулировки шага подробно описан в [4] и, в основном, направлен на уменьшение общего количества вычислений значений функции. Так как на каждой итерации $r(\alpha)$ -алгоритма $h_k > 0$, то для него маловероятно «Зацикливание» в точках, которые являются «ловушками» для минимизирующей последовательности $r_\mu(\alpha)$ -алгоритма [3].

Итак, фактически мы имеем дело с двумя типами алгоритмов: $r_\mu(\alpha)$ -алгоритм и предельный вариант r -алгоритма более удобны в теоретическом плане, $r(\alpha)$ -алгоритм — в практических приложениях. Их объединяет, во-первых, способ выбора шагового множителя, близкий к применяемому в методе наискорейшего спуска, что обеспечивает для первых монотонность, для $r(\alpha)$ -алгоритма — в определенном смысле почти монотонность по функционалу. Во-вторых, растяжение пространства в направлении разности последовательных субградиентов значительно уменьшает угол между ними, что позволяет «расширить» конус подходящих направлений убывания функции.

Несмотря на практическую эффективность $r(\alpha)$ -алгоритма, попытки обосновать его сходимость хотя бы для частных случаев оказались безуспешными. На примере функции [3] обосновать более сильные результаты о сходимости $r_\mu(\alpha)$ -алгоритма вряд ли возможно.

В этой связи возникает вопрос: что же мешает обосновать доказательство сходимости r -алгоритмов? Причина в обоих случаях одна и та же — отсутствие у них критерия останова, который бы гарантировал достаточные условия оптимальности. На самом деле, условие $\|x_{k+1} - x_k\| < \varepsilon$, где ε — достаточно малое число, используемое для $r(\alpha)$ -алгоритма в качестве критерия останова, означает сходимость к нулю длины отрезка траектории спуска и является естественным для гладких функций. Но оно практически ничего не дает для негладких функций. Например, если для $r_\mu(\alpha)$ -алгоритма $h_k = 0$, то это еще не значит, что процесс спуска следует прекратить. В то же время, как убеждает пример функции [5], используя для $r_\mu(\alpha)$ -алгоритма только

информацию о том, что $h_k = 0$ на протяжении какого-то числа итераций, пусть даже сколь угодно большого, невозможно что-то утверждать о полученной точке.

Обсуждаемые в [3] модификации $r_\mu(\alpha)$ -алгоритма фактически были направлены на то, чтобы дополнить его критерием останова для выпуклых функций, суть которого сводится к выяснению существования в точке направления убывания функции. Но практически они обладают рядом существенных недостатков: это необходимость достаточно точного поиска минимума функции по направлению; сложность нахождения кратчайшего вектора к выпуклой оболочке векторов; проблемы с ограничением числа хранимых субградиентов и т.д. Особо следует выделить тот факт, что при таком подходе не просматривается связь между коэффициентом растяжения и геометрией линий уровня $f(x)$. Даже, если для модифицированных $r_\mu(\alpha)$ -алгоритмов и можно доказать сходимость для класса выпуклых функций, то скорее всего речь может идти о сходимости в пределе без каких-либо оценок скорости сходимости. Поэтому для обоснования сходимости алгоритмов с растяжением пространства этот путь не рационален.

Поэтому подход, основанный на внешней аппроксимации области локализации множества экстремумов эллипсоидом, представляется значительно более перспективным. Именно эта идея дала теории оптимизации предложенный независимо в [5, 6] метод эллипсоидов – уникальный в своем роде алгоритм, для которого скорость сходимости не зависит от свойств функции. Тем более, что в [6] этот метод получен как частный случай методов с растяжением пространства в направлении субградиента.

В процессе использования оператора растяжения пространства в методах субградиентного типа для решения задач негладкой оптимизации сложилась определенная интерпретация преобразования пространства. В основном она связана с самим исходным определением $R_\alpha(\xi)$, а именно с тем фактом, что, применяя $R_\beta(\xi)$ к пространству субградиентов, мы уменьшаем проекции субградиентов на направление ξ и сохраняем неизменными их проекции на ортогональные к ξ направления. В основном, обусловило интерпретацию $R_\alpha(\xi)$, как возможность «расширить» конус подходящих направлений убывания функции. Этот факт подробно проиллюстрирован в [4], с. 112 и послужил основой эффективного r -алгоритма [1].

Более естественно придерживаться немного другой интерпретации $R_\alpha(\xi)$, связанной с преобразованием некоторой выпуклой замкнутой области, которая задается в виде эллипсоида специального типа. Пусть

в $Y_k = E^n$ задан специальный эллипсоид $\text{Ell}(y_j^k, \xi_k, a, b)$. Здесь y_j^k — точка в Y , задающая центр эллипсоида, $\xi_k \in Y_k^*$ — направление, в котором длина полуоси эллипсоида равна a , b длины его полуосей во всех остальных $(n - 1)$ ортогональных в ξ_k направлениях. Для того, чтобы преобразовать его в шар, достаточно применить операцию растяжения пространства в направлении ξ_k с коэффициентом $\alpha_k = b/a$. В случае, когда $b > a$, это будет растяжением пространства аргументов, если же $b < a$ — его сжатием. При этом к пространству субградиентов будет применен оператор $R_{\beta_k}(\xi_k)$, и соответственно в первом случае получим его сжатие, во втором — растяжение.

Очевидно, что переходя в пространстве Y_k , в точку y_j^k и растягивая пространство аргументов в направлении ξ_k с коэффициентом $\alpha_k = b/a$, в преобразованном пространстве $Y_{k+1} = R_{\alpha_k}(\xi_k)Y_k$ — получим шар радиусом $r_{k+1} = b$ с центром в точке $y_j^{k+1} = R_{\alpha_k}(\xi_k)y_j^k$, который будет образом $\text{Ell}(y_j^k, \xi_k, a, b)$ в Y_k . Для методов типа отсечений этот факт позволяет обеспечить однотипность работы с информацией о полупространствах в преобразованном пространстве на каждой итерации алгоритма. Если при этом задаться целью построения этих эллипсоидов для локализации области экстремумов так, чтобы гарантировать уменьшение их объема от итерации к итерации, автоматически получим для выпуклых задач в рамках методов отсечений с растяжением пространства сходящийся со скоростью геометрической прогрессии алгоритм в смысле сходимости рекордов по функции.

Для построения таких эллипсоидов в качестве более тонких локализаторов множества экстремумов подходят такие выпуклые тела в E^n , как шаровой слой и n -мерный круговой конус, построению которых следует уделить основное внимание. Для этих целей можно использовать, такие выпуклые тела в E^n , как n -мерный «угол» ($m \leq n$) и симплекс, которые позволяют эффективно решать эти задачи.

Для алгоритмов с растяжением пространства такого рода интерпретация представляется достаточно негладкой и позволяет связать для них шаговый множитель и коэффициент растяжения пространства с такими характеристиками выпуклых функций, как геометрия линий уровня $f(x)$ и геометрия субдифференциала в точке экстремума. Кроме того, эти алгоритмы применимы для более широкого класса задач, чем задачи выпуклого программирования, например: задача отыскания седловых точек выпукло-вогнутых функций; частные случаи задач решения вариационных неравенств; специальные классы задач линейной и нелинейной дополнителности.

Литература

1. ШОР Н. З., ЖУРБЕНКО Н. Г. Метод минимизации, использующий операцию растяжения пространства в направлении разности двух последовательных градиентов // Кибернетика. – 1971. – № 3. – С. 51–59.
2. ШОР Н. З. Методы минимизации недифференцируемых функций и их приложения – К.: Наук. думка, 1979. – 199 с.
3. СТЕЦЮК П. И. К вопросу сходимости r -алгоритмов // Кибернетика и системный анализ. – 1995. – № 3. – С. 141–147.
4. МИХАЛЕВИЧ В. С., ТРУБИН В. А., ШОР Н. З. Оптимизационные задачи производственно-транспортного планирования. – М.: Наука, 1986. – 264 с.
5. ЮДИН Д. Б. НЕМИРОВСКИЙ А. С. Информационная сложность и эффективные методы решения выпуклых экстремальных задач // Экономика и мат. методы. – 1976. – **12**, вып. 2. – С. 357–359.
6. ШОР Н. З. Метод отсечения с растяжением пространства для решения задач выпуклого программирования // Кибернетика. – 1977. – № 1. – С. 94–95.

r -алгоритмы и эллипсоиды

П. И. Стецюк

Кибернетика и системный анализ. – 1996. – № 1. – С. 113–134.

Рассматривается преобразование пространства, которое допускает внешнюю аппроксимацию множества экстремумов эллипсоидами специального типа с гарантированным уменьшением их объема. Основываясь на нем, для задачи минимизации выпуклой функции описаны два алгоритма фейеровского типа с преобразованием пространства, идейно близких к r -алгоритмам. Приведены численные эксперименты, подтверждающие эффективность этих алгоритмов, и обсуждаются пути их усовершенствования.

Об r -алгоритмах. Для решения задач негладкой минимизации одним из наиболее эффективных и практически проверенных средств рекомендовали себя r -алгоритмы [1, 2]. Но теоретическое обоснование сходимости этого класса алгоритмов проведено недостаточно полно. Отметим известные результаты в этом направлении.

Наиболее общий из них [2] относится к «идеализированному» варианту r -алгоритма ($r_\mu(\alpha)$ -алгоритм) и заключается в том, что для класса почти дифференцируемых кусочно-гладких функций получены достаточные условия, при которых $r_\mu(\alpha)$ -алгоритм сходится к локальному минимуму. Но эти условия слишком сильные и не выполняются даже для кусочно-линейных функций. Приведенный в [3] пример выпуклой кусочно-линейной функции показывает, что доказать сходимость $r_\mu(\alpha)$ -алгоритма для выпуклых функций не представляется возможным.

Ряд результатов имеет место для гладких функций и связан с предельным вариантом $r(\alpha)$ -алгоритма, т. е. $\beta_k = 0$ и h_k находится из условия минимума $f(x)$ по направлению. Первый из них [2] заключается в том, что для квадратичной положительно определенной функции $f(x) = (Ax, x)/2$, $x \in E^n$, предельный вариант $r(\alpha)$ -алгоритма сходится к точке минимума не более чем за n шагов. Его можно рассматривать как проективный вариант метода сопряженных градиентов. Второй из результатов [2] утверждает, что применительно к задаче минимизации выпуклой непрерывно дифференцируемой функции $f(x)$, $x \in E^n$, предельный вариант r -алгоритма с восстановлением матрицы B_k , после каждых n шагов обладает по отношению к большим циклам, состоящим из n шагов r -алгоритма, квадратичной скоростью сходимости при обычных условиях гладкости и регулярности для $f(x)$.

Все теоретические результаты о сходимости r -алгоритмов относятся к тем вариантам r -алгоритма, которые используют точный поиск минимума функции по направлению. Реализация этой операции с достаточно высокой точностью требует значительного количества вычислений значений $f(x)$ и ее субградиентов, поэтому в практических приложениях используется вариант r -алгоритма с адаптивной регулировкой шага ($r(\alpha)$ -алгоритм). В отличие от $r_\mu(\alpha)$ -алгоритма здесь шаговый множитель h_k выбирается из условия приближенного поиска минимума по направлению и так, чтобы выполнялось условие $h_k > h_k^*$, где h_k^* , соответствует минимуму функции по направлению. Этот способ регулировки шага подробно описан в [4] и, в основном, направлен на уменьшение общего числа вычислений $f(x)$ и $\partial f(x)$ с тем, чтобы на одну итерацию $r(\alpha)$ -алгоритма приходилось не более 2–3 вычислений $f(x)$ и $\partial f(x)$.

Итак, фактически мы имеем дело с двумя типами алгоритмов: $r_\mu(\alpha)$ -алгоритм и предельный вариант r -алгоритма более удобны в теоретическом плане, $r(\alpha)$ -алгоритм — в практических приложениях. Их объединяет, во-первых, способ выбора шагового множителя, близкий к применяемому в методе наискорейшего спуска, что обеспечивает для первых монотонность, для $r(\alpha)$ -алгоритма в определенном смысле почти монотонность по функционалу. Во-вторых, растяжение пространства в направлении разности последовательных субградиентов значительно уменьшает угол между ними, что позволяет «расширить» конус подходящих направлений убывания функции.

Несмотря на практическую эффективность $r(\alpha)$ -алгоритма, попытки обосновать его сходимость хотя бы для частных случаев оказались безуспешными. При этом основная трудность обусловлена тем, что адаптивная регулировка шагового множителя приводит к более «смелому» шагу по направлению, чем если шаг выбирать из условия минимума $f(x)$ по этому же направлению. Такого рода шаги объясняет подход [5], основанный на внешней аппроксимации области локализации множества экстремумов эллипсоидами специального типа, которые посредством однорангового оператора растяжения пространства можно преобразовать в шар. Но они связаны с переходом в центр тяжести очередного локализатора, и при этом теряется интерпретация метода как градиентного.

Желательно было бы получить метод субградиентного типа с растяжением пространства, идейно близкий к $r(\alpha)$ -алгоритму в том смысле, чтобы операция растяжения пространства позволяла «расширить» конус подходящих направлений убывания функции, и для которого можно было бы легко доказать сходимость для выпуклых функций, пусть

даже с некоторыми ограничениями. Оказывается, что такой алгоритм легко построить для задачи минимизации выпуклой функции при условии, что известно минимальное значение $f(x)$. При этом он сходится с любого начального приближения, не содержит никаких произвольных параметров и автоматически выбирает коэффициенты растяжения пространства, которые согласуются с геометрией линий уровня $f(x)$. Прежде чем переходить к алгоритму, объясним главную идею, в основе которой лежат простые геометрические рассуждения.

Об одном эллипсоиде минимального объема. Пусть имеется выпуклое тело в E^n , обозначим его $W(x_0, g_1, g_2, R)$, полученное в результате пересечения шара $S(x_0, R) = \{x : \|x - x_0\| \leq R\}$, и двух полупространств $P_1 = \{x : (x - x_0, g_1) \leq 0\}$ и $P_2 = \{x : (x - x_0, g_2) \leq 0\}$, и при этом $(g_1, g_2) < 0$. Проекция $W(x_0, g_1, g_2, R)$ на плоскость, определяемую g_1 и g_2 , представлена на рис.1. Здесь $(\xi_1, g_1) = 0$, $\|\xi_1\| = 1$, $(\xi_2, g_2) = 0$, $\|\xi_2\| = 1$, $(\xi_1, \xi_2) = -\left(\frac{g_1}{\|g_1\|}, \frac{g_2}{\|g_2\|}\right)$. Такая картина типична для негладких функций, когда в некоторой точке x_k имеется два субградиента и ни один из них не является направлением убывания функции из точки x_k .

Среди семейства эллипсоидов с центром в точке x_0 и таких, чтобы одна из осей совпадала с направлением $\eta_1 = \frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|}$, вторая – ортогональным к η_1 направлением $\eta_2 = \frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|}$, рассмотрим эллипсоид оптимального объема, содержащий тело $W(x_0, g_1, g_2, R)$. В системе координат, для которой начало совпадает с центром шара x_0 , ось x_1 – с направлением η_1 , ось x_2 – с направлением η_2 , а оси x_3, \dots, x_n образуют совместно с x_1 и x_2 взаимно-ортогональную систему, поверхность семейства этих эллипсоидов удовлетворяет следующему уравнению:

$$\frac{x_1^2}{a^2} + \frac{x_2^2}{b^2} + \sum_{i=3}^n \frac{x_i^2}{c_i^2} = 1. \quad (1)$$

Здесь a и b – длины полуосей эллипсоида в направлениях η_1 и η_2 соответственно, c_i – длины полуосей в остальных $(n - 2)$ ортогональных к η_1 и η_2 направлениях. Из (1) и условия, что эллипсоид должен проходить через точки с координатами Re_i^T , $i = 3, \dots, n$, где e_i – вектор, i -я компонента которого равна единице, а все остальные равны нулю, следует, что $c_i = R$, $i = 3, \dots, n$, поэтому искомый эллипсоид обозначим

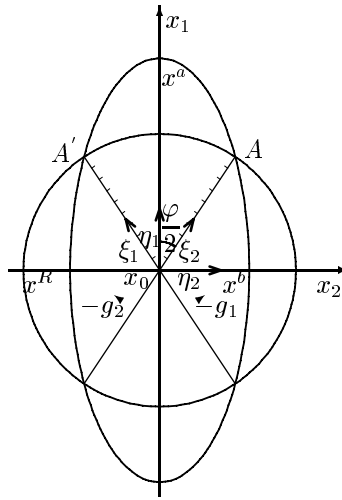


Рис. 1. Эллипсоид типа $\text{ell}(x_0, a, b, c)$. Здесь $\|x^R - x_0\| = R$, $\|x^a - x_0\| = a$, $\|x^b - x_0\| = b$.

$\text{ell}(x_0, a, b, c)$. Длины полуосей a и b связаны прохождением эллипсоида через точки A и A' , координаты которых $\left(R \cos \frac{\varphi}{2}, R \sin \frac{\varphi}{2}, 0, \dots, 0\right)$ и $\left(R \cos \frac{\varphi}{2}, -R \sin \frac{\varphi}{2}, 0, \dots, 0\right)$ и должны удовлетворять соотношению

$$\frac{\left(R \cos \frac{\varphi}{2}\right)^2}{a^2} + \frac{\left(R \sin \frac{\varphi}{2}\right)^2}{b^2} = 1,$$

откуда следует, что

$$a = \frac{bR \cos \frac{\varphi}{2}}{\sqrt{b^2 - \left(R \sin \frac{\varphi}{2}\right)^2}}.$$

Отношение объема $\text{ell}(x_0, a, b, c)$ к объему исходного шара $S(x_0, R)$ задается

$$\frac{\text{vol}(\text{ell}(x_0, a, b, c))}{\text{vol}(S(x_0, R))} = \frac{v_0 ab R^{n-2}}{v_0 R^n} = \frac{ab}{R^2} = \frac{\left(\frac{b}{R}\right)^2 \cos \frac{\varphi}{2}}{\sqrt{\left(\frac{b}{R}\right)^2 - \sin^2 \frac{\varphi}{2}}} = q(t).$$

Здесь $t = \frac{b}{R}$, $v_0 = \frac{\pi^{n/2}}{\Gamma(\frac{n}{2} + 1)}$ – объем единичного шара в E^n ,

$\Gamma(z)$ – гамма-функция Эйлера.

Приравняв к нулю производную $\frac{d}{dt}q(t)$, получим оптимальное значение параметра $t^* = \sqrt{2} \sin \frac{\varphi}{2}$, при котором гарантируется минимальный объем $\text{ell}(x_0, a, b, c)$. Длины его полуосей будут:

$$\begin{aligned} b = \sqrt{2}R \sin \frac{\varphi}{2} &= \sqrt{2}R \sqrt{\frac{1 - \cos \varphi}{2}} = R \sqrt{1 - (\xi_1, \xi_2)} = \\ &= R \sqrt{1 + \left(\frac{g_1}{\|g_1\|}, \frac{g_2}{\|g_2\|} \right)}, \end{aligned} \quad (2)$$

$$\begin{aligned} a = \sqrt{2}R \cos \frac{\varphi}{2} &= \sqrt{2}R \sqrt{\frac{1 + \cos \varphi}{2}} = R \sqrt{1 + (\xi_1, \xi_2)} = \\ &= R \sqrt{1 - \left(\frac{g_1}{\|g_1\|}, \frac{g_2}{\|g_2\|} \right)}, \end{aligned} \quad (3)$$

$$c_i = R, \quad i = 3, \dots, n. \quad (4)$$

При этом

$$\frac{\text{vol}(\text{ell}(x_0, a, b, c))}{\text{vol}(S(x_0, R))} = \sqrt{1 - \left(\frac{g_1}{\|g_1\|}, \frac{g_2}{\|g_2\|} \right)^2} < 1. \quad (5)$$

Следовательно, объем эллипсоида $\text{ell}(x_0, a, b, c)$ с длинами полуосей, вычисленными согласно (2)–(4), меньше, чем объем шара $S(x_0, R)$, и константа уменьшения объема (5) зависит от угла между векторами g_1 и g_2 . Для того, чтобы преобразовать его в новый шар радиуса R , достаточно «растянуть» пространство аргументов в направлении

$$\eta_1 = \frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|} \quad \text{с коэффициентом} \quad a_1 = \frac{1}{\sqrt{1 - \left(\frac{g_1}{\|g_1\|}, \frac{g_2}{\|g_2\|} \right)}}$$

и в направлении

$$\eta_2 = \frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|} \quad \text{с коэффициентом} \quad a_2 = \frac{1}{\sqrt{1 + \left(\frac{g_1}{\|g_1\|}, \frac{g_2}{\|g_2\|} \right)}}.$$

Но направления η_1 и η_2 совпадают с точностью до некоторого множителя с направлениями $\left(\frac{g_1}{\|g_1\|} + \frac{g_2}{\|g_2\|} \right)$ и $\left(\frac{g_1}{\|g_1\|} - \frac{g_2}{\|g_2\|} \right)$ соответственно.

Пусть g_1 и g_2 – субградиенты $f(x)$ в точке x_0 и $(g_1, g_2) < 0$. Преобразование эллипсоида $\text{ell}(x_0, a, b, c)$ в шар приводит к растяжению пространства аргументов в направлении разности нормированных субградиентов, т. к. $\alpha_2 > 1$, и к его «сжатию» в направлении суммы нормированных субградиентов, поскольку $\alpha_1 < 1$, которое ортогонально к предыдущему направлению, что фактически только усиливает растяжение пространства аргументов в направлении разности нормированных субградиентов. Таким образом, приходим к растяжению пространства, близкому к применяемому в $r(\alpha)$ -алгоритмах, в том смысле, что оно позволяет «расширить» конус подходящих направлений убывания $f(x)$, но само направление растяжения пространства может сильно отличаться от разности субградиентов, особенно если $\|g_1\| \gg \|g_2\|$ или $\|g_2\| \gg \|g_1\|$.

Итак, требуемое преобразование пространства задается посредством применения двух операторов растяжения пространства [2]:

$$\begin{aligned} T(\xi_1, \xi_2) &= R_{\alpha_1} \left(\frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|} \right) R_{\alpha_2} \left(\frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|} \right) = \\ &= R_{\alpha_2} \left(\frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|} \right) R_{\alpha_1} \left(\frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|} \right), \end{aligned} \quad (6)$$

где

$$\|\xi_1\| = \|\xi_2\| = 1, \quad (\xi_1, \xi_2)^2 < 1, \quad \alpha_1 = \frac{1}{\sqrt{1 - (\xi_1, \xi_2)}}, \quad \alpha_2 = \frac{1}{\sqrt{1 + (\xi_1, \xi_2)}},$$

$R_\alpha(\xi) = I + (\alpha - 1)\xi\xi^T$ – оператор растяжения пространства в направлении ξ ($\|\xi\| = 1$) с коэффициентом α . Перестановочность операторов растяжения пространства в (6) обеспечивается за счет ортогональности направлений растяжения $\frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|}$ и $\frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|}$. Этим свойством часто будем пользоваться в дальнейшем.

Обратное к $T(\xi_1, \xi_2)$ преобразование задается

$$\begin{aligned} T^{-1}(\xi_1, \xi_2) &= R_{\beta_1} \left(\frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|} \right) R_{\beta_2} \left(\frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|} \right) = \\ &= R_{\beta_2} \left(\frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|} \right) R_{\beta_1} \left(\frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|} \right), \end{aligned} \quad (7)$$

$$\text{где} \quad \beta_1 = \frac{1}{\alpha_1} = \sqrt{1 - (\xi_1, \xi_2)}, \quad \beta_2 = \frac{1}{\alpha_2} = \sqrt{1 + (\xi_1, \xi_2)}.$$

Отметим некоторые свойства преобразования (6), (7).

Лемма 1. Пусть B_k – невырожденная матрица размера $n \times n$, p_1, p_2 – n -мерные векторы, такие, что $\left(\frac{B_k^* p_1}{\|B_k^* p_1\|}, \frac{B_k^* p_2}{\|B_k^* p_2\|}\right)^2 < 1$, и пусть

$$B_{k+1} = B_k R_{\beta_1} \left(\frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|} \right) R_{\beta_2} \left(\frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|} \right), \quad (8)$$

где $\xi_1 = \frac{B_k^* p_1}{\|B_k^* p_1\|}$, $\xi_2 = \frac{B_k^* p_2}{\|B_k^* p_2\|}$, $\beta_1 = \sqrt{1 - (\xi_1, \xi_2)}$, $\beta_2 = \sqrt{1 + (\xi_1, \xi_2)}$.

Тогда матрица B_{k+1} невырождена и, кроме того, выполнены условия:

$$\mathbf{a)} \det(B_{k+1}) = \det(B_k) \sqrt{1 - (\xi_1, \xi_2)^2}; \quad \mathbf{б)} (B_{k+1}^* p_1, B_{k+1}^* p_2) = 0.$$

Доказательство. Невырожденность матрицы B_{k+1} и условие **а)** следуют из

$$\begin{aligned} \det(B_{k+1}) &= \det(B_k) \det \left(R_{\beta_1} \left(\frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|} \right) \right) \det \left(R_{\beta_2} \left(\frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|} \right) \right) = \\ &= \det(B_k) \beta_1 \beta_2 = \det(B_k) \sqrt{1 - (\xi_1, \xi_2)} \sqrt{1 + (\xi_1, \xi_2)} = \\ &= \det(B_k) \sqrt{1 - (\xi_1, \xi_2)^2} \neq 0. \end{aligned}$$

Для доказательства условия **б)**, используя свойство оператора растяжения пространства $R_\beta(\xi) = R_\beta^*(\xi)$ и перестановочность операторов $R_{\beta_1} \left(\frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|} \right)$ и $R_{\beta_2} \left(\frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|} \right)$, получаем

$$\begin{aligned} (B_{k+1}^* p_1, B_{k+1}^* p_2) &= \left(R_{\beta_1} \left(\frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|} \right) R_{\beta_2} \left(\frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|} \right) B_k^* p_1, \right. \\ &\quad \left. R_{\beta_1} \left(\frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|} \right) R_{\beta_2} \left(\frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|} \right) B_k^* p_2 \right) = \\ &= \left(B_k^* p_1, R_{\beta_1^2} \left(\frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|} \right) R_{\beta_2^2} \left(\frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|} \right) B_k^* p_2 \right) = \\ &= \left(B_k^* p_1, \left(I + (\beta_1^2 - 1) \left(\frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|} \right) \left(\frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|} \right)^T + \right. \right. \\ &\quad \left. \left. + (\beta_2^2 - 1) \left(\frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|} \right) \left(\frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|} \right)^T \right) B_k^* p_2 \right) = \end{aligned}$$

$$\begin{aligned}
&= \frac{1}{\|B_k^* p_2\|} \left(B_k^* p_1, \left(I - \frac{(\xi_1, \xi_2)}{2(1 + (\xi_1, \xi_2))} (\xi_1 + \xi_2)(\xi_1 + \xi_2)^T + \right. \right. \\
&\quad \left. \left. + \frac{(\xi_1, \xi_2)}{2(1 - (\xi_1, \xi_2))} (\xi_1 - \xi_2)(\xi_1 - \xi_2)^T \right) \xi_2 \right) = \\
&= \frac{1}{\|B_k^* p_2\|} \left(B_k^* p_1, \xi_2 - \frac{(\xi_1, \xi_2)}{2} (\xi_1 + \xi_2) - \frac{(\xi_1, \xi_2)}{2} (\xi_1 - \xi_2) \right) = \\
&= \frac{1}{\|B_k^* p_2\| \|B_k^* p_1\|} \left(\xi_1, \xi_2 - (\xi_1, \xi_2) \xi_1 \right) = \\
&= \frac{1}{\|B_k^* p_2\| \|B_k^* p_1\|} \left((\xi_1, \xi_2) - (\xi_1, \xi_2) \right) = 0.
\end{aligned}$$

Условие а) леммы 1 фактически отражает уменьшение объема эллипсоида типа $\text{ell}(x_0, a, b, c)$ с длинами полуосей, вычисленными согласно (2)–(4). Объясним геометрический смысл условия б). Пусть $g_f(x)$ – субградиент $f(x)$, тогда $g_{\varphi_k}(x) = B_k^* g_f(x)$ – субградиент от функции $\varphi_k(y) = f(B_k y)$, которая получается из $f(x)$ при линейном преобразовании пространства $y = A_k x$, где $A_k = B_k^{-1}$. Поэтому, если p_1 и p_2 – субградиенты $f(x)$ в точке x_k , то преобразование пространства $y = A_{k+1} x$, где $A_{k+1} = B_{k+1}^{-1}$, и B_{k+1} вычислена согласно (8), фактически ортогонализует их прообразы $B_{k+1}^* p_1$ и $B_{k+1}^* p_2$ для функции $\varphi_{k+1}(y) = f(B_{k+1} y)$.

При этом условие $\left(\frac{B_k^* p_1}{\|B_k^* p_1\|}, \frac{B_k^* p_2}{\|B_k^* p_2\|} \right)^2 < 1$ требуется для того, чтобы ортогонализация была возможной. Отметим, что здесь не используется $(B_k^* p_1, B_k^* p_2) \leq 0$, но в дальнейшем будем рассматривать только ситуации, когда угол между $B_k^* p_1$ и $B_k^* p_2$ тупой, так как при этом обеспечивается внешняя аппроксимация множества экстремумов эллипсоидом типа $\text{ell}(x_0, a, b, c)$.

В вычислительном отношении преобразование (6) требует в два раза больше арифметических операций, чем растяжение пространства в направлении разности двух последовательных субградиентов. Но оно обладает рядом преимуществ по отношению к нему. Во-первых, учитывает угол между субградиентами, что обеспечивает переменный коэффициент растяжения пространства. Во-вторых, гарантирует уменьшение объема области локализации множества экстремумов, что позволяет доказывать сходимость алгоритмов в смысле рекордов минимизируемой функции.

Если мы имеем для выпуклой функции $f(x)$ механизм нахождения в точке субградиентов, образующих тупой угол (в качестве такового может служить процедура достаточно точного спуска по направлению), то, не сдвигаясь из точки, можем преобразовывать линии уровня $f(x)$, гарантируя при этом уменьшение объема области локализации множества экстремумов таким образом, чтобы ускорить поиск подходящего направления убывания $f(x)$. Этот факт позволяет обеспечить выход из «угловой» точки для почти дифференцируемых функций, обсуждаемый в [3], посредством применения преобразования на каждом шаге метода, ограничившись при этом запоминанием не более чем $(n + 1)$ почти-градиентов.

Рассмотрим некоторые применения преобразования (6) в методах субградиентного типа.

Методы фейеровского типа с растяжением пространства.
Для задачи

$$\min f(x), \quad x \in E^n, \quad (9)$$

где $f(x)$ – выпуклая функция на E^n , множество точек минимума X^* которой непусто (для удобства дальнейших рассуждений предположим, что множество X^* состоит из единственной точки x^*) и f^* – минимальное значение $f(x)$, из приведенных выше соображений приходим к следующему алгоритму субградиентного типа с растяжением пространства.

Перед началом вычислений имеем $x_0 \in E^n$, $f(x_0)$, $g(x_0) = \partial f(x_0)$, $B_0 = I$ – единичная матрица размера $n \times n$. Пусть на k -м шаге получены $x_k \in E^n$, $f(x_k)$, $g(x_k) = \partial f(x_k)$, B_k – матрица $n \times n$. Тогда, если $f(x_k) = f^*$, то x_k – искомая точка. В противном случае переходим к $(k + 1)$ -му шагу

1) Вычислим очередное приближение

$$x_{k+1} = x_k - \frac{f(x_k) - f^*}{\|B_k^* g(x_k)\|} B_k \frac{B_k^* g(x_k)}{\|B_k^* g(x_k)\|} \quad (10)$$

2) Вычислим $f(x_{k+1})$, $g(x_{k+1}) = \partial f(x_{k+1})$.

3) Положим

$$\xi_1 = \frac{B_k^* g(x_k)}{\|B_k^* g(x_k)\|}, \quad \xi_2 = \frac{B_k^* g(x_{k+1})}{\|B_k^* g(x_{k+1})\|}. \quad (11)$$

- 4) Полагаем $B_{k+1} = B_k$ и переходим к 5, если $(\xi_1, \xi_2) \geq 0$, иначе вычисляем

$$\eta_1 = \frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|}, \quad \eta_2 = \frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|}, \quad (12)$$

$$\beta_1 = \sqrt{1 - (\xi_1, \xi_2)}, \quad \beta_2 = \sqrt{1 + (\xi_1, \xi_2)}, \quad (13)$$

$$B_{k+1} = B_k R_{\beta_1}(\eta_1) R_{\beta_2}(\eta_2), \quad (14)$$

здесь $R_\beta(\eta) = I + (\beta - 1)\eta\eta^T$, $\|\eta\| = 1$ и переходим к 5.

- 5) Переходим к очередному шагу с x_{k+1} , B_{k+1} , $g(x_{k+1})$, $f(x_{k+1})$.

Справедлива следующая теорема.

Теорема 1. Последовательность $\{x_{k+1}\}_{k=0}^\infty$, генерируемая алгоритмом (10)–(14), удовлетворяет неравенству

$$\|A_{k+1}(x_{k+1} - x^*)\|^2 \leq \|A_k(x_k - x^*)\|^2 - \frac{(f(x_k) - f^*)^2}{\|B_k^*g(x_k)\|^2}, \quad k = 0, 1, 2, \dots \quad (15)$$

Здесь $A_k = B_k^{-1}$, $A_{k+1} = B_{k+1}^{-1}$.

Доказательство. Доказательство проведем для случаев $(\xi_1, \xi_2) \geq 0$ и $(\xi_1, \xi_2) < 0$ отдельно.

1. $(\xi_1, \xi_2) \geq 0$. Тогда $A_{k+1} = A_k$ и справедливость (15) следует из

$$\begin{aligned} & \|A_{k+1}(x_{k+1} - x^*)\|^2 = \|A_k(x_{k+1} - x^*)\|^2 = \\ & = \|A_k\left(x_k - x^* - \frac{(f(x_k) - f^*)}{\|B_k^*g(x_k)\|} B_k \frac{B_k^*g(x_k)}{\|B_k^*g(x_k)\|}\right)\|^2 = \\ & = \|A_k(x_k - x^*) - \frac{(f(x_k) - f^*)}{\|B_k^*g(x_k)\|^2} B_k^*g(x_k)\|^2 = \\ & = \|A_k(x_k - x^*)\|^2 - \\ & \quad - 2 \frac{(f(x_k) - f^*)}{\|B_k^*g(x_k)\|^2} (A_k(x_k - x^*), B_k^*g(x_k)) + \frac{(f(x_k) - f^*)^2}{\|B_k^*g(x_k)\|^2} = \\ & = \|A_k(x_k - x^*)\|^2 - 2 \frac{(f(x_k) - f^*)}{\|B_k^*g(x_k)\|^2} (x_k - x^*, g(x_k)) + \frac{(f(x_k) - f^*)^2}{\|B_k^*g(x_k)\|^2} \leq \\ & \leq \|A_k(x_k - x^*)\|^2 - \frac{(f(x_k) - f^*)^2}{\|B_k^*g(x_k)\|^2}. \end{aligned} \quad (16)$$

Неравенство в (16) следует из того, что в силу выпуклости $f(x)$ справедливо $\left(x_k - x^*, g(x_k)\right) \geq f(x_k) - f^* \geq 0$.

2. $(\xi_1, \xi_2) < 0$. Прежде, чем переходить к доказательству, упростим оператор $R_{\alpha_1^2}(\eta_1)R_{\alpha_2^2}(\eta_2)$, $\left(\alpha_1 = 1/\beta_1, \alpha_2 = \frac{1}{\beta_2}\right)$, учитывая, что

$$\alpha_1^2 = \frac{1}{1 - (\xi_1, \xi_2)}, \quad \alpha_2^2 = \frac{1}{1 + (\xi_1, \xi_2)}, \quad \eta_1 = \frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|}, \quad \eta_2 = \frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|} :$$

$$\begin{aligned} R_{\alpha_1^2}(\eta_1)R_{\alpha_2^2}(\eta_2) &= \left(I + (\alpha_1^2 - 1)\eta_1\eta_1^T\right)\left(I + (\alpha_2^2 - 1)\eta_2\eta_2^T\right) = \\ &= I + (\alpha_1^2 - 1)\eta_1\eta_1^T + (\alpha_2^2 - 1)\eta_2\eta_2^T = \\ &= I + \frac{(\xi_1, \xi_2)}{1 - (\xi_1, \xi_2)}\left(\frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|}\right)\left(\frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|}\right)^T - \\ &\quad - \frac{(\xi_1, \xi_2)}{1 + (\xi_1, \xi_2)}\left(\frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|}\right)\left(\frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|}\right)^T = \\ &= I + \frac{(\xi_1, \xi_2)}{2(1 - (\xi_1, \xi_2)^2)}\left(\left(\xi_1 + \xi_2\right)\left(\xi_1 + \xi_2\right)^T - \left(\xi_1 - \xi_2\right)\left(\xi_1 - \xi_2\right)^T\right) = \\ &= I + \frac{(\xi_1, \xi_2)}{1 - (\xi_1, \xi_2)^2}\left(\xi_1\xi_2^T + \xi_2\xi_1^T\right). \end{aligned}$$

Учитывая, что $\xi_1 = \frac{B_k^*g(x_k)}{\|B_k^*g(x_k)\|}$ и $\xi_2 = \frac{B_k^*g(x_{k+1})}{\|B_k^*g(x_{k+1})\|}$, имеем

$$\begin{aligned} &\|A_{k+1}(x_{k+1} - x^*)\|^2 = \\ &= \left(R_{\alpha_1}(\eta_1)R_{\alpha_2}(\eta_2)A_k(x_{k+1} - x^*), R_{\alpha_1}(\eta_1)R_{\alpha_2}(\eta_2)A_k(x_{k+1} - x^*)\right) = \\ &= \left(A_k(x_{k+1} - x^*), R_{\alpha_1^2}(\eta_1)R_{\alpha_2^2}(\eta_2)A_k(x_{k+1} - x^*)\right) = \\ &= \left(A_k(x_{k+1} - x^*), \left(I + \frac{(\xi_1, \xi_2)}{1 - (\xi_1, \xi_2)^2}(\xi_1\xi_2^T + \xi_2\xi_1^T)\right)A_k(x_{k+1} - x^*)\right) = \\ &= \|A_k(x_{k+1} - x^*)\|^2 + \\ &\quad + \frac{2(\xi_1, \xi_2)}{1 - (\xi_1, \xi_2)^2}\left(\xi_1, A_k(x_{k+1} - x^*)\right)\left(\xi_2, A_k(x_{k+1} - x^*)\right) = \\ &= \|A_k(x_{k+1} - x^*)\|^2 + \\ &\quad + \frac{2(\xi_1, \xi_2)}{1 - (\xi_1, \xi_2)^2}\frac{\left(g(x_k), x_{k+1} - x^*\right)}{\|B_k^*g(x_k)\|}\frac{\left(g(x_{k+1}), x_{k+1} - x^*\right)}{\|B_k^*g(x_{k+1})\|}. \quad (17) \end{aligned}$$

Оценим оба слагаемых в (17). Первое из них допускает оценку аналогично, как в 1:

$$\|A_k(x_{k+1} - x^*)\|^2 \leq \|A_k(x_k - x^*)\|^2 - \frac{(f(x_k) - f^*)^2}{\|B_k^*g(x_k)\|^2}. \quad (18)$$

Для второго оценим все три его сомножителя. Первый из них

$$\frac{2(\xi_1, \xi_2)}{(1 - (\xi_1, \xi_2)^2)\|B_k^*g(x_k)\|\|B_k^*g(x_{k+1})\|} < 0, \quad (19)$$

так как $(\xi_1, \xi_2) < 0$, $\|\xi_1\| = 1$, $\|\xi_2\| = 1$. Для двух других, учитывая выпуклость $f(x)$, получаем

$$\begin{aligned} & (g(x_k), x_{k+1} - x^*) = \\ &= \left(g(x_k), x_k - x^* - \frac{(f(x_k) - f^*)}{\|B_k g(x_k)\|} B_k \frac{B_k^* g(x_k)}{\|B_k^* g(x_k)\|} \right) = \\ &= \left(g(x_k), x_k - x^* \right) - \frac{(f(x_k) - f^*)}{\|B_k^* g(x_k)\|^2} \left(g(x_k), B_k B_k^* g(x_k) \right) = \\ &= \left(g(x_k), x_k - x^* \right) - \frac{(f(x_k) - f^*)}{\|B_k^* g(x_k)\|^2} \left(B_k^* g(x_k), B_k^* g(x_k) \right) = \\ &= \left(g(x_k), x_k - x^* \right) - (f(x_k) - f^*) \geq 0; \end{aligned} \quad (20)$$

$$(g(x_{k+1}), x_{k+1} - x^*) \geq f(x_{k+1}) - f^* \geq 0. \quad (21)$$

Из (17), учитывая (18)–(21), следует

$$\|A_{k+1}(x_{k+1} - x^*)\|^2 \leq \|A_k(x_k - x^*)\|^2 - \frac{(f(x_k) - f^*)^2}{\|B_k^*g(x_k)\|^2},$$

что завершает доказательство теоремы.

Алгоритм (10)–(14) обладает двумя характерными чертами r -алгоритмов. Во-первых, движение из точки осуществляется в направлении субградиента, и, во-вторых, растяжение пространства производится в направлении разности нормированных последовательных субградиентов $B_k^*g(x_k)$ и $B_k^*g(x_{k+1})$ при условии, что угол φ_k между ними тупой, т. е. $\cos \varphi_k < 0$, и позволяет «расширить» конус подходящих направлений убывания $f(x)$. Поэтому его можно считать одним из вариантов r -алгоритмов с переменным коэффициентом растяжения пространства

и классической фейеровской регуляровкой шагового множителя [6]. При этом коэффициент растяжения пространства α_k в направлении разности нормированных последовательных субградиентов с учетом того, что «сжатие» в ортогональном направлении только усиливает α_k , вычисляется

$$\alpha_k = \frac{\sqrt{1 - \cos \varphi_k}}{\sqrt{1 + \cos \varphi_k}} = \sqrt{1 - 2 \frac{\cos \varphi_k}{1 + \cos \varphi_k}} > 1, \quad (22)$$

так как $\cos \varphi_k < 0$.

В результате работы алгоритма (10)–(14) на k -м шаге имеем точку x_k , для которой справедливо

$$\|A_k(x_k - x^*)\|^2 \leq \|x_0 - x^*\|^2 - \sum_{i=0}^{k-1} \frac{(f(x_i) - f^*)^2}{\|B_i^* g(x_i)\|^2}.$$

Множество точек x , удовлетворяющих неравенству

$$\|A_k(x_k - x)\| \leq \sqrt{\|x_0 - x^*\|^2 - \sum_{i=0}^{k-1} \frac{(f(x_i) - f^*)^2}{\|B_i^* g(x_i)\|^2}}, \quad (23)$$

представляет собой локализирующий точку x^* эллипсоид Φ_k , объем которого равен

$$\text{vol}(\Phi_k) = \frac{v_0 \left(\sqrt{\|x_0 - x^*\|^2 - \sum_{i=0}^{k-1} \frac{(f(x_i) - f^*)^2}{\|B_i^* g(x_i)\|^2}} \right)^n}{\det(A_k)}, \quad (24)$$

где v_0 – объем единичного n -мерного шара.

Итак, вопрос о сходимости алгоритма (10)–(14) сводится к оценке скорости уменьшения объема эллипсоида Φ_k , локализирующего x^* . Стремление к нулю $\text{vol}(\Phi_k)$, $k \rightarrow \infty$ согласно (24) обеспечивают два фактора: во-первых, рост $\det(A_k)$ при $k \rightarrow \infty$ и, во-вторых, сходимость ряда $\sum_{i=0}^{k-1} \frac{(f(x_i) - f^*)^2}{\|B_i^* g(x_i)\|^2}$ к $\|x_0 - x^*\|^2$ при $k \rightarrow \infty$. Для того, чтобы хорошо срабатывал первый, должны быть достаточно тупыми углы φ_k между последовательными субградиентами $B_k^* g(x_k)$ и $B_k^* g(x_{k+1})$. Так, $\varphi_k \leq -\frac{3}{2}\pi$ обеспечивает $\det(A_{k+1}) \geq \sqrt{2} \det(A_k)$. Для хорошей работы второго $\frac{f(x_k) - f^*}{\|B_k^* g(x_k)\|}$ должно быть сравнимо с $\|x_0 - x^*\|$. Оба этих фактора

зависят от конкретных свойств $f(x)$, и получить для задачи (9) гарантированные оценки скорости уменьшения объема в (24) без каких-либо ограничений на $f(x)$ не представляется возможным.

Кроме того, из монотонного уменьшения $\text{vol}(\Phi_k)$ не совсем очевидно на сходимость метода, так как для ряда $\sum_{i=0}^k \frac{(f(x_i) - f^*)^2}{\|B_i^* g(x_i)\|^2}$ при $k \rightarrow \infty$ не следует расходимость, поскольку максимальное собственное число матрицы преобразования пространства субградиентов может увеличиваться в $\sqrt{2}$ раз при переходе к очередному шагу. Для доказательства сходимости метода типа (10)–(14) можно поступить следующим образом. На некоторой итерации K прекратить преобразования пространства. При этом для всех k будет выполнено

$$\|B_k^* g(x_k)\|^2 \leq \lambda_{\max}(B_k B_k^*) \|g(x_k)\|^2 \leq \max_{i=0}^K \{\lambda_{\max}(B_i B_i^*)\} \|g(x_k)\|^2,$$

где $\lambda_{\max}(B_i B_i^*)$ – максимальное собственное число матрицы $B_i B_i^*$. Тогда для выпуклой функции $f(x)$ в силу ограниченности $\|g(x_k)\|$, если последовательность x_k ограничена, следует расходимость ряда

$$\sum_{i=0}^k \frac{(f(x_i) - f^*)^2}{\|B_i^* g(x_i)\|^2} \quad \text{при } k \rightarrow \infty,$$

т. е. доказательство для метода – обеспечить за счет классической фейереровской регулировки шагового множителя, но в некотором преобразованном пространстве аргументов $Y = A_K X$.

Но, во-первых, в вычислительном отношении предельная скорость сходимости не столь актуальна. Во-вторых, алгоритм (10)–(14) вычислительно не конструктивен в том смысле, что критерий останова $f(x_k) = f^*$ никогда не сработает. В-третьих, на этом пути теряется r -алгоритмовская природа метода, поэтому поступим по другому. Построим метод типа (10)–(14), конструктивный в вычислительном плане, и при этом будем ориентироваться на те его характеристики, которые вычислительно легко проверять. Условимся называть его вычислительной схемой метода типа (10)–(14).

Для того, чтобы обеспечить конструктивность критерия останова для задачи (9), более корректно сформулируем, что же будем считать ее решением. Пусть $X_\varepsilon^* = \{x : f(x) - f^* \leq \varepsilon\}$, где $\varepsilon > 0$. Точку $x_k \in X_\varepsilon^*$, т. е. такую, для которой выполняется $f(x_k) - f^* \leq \varepsilon$, будем считать решением задачи (9) с точностью ε по функционалу. При этом входным

параметром алгоритма будет точность ε , с которой требуется решить задачу (9) по функционалу, и останов алгоритма будем производить, как только достигнем точки $x_k \in X_\varepsilon^*$.

Кроме этого, отметим, что в алгоритме (10)–(14) дважды реализуется операция умножения матрицы на один и тот же вектор. Это имело смысл для удобства доказательства теоремы 1, но в вычислительной схеме избавимся от лишних операций. Для этого понадобится следующая лемма, которую приведем без доказательства.

Лемма 2. Пусть B_k – невырожденная матрица размера $n \times n$, p_k, p_{k+1} – n -мерные векторы, такие, что $\left(\frac{B_k^* p_k}{\|B_k^* p_{k+1}\|}, \frac{B_k^* p_{k+1}}{\|B_k^* p_{k+1}\|} \right)^2 < 1$, и пусть B_{k+1} вычисляется согласно (8), где $\xi_1 = \xi_k = \frac{B_k^* p_k}{\|B_k^* p_k\|}$, $\xi_2 = \xi_{k+1} = \frac{B_k^* p_{k+1}}{\|B_k^* p_{k+1}\|}$, $\beta_1 = \sqrt{1 - (\xi_1, \xi_2)}$, $\beta_2 = \sqrt{1 + (\xi_1, \xi_2)}$. Тогда:

$$\frac{B_{k+1}^* p_{k+1}}{\|B_{k+1}^* p_{k+1}\|} = \left(\frac{1}{\beta_2} - \frac{1}{\beta_1} \right) \frac{\xi_k}{2} + \left(\frac{1}{\beta_2} + \frac{1}{\beta_1} \right) \frac{\xi_{k+1}}{2}, \quad (25)$$

$$\frac{B_{k+1}^* p_k}{\|B_{k+1}^* p_k\|} = \left(\frac{1}{\beta_2} + \frac{1}{\beta_1} \right) \frac{\xi_k}{2} + \left(\frac{1}{\beta_2} - \frac{1}{\beta_1} \right) \frac{\xi_{k+1}}{2}, \quad (26)$$

$$\|B_{k+1}^* p_i\| = \beta_1 \beta_2 \|B_k^* p_i\|, \quad i = k, k+1. \quad (27)$$

Для двух векторов: $B_k^* p_k$ и $B_k^* p_{k+1}$ в преобразованном пространстве соотношения (25)–(27) леммы 2 обеспечивают пересчет их нормированных направлений, а также их норм при переходе в очередное преобразованное пространство аргументов, полученное в результате применения оператора (6), определяемого этими векторами.

Используя ε_f для обозначения точности решения задачи (9) по функционалу и учитывая сказанное выше, приходим к следующей вычислительной схеме алгоритма (10)–(14).

Перед началом вычислений имеем $\varepsilon_f > 0$, $x_0 \in E^n$, $h_0 = \frac{f(x_0) - f^*}{\|\partial f(x_0)\|}$,

$\xi_0 = \frac{\partial f(x_0)}{\|\partial f(x_0)\|} \in E^n$, $B_0 = I$ – единичная матрица размера $n \times n$. Пусть на k -м шаге получены $x_k \in E^n$, $h_k, \xi_k \in E^n$, B_k – матрица $n \times n$. Тогда, если $f(x_k) - f^* \leq \varepsilon_f$, то x_k – искомая точка. В противном случае переходим к $(k+1)$ -му шагу.

1) Вычислим очередное приближение

$$x_{k+1} = x_k - h_k B_k \xi_k. \quad (28)$$

2) Вычислим $f(x_{k+1})$, $\partial f(x_{k+1})$ и положим

$$\xi_{k+1} = \frac{B_k^* \partial f(x_{k+1})}{\|B_k^* \partial f(x_{k+1})\|}, \quad h_{k+1} = \frac{f(x_{k+1}) - f^*}{\|B_k^* \partial f(x_{k+1})\|}. \quad (29)$$

3) Если $(\xi_k, \xi_{k+1}) \geq 0$, полагаем $B_{k+1} = B_k$ и переходим к 4, иначе вычисляем

$$\eta_1 = \frac{\xi_k + \xi_{k+1}}{\|\xi_k + \xi_{k+1}\|}, \quad \eta_2 = \frac{\xi_k - \xi_{k+1}}{\|\xi_k - \xi_{k+1}\|}, \quad (30)$$

$$\beta_1 = \sqrt{1 - (\xi_k, \xi_{k+1})}, \quad \beta_2 = \sqrt{1 + (\xi_k, \xi_{k+1})}, \quad (31)$$

$$B_{k+1} = B_k R_{\beta_1}(\eta_1) R_{\beta_2}(\eta_2), \quad (32)$$

$$\xi_{k+1} = \left(\frac{1}{\beta_2} - \frac{1}{\beta_1} \right) \frac{\xi_k}{2} + \left(\frac{1}{\beta_2} + \frac{1}{\beta_1} \right) \frac{\xi_{k+1}}{2}, \quad h_{k+1} = \frac{h_{k+1}}{\beta_1 \beta_2}. \quad (33)$$

Здесь $R_\beta(\eta) = I + (\beta - 1)\eta\eta^T$, $\|\eta\| = 1$, и переходим к 4.

4) Переходим к очередному шагу с x_{k+1} , B_{k+1} , ξ_{k+1} , h_{k+1} .

Теорема 2. Пусть на каждом шаге алгоритма (28)–(33) выполнены условия $\|B_k\| \leq c_1$ и $\|\partial f(x_k)\| \leq c_2$, где $\|B_k\|$ – евклидова норма матрицы B_k . Тогда объем эллипсоида Φ_k , в котором локализована x^* , допускает оценку

$$\text{vol}(\Phi_k) \leq v_0 \det(B_k) \left(\|x_0 - x^*\|^2 - \frac{1}{c_1^2} \sum_{i=0}^{k-1} \frac{(f(x_i) - f^*)^2}{\|g(x_i)\|^2} \right)^{n/2}, \quad (34)$$

где v_0 – объем единичного шара в E^n и

$$\det(B_k) = \prod_{i=1}^k \sqrt{1 - \left(\frac{B_i^* g(x_{i-1})}{\|B_i^* g(x_{i-1})\|}, \frac{B_i^* g(x_i)}{\|B_i^* g(x_i)\|} \right)^2}.$$

$$(B_i^* g(x_{i-1}), B_i^* g(x_i)) < 0$$

Алгоритм (28)–(33) позволяет найти точку $x_k \in X_{\varepsilon_f}^*$ не более чем за K шагов, где $K = \left\lceil \left(\frac{c_1 c_2 \|x_0 - x^*\|}{\varepsilon_f} \right)^2 \right\rceil + 1$. Здесь $\lceil a \rceil$ – целая часть действительного числа a .

Доказательство. Неравенство (34) следует из (24), так как с учетом леммы 1, имеем

$$\frac{1}{\det(A_k)} = \det(B_k) = \prod_{i=1}^k \sqrt{1 - \left(\frac{B_i^* g(x_{i-1})}{\|B_i^* g(x_{i-1})\|}, \frac{B_i^* g(x_i)}{\|B_i^* g(x_i)\|} \right)^2} \\ (B_i^* g(x_{i-1}), B_i^* g(x_i)) < 0$$

и, кроме того,

$$\sum_{i=0}^{k-1} \frac{(f(x_i) - f^*)^2}{\|B_i^* g(x_i)\|^2} \geq \sum_{i=0}^{k-1} \frac{(f(x_i) - f^*)^2}{\|B_i\|^2 \|g(x_i)\|^2} \geq \sum_{i=0}^{k-1} \frac{(f(x_i) - f^*)^2}{c_1^2 \|g(x_i)\|^2}.$$

Учитывая, что

$$\|x_0 - x^*\|^2 - \frac{1}{c_1^2} \sum_{i=0}^{k-1} \frac{(f(x_i) - f^*)^2}{\|g(x_i)\|^2} \leq \|x_0 - x^*\|^2 - \frac{1}{c_1^2 c_2^2} \sum_{i=0}^{k-1} \varepsilon_f^2 = \\ = \|x_0 - x^*\|^2 - \frac{k \varepsilon_f^2}{c_1^2 c_2^2},$$

получаем, что не более чем за $K = \left\lceil \left(\frac{c_1 c_2 \|x_0 - x^*\|}{\varepsilon_f} \right)^2 \right\rceil + 1$ шагов алгоритм (28)–(33) решит задачу (9) с точностью ε_f по функционалу, так как в противном случае приходим к отрицательному объему эллипсоида, локализирующего точку x^* .

Итак, алгоритм (28)–(33) позволяет решить задачу (9) с точностью ε_f по функционалу, но оценка максимального числа итераций, необходимых для достижения точности ε_f , очень грубая и при реальной работе алгоритма следует ожидать, что критерий останова будет срабатывать из условия

$$f(x_k) - f^* \leq \|B_k^* g(x_k)\| \sqrt{\|x_0 - x^*\|^2 - \sum_{i=0}^{k-1} \frac{(f(x_i) - f^*)^2}{\|B_i^* g(x_i)\|^2}} \quad (35)$$

в силу малости $\|B_k^* g(x_k)\|$ при больших k .

Неравенство (35) следует из (23) с учетом того, что

$$\|A_k(x_k - x^*)\| \geq \\ \geq \left(A_k(x_k - x^*), \frac{B_k^* g(x_k)}{\|B_k^* g(x_k)\|} \right) \geq \frac{(x_k - x^*, g(x_k))}{\|B_k^* g(x_k)\|} \geq \frac{f(x_k) - f^*}{\|B_k^* g(x_k)\|}.$$

Алгоритм (28)–(33) можно также рассматривать и как метод фейеровского типа с преобразованием пространства, которое по смыслу близкое к тому, которое применяется в r -алгоритмах. При этом преобразование определяют два последовательных субградиента в точках полученных согласно фейеровскому шагу.

Рассмотрим еще один вариант методов фейеровского типа с преобразованием пространства, который отличается от предыдущего тем, что преобразование пространства будет определяться субградиентом в точке и некоторым вектором агрегатного типа, являющегося выпуклой комбинацией субградиентов, вычисленных ранее. Как и для алгоритма (28)–(33), движение из точки будет осуществляться в направлении антисубградиента с классическим фейеровским шагом. Кратко опишем основную идею на примере алгоритма (28)–(33).

Заметим, что в преобразованном пространстве $y = A_{k+1}x$, если на k -м шаге была реализована операция преобразования пространства, то согласно лемме 1 для функции $\varphi_{k+1}(y) = f(B_{k+1}y)$ субградиенты $B_{k+1}^*g(x_k)$ и $B_{k+1}^*g(x_{k+1})$ будут ортогональны. При этом силу классического фейеровского шага и тупого угла между последовательными субградиентами $B_{k+1}^*g(x_k)$ и $B_{k+1}^*g(x_{k+1})$, субградиент $B_{k+1}^*g(x_k)$ допускает параллельный перенос в точку $y_{k+1} = A_{k+1}x_{k+1}$, не отсекая при этом точку $y^* = A_{k+1}x^*$. Поэтому в точке $y_{k+1} = A_{k+1}x_{k+1}$ мы имеем два ортогональных субградиента: $B_{k+1}^*g(x_k)$ и $B_{k+1}^*g(x_{k+1})$ и пересечение определяемых ими полупространств в преобразованном пространстве переменных локализует точку $y^* = A_{k+1}x^*$. Движение из точки y_{k+1} осуществляется согласно классическому фейеровскому шагу в направлении, обратном к $B_{k+1}^*g(x_{k+1})$, в некоторую точку $y_{k+2} = A_{k+1}x_{k+2}$, где x_{k+2} вычисляется согласно (28), вследствие чего субградиент $B_{k+1}^*g(x_k)$ допускает параллельный перенос и в точку y_{k+2} .

Очевидно, что если субградиент $B_{k+1}^*g(x_{k+2})$, вычисленный в точке y_{k+2} , образует тупой угол с двумя предыдущими, то можно уменьшить объем области локализации x^* , выбрав в качестве векторов, определяющих преобразование (6) в точке y_{k+2} , субградиент в ней $B_{k+1}^*g(x_{k+2})$ и вектор, являющийся выпуклой комбинацией первых двух, так, чтобы угол между ними был максимально тупым. Для нормированных векторов p_1, p_2, p_3 в E^n это приводит к следующей задаче

$$\min(p, p_3),$$

$$p = \lambda_1 p_1 + \lambda_2 p_2, \quad \lambda_1 \geq 0, \quad \lambda_2 \geq 0,$$

$$\|p\| = 1, \quad (p_1, p_2) = 0,$$

которая легко решается аналитически. Ее решение характеризует следующая лемма.

Лемма 3. Пусть p_1, p_2, p_3 – векторы в E^n , такие, что $(p_1, p_2) = 0$, $\|p_1\| = \|p_2\| = \|p_3\| = 1$, и условия $(p_1, p_3) \geq 0$ и $(p_2, p_3) \geq 0$ не выполняются одновременно. Тогда вектор

$$p = \begin{cases} p_1, & \text{если } (p_1, p_3) < 0 \text{ и } (p_2, p_3) \geq 0, \\ p_2, & \text{если } (p_1, p_3) \geq 0 \text{ и } (p_2, p_3) < 0, \\ \lambda_1 p_1 + \lambda_2 p_2, & \text{если } (p_1, p_3) < 0 \text{ и } (p_2, p_3) < 0, \end{cases} \quad (36)$$

доставляет минимум скалярному произведению (p, p_3) . Здесь

$$\lambda_1 = -\frac{(p_1, p_3)}{\sqrt{(p_1, p_3)^2 + (p_2, p_3)^2}} \quad \text{и} \quad \lambda_2 = -\frac{(p_2, p_3)}{\sqrt{(p_1, p_3)^2 + (p_2, p_3)^2}}.$$

Для очередного шага эта ситуация повторяется, только в качестве первого вектора уже будет использоваться не субградиент, а вектор, являющийся выпуклой комбинацией двух предыдущих субградиентов в очередном преобразованном пространстве. Это позволяет замкнуть цикл вычислений и построить конструктивное правило использования в качестве одного из векторов, определяющих преобразование (6), некоторого агрегированного вектора, полученного в результате выпуклой комбинации вычисленных ранее субградиентов.

Основываясь на леммах 2, 3, приходим к следующей вычислительной схеме алгоритма.

Перед началом вычислений имеем $\varepsilon_f > 0$, $x_0 \in E^n$, $h_0 = \frac{f(x_0) - f^*}{\|\partial f(x_0)\|}$,

$\xi_0 = \frac{\partial f(x_0)}{\|\partial f(x_0)\|} \in E^n$, $p_0 = 0 \in E^n$, $B_0 = I$ – единичная матрица размера $n \times n$. Пусть на k -м шаге получены $x_k \in E^n$, $h_k, \xi_k \in E^n$, $p_k \in E^n$, B_k – матрица $n \times n$. Тогда, если $f(x_k) - f^* \leq \varepsilon_f$, то x_k – искомая точка. В противном случае переходим к $(k+1)$ -му шагу

1) Вычислим очередное приближение

$$x_{k+1} = x_k - h_k B_k \xi_k. \quad (37)$$

2) Вычислим $f(x_{k+1}), \partial f(x_{k+1})$ и положим

$$\xi_{k+1} = \frac{B_k^* \partial f(x_{k+1})}{\|B_k^* \partial f(x_{k+1})\|}, \quad h_{k+1} = \frac{f(x_{k+1}) - f^*}{\|B_k^* \partial f(x_{k+1})\|}. \quad (38)$$

3) Вычислим

$$\lambda_1 = -\frac{(p_k, \xi_{k+1})}{\sqrt{(p_k, \xi_{k+1})^2 + (\xi_k, \xi_{k+1})^2}}, \quad \lambda_2 = -\frac{(\xi_k, \xi_{k+1})}{\sqrt{(p_k, \xi_{k+1})^2 + (\xi_k, \xi_{k+1})^2}}$$

и положим

$$p_{k+1} = \begin{cases} \lambda_1 p_k + \lambda_2 \xi_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 > 0, \\ p_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 \leq 0, \\ \xi_k, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 > 0, \\ 0, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 \leq 0. \end{cases} \quad (39)$$

4) Если $(p_{k+1}, \xi_{k+1}) \geq 0$, полагаем $B_{k+1} = B_k$ и переходим к 5, иначе вычисляем

$$\eta_1 = \frac{p_{k+1} + \xi_{k+1}}{\|p_{k+1} + \xi_{k+1}\|}, \quad \eta_2 = \frac{p_{k+1} - \xi_{k+1}}{\|p_{k+1} - \xi_{k+1}\|}, \quad (40)$$

$$\beta_1 = \sqrt{1 - (p_{k+1}, \xi_{k+1})}, \quad \beta_2 = \sqrt{1 + (p_{k+1}, \xi_{k+1})}, \quad (41)$$

$$B_{k+1} = B_k R_{\beta_1}(\eta_1) R_{\beta_2}(\eta_2), \quad (42)$$

$$\xi_{k+1} = \left(\frac{1}{\beta_2} - \frac{1}{\beta_1}\right) \frac{p_{k+1}}{2} + \left(\frac{1}{\beta_2} + \frac{1}{\beta_1}\right) \frac{\xi_{k+1}}{2}, \quad h_{k+1} = \frac{h_{k+1}}{\beta_1 \beta_2}, \quad (43)$$

$$p_{k+1} = \left(\frac{1}{\beta_2} + \frac{1}{\beta_1}\right) \frac{p_{k+1}}{2} + \left(\frac{1}{\beta_2} - \frac{1}{\beta_1}\right) \frac{\xi_{k+1}}{2}. \quad (44)$$

Здесь $R_\beta(\eta) = I + (\beta - 1)\eta\eta^T$, $\|\eta\| = 1$ и переходим к 5.

5) Переходим к очередному шагу с x_{k+1} , B_{k+1} , ξ_{k+1} , h_{k+1} , p_{k+1} .

Для алгоритма (37)–(44) справедлива теорема, аналогичная к теореме 2 для алгоритма (28)–(33).

Теорема 3. Пусть на каждом шаге алгоритма (37)–(44) выполнены условия $\|B_k\| \leq c_1$ и $\|\partial f(x_k)\| \leq c_2$. Тогда объем эллипсоида Φ_k , в котором локализована x^* , допускает следующую оценку:

$$\text{vol}(\Phi_k) \leq v_0 \det(B_k) \left(\|x_0 - x^*\|^2 - \frac{1}{c_1^2} \sum_{i=0}^{k-1} \frac{(f(x_i) - f^*)^2}{\|g(x_i)\|^2} \right)^{n/2}. \quad (45)$$

Здесь v_0 – объем единичного шара в E^n и

$$\det(B_k) = \prod_{i=1}^k \sqrt{1 - \left(p_i, \frac{B_i^* g(x_i)}{\|B_i^* g(x_i)\|}\right)^2}.$$

$$(p_i, B_i^* g(x_i)) < 0$$

Алгоритм (37)–(44) позволяет найти точку $x_k \in X_{\varepsilon_f}^*$ не более чем за K шагов, где $K = \left\lceil \left(\frac{c_1 c_2 \|x_0 - x^*\|}{\varepsilon_f} \right)^2 \right\rceil + 1$.

Алгоритм (37)–(44) по тем же соображениям, что и предыдущий, можно отнести к классу r -алгоритмов, но в отличие от алгоритма (28)–(33) он использует некий агрегированный вектор, являющийся выпуклой комбинацией субградиентов, вычисленных в предыдущих точках, что дает основание надеяться на более сильное уменьшение объема области локализации множества экстремумов.

Следует заметить, что такого рода векторы характерны, например, для методов агрегатного типа [7], где они определяют направление движения из точки. Но, использование агрегированного вектора в алгоритме (37)–(44) выгодно отличается от его использования в алгоритмах агрегатного типа в том смысле, что при этом автоматически решается проблема обновления агрегированного вектора.

Итак, в рамках алгоритмов субградиентного типа с преобразованием пространства мы рассмотрели два достаточно простых алгоритма с классической фейеровской регуляризкой шагового множителя, основанных на преобразовании (6), которые позволяют с точностью до ε по функционалу решить задачу (9). Для них характерно использование на каждом шаге незначительной информации о гиперплоскостях, отсекающих множество X^* . Так, первый использует не более двух, второй – не более трех отсекающих гиперплоскостей. Очевидно, что, используя в точке более обширную информацию об отсекающих множество X^* гиперплоскостях, можно существенно ускорить эти алгоритмы; дальше обсудим некоторые из путей в этом направлении.

Но сам факт сходимости алгоритмов (28)–(33) и (37)–(44) для задачи (9) ровно ничего не говорит об их практической работоспособности, тем более что для установления сходимости используется уменьшение объема области локализации множества экстремумов X^* . Так, классический вариант фейеровского метода без преобразования пространства [6], где

очередное приближение вычисляется по формуле

$$x_{k+1} = x_k - \frac{(f(x_k) - f^*)}{\|\partial f(x_k)\|^2} \partial f(x_k), \quad (46)$$

также сходится для задачи (9) к $x^* \in X^*$, но для «овражных» функций очень плохо работает, поэтому постараемся подтвердить их эффективность численными расчетами для различного рода гладких и негладких выпуклых задач.

Численные эксперименты. Для численных экспериментов в качестве тестовых гладких задач выберем квадратичные положительно определенные функции $f(x) = \frac{1}{2}(x, Ax)$, $x \in E^n$, где матрица A размера $n \times n$ диагональная и такова, что $a_{ii} = t^{i-1}$, $i = 1, \dots, n$, $t > 1$. Будем обозначать их $Quad(t)$. Для тестовых негладких задач рассмотрим $f(x) = \sum_{i=1}^n t^{i-1} |x_i|$, $x \in E^n$ и будем обозначать их $Sabs(t)$. Начальное приближение как для $Quad(t)$, так и для $Sabs(t)$ выберем $x_0 = (1, 1, \dots, 1, 1)$. Рассмотрим следующие значения параметра $t = \{2.0, 1.2, 1.1\}$, что позволяет, управляя числом переменных задачи n , проверить работу алгоритмов как для плохо обусловленных, так и для хорошо обусловленных задач этого типа. Кроме $Sabs(t)$ в качестве тестовых негладких задач рассмотрим и два классических примера кусочно-квадратичных функций [2, с. 176] (*Shor*) и [2, с. 151] (*Maxquad*).

Алгоритмы (28)–(33) и (37)–(44) условимся называть алгоритмами A и B соответственно, и для них под итерацией будем понимать те их шаги, на которых производится операция преобразования пространства. Для сравнения рассмотрим также классический вариант фейеровского метода без преобразования пространства [6] (Fejer), где шаг вычисляется согласно (46) и $r(\alpha)$ -алгоритм [1, 2], который зарекомендовал себя практически эффективным алгоритмом для решения задач негладкой оптимизации. Для $r(\alpha)$ -алгоритма коэффициент растяжения пространства α и параметры адаптивной регулировки шагового множителя q_1, q_2, m выберем согласно рекомендуемым в [4], т. е. для гладких функций: $\alpha = 2$, $q_1 = 0.9$, $q_2 = 1.1$, $m = 3$ и для негладких: $\alpha = 2$, $q_1 = 1.0$, $q_2 = 1.1$, $m = 3$.

В качестве основной характеристики для сравнения выберем число вычислений $f(x)$ и $\partial f(x)$, не принимая в расчет время работы алгоритмов, поэтому отметим, что в вычислительном отношении итерация алго-

ритмов A и B в два раза более трудоемка, чем итерация $r(\alpha)$ -алгоритма. Кроме этого, $r(\alpha)$ -алгоритм по отношению к остальным обладает еще и тем преимуществом, что он не использует f^* .

Для того, чтобы обеспечить равноправность алгоритмов, их останков будем производить по одному и тому же критерию, а именно, при достижении точки $x_k \in X_{\varepsilon_f}^*$, где $\varepsilon_f = \varepsilon(1 + |f^*|)$. Так, для задач $Quad(t)$ и $Sabs(t)$ это равносильно $f(x_k) - f^* \leq \varepsilon$, так как f^* для них равно нулю. Для $Shor$ это будет $f(x_k) - f^* < 23.6\varepsilon$ и для $Maxquad - f(x_k) - f^* < 1.85\varepsilon$. Кроме этого, для решения тестовых задач отведем каждому из алгоритмов не более 10000 итераций, и если за это число итераций не достигнута требуемая точность, то будем считать, что алгоритм не решил данную задачу.

Численные эксперименты для квадратичных задач при различных значениях n приведены в табл. 1. Здесь N_{fg} – число вычислений $f(x)$ и $\partial f(x)$, N_{iter} – число итераций алгоритмов. При этом все алгоритмы заканчивали работу по одному и тому же критерию $\varepsilon_f \leq 10^{-10}$, с тем, чтобы обеспечить $\|x_k - x^*\| \leq 10^{-5}$. Для классического фейеровского метода прочерк обозначает, что он не справился с задачей за 10000 итераций. Как видно из табл. 1, алгоритмы A и B составляют достойную конкуренцию $r(\alpha)$ -алгоритму.

Таблица 1. Квадратичные задачи

Задача	n	<i>Fejer</i>	$r(\alpha)$ -алгоритм		Алгоритм A		Алгоритм B	
		N_{fg}	N_{iter}	N_{fg}	N_{iter}	N_{fg}	N_{iter}	N_{fg}
$Quad(1.1)$	75	738	147	189	123	143	117	124
$Quad(1.1)$	150	–	515	920	733	843	543	548
$Quad(1.1)$	200	–	975	1841	1566	1781	1083	1088
$Quad(1.2)$	30	261	110	135	49	64	52	65
$Quad(1.2)$	60	5407	199	297	230	259	198	205
$Quad(1.2)$	100	–	508	917	807	932	614	619
$Quad(2.)$	10	427	79	98	31	54	28	55
$Quad(2.)$	20	–	137	186	135	164	124	140
$Quad(2.)$	30	–	244	396	326	386	267	284

Результаты для негладких задач ($\varepsilon \leq 10^{-5}$) приведены в табл. 2. Здесь отсутствует классический фейеровский метод, так как он не спра-

вился ни с одной из задач за 10000 итераций. Для алгоритма A здесь ситуация не столь хороша, как для квадратичных функций. Если при небольших n и не слишком «овражных» задачах он еще составляет конкуренцию $r(\alpha)$ -алгоритму, то при больших n или сильной «овражности» это не так. В этом убеждает пример $Sabs(1.1)$ при $n=200$, где за 10000 итераций было получено $f(x)$ порядка 1. Но алгоритм B оказался конкурентноспособным с $r(\alpha)$ -алгоритмом на значительном числе задач даже по времени счета, так как он требует существенно меньшего числа итераций.

Таблица 2. Негладкие задачи

Задача	n	$r(\alpha)$ -алгоритм		Алгоритм A		Алгоритм B	
		N_{iter}	N_{fg}	N_{iter}	N_{fg}	N_{iter}	N_{fg}
$Sabs(1.1)$	75	1826	1832	1693	1695	516	518
$Sabs(1.1)$	150	4510	4516	8196	8198	2665	2667
$Sabs(1.1)$	200	6743	6749	—	—	4679	4681
$Sabs(1.2)$	30	663	671	432	434	143	145
$Sabs(1.2)$	60	1604	1611	2133	2135	707	709
$Sabs(1.2)$	100	3242	3249	6687	6689	2343	2345
$Sabs(2.)$	10	226	259	160	164	75	78
$Sabs(2.)$	20	561	590	791	796	306	308
$Sabs(2.)$	30	997	1025	1876	1878	784	786
$Shor$	5	61	64	74	78	30	33
$Maxquad$	10	112	123	109	117	34	41

В табл. 3 для алгоритмов A и B приведена статистика для гладких и негладких задач, вычисленных согласно (22), коэффициентов растяжения пространства в направлении разности нормированных векторов, определяющих преобразование пространства. Здесь α_{cp} — средний на одну итерацию коэффициент растяжения пространства, α_{min} , α_{max} — минимальный и максимальный коэффициенты, с которыми реализовывалось растяжение пространства в ходе процесса оптимизации. Как видно из табл. 3 для квадратичных овражных функций средний коэффициент около двух, что вполне соответствует выбору коэффициента растяжения пространства для $r(\alpha)$ -алгоритма. Для негладких

Таблица 3. Коэффициенты растяжения пространства.

Задача	n	Алгоритм A			Алгоритм B		
		$\alpha_{ср.}$	α_{\min}	α_{\max}	$\alpha_{ср.}$	α_{\min}	α_{\max}
$Quad(1.1)$	75	1.885	1.041	3.491	1.854	1.009	2.872
$Quad(1.1)$	150	2.304	1.002	8.229	2.511	1.155	4.424
$Quad(1.1)$	200	2.328	1.002	9.944	2.645	1.135	4.475
$Quad(1.2)$	30	1.519	1.010	2.199	1.493	1.037	2.199
$Quad(1.2)$	60	2.107	1.017	5.391	2.166	1.040	4.098
$Quad(1.2)$	100	2.290	1.001	10.704	2.505	1.002	5.377
$Quad(2.)$	10	1.526	1.003	2.364	1.552	1.045	2.182
$Quad(2.)$	20	2.010	1.018	5.281	2.013	1.001	5.313
$Quad(2.)$	30	2.216	1.001	8.462	2.344	1.007	5.447
$Sabs(1.1)$	75	1.679	1.136	2.369	2.207	1.383	3.199
$Sabs(1.1)$	150	1.640	1.136	2.480	2.263	1.383	3.140
$Sabs(1.1)$	200	1.637	1.136	2.480	2.264	1.383	3.160
$Sabs(1.2)$	30	1.722	1.082	2.799	2.119	1.286	3.565
$Sabs(1.2)$	60	1.694	1.082	2.503	2.261	1.279	3.711
$Sabs(1.2)$	100	1.665	1.082	2.494	2.270	1.500	3.436
$Sabs(2.)$	10	1.771	1.097	4.805	2.190	1.015	5.960
$Sabs(2.)$	20	1.746	1.048	3.450	2.345	1.089	5.030
$Sabs(2.)$	30	1.748	1.071	3.395	2.352	1.303	5.030
$Shor$	5	1.502	1.146	2.435	1.789	1.069	3.185
$Maxquad$	10	1.435	1.006	5.138	2.097	1.275	5.138

функций можно отметить, что если мы для алгоритмов A и B гарантируем средний коэффициент растяжения пространства порядка двух, то это обеспечивает их работу не хуже $r(\alpha)$ -алгоритма.

Следует также отметить, что алгоритмы A и B оказались не слишком чувствительными к точности ε_f , в том смысле, что для решения задачи с точностью по функционалу $\varepsilon_f = 10^{-10}$ они требуют незначительно большего числа итераций, чем с точностью $\varepsilon_f \leq 10^{-5}$. Это объясняется, по всей видимости, тем, что на последних шагах методы работают в преобразованном пространстве переменных, которое является хорошим для субградиентного метода с классической фейеровской регуляризкой шагового множителя.

Итак, численные эксперименты демонстрируют достаточно хоро-

шую работу алгоритмов A и B и, учитывая тот факт, что их можно улучшить, дают основания надеяться на создание для решения задачи (9) достаточно эффективных алгоритмов на базе преобразования пространства типа (6).

Об усовершенствовании алгоритмов. Для поиска безусловного минимума выпуклой функции алгоритмы A и B носят неконструктивный характер в том смысле, что они предполагают знание f^* . Тем не менее они имеют конкретную область приложений и могут быть использованы для нахождения допустимой точки системы неравенств

$$f_i(x) \leq 0, \quad i = 1, 2, \dots, m, \quad (47)$$

где $f_i(x)$, $i = 1, 2, \dots, m$, — выпуклые функции векторного аргумента $x \in E^n$. При условии, что система неравенств (47) совместна, достаточно решить задачу типа (9), где

$$f(x) = \max\{0, \max_{i=1}^m f_i(x)\}. \quad (48)$$

При этом $f^* = 0$ и X^* непусто.

Для функции вида (48), отсекающую множество X^* гиперплоскость в точке x_k будет задавать любая выпуклая комбинация субградиентов к нарушенным в точке x_k неравенствам, т. е. тех $\partial f_j(x_k)$, $j \in \{1, 2, \dots, m\}$, для которых выполняется $f_j(x_k) > 0$. Этот факт подсказывает направления для улучшения алгоритмов в том смысле, что прежде чем двигаться из точки x_k , целесообразно преобразование пространства направить на «расширение» конуса подходящих направлений убывания функции, т. е. избавиться от слишком «тупых» углов между субградиентами.

При этом целесообразно рассматривать и вычисленные ранее субградиенты, которые при параллельном переносе в эту точку не отсекают множества X^* и позволяют усилить коэффициент растяжения пространства. В качестве направления движения из точки можно ориентироваться как на субградиент в точке, так и на переход в вершину конуса, локализирующего множество X^* , как в методе ортогонального спуска [9].

Кроме того, предположение о совместности системы (47) слишком сильное, но оно легко снимается за счет задания априорной информации о расстоянии до оптимума, т. е. $\|x_0 - x^*\| \leq R$. При этом мы или находим с точностью до ε решение системы (47), или получаем достаточные условия ее несовместности в шаре $\|x_0 - x^*\| \leq R$, основываясь на расходимости $\sum_{i=0}^k \frac{(f(x_i) - f^*)^2}{\|B_i^* g(x_i)\|^2}$. При этом целесообразно проводить и

растяжения пространства в направлении градиента, если сдвиг из центра шара в преобразованном пространстве (т. е. $\frac{f(x_k) - f^*}{\|B_k^* g(x_k)\|}$) сравним с его радиусом.

А теперь обсудим, как избавиться от основного недостатка алгоритмов A и B – необходимости знания f^* . Поскольку, учитывая сказанное выше, мы имеем алгоритм, который за конечное число шагов находит с точностью ε по функционалу решение задачи (9) или же отвечает на вопрос о ее несовместности, то для корректировки f^* можно избрать путь деления интервала, в котором содержится значение f^* , методом дихотомии или «золотого» сечения. Это не представляет особых проблем, поскольку нижняя граница интервала, в котором содержится f^* , легко определяется на основании известной априорной информации о радиусе локализации множества X^* .

Но более рациональным в вычислительном плане представляется корректировать f^* за счет процедуры спуска, т. е. в качестве f^* использовать f^{rec} , где f^{rec} – рекордное значение $f(x)$ и, например, такое, что $f^{rec} - f^* < f(x_k) - f^{rec}$. Для поиска f^{rec} перспективной представляется процедура, близкая к процедуре ε -наискорейшего спуска [8, 10], которая позволяет найти значение минимума $f(x)$ с точностью до ε за конечное число шагов, с тем отличием, что применяться она должна в преобразованном пространстве, т. е. не к функции $f(x)$, а к ее образу в преобразованном на k -м шаге пространстве, т. е. к функции $\varphi_k(y) = f(B_k y)$. При этом корректировка ε не представляет особых проблем, так как в преобразованном пространстве известны радиус локализации множества экстремумов и нормы субградиентов функции $\varphi_k(y)$. Для процедуры такого типа самая плохая будет ситуация, когда мы находимся в точке, близкой к оптимуму, т. е. когда не можем понизить f^{rec} . Но при этом можно воспользоваться идеей классического метода эллипсоидов [11, 12], что, во-первых, гарантирует уменьшение объема области локализации множества экстремумов, и, во-вторых, обеспечивает переход в точку, достаточно далекую от рекордной, которая хороша для фейеровских шагов. Это приводит к сходящемуся алгоритму в смысле рекордов по функционалу и позволяет надеяться на его достаточно хорошую практическую работоспособность.

Заключение. В смысле выравнивания линий уровня $f(x)$ преобразование (6) объясняет эффективность r -алгоритмов с наглядных геометрических позиций и его можно считать вполне удовлетворительной интерпретацией растяжения пространства в направлении разности двух последовательных субградиентов. При этом оно позволяет сохранить

градиентную природу метода в отличие от классического метода эллипсоидов [11, 12], который является удачной интерпретацией субградиентного метода с растяжением пространства в направлении субградиента. Разумное сочетание эллипсоидов специального типа [5] с эллипсоидами типа $\text{ell}(x_0, a, b, c)$ вполне способно обеспечить создание на базе одномерного оператора растяжения пространства практически эффективных алгоритмов, применимых для решения более широкого класса задач, чем задачи выпуклого программирования, например, таких, как задача отыскания седловых точек выпукло-вогнутых функций, частные случаи задач решения вариационных неравенств, а также специальные классы задач линейной и нелинейной дополнителности.

Автор признателен профессору Н. З. Шору за внимание и поддержку данной работы.

Литература

1. Шор Н. З., ЖУРБЕНКО Н. Г. Метод минимизации, использующий операцию растяжения пространства в направлении разности двух последовательных градиентов // Кибернетика. – 1971. – № 3. – С. 51–59.
2. Шор Н. З. Методы минимизации недифференцируемых функций и их приложения. – К.: Наук. думка, 1979. – 199 с.
3. СТЕЦЮК П. И. К вопросу сходимости r -алгоритмов // Кибернетика и системный анализ. – 1995. – № 6. – С. 173–177.
4. Шор Н. З., СТЕЦЕНКО С. И. Квадратичные экстремальные задачи и недифференцируемая оптимизация. – К.: Наук. думка, 1989. – 208 с.
5. СТЕЦЮК П. И. К обоснованию сходимости алгоритмов с растяжением пространства // Теория оптимальных решений. – 1995. – С. 4–8.
6. ПОЛЯК Б. Т. Введение в оптимизацию. – М.: Наука, 1983. – 384 с.
7. KIEWIEL K. C. An aggregate subgradient method for nonsmooth convex programming // Math. Programming. – 1983. – N 27/3. – P. 320–341.
8. Nonsmooth optimization / Eds. Lemaréchal C., Mifflin R. – Oxford: Pergamon Press, 1978. – 186 p.

9. ЩЕПАКИН М. Б. О методе ортогонального спуска // Кибернетика. – 1987. – № 1. – С. 58–62.
10. LEMARÉCHAL C. An extension of Davidon methods to nondifferentiable problems // Math. Progr. Study 3. – North-Holland, Amsterdam, – 1975. – P. 95–109.
11. Юдин Д. Б., НЕМИРОВСКИЙ М. С. Информационная сложность и эффективные методы решения выпуклых экстремальных задач // Экономика и мат. методы. – 1976. – Вып. 2. – С. 357–359.
12. ШОР Н. З. Методы отсечения с растяжением пространства для решения задач выпуклого программирования // Кибернетика. – 1977. – № 1. – С. 94–95.

Метод центров тяжести простых тел

П. И. Стецюк

Кибернетика и системный анализ. — 1996. — № 5. — С. 117–138.

В рамках методов отсекающих плоскостей рассмотрена довольно общая конструкция с использованием линейных преобразований пространства, которая для задач выпуклого программирования позволяет обосновать сходящиеся в терминах рекордов по функции алгоритмы. Она основывается на внешней аппроксимации множества экстремумов простыми геометрическими телами с гарантированным уменьшением их объема на каждом шаге метода и позволяет создавать экономные по сложности обработки информации алгоритмы.

Информационная сложность методов первого порядка для задачи выпуклого программирования рассматривается в терминах числа обращений к оракулу первого порядка, который предоставляет значения функции и субградиента в точке. Но при этом не принимается во внимание объем вспомогательной вычислительной работы (сложность обработки информации), необходимый для того, чтобы задать очередной вопрос оракулу. Так, для методов отсечений, основанных на уменьшении объема области локализации множества экстремумов, эта характеристика колеблется от незначительного числа арифметических операций в методе эллипсоидов до практически невыполнимой задачи в методе центров тяжести.

Цель работы — обсудить возможность построения для задачи выпуклого программирования методов первого порядка, для которых можно было бы разумно оценить сложность обработки информации. В рамках методов отсечений с использованием линейных преобразований пространства рассмотрим достаточно общую конструкцию, которая для выпуклых задач позволяет обосновывать сходящиеся (в смысле рекордов по функции) со скоростью геометрической прогрессии алгоритмы. Она основана на внешней аппроксимации множества экстремумов простыми геометрическими телами с тем, чтобы обеспечить достаточное уменьшение их объема на каждом шаге методов, и позволяет в ее рамках создавать экономные по сложности обработки информации алгоритмы.

Не ограничивая общности для случая общей задачи выпуклого программирования, рассуждения проведем применительно к задаче:

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

где $f(x)$ – выпуклая функция векторного аргумента $x \in X$, $X = R^n$, $Q \subset R^n$ – выпуклое ограниченное многогранное множество, R^n – евклидово пространство размерности n со скалярным произведением (x, y) . Оптимальное значение $f(x)$ для нее обозначим f^* , множество экстремумов – X^* . X будем называть исходным пространством аргументов.

Значительная часть методов для решения задачи (1) обладает «предельной» сходимостью, т. е. теоремы об их сходимости построены на том факте, что в пределе удовлетворяются некоторые условия экстремума. К ним можно отнести: ε -субградиентные методы, методы возможных направлений, методы штрафных функций и барьеров и др. Для этих методов наличие оценок скорости сходимости обусловлено, как правило, значительным сужением задачи (1).

Но в то же время существует ряд методов, которые для задачи (1) обладают «гарантированной» скоростью сходимости, т. е. существует некоторая скалярная величина, для которой можно гарантировать ее уменьшение на каждом шаге алгоритма в определенное число раз и, кроме того, она является мажорантой сверху для рекордов $f(x)$. Эти методы относятся к классу методов отсечений и, чаще всего, такой скалярной величиной для них является объем области локализации множества экстремумов.

Методы этого типа для задачи (1) гарантируют сходимость по функции в следующем смысле:

$$f_k^* - f^* \leq q^k \max_{x \in Q} (f(x) - f^*), \quad (2)$$

где f_k^* – рекордное значение $f(x)$, полученное к k -ому шагу метода. Здесь $q < 1$ – знаменатель геометрической прогрессии, характеризующей скорость сходимости метода. Сходимость согласно (2) принято называть сходимостью по последовательности рекордов минимизируемой функции, и она является следствием уменьшения объема некоторых тел в R^n , локализующих X^* для задачи (1).

Центральный среди этих методов – метод центров тяжести (МЦТ) [1], основанный на внешней аппроксимации множества экстремумов выпуклым многогранником общего вида с уменьшением его объема в $(1 - 1/e)$ раз на каждой итерации. МЦТ сходится для задачи (1) в смысле (2) с $q_{\text{МЦТ}} = (1 - 1/ne)$ и является субоптимальным методом в смысле информационной трудоемкости для всего класса выпуклых задач [2]. Здесь в качестве трудоемкости метода рассматривается минимальное число обращений к оракулу первого порядка (поставляет значение функции и субградиент в точке), с тем, чтобы обеспечить решение всякой задачи класса с заданной относительной погрешностью

по функции. Следовательно, МЦТ существенно не улучшаем для задачи (1) в смысле информационной трудоемкости. Но его основной недостаток заключается в том, что при $n \geq 3$ отыскание центра тяжести выпуклого многогранника общего вида в R^n чрезвычайно громоздкая задача и метод вряд ли может быть реализован. Таким образом, для МЦТ сложность обработки информации (ОИ-трудоемкость) на итерации не поддается разумной оценке.

Этого недостатка лишены метод эллипсоидов (МЭ) [3, 4], и методы симплексов (МС) [5, 6]. Они основаны на внешней аппроксимации множества экстремумов более простыми телами, чем общий выпуклый многогранник, с гарантированным уменьшением их объема. Так, в МЭ переход к следующему эллипсоиду происходит с постоянной скоростью уменьшения объема в $(1 - 1/2n)$ раз. В МС переход к очередному симплексу происходит с уменьшением объема в наихудшем случае в $(1 - 1/2n^2)$ раз. Для задачи (1) это равносильно сходимости в смысле (2) с $q_{\text{мэ}} = (1 - 1/2n^2)$ для МЭ и $q_{\text{мс}} = (1 - 1/2n^3)$ для МС. В отличие от МЦТ итерации МЭ и МС легко реализуемы и их ОИ-трудоемкость составляет $O(n^2)$ арифметических операций. Однако МЭ и МС обладают достаточно слабыми оценками скорости сходимости по отношению к МЦТ. Например, чтобы на порядок улучшить значение функции, в МЭ требуется $4.6n^2$ итераций, а это не так мало даже при небольших значениях n .

Следовательно, вполне естественно попытаться построить алгоритм, который бы обладал скоростью сходимости, как в МЦТ и ОИ-трудоемкостью, как в МЭ и МС. Поэтому основной вопрос, на который попытаемся ответить, сформулируем следующим образом: Можно ли в рамках методов первого порядка построить алгоритм для решения задачи (1), который бы обладал нетрудоемкой итерацией в смысле ОИ-трудоемкости и на каждой из них практически реализовывал теоретически оптимальную оценку скорости сходимости или даже лучше? Пока условимся называть его искомым алгоритмом.

Как это сделать или об МЦТПТ. Для того, чтобы искомый алгоритм применительно к задаче (1) обладал геометрической скоростью сходимости в смысле (2), естественно для него в качестве критерия выбрать уменьшение объема области локализации множества экстремумов. При этом уменьшение объема в $(1 - 1/e)$ раз на каждом шаге искомого алгоритма равносильно теоретически оптимальной оценке скорости сходимости по последовательности рекордов минимизируемой функции для задачи (1).

Для того, чтобы обеспечить для него простоту итерации, целесообразно остановиться на идее внешней аппроксимации множества экстремумов более простыми выпуклыми телами, чем общий выпуклый многогранник. В качестве таковых разумно использовать в R^n эллипсоид, симплекс или параллелепипед. Далее будем называть их локализаторами, подразумевая под этим, что они содержат множество экстремумов задачи (1) — X^* .

Очевидно, в отличие от МЭ и МС, искомый алгоритм должен использовать информацию о функции в существенно большем числе точек, чем одна, что характерно для субоптимального МЦТ. Поэтому при построении локализаторов будем использовать значения функции и субградиента $f(x)$ в ряде точек, где они задают отсекающие множество X^* гиперплоскости вида

$$(x - x_i, \partial f(x_i)) \leq f^* - f(x_i) \leq f_k^* - f(x_i), \quad x \in X^*, \quad i = 1, 2, \dots, m. \quad (3)$$

Здесь f_k^* — рекордное значение $f(x)$, полученное к k -ому шагу искомого алгоритма.

Отметим, что наличие локализирующего X^* тела сколь угодно малого объема еще не дает оснований говорить о сходимости метода по последовательности рекордов минимизируемой функции. Но ситуация упрощается, когда это тело служит хорошим локализатором ограниченного многогранного множества, полученного в результате пересечения полупространств вида (3). Техника такого рода оценок не представляет особых проблем и достаточно хорошо разработана как для выпуклых, так и для квазивыпуклых функций (см., например, [7]). Поэтому в дальнейшем не будем уделять внимание этим вопросам, а ограничимся принципиальной схемой построения искомого алгоритма, подразумевая, что мы строим достаточно хорошо аппроксимирующие множество экстремумов локализаторы.

Пусть уже выбран локализатор одного из указанных типов, обозначим его L_1 . Тогда возникает вопрос о построении очередного локализатора, например, этого же типа (обозначим его L_2), который был бы по объему меньше, чем L_1 . Для этого естественным представляется в качестве очередной точки, относительно которой можно строить L_2 , выбрать точку, совпадающую с «центром» локализатора L_1 и обеспечить в каком-то смысле равноправность этой точки относительно его границ. Это легко сделать, преобразовав локализатор L_1 в геометрически простое тело в R^n : эллипсоид — в шар, параллелепипед — в куб, произвольный симплекс — в правильный симплекс. Шар, куб и правильный симплекс будем называть геометрически простыми, а эллипсоид,

параллелепипед и неправильный симплекс – геометрически сложными телами в R^n . Операцию преобразования геометрически сложного тела в геометрически простое тело условимся называть регуляризацией локализатора. Для того, чтобы ее осуществить, достаточно использовать линейное преобразование пространства. Поскольку для указанных тел регуляризация определена с точностью до константы, условимся среди определяющих их регуляризацию матриц выбирать ту, для которой детерминант равен отношению объемов текущего и очередного локализаторов. Это не имеет принципиального значения, так как субградиентные методы инвариантны относительно однородного масштабирования всего пространства R^n .

Пусть L_1 – геометрически простой локализатор одного из указанных типов и пусть построен очередной локализатор L_2 одного из этих типов, не обязательно геометрически простой, но такой, чтобы выполнялось условие $\text{vol}(L_1) > \text{vol}(L_2)$. При этом совершенно не обязательно, чтобы L_2 был вложенным в L_1 . Тогда переходим в центр тяжести L_2 и регуляризуем L_2 посредством линейного преобразования пространства. Повторяя этот же цикл для L_2 и L_3 , L_3 и L_4 и т. д., получим замкнутую в вычислительном отношении процедуру в том смысле, что на каждом ее шаге обеспечивается возможность однотипного построения очередного локализатора.

Эта процедура приводит к тому, что на k -ом ее шаге необходимо работать в том из преобразованных пространств $Y_k = A_k X$, где текущий локализатор L_k является телом геометрически простой формы. Здесь A_k – невырожденная матрица размера $n \times n$, полученная в результате последовательных регуляризаций локализаторов к k -ому шагу, поэтому отсекающие гиперплоскости вида (3) нужны в пространстве Y_k . Это не представляет особых проблем, так как соотношение (3) может быть записано

$$\left(y - y_i, (A_k^{-1})^T \partial f(x_i) \right) \leq f_k^* - f(x_i), \quad y \in Y^*, \quad i = 1, 2, \dots, m, \quad (4)$$

где $y = A_k x$ и $y_i = A_k x_i$ – образы точек из X в преобразованном пространстве Y_k , $Y^* = A_k X^*$ – образ множества экстремумов в Y_k .

Согласно (4) $(A_k^{-1})^T \partial f(x_i)$ можно рассматривать как субградиент в точке $y_i = A_k x_i$ для функции $\varphi_k(y) = f(A_k^{-1} y)$, определенной в преобразованном пространстве Y_k . Переход из Y_k в Y_{k+1} осуществляется посредством пересчета матрицы $A_{k+1} = T_{k+1} A_k$, где T_{k+1} – невырожденная матрица, задающая для локализатора L_{k+1} преобразование из геометрически сложной формы в геометрически простую. Следова-

но, субградиенты при переходе от одного пространства к другому будут пересчитываться $\partial\varphi_{k+1}(y) = (T_{k+1}^{-1})^T \partial\varphi_k(y)$.

Отсекающие гиперплоскости вида (3) в X и их образы (4) в Y_k позволяют легко ориентироваться в преобразованном пространстве Y_k относительно любой точки, в том числе и относительно точки y_k — центра тяжести локализатора L_k в Y_k . Пусть переход в центр тяжести L_{k+1} в Y_k задает соотношение

$$y_{k+1} = y_k + \sum_{i=1}^m \alpha_i \partial\varphi_k(y_i), \quad (5)$$

где α_i , $i = 1, 2, \dots, m$ — некоторые скаляры. Тогда процесс (5) в Y_k равносильно в X соотношению

$$x_{k+1} = x_k + A_k^{-1} \sum_{i=1}^m \alpha_i (A_k^{-1})^T \partial f(x_i), \quad (6)$$

где x_k и x_{k+1} — прообразы в X центров тяжести локализаторов L_k и L_{k+1} в Y_k .

Для реализации процесса типа (6) на k -ом шаге достаточно знать матрицу $B_k = A_k^{-1}$, x_k — прообраз в X центра тяжести локализатора L_k в Y_k , субградиенты $\partial\varphi_k(y_i) = B_k^T \partial f(x_i)$, $i = 1, 2, \dots, m$, а также правило, согласно которому вычисляется очередной центр тяжести L_{k+1} , т. е. α_i , $i = 1, 2, \dots, m$. Для перехода в очередное преобразованное пространство аргументов Y_{k+1} достаточно знать матрицу T_{k+1}^{-1} , где T_{k+1} — невырожденная матрица размера $n \times n$, которая задает преобразование L_{k+1} из геометрически сложной в геометрически простую форму. В частности, если такое преобразование не требуется, $T_{k+1} = I_n$, где I_n — единичная матрица размера $n \times n$.

Основанный на внешней аппроксимации множества экстремумов простыми телами процесс (5) имеет достаточно простую и геометрически наглядную интерпретацию на k -ом шаге в соответствующем ему преобразованном пространстве аргументов Y_k . Но эта простота отсутствует в исходном пространстве аргументов X и его можно согласно (6) рассматривать как зеркальное отображение в X процесса в Y_k . В этом смысле такая интерпретация процесса близка к некоторой достаточно общей конструкции, так называемых методов зеркального спуска [2].

Итак, искомый алгоритм будем интерпретировать как метод отсечений в некотором преобразованном пространстве аргументов, основываясь на информации о функции и ее субградиенте в ряде точек. В

качестве локализатора множества экстремумов выберем одно из указанных выпуклых тел в R^n , обеспечив достаточное уменьшение его объема по отношению к предыдущему локализатору, а также переход в центр тяжести очередного локализатора и его регуляризацию. Поэтому откажемся от названия «искомый алгоритм» и заменим его более подходящим для такого процесса названием – метод центров тяжести простых тел (МЦТПТ). При этом под МЦТПТ будем понимать не столько конкретный метод, сколько некоторую общую конструкцию, в рамках которой для задачи (1) можно описывать сходящиеся по последовательности рекордов минимизируемой функции алгоритмы.

При дальнейшем обсуждении МЦТПТ удобно разделить основные его функции. Выбор локализующих множество экстремумов тел и их регуляризацию условимся называть внешним алгоритмом. Сюда же отнесем и пересчет субградиентов при переходе в очередное преобразованное пространство аргументов. При этом под итерацией МЦТПТ в смысле перехода в очередную точку будем понимать именно итерацию внешнего алгоритма, т. е. переход в центр тяжести построенного локализатора. Построение же эффективных для внешнего алгоритма отсекающих гиперплоскостей, т. е. фактически определение точек, в которых требуется вычислять $f(x)$ и $\partial f(x)$, будем называть внутренним алгоритмом. Естественно и внутренний алгоритм также наделить своим критерием останова и в качестве такового рассмотрим некий аналог необходимого и достаточного условия экстремума выпуклой функции, а именно $0 \in \partial f(x)$, где $\partial f(x)$ – субдифференциал. Если для внешнего алгоритма все рассуждения должны быть математически строгими, то для внутреннего это не обязательно, и он может допускать различного рода нестрогие эвристические процедуры.

На самом деле функции внешнего и внутреннего алгоритма не так легко разделить и в МЦТПТ их естественно в значительной мере совместить, особенно в той их части, что касается построения очередного локализатора. Поэтому от внутреннего алгоритма потребуем, чтобы он на себя взял часть функций по подготовке информации (обозначим ее Ω), используемой внешним алгоритмом при выборе очередного локализатора. К Ω отнесем информацию о текущем геометрически простом локализаторе, об отсекающих гиперплоскостях, определяющих образ выпуклого многогранного множества Q в преобразованном пространстве аргументов, а также информацию, обеспечивающую конструктивный критерий останова для МЦТПТ, о которой речь пойдет ниже.

Итак, МЦТПТ для решения задачи (1) фактически должен обладать двумя критериями останова. Первый из них – достаточное уменьше-

ние объема области локализации множества экстремумов, второй связан с приближенным выполнением условий оптимальности для задачи выпуклого программирования. При этом первый критерий должен быть нацелен на то, чтобы гарантировать теоретически оптимальную оценку скорости сходимости для задачи (1) в смысле рекордов по функции.

С учетом вышесказанного, опуская подробную детализацию ряда вопросов для внешнего и внутреннего алгоритмов, опишем общую вычислительную схему МЦТПТ применительно к решению задачи (1), в основном с целью оценить для него сложность обработки информации (ОИ-трудоемкость). В преобразованном пространстве аргументов локализаторы геометрически простой формы (шар, куб, правильный симплекс) будем обозначать через L . Для обозначения локализаторов не обязательно геометрически простой формы (эллипсоид, параллелепипед или неправильный симплекс) будем использовать L' .

Перед началом вычислений имеем: x_0 – центр тяжести L_0 и $X^* \subset L_0$, $q = 1 - 1/e$ – константа, характеризующая достаточное уменьшение объема на итерации МЦТПТ, m_0 – максимальное число хранимых субградиентов ($m_0 \leq 2n$), $B_0 = I_n$ – единичная матрица размера $n \times n$.

МЦТПТ строит итерационную последовательность $\{x_k\}_{k=0}^{\infty}$ следующим образом:

Шаг 1. Вычислим согласно некоторым правилам (внутренний алгоритм) $(x_i, f(x_i) \partial f(x_i), \partial \varphi_k(y_i) = B_k^T \partial f(x_i), i = 1, 2, \dots, m)$, $m \leq m_0$ и множество Ω . Если получены необходимые и достаточные условия экстремума, то останов.

Шаг 2. В пространстве аргументов $Y_k = B_k^{-1}X$ относительно центра тяжести L_k (т. е. точки $y_k = B_k^{-1}x_k$) выбираем локализатор L'_{k+1} , такой, чтобы $Y_k^* = B_k^{-1}X^* \subset L'_{k+1}$ и $q^* = \text{vol}(L'_{k+1}) / \text{vol}(L_k)$ было минимально. Если $q^* \geq q$, то переходим к шагу 1.

Шаг 3. Вычислим очередное приближение

$$x_{k+1} = x_k + B_k \sum_{i=1}^m \alpha_i \partial \varphi_k(y_i),$$

которое задает прообраз в X центра тяжести L'_{k+1} в Y_k . Здесь α_i , $i = 1, 2, \dots, m$ – скаляры, задающие в Y_k центр тяжести L'_{k+1} относительно центра тяжести L_k .

Шаг 4. $B_{k+1} = B_k T_{k+1}^{-1}$ (регуляризация L'_{k+1}). T_{k+1} – неособенная матрица размера $n \times n$, задающая преобразование L'_{k+1} в L_{k+1} и такая, что $\det(T_{k+1}^{-1}) = q^*$.

Шаг 5. $\partial\varphi_{k+1}(y_i) = (T_{k+1}^{-1})^T \partial\varphi_k(y_i)$, $i = 1, 2, \dots, m$ (пересчет субградиентов при переходе в очередное преобразованное пространство аргументов).

Шаг 6. Положим $k = k + 1$ и переходим к шагу 2, модифицируя при этом Ω или некоторую его часть.

Заметим, что в приведенной схеме МЦТПТ фактически отсутствует останов по уменьшению объема. Это не совсем случайно. Дело в том, что принятая схема аппроксимации множества экстремумов простыми телами позволяет достаточно хорошо рассчитывать нижнюю оценку f^* посредством последовательных уточнений ее в каждом из преобразованных пространств. Следовательно, используя рекордную точку и нижнюю оценку f^* , внутренний алгоритм для задачи (1) обеспечивает конструктивный критерий останова в терминах точности ее решения по функционалу.

Кроме этого, к Ω отнесены и определяющие многогранное множество Q в X гиперплоскости, образы которых в Y_k задают отсекающие гиперплоскости для его образа $Q_k = B_k^{-1}Q$ в Y_k и также требуют соответствующего пересчета и хранения. Но для оценки ОИ-трудоемкости МЦТПТ это не существенно, поскольку для линейных неравенств общего вида это увеличивает число отсекающих гиперплоскостей на некоторую константу порядка n . Когда многогранник Q в X задан параллелепипедом $Q = \{x : l_i \leq x_i \leq u_i, i = 1, \dots, n\}$, то такого хранения не требуется, так как в этом случае нормали к отсекающим образ Q в Y_k гиперплоскостям будут задавать строки матрицы B_k .

Для итерации МЦТПТ ОИ-трудоемкость определяется ОИ-трудоемкостью шагов 1–6, где шаг 1 определяет внутренний алгоритм, шаги 2–6 – внешний. Шаг 3 требует $O(n^2)$ арифметических операций, шаги 4 и 5 – $O(n^3)$ в самом наихудшем случае, т. е. при использовании в качестве локализаторов эллипсоидов, параллелепипедов или симплексов самого общего вида. Поэтому обеспечив легкий выбор внешним алгоритмом очередного локализатора (шаг 2) и для внутреннего алгоритма ОИ-трудоемкость порядка $O(n^3)$, получим общую ОИ-трудоемкость итерации МЦТПТ не более $O(n^3)$. Если при этом для МЦТПТ на каждом шаге сможем гарантировать уменьшение объема в $(1 - 1/e)$ раз, то он будет реализовывать теоретически оптимальную оценку по последовательности рекордов минимизируемой функции, и его общая ОИ-трудоемкость будет – $O(n^4)$ арифметических операций. Используя специальные локализаторы одного из указанных типов и учитывая тот факт, что работа внутреннего алгоритма нужна не на каждой итерации, для МЦТПТ практически можно обеспечить в среднем ОИ-

трудоемкость итерации – $O(n^2)$.

Приведенная достаточно общая схема МЦТПТ позволяет, конкретизируя внутренний и внешний алгоритмы, получать для решения задачи (1) вполне конструктивные, сходящиеся в смысле рекордов по функции методы. Так, если ориентироваться на выбор локализатора одного и того же типа, то получим метод описанных эллипсоидов (МОЭ), метод описанных параллелепипедов (МОП) или же метод описанных симплексов (МОС). Кроме выбора локализаторов она требует и определенного механизма построения отсекающих гиперплоскостей (внутренний алгоритм), который бы обеспечивал легкое построение эффективных требуемых локализаторов. Ниже эти вопросы рассмотрим более подробно, и при их обсуждении главное внимание будем уделять легко реализуемым алгоритмам в рамках этой схемы, ориентируясь на использование локализаторов специального типа, с тем, чтобы обеспечить практическую ОИ-трудоемкость итерации – $O(n^2)$ арифметических операций.

Принятый подход к созданию искомого алгоритма не является чем-то совершенно новым. Такого рода алгоритмы, для которых в качестве локализаторов используются эллипсоиды, рассматривались в [8], [9]. Их принято называть методами эллипсоидов с более глубокими отсечениями, и в отличие от МЦТПТ они используют только один критерий для установления их сходимости – достаточное уменьшение объема эллипсоидов, локализующих множество экстремумов. Так, в [8] рассмотрены алгоритмы, основанные на эллипсоидах минимального объема описанных вокруг следующих выпуклых тел: сегмент, слой и s -пирамида. Но они не используют информацию, полученную на предыдущих шагах. Алгоритм в [9] также основан на эллипсоиде минимального объема, содержащего шаровой слой, и уже использует полученную ранее информацию. Но параметры шарового слоя для него вычисляются на основании информации только о двух гиперплоскостях и вряд ли при этом следует ожидать хороших оценок скорости сходимости. Поэтому и не удивительно, что вычислительные эксперименты давали незначительные преимущества этих методов по отношению к классическому варианту МЭ.

Что касается методов эллипсоидов с более глубокими отсечениями, их основной недостаток заключается в том, что, уделяя главным образом внимание технике описания эллипсоидов оптимального объема, содержащих различного рода выпуклые тела, они практически игнорируют вопросы «хорошего» построения этих тел. Так константа, характеризующая уменьшение объема для эллипсоида оптимального объема, содержащего шаровой слой, существенно зависит от двух параметров

– расстояния от центра шара до нижней границы шарового слоя и от самой его толщины. В этой связи естественно уделить внимание увеличению первого и уменьшению второго параметров, для чего явно недостаточно использования только двух отсекающих гиперплоскостей. Поэтому в дальнейшем при обсуждении конкретных вариантов МОЭ ограничимся достаточно простыми локализаторами и главное внимание уделим вопросам их эффективного построения.

О внешнем алгоритме. Что из указанных тел выбрать в качестве локализатора множества экстремумов? Предпочтительнее остальных является эллипсоид, так как при этом обеспечивается возможность на каждой итерации внешнего алгоритма уменьшить объем области локализации множества экстремумов, используя информацию о субградиенте только в одной точке – центре шара в преобразованном пространстве, содержащего образ множества экстремумов (классический МЭ). Из этих соображений в качестве локализатора можно избрать и симплекс.

Но, кроме того, что он теоретически хуже позволяет уменьшать объем, чем эллипсоид, он также обладает и рядом недостатков, которые проявляются в МС. Так, основная проблема в МС связана со сложностью перехода от симплекса в n -мерном пространстве к симплексу в пространстве меньшей размерности, когда решение находится в подпространстве пространства R^n . В то же время этот недостаток легко устраним для эллипсоида и фактически приводит к проектированию на подпространство меньшей размерности. При этом сохраняется возможность рассуждений на уровне шара в подпространстве пространства R^n .

Кроме этого, локализирующий множество экстремумов эллипсоид преобразовывается в шар, для которого обеспечивается минимум дополнительной информации об локализаторе, в том смысле, что достаточно знать центр шара в преобразованном пространстве (на самом деле достаточно знать его прообраз в X) и его радиус, что немаловажно с точки зрения информационной трудоемкости МЦТТТ.

Итак, пусть мы выбрали в качестве локализатора эллипсоид. Тогда для обеспечения простоты итерации внешнего алгоритма достаточно отказаться от рассмотрения сложных оптимальных эллипсоидов, порожденных, например, таким телом, как S -пирамида [8], и ограничиться эллипсоидами специального вида $\text{Ell}(\xi, a, b)$, т. е. такими, для которых в R^n длина полуоси в направлении ξ равна a , а длины полуосей в $(n-1)$ ортогональных к ξ направлениях одинаковы и равны b . Это обусловлено тем, что для его преобразования в шар достаточно одномерного оператора растяжения пространства [10].

Простое построение таких эллипсоидов в R^n для шара $S(y_0, r)$ допускают такие тела, как шаровой слой – тело, полученное в результате пересечения шара $S(y_0, r)$ и множества вида

$$PU(y_0, \xi, a_1, a_2) = \left\{ y : a_1 \leq (y - y_0, \xi) \leq a_2 \right\},$$

и n -мерный ограниченный круговой конус – пересечение шара $S(y_0, r)$, $PU(y_0, \xi, a_1, a_2)$ и множества вида

$$KS(y_0, \xi, a) = \left\{ y : (y - y_0, \xi) \geq a \|y - y_0\| \right\}.$$

Здесь ξ – ненулевой вектор единичной длины, a , a_1 и a_2 – скаляры, такие, что $0 \leq a_1 \leq a_2 \leq r$ и $0 < a < 1$. $PU(y_0, \xi, a_1, a_2)$ условимся называть плоским слоем в R^n в направлении ξ , $KS(y_0, \xi, a)$ – n -мерным круговым конусом в направлении ξ . Рациональному построению этих тел и следует уделить главное внимание. Для этого целесообразно использовать такие выпуклые тела в R^n , как $S(m)$ -пирамида [8] (тело, являющееся пересечением шара и полупространств, определяемых m ($m \leq n$) линейно независимыми субградиентами) и симплекс, которые позволяют эффективно решать эти задачи.

Пусть $S(2)$ -пирамида в R^n получена в результате пересечения шара $S(y_0, r)$ и двух полупространств P_1 и P_2 , определяемых векторами нормалей p_1 и p_2 . Пусть центр шара y_0 принадлежит границам полупространств P_1 и P_2 и угол между векторами p_1 и p_2 – тупой, т. е. $(p_1, p_2) < 0$. Тогда, согласно [11], существует эллипсоид типа $\text{Ell}(\xi_1, \xi_2, a, b, r)$ с центром в y_0 , содержащий эту $S(2)$ -пирамиду, и его объем меньше, чем объем шара $S(y_0, r)$. Константа, характеризующая уменьшение его объема по отношению к объему шара, зависит только от косинуса угла между векторами p_1 и p_2 . Здесь a и b – длины полуосей эллипсоида в взаимно-ортогональных направлениях ξ_1 и ξ_2 , r – длины полуосей в $(n - 2)$ – ортогональных к ξ_1 и ξ_2 направлениях. Преобразование $\text{Ell}(\xi_1, \xi_2, a, b, r)$ в шар требует двух последовательных растяжений пространства и по смыслу оно близкое к преобразованию, применяемому в r -алгоритмах [10].

В отличие от эллипсоида типа $\text{Ell}(\xi, a, b)$ он позволяет уменьшать объем области локализации множества экстремумов, не сдвигаясь из центра шара, и это обстоятельство обеспечивает достаточно эффективный механизм для уменьшения объема $S(m)$ -пирамиды, когда между определяющими ее субградиентами взаимно тупые углы, посредством погружения ее в различного рода $S(2)$ -пирамиды. Этот факт предопределяет преимущество $\text{Ell}(\xi_1, \xi_2, a, b, r)$ по отношению к эллипсоиду оптимального объема, содержащему $S(2)$ -пирамиду, центр которого будет

находиться внутри $S(2)$ -пирамиды, что приводит к необходимости каждый раз искать новую $S(m)$ -пирамиду.

Если ограничиться для внешней аппроксимации множества экстремумов локализаторами типа $\text{Ell}(\xi, a, b)$ и $\text{Ell}(\xi_1, \xi_2, a, b, r)$, в рамках МЦТПТ получим МОЭ, обеспечив при этом достаточную простоту итерации внешнего алгоритма. Так, шаги 3–5 потребуют $3n(n + (2/3)m)$ арифметических операций умножения и приблизительно столько же сложений для $\text{Ell}(\xi, a, b)$ и в два раза больше для $\text{Ell}(\xi_1, \xi_2, a, b, r)$. Здесь m – число хранимых субградиентов.

Насколько реально при этом создание МОЭ, обладающего теоретически оптимальной оценкой скорости сходимости или даже лучше? Чтобы обеспечить уменьшение объема области локализации множества экстремумов на каждой из итераций внешнего алгоритма не менее чем в $(1 - 1/e)$ раз, вполне достаточно для шара радиуса r гарантировать выполнение одного из следующих условий. Для шарового слоя – либо расстояние от нижней его границы до центра шара будет порядка r/n , либо его ширина будет порядка r/\sqrt{n} . Для $S(2)$ -пирамид, которые допускают построение эллипсоида типа $\text{Ell}(\xi_1, \xi_2, a, b, r)$, достаточно, чтобы косинус угла между определяющими их векторами был меньше, чем $-\sqrt{\frac{2}{e} - \frac{1}{e^2}} = -0.775$. Выполнение одного из этих условий представляется не таким уж невозможным, если использовать информацию о функции в ряде точек. Построение же эллипсоида, содержащего n -мерный ограниченный круговой конус, или эллипсоида содержащего шаровой слой в расчете на «хорошее» отсечение от центра шара для следующего шага внешнего алгоритма позволяют в значительной мере улучшить эту оценку.

Такой вариант МОЭ может служить одной из возможных интерпретаций, которая геометрически достаточно наглядна, для методов с растяжением пространства [10], если, согласно [12] операцию растяжения пространства понимать как преобразование такого рода эллипсоидов в шары. На этом пути легко объясняются как растяжение пространства аргументов, так и его сжатие. Так для $\text{Ell}(\xi_1, \xi_2, a, b, r)$ присутствуют оба этих элемента. Для $\text{Ell}(\xi, a, b)$, когда он содержит шаровой слой, всегда требуется растяжение пространства аргументов. Когда он содержит n -мерный ограниченный круговой конус, при одних условиях требуется растяжение пространства аргументов, а при других – его сжатие.

Следует отметить, что, исходя из уменьшения объема области локализации множества экстремумов, главенствующая роль должна принадлежать эллипсоидам типа $\text{Ell}(\xi, a, b)$. Привлечение же эллипсоидов

типа $\text{Ell}(\xi_1, \xi_2, a, b, r)$ целесообразно только тогда, когда они очень хороши для уменьшения объема, как в упомянутых выше случаях. Положить $\text{Ell}(\xi_1, \xi_2, a, b, r)$ в основу методов разумнее при интерпретации их как методов типа отсечений в рамках методов феевского типа, как в [11], используя при этом $\text{Ell}(\xi, a, b)$ в том случае, когда не удастся легко улучшить рекордное значение $f(x)$. При этом можно ограничиться хранением незначительного числа субградиентов.

Но приведенные в пользу эллипсоида доводы не значат, что остальные тела из указанных нецелесообразно использовать в качестве локализаторов. Так, например, естественно использовать в качестве локализаторов параллелепипед, грани которого параллельны осям координат, или же симплекс, n граней которого параллельны осям координат, пусть они даже будут служить более грубыми локализаторами множества экстремумов. В этом случае регуляризация локализаторов использует линейное преобразование пространства, которое фактически сводится к масштабированию переменных задачи. Это позволяет обойтись запоминанием n элементов, в отличие от того, что при использовании эллипсоида их требуется $n \times n$, и, кроме того, при решении внутренних подзадач использовать мощный арсенал средств для работы с разреженными матрицами.

Что касается рекомендаций относительно выбора локализаторов для внешней аппроксимации множества экстремумов, то здесь сложно давать какие-то готовые рецепты. Ответ на вопрос о том, на какой из методов МОЭ, МОП или МОС в рамках МЦТПТ ориентироваться, далеко не однозначен. У каждого из них имеется ряд преимуществ, которые могут служить недостатками для других. Так, например, с общих геометрических позиций более рационально использовать эллипсоид или параллелепипед, так как они являются центрально-симметричными телами в R^n . Симплекс таковым не является. При этом, если для эллипсоида простота его построения связана с переходом к шару, то в случае параллелепипеда такой простоты не наблюдается при переходе к кубу. Но если ориентироваться на построение параллелепипеда даже самого общего вида с тем, чтобы его осями всегда были образы нормалей к отсекающим гиперплоскостям в X , то можно сохранить возможность работы с разреженными матрицами, т. е. максимально учесть структурные особенности задачи. В то же время симплекс позволяет легко работать с неправильными симплексами в X , что характерно для различных модификаций метода симплексов Нелдера-Мида [13], сохраняя при этом возможность рассуждений в преобразованном пространстве на уровне правильного симплекса.

Интересным также представляется использование в качестве локализаторов всех рассмотренных тел в различных комбинациях. На одном из таких вариантов, который использует в качестве локализаторов все три тела, немного подробнее остановимся в дальнейшем при обсуждении проблемы «рестарта» МЦТПТ.

О внутреннем алгоритме. Для создания алгоритмов в рамках МЦТПТ требуется и механизм построения эффективных отсекающих гиперплоскостей, поэтому попытаемся прояснить некоторые моменты, связанные с внутренним алгоритмом. При этом будем учитывать, что от его работы требуется подготовка информации для внешнего алгоритма с тем, чтобы обеспечить легкое построение локализаторов. Так, например, для эллипсоида типа $\text{Ell}(\xi, a, b)$ требуется построение «хороших» шаровых слоев и n -мерных ограниченных круговых конусов, для эллипсоида типа $\text{Ell}(\xi_1, \xi_2, a, b, r)$ – эффективных $S(m)$ -пирамид специального типа.

При обсуждении внутреннего алгоритма будем исходить из следующих соображений.

Во-первых, он должен строить эффективные отсекающие гиперплоскости для внешнего алгоритма.

Во-вторых – известна некоторая точка x_k , образ которой $y_k = A_k x_k$ является центром тяжести геометрически простого локализатора множества экстремумов в преобразованном пространстве аргументов $Y_k = A_k X$.

Здесь A_k – невырожденная матрица размера $n \times n$, полученная в результате последовательных регуляризаций локализаторов, аппроксимирующих множество экстремумов. Поэтому вполне естественно вопрос о построении эффективных отсекающих гиперплоскостей рассматривать в преобразованном пространстве относительно точки y_k .

Не ограничивая общности, основные принципы внутреннего алгоритма для задачи (1) рассмотрим на примере исходного пространства аргументов X – области определения $f(x)$. Тем более, что эта процедура требуется на начальном шаге МЦТПТ, когда никакой информации об поведении $f(x)$ еще не имеется. Перенос же ее в преобразованное посредством линейного оператора пространство аргументов осуществляется сравнительно просто и приводит к этим же задачам на наборе векторов, которые получены в результате применения линейного оператора к векторам исходного пространства.

С учетом вышесказанного построим для нашего случая некоторую «идеализированную» процедуру, позволяющую строить эффективные отсекающие гиперплоскости для задачи (1), используя для этого значе-

ния функции и ее субградиента в точке. Назовем ее ИЗМЕРИТЕЛЬ и под измерением будем понимать вычисление в точке x_i значений $f(x_i)$ и $\partial f(x_i)$, т. е. фактически обращение к оракулу первого порядка.

Пусть перед началом вычислений для МЦТПТ мы имеем некоторую точку x_0 , которая принадлежит многограннику Q и является центром тяжести геометрически простого локализатора L_0 , который содержит множество X^* задачи (1). $f(x_0)$, $\partial f(x_0)$ – вычисленные в x_0 значения функции и субградиента, $f_0^* = f(x_0)$ – рекордное значение $f(x)$.

Опишем шаг процедуры ИЗМЕРИТЕЛЬ. Пусть уже имеется m гиперплоскостей, заданных субградиентами, вычисленными ранее, т. е. $(x_i, \partial f(x_i), f(x_i), i = 0, 1, 2, \dots, m-1)$, f_0^* – достигнутое рекордное значение $f(x)$ для задачи (1). Тогда, в качестве первой из точек возьмем решение задачи, как в методе отсекающей гиперплоскости [14], с тем отличием, чтобы точка минимума кусочно-линейной аппроксимации $f(x)$, построенная по имеющейся информации, принадлежала локализатору L_0 :

$$\min t, \quad (7)$$

$$f(x_i) + (\partial f(x_i), x - x_i) \leq t, \quad i = 0, 1, \dots, m-1, \quad (8)$$

$$x \in L_0, \quad x \in Q. \quad (9)$$

Обозначим эту точку через x_A и оптимальное значение t^* через f_A . Так как $f^* \geq f_A$, то f_A может служить нижней оценкой f^* для задачи (1). Кроме того, точка x_A – строго внутренняя точка многогранного множества, заданного ограничениями вида (8), если параметр $t = f_0^*$.

Используя неравенство $f_0^* \geq f^* \geq f_A$, определим некий «зазор» $\Delta = f_0^* - f_A$, который локализует оптимальное значение f^* задачи (1). Его целесообразно использовать для построения эффективных отсекающих гиперплоскостей относительно точки x_0 . Здесь напрашивается проектировать точку x_0 на множество линий уровня полученной аппроксимации $f(x)$, задавшись некоторым $f_\Delta = f_0^* - \Delta/2$. Поэтому в качестве второй точки (обозначим ее x_B) возьмем решение следующей задачи:

$$\min \|x_0 - x\|^2, \quad (10)$$

$$f(x_i) + (\partial f(x_i), x - x_i) \leq f_\Delta, \quad i = 0, 1, \dots, m-1, \quad (11)$$

$$x \in L_0, \quad x \in Q. \quad (12)$$

Очевидно, что отрезок в R^n , который соединяет точки x_B и x_A , целиком принадлежит многогранному множеству, описанному ограничениями вида (8), если параметр $t = f_0^*$. Поэтому целесообразно проводить измерения в точках этого отрезка. Для определенности выберем три измерения и будем проводить их последовательно в точках x_B , $(x_A + x_B)/2$ и x_A . Очевидно, что если гиперплоскость с нормалью $\xi = -\frac{\partial f(x_B)}{\|\partial f(x_B)\|}$, проведенная в точке x_B со сдвигом в направлении ξ на величину $\frac{f(x_B) - f_A}{\|\partial f(x_B)\|}$ отсекает точку x_A , то измерения можно прекратить, так как в этом случае можно улучшить f_A и, следовательно, уменьшить «зазор» Δ . Это же касается и точки $(x_B + x_A)/2$, если измерения не прекратились ранее. В наихудшем случае это приводит к трем вычислениям $f(x)$ и $\partial f(x)$. К очередному шагу процедуры ИЗМЕРИТЕЛЬ переходим, добавив к имеющимся субградиентам вычисленные описанным выше способом и уточнив при этом f_0^* , если это возможно.

Измерения, производимые процедурой ИЗМЕРИТЕЛЬ, эффективные в том смысле, что $f(x)$ и $\partial f(x)$ вычисляются в той области многогранного множества Q , в которой имеется недостаточно информации об поведении $f(x)$, и не вычисляются там, где заранее известно, что значение функции больше полученного рекорда. Более того, если из задач (7)–(9) и (10)–(12) убрать ограничение $x \in L_0$ и дополнить модифицированную таким образом процедуру ИЗМЕРИТЕЛЬ некоторым правилом перехода в точку x_{k+1} (например, в точку улучшения рекорда f_k^*) получим сходящийся «в пределе» алгоритм для решения задачи (1) без преобразования пространства.

Для точки x_0 , если в ней достигается $f(x_0) = f_0^*$, процедура ИЗМЕРИТЕЛЬ фактически реализует поиск направления убывания $f(x)$ из этой точки, если оно существует, и в пределе напоминает процедуры, которые используются в ε -субградиентных методах [15]. В процессе вычислений она существенно использует значения функции в точках вычисления субградиентов и нижнюю оценку f^* – параметры, использование которых для процедур в [15] сильно затрудняет регулировку ε . Если остановиться на использовании процедуры типа ИЗМЕРИТЕЛЬ для внутреннего алгоритма, то естественно при этом для него в качестве критерия останова выбрать условие, аналогичное, как и в ε -субградиентных методах – $0 \in \partial_\varepsilon f(x)$, где ε легко подсчитывается на основании полученной в процессе вычислений информации.

Но в практическом плане ИЗМЕРИТЕЛЬ приводит к решению трудоемких подзадач (7)–(9) и (10)–(12) и обеспечить при этом просто-

ту итерации МЦТПТ невозможно. Поэтому целесообразно заменить ее процедурой, которая пусть делала бы измерения более грубо, но была просто реализуема. При этом желательно, чтобы она реализовывала их идейно, как можно ближе к процедуре ИЗМЕРИТЕЛЬ. Так как сходимость МЦТПТ обеспечивается за счет итераций внешнего алгоритма, то нет особой надобности в точном решении подзадач (7)–(9) и (10)–(12), поэтому логично заменить их некоторой процедурой, которая позволяла бы одновременно получить приближенные решения как первой, так и второй подзадач. При этом она должна поставлять содержательную информацию для выбора локализаторов внешним алгоритмом.

Такой механизм легко обеспечить на уровне анализа некоторых выпуклых ограниченных тел в R^n , полученных в результате пересечения локализатора и $K(m)$ -конуса в R^n , который задан пересечением не более, чем m ($m \leq n$) полупространств вида (8), определяемых линейно независимыми субградиентами $\partial f(x_i)$, $i = 1, 2, \dots, m$. В качестве полупространств, задающих $K(m)$ -конус, могут использоваться и полупространства, которые задают выпуклый многогранник Q . Для $K(m)$ -конуса точку, которая принадлежит пересечению границ всех m полупространств и при этом находится на кратчайшем расстоянии от центра локализатора x_0 , условимся называть его вершиной относительно x_0 .

Кратко объясним смысл производимых при этом измерений на примере $K(m)$ -конуса, построенного по субградиентам $f(x)$, т. е. фактически в предположении, что $X^* \subset \text{int } Q$. В случае, если решение задачи (1) достигается на границе Q , механизм измерений остается таким же, с тем отличием, что формирование $K(m)$ -конуса требует привлечения полупространств, задающих некоторые из границ Q .

Пусть выбраны m линейно независимых субградиентов, вычисленных ранее, т. е. $(x_i, \partial f(x_i), f(x_i), i = 1, 2, \dots, m)$. Рассмотрим систему линейных уравнений

$$f(x_i) + (\partial f(x_i), x - x_i) = t, \quad i = 1, 2, \dots, m, \quad (13)$$

где t – параметр. Решение системы (13) имеет следующий вид относительно точки x_0 :

$$x(t) = x_0 + G(G^T G)^{-1}(te + a), \quad (14)$$

и для определяемого этими субградиентами $K(m)$ -конуса задает его вершину относительно точки x_0 при разных значениях t . Здесь G – матрица размера $n \times m$, столбцами которой являются $\partial f(x_i)$, $i = 1, 2, \dots, m$;

a – m -мерный вектор, i -ая компонента которого для всех $i = 1, 2, \dots, m$ равна $(\partial f(x_i), x_i - x_0) - f(x_i)$; e – m -мерный вектор, все компоненты которого равны единице.

Управление параметром t позволяет легко анализировать $K(m)$ -конус при разных значениях t . Так, при $t = f_0^*$ он будет содержать множество X^* . Формально это равносильно анализу упрощенной задачи квадратичного программирования типа (10)–(12) с m ограничениями вида (11) без учета ограничений (12) при различных значениях параметра f_Δ . При этом вектор $u(t) = (G^T G)^{-1}(te + a)$ для нее играет роль множителей Лагранжа и позволяет легко проверять, является ли вершина $x(t)$, вычисленная согласно (14), ее решением. Если $u(f_0^*) \geq 0$, то гиперплоскость с нормалью $\frac{x(f_0^*) - x_0}{\|x(f_0^*) - x_0\|}$, проведенная на расстоянии $\|x(f_0^*) - x_0\|$ от точки x_0 отсекает множество X^* задачи (1). Минимальное значение параметра $t = t^*$, при котором $u(t^*) \geq 0$, и пространство, заданное гиперплоскостью с нормалью $\frac{x(t^*) - x_0}{\|x(t^*) - x_0\|}$ на расстоянии $\|x(t^*) - x_0\|$ от точки x_0 , не имеет общих точек с локализатором L_0 , может служить уточнением нижней оценки f_A для f^* . Для проведения измерений подходящим является отрезок $[x(f_0^* - \delta), x(f_A)]$, где $0 < \delta \leq \Delta/2$, если вершина $x(f_0^* - \delta)$ не отсекается от многогранного множества типа (8) при $t = f_0^* - \delta$ какой-либо из гиперплоскостей, не включенных в рассматриваемый $K(m)$ -конус. В противном случае можно перейти к очередному $K(m_1)$ -конусу, увеличивая при этом расстояние $\|x_0 - x(f_0^* - \delta)\|$.

Такой механизм позволяет строить эффективные отсекающие гиперплоскости (в том смысле, что и процедура ИЗМЕРИТЕЛЬ), обойдясь незначительным числом арифметических операций. Так, самая трудоемкая из них, обеспечивающая переход от $K(m)$ -конуса к $K(m+1)$ -конусу, требует порядка $m \times n$ арифметических операций. При этом правило, согласно которому следует включить очередной субградиент с тем, чтобы измерения были эффективными, геометрически достаточно «прозрачно» и использует как значения функции, так и субградиенты в точках. Этот механизм легко переносится в преобразованное посредством линейного оператора A_k пространство аргументов $Y_k = A_k X$, где A_k – невырожденная матрица размера $n \times n$. Для $K(m)$ -конуса в Y_k его вершина $y(t)$ относительно точки y_k , которая является центром геометрически простого локализатора $L_k \in Y_k$, вычисляется как в (14) с тем отличием, что векторы $\partial f(x_i)$ заменяются на векторы $\partial \varphi_k(y_i) = (A_k^{-1})^T \partial f(x_i)$ для всех $i = 1, 2, \dots, m$.

Основанная на таком принципе процедура может быть интерпретирована, как ε -субградиентная в соответствующем пространстве аргументов, где ε выбирается, исходя из рекорда по функции и нижней оценки f^* , с тем отличием, что решение внутренних подзадач (переход от $K(m)$ -конуса к $K(m_1)$ -конусу) обеспечивают итерации симплексного типа. При этом в качестве критерия останова для нее может быть использовано как одно из условий:

- а) $f_k^* - f_A \leq \gamma_1 r \|\partial\varphi_k(y_k)\|$;
- б) $\|y_k - y(f_k^* - \varepsilon)\| \geq \gamma_2 r$;
- в) максимальное число вычисленных субградиентов;
- г) ограничение на трудоемкость внутреннего алгоритма — $O(n^3)$;

так и различного рода их комбинации. Здесь r — радиус шара минимального объема в преобразованном пространстве аргументов, содержащего локализатор, γ_1, γ_2 — скаляры, такие что $0 < \gamma_1 \leq 0.1, 1/n \leq \gamma_2 < 1$. Но так как эта процедура реализует измерения более грубо, чем ИЗМЕРИТЕЛЬ, то ее можно рассматривать относительно нескольких точек в R^n , например, центра тяжести локализатора и точки, в которой достигается рекорд f_k^* .

Так как процедура построения эффективных отсекающих гиперплоскостей основана на работе с ограниченными выпуклыми телами, то отсюда автоматически следует ответ на вопрос об информации для внешнего алгоритма. При использовании внешнего локализатора типа параллелепипеда или симплекса таковой может служить информация о вершинах некоторых локализующих множество экстремумов $K(n)$ -конусов и списке образующих их субградиентов. Но в качестве основной целесообразно ориентироваться на информацию о ширине множества экстремумов в R^n в направлениях, заданных вычисленными субградиентами, т. е. на плоские слои вида $PU(y_0, \xi, a_1, a_2)$, локализующие множество X^* . Она легко переносится в преобразованное пространство аргументов Y_k для образа множества X^* , т. е. $Y_k^* = A_k X^*$, и ее вычисление и уточнение не представляет особых проблем для ряда выпуклых тел в R^n , локализующих множество экстремумов. Так, например, это очень легко для шара. В ряде случаев для этих целей подходит $S(m)$ -пирамида, которая является пересечением шара и $K(m)$ -конуса. $S(m)$ -пирамиды легко строить на уровне правильных локализаторов в R^n . Но наиболее подходящим телом для вычисления ширины множества в различных направлениях в R^n является локализующий это

множество симплекс. $K(n)$ -конус с вершиной $y(f_k^*)$ относительно точки y_k позволяет посредством добавления $(n + 1)$ -ой отсекающей гиперплоскости легко анализировать различные симплексы, локализирующие $Y^* = A_k X^*$ для задачи (1).

Вычисление ширины множества экстремумов в некотором направлении требует для симплекса $O(n^2)$, для $S(m)$ -пирамиды — $O(nm)$ арифметических операций. Следовательно, если вычислять ее для хранимых субградиентов всего несколько раз для построенных внутренних локализаторов, то это требует не более $O(n^3)$ арифметических операций. На самом деле в практическом плане вычисление ширины множества в различных направлениях упрощается тем, что ее легко совместить с операцией выбора очередного конуса.

Для задачи (1) информация о ширине множества X^* и его образа Y^* в направлениях вычисленных ранее субградиентов содержательна при работе внутреннего алгоритма, как на предмет более богатого выбора отсекающих гиперплоскостей для $K(m)$ -конусов, так и получения более точной нижней оценки f^* . Кроме этого она может быть использована непосредственно для построения локализаторов типа параллелепипедов, оси которого совпадают с n -линейно независимыми субградиентами, а также для построения шарового слоя в направлении субградиента.

Общие замечания. Наиболее приемлемым в рамках схемы МЦТПТ выглядит МОЭ с использованием для внешней аппроксимации множества экстремумов эллипсоидов типа $\text{Ell}(\xi, a, b)$ и $\text{Ell}(\xi_1, \xi_2, a, b, r)$, ориентируясь на работу внутреннего алгоритма, состоящего в построении эффективных локализирующих множество экстремумов плоских слоев в направлениях субградиентов как на базе $S(m)$ -пирамид, так и симплексов. Для начала этого процесса достаточно знать только стартовую точку x_0 и радиус r шара с центром в x_0 , в котором содержится множество экстремумов. При этом априорное задание радиуса даже с большим запасом не столь существенно, так как он легко уточняется на уровне локализирующего множество экстремумов симплекса, содержащегося внутри шара. Это легко сделать как непосредственным уточнением радиуса шара с центром в точке x_0 , так и с помощью описания эллипсоида типа $\text{Ell}(\xi, a, b)$, содержащего n -мерный ограниченный круговой конус. Шаг 2 МЦТПТ очень прост, если ориентироваться на выбор направлений движения, совпадающими с субградиентами в преобразованном пространстве. Если в качестве направления движения выбирать некоторые направления, являющиеся линейной (не обязательно выпуклой) комбинацией субградиентов, то он незначительно усложнится.

При этом вполне реально создание практически эффективных алго-

ритмов для решения задачи (1) со средней оценкой ОИ-трудоемкости итерации $O(n^2)$. Их можно интерпретировать как методы ε -субградиентного типа с растяжением пространства, которое используется для преобразования эллипсоидов в шары и направлено как на обеспечение гарантированной оценки скорости сходимости в смысле рекордов по функции, так и на обеспечение эффективности ε -субградиентной процедуры в преобразованном пространстве аргументов.

Однако, несмотря на близость к ε -субградиентным процедурам, все-таки направленность у внутреннего алгоритма немного другая, в том смысле, что основная цель — не найти направление убывания функции из данной точки, а построить гиперплоскости, позволяющие хорошо аппроксимировать множество экстремумов, и только предельный его вариант можно рассматривать как ε -субградиентную процедуру. Хорошо это или плохо? В качестве аргументации попытаемся привести некоторые из проблем в выпуклом программировании, решение которых вытекает из такой схемы МЦТПТ. В основном, они связаны с известными проблемами в методах отсечений, которые или не разрешены, или очень расплывчато описаны в литературе, а также с некоторыми достаточно общими проблемами для методов в выпуклом программировании.

1. Для методов типа отсечений не решен вопрос об «отсеве» лишних отсекающих гиперплоскостей. Для метода отсекающей гиперплоскости хотя и применяются некоторые правила «отсева» [16], однако они носят эвристический характер и не обоснованы теоретически. Наиболее обоснован отсев гиперплоскостей в агрегатных методах [17], хотя и он не дает гарантий, что мы не выбросим активные. С позиций же МЦТПТ этот вопрос решается достаточно просто. Так, для отсева явно лишней отсекающей гиперплоскости внешним алгоритмом достаточно убедиться, что локализатор полностью содержится в полупространстве, которое она определяет. Эта задача тривиально решается для симплекса и шара в R^n , и чуть сложнее она для куба. Более тонкий отсев лишних гиперплоскостей можно производить и на уровне внутреннего алгоритма при анализе $S(m)$ -пирамид и симплексов, и он будет тем лучше, чем точнее нижняя оценка f^* .

2. Обычно предполагают, что система ограничений, задающая выпуклый многогранник Q , совместна, но это далеко не всегда так. Поэтому желательно иметь возможность на уровне алгоритма получать ответ на вопрос: «Совместна ли система ограничений?». Эта проблема легко решается для многогранника Q , заданного системой линейных неравенств, и сигналом об ее несовместности служит тот факт, что построенный локализатор и полупространство, определяемое каким-то из нера-

венств, не имеют общих точек. В случае, если выпуклое множество Q более сложной структуры, например, $Q = \{f_i(x) \leq 0, i = 1, 2, \dots, m\}$, где $f_i(x)$ – выпуклые функции, этот вопрос решается на уровне отсекающих гиперплоскостей вида $(\partial f_i(x_k), x - x_k) \leq f_i(x_k)$ для $x \in X^*$.

3. Автоматически решается проблема выбора шагового множителя в направлении движения для субградиентных методов. Шаг и направление движения связаны между собой в преобразованном пространстве аргументов и определяются переходом в центр тяжести выбранного локализатора множества экстремумов с меньшим объемом. Так, при использовании эллипсоидов типа $\text{Ell}(\xi, a, b)$ и $\text{Ell}(\xi_1, \xi_2, a, b, r)$ автоматически связываются шаг в направлении движения и коэффициент растяжения пространства аргументов в этом или некоторых других направлениях с такими характеристиками выпуклых задач, как геометрия линий уровня $f(x)$ и геометрия субдифференциального множества в экстремуме.

4. С точки зрения параллельных вычислений МЦТПТ допускает легкое распараллеливание процесса на уровне содержательных подзадач. Так, при анализе $S(m)$ -пирамиды или симплекса каждому из подпроцессов можно поручить работу по уточнению ширины множества экстремумов в определенных для него направлениях, при анализе $K(m)$ -конуса – определение того, какой из $K(m+1)$ -конусов выбрать для проведения очередных измерений. Можно легко распараллелить и пересчет субградиентов при переходе в очередное преобразованное пространство аргументов.

Этот список вопросов можно было бы и продолжить, как, например, восстановление решения прямой задачи по решению двойственной при использовании декомпозиционного подхода, или построение достаточно хорошей нижней оценки для различного рода схем метода ветвей и границ, которые работают в прямом пространстве переменных, и др. Но не будем этим заниматься, а сосредоточим внимание только на одной достаточно общей проблеме – «рестарта» МЦТПТ, так как она приводит к несколько неожиданным выводам.

5. Проблема «рестарта» связана не столько с методами типа отсечений, сколько с методами, использующими линейные преобразования пространства, и для МЦТПТ она обусловлена накоплением ошибок вычислений в матрице преобразования пространства при последовательной регуляризации внешних локализаторов. Для ее решения на уровне МЦТПТ можно привлекать как шар, так и симплекс. Но мы рассмотрим эту проблему на примере использования описанного выше вари-

анта МОЭ для решения задачи (1), когда многогранник Q является параллелепипедом, т. е. $Q = \{x : l_i \leq x_i \leq u_i, i = 1, \dots, n\}$.

Для того, чтобы обеспечить стартовые условия для МОЭ, преобразуем пространство аргументов X , чтобы образ Q в нем был кубом Q_0 . Для этого применим последовательные растяжения пространства в направлениях e_i с коэффициентами $\frac{\max_{i=1}^n (u_i - l_i)}{u_i - l_i}$ для всех $i = 1, \dots, n$. Здесь e_i — n -мерный вектор, i -ая компонента которого равна единице, а все остальные — нулю. В преобразованном пространстве аргументов содержащий Q_0 шар радиуса $r = \max_{i=1}^n (u_i - l_i) \frac{\sqrt{n}}{2}$ с центром, совпадающим с центром куба Q_0 , будет обеспечивать стартовую позицию для МОЭ. Все эти приготовления сводятся к тому, что для старта МОЭ достаточно положить

$$x_0 = \left\{ \frac{l_1 + u_1}{2}, \dots, \frac{l_n + u_n}{2} \right\}; \quad B_0 = \text{diag} \left\{ \frac{u_i - l_i}{\max_{i=1}^n (u_i - l_i)} \right\};$$

$$r_0 = \max_{i=1}^n (u_i - l_i) \frac{\sqrt{n}}{2}.$$

Стартуем в преобразованном пространстве аргументов с внутреннего алгоритма, используя в качестве отсекающих гиперплоскостей и границы параллелепипеда Q , задаваемые векторами нормалей $\frac{B_0^T e_i}{\|B_0^T e_i\|}$. При его работе будем определять ширину $Y^* = B_0^{-1} X^*$ как в направлениях субградиентов, так и в направлениях $\frac{B_0^T e_i}{\|B_0^T e_i\|}$ для всех $i = 1, \dots, n$. Это приводит к незначительному числу дополнительных арифметических операций как для $S(m)$ -пирамиды, так и для симплекса. В результате возможны два исхода: первый — существенное улучшение ширины Y^* в направлениях $\frac{B_0^T e_i}{\|B_0^T e_i\|}$, что равносильно уточнению параллелепипеда Q в X , и второй — этого не произошло. Будем ориентироваться на первый, так как при втором, применяя МОЭ с уточнением ширины $Y^* = B_k^{-1} X^*$ в направлениях $\frac{B_k^T e_i}{\|B_k^T e_i\|}$, всё равно придем к улучшению границ параллелепипеда Q в X .

Применение МОЭ в сочетании с такой схемой уточнения границ параллелепипеда, локализирующего X^* в X , позволяет решить проблему «рестарта» для МЦТТ в том смысле, что уточненный параллелепи-

пед в X может служить разумным начальным приближением для повторного запуска МОЭ, как с полным обновлением информации об отсекающих гиперплоскостях, так и с ее отсеком. При этом такие комбинированные варианты МЦПТ имеют ряд преимуществ при решении стандартных задач математического программирования.

Во-первых, они более конструктивны именно для целочисленных переменных задачи (1), в том смысле, что это позволяет более сильно «зажимать» границы параллелепипеда. Так, например, если для некоторой компоненты x , скажем x_j , получен диапазон ее изменения $0.01 \leq x_j \leq 1.99$, то очевидно, что она может принимать только единственное целое значение $x_j = 1$. В их распоряжении имеется также и нижняя оценка f^* , которую можно только улучшать при переходе в очередное преобразованное пространство аргументов. Целочисленные точки, в которых следует проводить измерения, можно определять посредством округления точек отрезка в X , который является образом отрезка $[y(f_k^*) - y(f_A)]$ в Y_k , с тем, чтобы они принадлежали пересечению локализатора и многогранного множества, заданного ограничениями вида (8), если параметр $t = f_k^*$.

Во-вторых, в рамках такой схемы можно строить более интересные схемы для параллельных компьютеров, которые подразумевают для алгоритмов отказ от привычной итеративной схемы $x_{k+1} = x_k + h_k p_k$. Так, например, центральный процессор, управляя методом типа отсечений, берет на себя функции контроля в X за границами параллелепипеда, нижней оценкой f^* и за шириной множества экстремумов в направлении вычисленных субградиентов. В нужный момент он может использовать для уточнения этих параметров несколько параллельных МОЭ, которые будут работать по различным центрам тяжести. Каждому из подпроцессоров поручается обработка своего локализатора типа эллипсоида. Он снабжается своей матрицей преобразования и его цель – уточнить параметры для центрального процессора. При этом он может время от времени обращаться к центральному процессору как на предмет обмена информацией по параметрам, так и по поводу решения вопросов об очередных измерениях.

В-третьих, на таком пути просматривается и выход из тупика «больших» размерностей для задач математического программирования. Например, для решения задач линейного программирования применение симплекс-метода ограничивается размерами памяти требуемой для хранения матрицы, обратной к базисной. Работа с разреженными матрицами позволяет увеличить размеры решаемых задач симплекс-алгоритмом. Но это характерно только для хорошо структурированных задач,

и в целом не снимает проблем памяти для симплекс-метода. Надежды в этом смысле на проективный алгоритм Дикина-Кармаркара не оправдались. Применение МОЭ с приведенной выше схемой уточнения границ параллелепипеда в X , учитывая, что ограничения типа равенств допускают построение эллипсоидов типа $E(\xi, a, b)$ объема сколь угодно близкого к нулевому, позволяют надеяться на преодоление проблем памяти для алгоритмов при решении задач линейного программирования.

Следовательно, отказ от одной из центральных идей в математическом программировании – идеи наискорейшего спуска и замена ее как можно лучшей аппроксимацией множества экстремумов – не столь уж и бесперспективна.

Кроме этого, следует заметить, что, методы в рамках схемы МЦТПТ применимы для более широкого класса задач, чем задачи выпуклого программирования. Они могут быть использованы для поиска точек, принадлежащих некоторому стационарному множеству, если можно в произвольной точке R^n определять, принадлежит эта точка этому множеству или строить достаточно хорошие отсекающие это множество гиперплоскости. При этом не обязательно даже, чтобы они отсекали и саму эту точку, и тем более, для построения отсекающих гиперплоскостей использовать именно субградиент в точке. В этой связи эти методы применимы для таких задач, как задача отыскания седловых точек выпукло-вогнутых функций, частные случаи задач решения вариационных неравенств, а также специальные классы задач линейной и нелинейной дополнителности.

Автор благодарен профессору Н. З. Шору за внимание к данной работе и за ряд полезных замечаний в процессе ее написания.

Литература

1. Левин А. Ю. Об одном алгоритме минимизации выпуклых функций // Докл. АН СССР. – 1965. – **160**, № 6. – С. 1224–1227.
2. НЕМИРОВСКИЙ А. С., ЮДИН Д. Б. Сложность задач и эффективность методов оптимизации. – М.: Наука, 1979. – 383 с.
3. ЮДИН Д. Б., НЕМИРОВСКИЙ А. С. Информационная сложность и эффективные методы решения выпуклых экстремальных задач // Экономика и мат. методы. – 1976. – Вып.2. – С. 357–359.

4. ШОР Н. З. Метод отсечения с растяжением пространства для решения задач выпуклого программирования // Кибернетика. – 1977. – № 1. – С. 94–95.
5. АЩЕПКОВ Л. Т., БЕЛОВ Б. И., БУЛАТОВ В. П., ВАСИЛЬЕВ О. В., СРОЧКО В. И., ТАРАСЕНКО Н. В. Методы решения задач математического программирования и оптимального управления. – Новосибирск: Наука, 1984. – 323 с.
6. YAMNITSKY B., LEVIN L. A. An old linear programming algorithm runs in polynomial time // 23rd Annual. Symp. Found. Comput. Sci. Chicago III, 3–5 Nov. – Selver Spring, 1982. – P. 327–328.
7. НЕСТЕРОВ Ю. Е. Эффективные методы в нелинейном программировании. – М.: Радио и связь, 1989. – 304 с.
8. ШОР Н. З., ГЕРШОВИЧ В. И. Об одном семействе алгоритмов для решения задач выпуклого программирования // Кибернетика. – 1979. – № 4. – С. 62–67.
9. KONIG H. AND PALLASCHKE D. On Khachian's algorithm and minimal ellipsoids // Numerische Mathematik. – 1981. – **36**. – P. 211–223.
10. ШОР Н.З. Методы минимизации недифференцируемых функций и их приложения. – К.: Наук. думка, 1979. – 199 с.
11. СТЕЦЮК П. И. r -алгоритмы и эллипсоиды // Кибернетика и системный анализ. – 1996. – № 1. – С. 113–114.
12. СТЕЦЮК П. И. К обоснованию сходимости алгоритмов с растяжением пространства // Теория оптимальных решений. – 1995. – С. 4–8.
13. NELDER J. A., MEAD R. A simplex method for function minimization // The Comput. Journal. – 1965. – 7. – P. 308–313.
14. KELLEY J. E. The cutting-plane method for solving convex programs // SIAM. – 1960. – 8, N 4. – P. 703–712.
15. HIRIART-URRUTY J. B., LEMARÉCHAL C. Convex analysis and minimization Algorithms. – Vol. I–II. – Berlin: Springer-Verlag, 1994.
16. ПОЛЯК Б. Т. Введение в оптимизацию. – М.: Наука, 1983. – 384 с.
17. KIWIŁ K. C. Methods of descent for nondifferentiable optimization. – Berlin: Springer-Verlag, 1985. – 1133; Lecture Notes in Math. – 362 p.

К методам эллипсоидов

П. И. Стецюк

Теория оптимальных решений. – 1999. – С. 27–33.

Дана геометрическая интерпретация процессов описанных эллипсоидов, содержащих пересечение шара и гиперплоскости. Они позволяют реализовывать монотонные по объему методы для нахождения стационарных точек специальных задач на базе оператора растяжения пространства.

Идея внешней аппроксимации некоторого стационарного множества монотонно уменьшающимися по объему эллипсоидами дает достаточно универсальный механизм для построения методов решения широкого класса задач. Среди них задача отыскания седловых точек выпукловогнутых функций, частные случаи задач решения вариационных неравенств, специальные классы задач линейной дополнительной, общая задача выпуклого программирования а также задача безусловной минимизации выпуклой функции. Для двух последних задач в [1] обсуждались простые схемы построения методов описанных эллипсоидов (МОЭ), гарантирующие геометрическую скорость сходимости по последовательности рекордов минимизируемой функции $f(x)$, в предположении, что $f(x)$ – произвольная выпуклая функция из класса выпуклых функций.

В то же время многие задачи, которые приводят к необходимости минимизации выпуклых функций, гарантируют выполнение более сильных свойств, чем просто выпуклость. Наряду с классом всех выпуклых функций часто приходится иметь дело с более узкими классами выпуклых функций. Так, например, класс выпуклых функций с ограниченной константой роста [2] содержит выпуклые функции $f(x)$, определенные в конечномерном евклидовом пространстве E^n , для которых найдутся точка $x^* \in X^*$ и константы $M \geq m \geq 1$, такие, что для всех $x \in E^n$ выполняется неравенство

$$m(f(x) - f^*) \leq (x - x^*, \partial f(x)) \leq M(f(x) - f^*), \quad (1)$$

где $\partial f(x)$ – субградиент $f(x)$ в точке x , $f^* = f(x^*)$ – значение минимума $f(x)$. Представляет интерес и частный случай этого класса ($m = M$), к которому относятся: дифференцируемые однородные (с показателем σ) выпуклые функции ($M = m = \sigma$), квадратичные вы-

пуклые функции ($M = m = 2$), кусочно-линейные выпуклые функции с острым минимумом ($M = m = 1$).

Интересен следующий вопрос. Что может дать идея внешней аппроксимации некоторого стационарного множества монотонно уменьшающимися по объему эллипсоидами для специальных классов функций? Если на каждом шаге итерационного процесса придерживаться определенной схемы построения эллипсоида, содержащего пересечение шара и гиперплоскости, то ответ на этот вопрос можно дать как для задачи нахождения точки минимума x^* выпуклой функции, удовлетворяющей (1) при $m = M$, при условии знания f^* , так и для задачи нахождения стационарной точки произвольной квадратичной функции. Ответ оказывается довольно интересен. Для обеих задач можно указать бесконечное число МОЭ, которые на каждом шаге гарантируют уменьшение объема области локализации множества точек минимума (стационарных точек) со сколь угодно большим коэффициентом. При этом «предельные» варианты таких методов, т. е. которые используют бесконечный коэффициент, находят решение указанных задач не более, чем за n шагов, где n – размерность пространства переменных. Почему это так, что лежит в основе указанных процессов, какие схемы методов при этом получаются и будет обсуждаться дальше.

1. Геометрическая интерпретация процессов простых описанных эллипсоидов. Пусть имеется шар

$$S_X(x_0, r_0) = \left\{ x \in X = E^n : \|x - x_0\|^2 \leq r_0^2 \right\}$$

и пусть задана гиперплоскость

$$P_X(p, t) = \left\{ x \in X = E^n : (p, x) - t = 0 \right\},$$

где E^n – евклидово пространство размерности n ($n \geq 2$) со скалярным произведением (\cdot, \cdot) . Пространство X условимся называть исходным пространством аргументов или переменных. Здесь параметры шара $S_X(x_0, r_0)$ – $x_0 \in E^n$ – центр шара, $r_0 \in E^1$ – радиус шара; параметры гиперплоскости $P_X(p, t)$ – $p \in E^n$ – ненулевой вектор (будем называть его вектором нормали к гиперплоскости $P_X(p, t)$), $t \in E^1$ – некоторый скаляр. Для удобства изложения (упрощение формул в ряде случаев) нормированный вектор нормали к гиперплоскости $P_X(p, t)$ – $p / \|p\|$ будем обозначать также через ξ , подразумевая при этом что $\xi = p / \|p\|$.

Тело, полученное в результате пересечения шара $S_X(x_0, r_0)$ и гиперплоскости $P_X(p, t)$, обозначим $W_X(x_0, r_0, p, t)$, т. е.

$$W_X(x_0, r_0, p, t) = \left\{ x \in E^n : x \in S_X(x_0, r_0) \cap P_X(p, t) \right\}.$$

Кроме того предположим, что оно непусто ($W_X(x_0, r_0, p, t) \neq \emptyset$). Это не принципиально, так как факт пересечения шара $S_X(x_0, r_0)$ и гиперплоскости $P_X(p, t)$ (увидим дальше) легко устанавливается путем анализа параметров x_0 , r_0 , p и t .

Что же такое тело $W_X(x_0, r_0, p, t)$? Пусть $S_X(x_1, r_1)$ – шар с центром в точке x_1 , которая есть проекцией точки x_0 на гиперплоскость $P_X(p, t)$, и радиусом $r_1 = \sqrt{r_0^2 - \rho_0^2}$, где ρ_0 – расстояние между точками x_0 и x_1 ($\rho_0 = \|x_0 - x_1\|$). $S_X(x_1, r_1) \equiv S_X(x_0, r_0)$, когда гиперплоскость $P_X(p, t)$ проходит через центр шара $S_X(x_0, r_0)$, т. е. $x_1 = x_0$ и $r_1 = r_0$. Когда $\rho_0^2 > r_0^2$ (задает достаточное условие для останова алгоритмов), то это означает, что $W_X(x_0, r_0, p, t) = \emptyset$, т. е. шар $S_X(x_0, r_0)$ и гиперплоскость $P_X(p, t)$ не пересекаются. В силу того, что точка x_1 есть проекцией точки x_0 на гиперплоскость $P_X(p, t)$, то данная гиперплоскость проходит через центр шара $S_X(x_1, r_1)$. Тело $W_X(x_0, r_0, p, t)$ является пересечением шара $S_X(x_1, r_1)$ и гиперплоскости с нормалью ξ , проходящей через его центр, и будет представлять собой некоторый шар (с центром в x_1 и радиусом r_1) в подпространстве E^{n-1} пространства E^n , причем таким, что ортонормированный базис E^{n-1} лежит в ортогональном дополнении к вектору ξ . Точку x_1 условимся считать центром тела $W_X(x_0, r_0, p, t)$, r_1 – его радиусом, а само тело $W_X(x_0, r_0, p, t)$ еще обозначим $V_X(x_1, r_1, \xi)$. Тело $V_X(x_1, r_1, \xi) \equiv W_X(x_0, r_0, p, t)$ можно заключить в эллипсоид специального типа $\text{ell}(x_1, \xi, a, b)$ с центром в точке x_1 , для которого длина полуоси в направлении ξ равна a ($a > 0$ – произвольно), а длины полуосей в $(n-1)$ – ортогональных к ξ и взаимно ортогональных направлениях одинаковы и равны b , где $b \geq r_1$.

Для преобразования эллипсоида $\text{ell}(x_1, \xi, a, b)$ в шар нужно «растянуть» его в направлении ξ с таким коэффициентом, чтобы длины всех полуосей стали одинаковыми. Для этого достаточно использовать оператор растяжения пространства [2] в направлении ξ с коэффициентом равным отношению длины полуоси b к длине полуоси a :

$$R_\alpha(\xi) = I + (\alpha - 1)\xi\xi^T, \quad \alpha = \frac{b}{a}, \quad (2)$$

где I – единичная матрица размера $n \times n$. В преобразованном оператором (2) пространстве аргументов $Y = R_\alpha(\xi)X$ будем иметь некоторый

новый шар $S_Y(y_1, b)$ радиуса b с центром в точке $y_1 = R_\alpha(\xi)x_1$, которая есть образом точки x_1 из X . Поскольку шар $S_Y(y_1, b)$ есть образом в Y эллипсоида $\text{ell}(x_1, \xi, a, b)$ из X , то он будет содержать образ тела $V_X(x_1, r_1, \xi)$ в пространстве Y , если длины полуосей $\text{ell}(x_1, \xi, a, b)$ выбрать как вышеуказано, т. е. $a > 0$, $b \geq r_1$.

Объем эллипсоида $\text{ell}(x_1, \xi, a, b)$ $\left(\text{vol}(\text{ell}(x_1, \xi, a, b))\right)$ будет равен $v_0 ab^{n-1}$, где v_0 — объем единичного шара в E^n . Отношение объема эллипсоида $\text{ell}(x_1, \xi, a, b)$ к объему исходного шара $S_X(x_0, r_0)$ равно:

$$q = \frac{\text{vol}(\text{ell}(x_1, \xi, a, b))}{\text{vol}(S_X(x_0, r_0))} = \frac{v_0 ab^{n-1}}{v_0 r_0^n} = \left(\frac{a}{r_0}\right) \left(\frac{b}{r_0}\right)^{n-1}. \quad (3)$$

Второй сомножитель в формуле (3) можно сделать меньше или равным единице, выбирая для $\text{ell}(x_1, \xi, a, b)$ длину полуоси b из условия: $r_1 \leq b \leq r_0$. Однако надеяться на существенное уменьшение q за счет сомножителя $\left(\frac{b}{r_0}\right)^{n-1}$ не имеет смысла, так как r_1 может быть очень близким к r_0 . В то же время первый сомножитель $\left(\frac{a}{r_0}\right)$ можно сделать сколь угодно малым в силу произвольного выбора для эллипсоида $\text{ell}(x_1, \xi, a, b)$ длины полуоси a . Это позволяет сделать q сколь угодно малым и означает возможность гарантировать посредством выбора длины полуоси a сколь угодно большое уменьшение объема эллипсоида вида $\text{ell}(x_1, \xi, a, b)$, локализирующего тело $W_X(x_0, r_0, p, t)$.

Учитывая, что при линейном невырожденном преобразовании пространства гиперплоскость переходит в гиперплоскость, и то, что в преобразованном пространстве мы имеем шар некоторого заданного радиуса, цикл таких построений можно замкнуть. Для нахождения точки из множества точек, принадлежащих пересечению некоторого набора гиперплоскостей, это дает возможность строить итерационные процессы описанных эллипсоидов и обосновывать их сходимость по такой характеристике, как монотонное уменьшение объема эллипсоидов, локализирующих это множество. Каждый шаг такого процесса интерпретируется как переход от эллипсоида вида $\text{ell}(y_1, \xi, a, b)$ в текущем преобразованном пространстве к шару в очередном преобразованном пространстве посредством применения оператора растяжения пространства. При этом очередная точка однозначно определяется центром тела $V_Y(y_1, r_1, \xi)$ в преобразованном пространстве переменных.

Следует отметить, что с позиций монотонного уменьшения объема длины полуосей эллипсоида $\text{ell}(x_1, \xi, a, b)$ не обязательно выбирать из

вышеуказанных условий. Главное чтобы $a < r_0 \left(\frac{r_0}{b}\right)^{n-1} = b \left(\frac{r_0}{b}\right)^n$, что гарантирует $q < 1$ в формуле (3). Это значит, что не следует бояться увеличения радиуса шара в преобразованном пространстве переменных, если за счет этого можно добиться более сильного уменьшения объема. Однако такая ситуация более уместна при рассмотрении методов, основанных на оптимальных эллипсоидах, содержащих такие выпуклые области как полушар (в классическом методе эллипсоидов) и «шаровой слой» (в методах эллипсоидов с «глубокими отсечениями») [3]. Вышерассматриваемый случай, хотя и близок к очень тонким шаровым слоям в E^n , все же отличается от последнего в том смысле, что «предельные» формулы оптимального эллипсоида, содержащего такие слои, не дают гарантии неувеличения радиуса шара в преобразованном пространстве аргументов.

При выборе длин полуосей эллипсоида $\text{ell}(x_1, \xi, a, b)$, содержащего тело $V_X(x_1, r_1, \xi)$, естественно следовать такой стратегии. Для полуоси b выбираем ее оптимальное значение ($b = r_1$), т. е. равное минимальному радиусу шара, локализирующего тело $V_X(x_1, r_1, \xi)$. Выбор длины полуоси a осуществляется из условия $0 < a \ll r_1$. Особо следует выделить «предельный» случай ($a = 0$), который допустим в силу произвольного выбора длины полуоси a . В данном случае оператор $R_\alpha(\xi)$ – «вырожден», так как $\alpha = \infty$. Однако это не мешает реализации указанной схемы процессов описанных эллипсоидов, поскольку при реализации методов нужен не сам оператор $R_\alpha(\xi)$, а обратный к нему оператор $R_\beta(\xi)$, $\beta = 1/\alpha$, который определен при $\alpha = \infty$. «Предельный» случай фактически равносителен проектированию на гиперплоскость с нормалью ξ , и именно он гарантирует конечную сходимость методов описанных эллипсоидов, т. е. не более чем за n шагов, где n – размерность пространства переменных. Такие варианты методов удобнее рассматривать как «предельные» при $\alpha = \infty$ и для них характерно бесконечное уменьшение объема.

2. Вышеизложенного достаточно, чтобы строить методы простых эллипсоидов для нахождения стационарных точек x^* некоторого множества X^* , если умеем строить гиперплоскости, проходящие через $x^* \in X^*$. В качестве примеров рассмотрим задачи, когда это просто сделать и когда «предельные» варианты методов простых эллипсоидов сходятся за конечное число шагов.

Пример 1. Пусть требуется решить систему линейных уравнений $Ax = b$, где A – матрица размера $m \times n$, $x \in E^n$, $b \in E^m$. Тогда очевидно, что семейство X^* будут те точки, где $Ax^* = b$. Поэтому, за-

давшись некоторой стартовой точкой x_0 и сделав предположение, что $\|x_0 - x^*\| \leq r_0$, получаем все стартовые условия для реализации процесса описанных эллипсоидов для нахождения точки $x^* \in X^*$, удовлетворяющей следующим ограничениям:

$$(a_i, x^*) = b_i, \quad i = 1, \dots, m, \quad (4)$$

$$\|x_0 - x^*\| \leq r_0, \quad (5)$$

где a_i – векторы строк матрицы A . Выбирая уравнения гиперплоскостей из (4), причем в произвольном порядке, в случае $\alpha = \infty$, не более чем за $\min(m, n)$ шагов, либо придем к точке $x^* \in X^*$, либо получим достаточное условие несовместности системы $Ax = b$ в шаре $S(x_0, r_0)$, задаваемым ограничением (5). Конкретный метод определяется последовательностью выбора гиперплоскостей из (4). Направления растяжения пространства будут совпадать с направлением нормалей гиперплоскостей в преобразованном пространстве аргументов.

Пример 2. Пусть x^* – точка, где достигается минимум выпуклой функции, удовлетворяющей (1) при $m = M$. Тогда равенство

$$m(f(x) - f^*) = (x - x^*, \partial f(x)),$$

которое справедливо для всех $x \in E^n$, дает возможность строить требуемые гиперплоскости, при условии знания f^* . Поэтому данная ситуация такая же как и в примере 1, т. е. задавшись x_0 и $r_0 > \|x_0 - x^*\|$ можно строить семейство методов простых эллипсоидов, предельные варианты которых находят точку x^* не более чем за n шагов. Конкретный метод определяется тем набором точек, где будут вычисляться субградиенты, а растяжения пространства будут производиться в направлении вычисленных субградиентов. Метод, на каждом шаге которого субградиент вычисляется в центре шара текущего пространства переменных, будет соответствовать известному методу с классическим фейеровским шагом в направлении субградиента, который является предельным вариантом семейства методов эллипсоидов с растяжением пространства в направлении субградиента для класса выпуклых функций с ограниченной константой роста [3].

Пример 3. Пусть X^* – множество стационарных точек квадратичной функции $f(x)$, $x \in E^n$. Тогда для всех $x^* \in X^*$ справедливо следующее равенство:

$$(x_h^* - x^*, \nabla f(x) - \nabla f(x(h))) = 0, \quad (6)$$

где $\nabla f(x)$ и $\nabla f(x(h))$ – градиенты $f(x)$ в точках x и $x(h) = x + hp$, а x_h^* – экстремальная точка, полученная из x в направлении p , т. е. точка, где реализуется экстремум $f(x)$ по направлению p (она вычисляется аналитически). Для квадратичных функций соотношение (6) дает возможность строить методы простых эллипсоидов для нахождения $x^* \in X^*$, предельные варианты которых находят x^* не более чем за n шагов. Структура таковых методов будет определяться как набором точек, в которых будет вычисляться градиент, так и некоторым набором линейно независимых направлений. Операция растяжения пространства переменных будет реализовываться по разности двух последовательных градиентов, где первый – вычислен в выбранной для шага точке, а второй – вычислен в точке, где $f(x)$ достигает экстремума из данной точки по выбранному для шага направлению. Применительно к нахождению точек минимума неотрицательно определенных квадратичных функций такие методы не требуют априорного знания f^* . Кроме того r -алгоритм также оказывается методом простых эллипсоидов, только нужно учесть тот факт, что переход в экстремальную точку в направлении градиента дает возможность уменьшить радиус шара, локализирующего множество точек минимума.

Приведенные в примерах 2 и 3 свойства процессов простых эллипсоидов дают основания надеяться на создание эффективных методов минимизации некоторых классов выпуклых функций на базе оператора растяжения пространства. Так, например, если на первых шагах использовать сравнительно небольшие коэффициенты растяжения и увеличивать их по мере уточнения информации об свойствах функции, то для таких процессов, начиная с какого-то момента, можно обеспечить квадратичную скорость сходимости как для кусочно-линейных так и для гладких выпуклых функций. Для этого достаточно посредством использования небольших коэффициентов растяжения обеспечить такую окрестность точки x^* , в которой кусочно-линейная выпуклая функция удовлетворяет (1) при $M \simeq m = 1$, а гладкая выпуклая функция хорошо аппроксимируется квадратичной. В первом случае методы будут использовать операцию растяжения пространства в направлении субградиента, а во втором – операцию растяжения в направлении разности двух последовательных субградиентов, второй из которых получен согласно процедуры (точной или приближенной) спуска по направлению. Если при реализации таких методов использовать B -форму (т. е. корректируется матрица обратного преобразования), то они должны быть численно устойчивыми и при работе с существенно овражными функциями.

В заключение отметим, что в рамках методов простых эллипсоидов можно найти простое объяснение ряду методов по типу сопряженных направлений для квадратичных функций, при этом коэффициенты растяжения пространства будут ограничены как снизу так и сверху. Кроме того, при создании методов можно использовать как априорную оценку расстояния до множества минимумов, так и выбор начальной стартовой точки.

Литература

1. СТЕЦЮК П. И. Метод центров тяжести простых тел // Кибернетика и системный анализ. – 1996. – №5. – С. 117–138.
2. ШОР Н. З. Методы минимизации недифференцируемых функций и их приложения. – К.: Наук. думка, 1979. – 199 с.
3. МИХАЛЕВИЧ В. С., ТРУБИН В. А., ШОР Н. З. Оптимизационные задачи производственно-транспортного планирования. – М.: Наука, 1986. – 264 с.

Об ускорении сходимости методов ЭЛЛИПСОИДОВ

П. И. Стецюк

*Труды XII Байкальской международной конференции. Том 1.
Математическое программирование. – Иркутск, 2001. – С. 61–66.*

Приведен оптимальный эллипсоид, с помощью которого можно строить ускоренные варианты методов эллипсоидов и обосновывать их сходимость по монотонному уменьшению объема эллипсоида, аппроксимирующего множество решений.

Введение

В работах [1, 2] предложен метод эллипсоидов (МЭ). Он основан на том факте, что вокруг полушара в E^n можно описать содержащий этот полушар эллипсоид, объем которого меньше, чем объем самого шара. Здесь и далее E^n – евклидово пространство размерности n со скалярным произведением (\cdot, \cdot) . Коэффициент уменьшения объема на каждом шаге МЭ приблизительно равен $(1 - 1/2n)$.

В работах [3]–[5] были предприняты усилия построить варианты методов эллипсоидов с более сильными коэффициентами уменьшения объема на шаге. Для этого использовались оптимальные эллипсоиды, описанные вокруг сложных выпуклых тел (сегмент, «шаровой» слой и s -пирамида). Однако не были получены эффективные варианты алгоритмов. Так, например, незначительные преимущества над классическим МЭ имеют методы, основанные на использовании оптимального эллипсоида, содержащего «шаровой» слой (тело, полученное в результате пересечения шара $S(x_0, r) = \{x : \|x - x_0\| \leq r\}$ и множества вида $PU(x_0, \xi, h, H) = \{x : h \leq (x - x_0, \xi) \leq H\}$). Здесь $\xi \in E^n$ – вектор единичной длины, h и H – скаляры такие, что $0 \leq h \leq H \leq r$. Это связано с тем, что для того, чтобы на каждом шаге этих методов обеспечить уменьшение объема приблизительно в два раза, требуется найти значение H порядка r/\sqrt{n} , что маловероятно при больших n .

В данной работе приведем оптимальный эллипсоид, с помощью которого можно строить ускоренные варианты методов эллипсоидов, и в ряде случаев он позволяет обеспечить хорошие коэффициенты уменьшения объема аппроксимирующего эллипсоида.

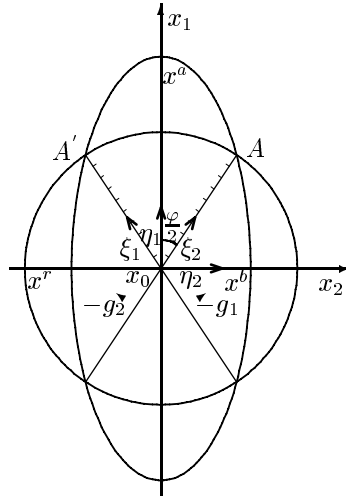


Рис. 1. Эллипсоид $\text{ell}(x_0, a, b, r)$. Здесь $\|x^r - x_0\| = r$, $\|x^a - x_0\| = a$, $\|x^b - x_0\| = b$.

1. Об одном эллипсоиде минимального объема.

Пусть $W(x_0, g_1, g_2, r)$ — выпуклое тело в E^n , полученное в результате пересечения шара $S(x_0, r) = \{x : \|x - x_0\| \leq r\}$ и двух полупространств $P(x_0, g_1) = \{x : (x - x_0, g_1) \leq 0\}$ и $P(x_0, g_2) = \{x : (x - x_0, g_2) \leq 0\}$, где $g_1, g_2 \in E^n$. Кроме того, пусть выполнено условие: $(g_1, g_2) < 0$. Проекция $W(x_0, g_1, g_2, r)$ на плоскость, определяемую g_1 и g_2 , представлена на рисунке 1. Здесь $(\xi_1, g_1) = 0$, $\|\xi_1\| = 1$, $(\xi_2, g_2) = 0$, $\|\xi_2\| = 1$, $(\xi_1, \xi_2) = -\left(\frac{g_1}{\|g_1\|}, \frac{g_2}{\|g_2\|}\right)$.

Среди семейства эллипсоидов с центром в точке x_0 , для которых одна из осей совпадает с направлением $\eta_1 = \frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|}$, а вторая с ор-

тогональным к η_1 направлением $\eta_2 = \frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|}$, рассмотрим эллипсоид оптимального объема, содержащий тело $W(x_0, g_1, g_2, r)$. В системе координат, для которой начало совпадает с центром шара x_0 , ось x_1 с направлением η_1 , ось x_2 с направлением η_2 , а оси x_3, \dots, x_n образуют совместно с x_1 и x_2 взаимно-ортогональную систему, поверхность семейства этих эллипсоидов удовлетворяет следующему уравнению:

$$\frac{x_1^2}{a^2} + \frac{x_2^2}{b^2} + \sum_{i=3}^n \frac{x_i^2}{c_i^2} = 1,$$

где a и b — длины полуосей эллипсоида в направлениях η_1 и η_2 соответственно, c_i — длины полуосей в остальных $(n-2)$ ортогональных к η_1 и η_2 направлениях. Длины полуосей a и b связаны между собой прохождением эллипсоида через точки A и A' , координаты которых $(R \cos \frac{\varphi}{2}, R \sin \frac{\varphi}{2}, 0, \dots, 0)$ и $(R \cos \frac{\varphi}{2}, -R \sin \frac{\varphi}{2}, 0, \dots, 0)$.

Из условия, что эллипсоид должен проходить через точки с координатами re_i^T , $i = 3, \dots, n$, где e_i — вектор, i -ая компонента которого равна единице, а все остальные равны нулю, следует, что $c_i = r$, $i = 3, \dots, n$, поэтому эллипсоид из указанного семейства обозначим $\text{ell}(x_0, a, b, r)$. В [6] показано, что эллипсоид минимального объема $\text{ell}(x_0, a^*, b^*, r)$ имеет следующие длины полуосей

$$\begin{aligned} a^* &= \sqrt{2} r \cos \frac{\varphi}{2} = \\ &= \sqrt{2} r \sqrt{\frac{1 + \cos \varphi}{2}} = r \sqrt{1 + (\xi_1, \xi_2)} = r \sqrt{1 - \left(\frac{g_1}{\|g_1\|}, g_2 \|g_2\| \right)}, \end{aligned}$$

$$\begin{aligned} b^* &= \sqrt{2} r \sin \frac{\varphi}{2} = \\ &= \sqrt{2} r \sqrt{\frac{1 - \cos \varphi}{2}} = r \sqrt{1 - (\xi_1, \xi_2)} = r \sqrt{1 + \left(\frac{g_1}{\|g_1\|}, \frac{g_2}{\|g_2\|} \right)}. \end{aligned}$$

Объем эллипсоида $\text{ell}(x_0, a^*, b^*, r)$ меньше, чем объем шара $S(x_0, r)$, и коэффициент уменьшения объема равен:

$$q^* = \frac{\text{vol}(\text{ell}(x_0, a^*, b^*, r))}{\text{vol}(S(x_0, r))} = \frac{a^* b^*}{r^2} = \sqrt{1 - \left(\frac{g_1}{\|g_1\|}, \frac{g_2}{\|g_2\|} \right)^2} < 1.$$

Следовательно, когда угол между векторами g_1 и g_2 тупой (т. е. $(g_1, g_2) < 0$) то эллипсоид $\text{ell}(x_0, a^*, b^*, r)$ можно использовать для внешней аппроксимации выпуклого тела $W(x_0, g_1, g_2, r)$. Коэффициент q^* зависит от угла между векторами g_1 и g_2 , и чем более тупой этот угол, тем меньше q^* . Легко видеть, что для уменьшения объема в два раза достаточно иметь угол, для которого $\cos \varphi = -\sqrt{3}/2$. Заметим, что такого уменьшения объема не добиться, если при таком угле строить «шаровой слой», содержащий тело $W(x_0, g_1, g_2, r)$, как в направлении $g_1/\|g_1\|$, так и в направлении $g_2/\|g_2\|$.

2. Преобразование эллипсоида

$\text{ell}(x_0, a^*, b^*, r)$ в шар $S(y_0, r)$.

Для того чтобы эллипсоид $\text{ell}(x_0, a^*, b^*, r)$ преобразовать в шар $S(y_0, r)$ радиуса r с центром в точке $y_0 = Tx_0$ преобразованного пространства аргументов (здесь T линейный оператор, действующий из E^n в E^n), достаточно последовательно применить два оператора растяжения пространства, т. е. «растянуть» пространство аргументов в направлении $\eta_1 = \frac{\xi_1 + \xi_2}{\|\xi_1 + \xi_2\|}$ с коэффициентом

$$\alpha_1 = \frac{1}{\sqrt{1 - \left(\frac{g_1}{\|g_1\|}, \frac{g_2}{\|g_2\|} \right)}}$$

и в направлении $\eta_2 = \frac{\xi_1 - \xi_2}{\|\xi_1 - \xi_2\|}$ с коэффициентом

$$\alpha_2 = \frac{1}{\sqrt{1 + \left(\frac{g_1}{\|g_1\|}, \frac{g_2}{\|g_2\|} \right)}}.$$

Направления η_1 и η_2 совпадают с точностью до некоторого множителя с направлениями $\xi = \left(\frac{g_1}{\|g_1\|} + \frac{g_2}{\|g_2\|} \right)$ и $\eta = \left(\frac{g_1}{\|g_1\|} - \frac{g_2}{\|g_2\|} \right)$ соответственно. Следовательно, линейный оператор преобразования пространства имеет следующий вид:

$$T_1(g_1, g_2) = R_{\alpha_1} \left(\frac{\xi}{\|\xi\|} \right) R_{\alpha_2} \left(\frac{\eta}{\|\eta\|} \right),$$

где $R_\alpha(\cdot)$ – оператор растяжения пространства (см. [7]).

Однако для преобразования эллипсоида $\text{ell}(x_0, a^*, b^*, r)$ в шар $S(y_0, r)$ можно применить «одноранговый эллипсоидальный оператор» (см. [8]):

$$T_2(g_1, g_2) = I - \frac{1}{1 - (\xi, \eta)^2} \left(\left(1 - \sqrt{1 - (\xi, \eta)^2} \right) \eta - (\xi, \eta) \xi \right) \eta^T,$$

где $\xi = g_1 / \|g_1\|$ и $\eta = g_2 / \|g_2\|$. Преимущество последнего состоит в том, что с его помощью можно более экономно реализовать методы в B -форме, которые требуют хранения матрицы обратного преобразования.

Заключение

Эллипсоид $\text{ell}(x_0, a^*, b^*, r)$ можно использовать при создании ускоренных вариантов методов эллипсоидов для широкого класса задач: задача выпуклого программирования, задача отыскания седловых точек выпукло-вогнутых функций, частные случаи задач решения вариационных неравенств, специальные классы задач линейной и нелинейной дополнителности. При этом для них вполне возможно обеспечить достаточно хорошую скорость сходимости. Это подтверждают два варианта субградиентных методов [6, 8] для нахождения точки минимума выпуклой функции при априорном знании значения функции в точке минимума. Они оказались достаточно эффективными при работе с существенно овражными функциями, о чем свидетельствуют результаты численных экспериментов [6, 9].

Литература

1. Юдин Д. Б., НЕМИРОВСКИЙ А. С. Информационная сложность и эффективные методы решения выпуклых экстремальных задач // Экономика и мат. методы. — 1976, вып.2. — С. 357–359.
2. ШОР Н. З. Метод отсечения с растяжением пространства для решения задач выпуклого программирования // Кибернетика. — 1977. — № 1. — С. 94–95.
3. ШОР Н. З., ГЕРШОВИЧ В. И. Об одном семействе алгоритмов для решения задач выпуклого программирования // Кибернетика. — 1979. — № 4. — С. 62–67.

4. KONIG H. AND PALLASCHKE D. On Khachian's algorithm and minimal ellipsoids // Numerische Mathematik. – 1981. – **36**. – P. 211–223.
5. TODD M. J. On minimum volume ellipsoids containing part of a given ellipsoid // Mathematics of Operations Research. – 1982. – **7**. – P. 253–261.
6. СТЕЦЮК П. И. r -алгоритмы и эллипсоиды // Кибернетика и систем. анализ. – 1996. – №1. – С. 113–134.
7. ШОР Н. З. Методы минимизации недифференцируемых функций и их приложения. – К.: Наук. думка, 1979. – 199с.
8. СТЕЦЮК П. И. Ортогонализующие линейные операторы в выпуклом программировании // Кибернетика и систем. анализ. – 1997. – №3. – С.97-119.
9. СТЕЦЮК П. И. Об одном методе для нахождения допустимой точки выпуклого неравенства // Теория оптимальных решений. – 2000. – С. 3–10.

Р А З Д Е Л IV

Ортогонализующие линейные операторы

В раздел включены статьи:

- 4.1. Ортогонализующие линейные операторы в выпуклом
программировании I (*Стецюк, 1997*) 192
- 4.2. Ортогонализующие линейные операторы в выпуклом
программировании II (*Стецюк, 1997*) 220
- 4.3. Классические фейеровские методы с преобразованием
пространства (*Стецюк, 1996*) 239
- 4.4. Об одном способе ускорения метода ортогонального
субградиентного спуска (*Стецюк, 1998*) 245
- 4.5. Об одном методе для нахождения допустимой точки выпуклого ..
неравенства (*Стецюк, 2000*) 253
- 4.6. К методам решения плохообусловленных систем линейных
уравнений (*Стецюк, 2001*) 261

Ортогонализующие линейные операторы в выпуклом программировании. I.

П. И. Стецюк

Кибернетика и системный анализ. – 1997. – № 3. – С. 97–119.

Рассмотрены два одноранговых линейных оператора преобразования пространства. Их использования позволяет обосновать сходимость семейства методов переменной метрики для решения задачи выпуклого программирования. На основе этих операторов построены субградиентные методы фееровского типа для нахождения минимума выпуклой функции при известном оптимальном значении функции. Приведены численные эксперименты с этими методами.

Введение

Методам субградиентного типа свойственна медленная сходимость при минимизации выпуклых функций овражного типа. Это связано с тем, что антисубградиент образует угол, близкий к $\pi/2$, с направлением на точку минимума. Оригинальный выход из этой ситуации был впервые осознан и предложен Н. З. Шором в [1]. Он заключается в использовании линейных неортогональных преобразований пространства, (более конкретно – оператора растяжения пространства), которые позволяют изменять углы между субградиентом и направлением на минимум. В частности, это приводит к методам субградиентного типа в преобразованном линейным оператором пространстве аргументов. Поэтому такие методы можно считать методами переменной метрики.

Исторически первым методом переменной метрики был предложенный Давидоном в [2] и развитый Флетчером и Пауэллом в [3] метод для минимизации дважды непрерывно дифференцируемых функций. Он основан на идее квадратической аппроксимации минимизируемой функции и фактически имитирует метод Ньютона-Рафсона без явного вычисления вторых производных [4]. В силу этого за ним закрепилось название метода квазиньютоновского типа, и нечасто в литературе можно встретить его интерпретацию как метода переменной метрики. Это еще обусловлено и формой записи метода Давидона-Флетчера-Пауэлла,

которая использует коррекцию симметричной матрицы, что не способствует интерпретации метода в преобразованном пространстве аргументов. Последнее обстоятельство имеет место и для других методов квазиньютоновского типа [5], а также для ряда методов сопряженных градиентов [4].

Поэтому условимся методами переменной метрики считать те методы, которые используют линейные неортогональные преобразования пространства, и результаты сходимости которых основаны на исследовании поведения характеристик минимизируемой функции в преобразованном пространстве аргументов. Наиболее обширную часть методов переменной метрики составляют методы субградиентного типа, использующие операцию растяжения пространства как в направлении субградиента, так и в направлении разности двух последовательных субградиентов (r -алгоритмы) [6, 7]. Причем, к первым относится и метод эллипсоидов, а r -алгоритмы зарекомендовали себя эффективным средством для решения негладких задач. К методам переменной метрики можно отнести и два метода [8], которые по сути являются вариантами r -алгоритмов с переменным коэффициентом растяжения пространства и классической фейеровской регуляризкой шагового множителя. Семейство экономных в смысле обработки информации методов переменной метрики для задач выпуклого программирования позволяет получить достаточно общую схему методов центров тяжести простых тел [9], которая наряду с растяжением пространства предполагает использование и других линейных неортогональных преобразований пространства.

Результаты сходимости приведенных выше методов основаны на изучении поведения нормы субградиента минимизируемой функции в растянутом пространстве, исследовании поведения функции расстояния до точки минимума в преобразованном пространстве, использовании монотонного уменьшения объема области локализации множества экстремумов. Очевидно, что имеется простой механизм, позволяющий расширить класс методов переменной метрики. Это можно сделать как посредством увеличения числа линейных неортогональных преобразований, так и числа исследуемых характеристик минимизируемой функции в преобразованном пространстве аргументов. При этом, сохраняя градиентную природу методов в преобразованном пространстве, можно строить методы, близкие по эффективности к квазиньютоновским, но применимые для более широкого класса функций, чем гладкие, в том числе и для негладких.

Практическая эффективность r -алгоритмов для овражных задач обусловлена тем, что растяжение пространства направлено на улучше-

ние структуры поверхностей уровня минимизируемой функции в очередном пространстве аргументов. Однако, кроме растяжения пространства, для этих целей можно использовать и другие линейные неортогональные операторы. Два из них, которые позволяют улучшать структуру поверхностей уровня овражных функций, и будут предметом обсуждения в данной работе. Эти преобразования достаточно просты и для минимизации выпуклых функций позволяют строить и обосновывать методы переменной метрики, которые обладают наглядной геометрической интерпретацией в преобразованном пространстве аргументов. Кроме преобразования пространства построение таких методов требует как выбора направления движения из текущей точки, так и определенного способа регулировки шагового множителя в этом направлении. Чтобы свести к минимуму обсуждение этих вопросов, ограничимся рассмотрением методов субградиентного типа с классической фейеровской регулировкой шагового множителя в направлении антисубградиента применительно к задаче минимизации выпуклой функции при известном значении функции в точке оптимума.

Изложение материала данной статьи следующее. В разд. 1 приведена постановка задачи, кратко проанализированы методы фейеровского типа. В разд. 2 рассмотрено одноранговое эллипсоидальное преобразование пространства и приведены на его основе два простых метода с классической фейеровской регулировкой шага. В разд. 3 рассмотрены доортогонализирующее преобразование пространства и основанное на нем семейство методов ортогонального субградиентного спуска с классической фейеровской регулировкой шага. Практическая эффективность приведенных методов подкреплена численными экспериментами. Попутно в разд. 2 и разд. 3 обсуждаются как возможные схемы ряда других алгоритмов на базе этих преобразований, так и конструктивное решение ряда принципиальных проблем для методов фейеровского типа применительно к решению задачи выпуклого программирования.

1. Постановка задачи и фейеровские методы

Постановка задачи. Пусть имеется задача безусловной минимизации:

$$\min f(x), \quad (1)$$

где $f(x)$ – выпуклая функция векторного аргумента $x \in X$, $X = R^n$; R^n – евклидово пространство размерности n со скалярным произведением (x, y) . X будем называть исходным пространством аргументов,

n – размером задачи. Пусть множество экстремумов задачи (1) X^* непусто и известно значение минимума $f(x)$: $f^* = f(x^*)$, $x^* \in X^*$. Не ограничивая общности, для удобства дальнейших рассуждений предположим, что множество X^* состоит из единственной точки x^* .

Такая постановка задачи имеет ряд приложений. Так, к задаче (1) легко свести задачу поиска допустимой точки совместной системы выпуклых неравенств. В случае, когда система выпуклых неравенств несовместна, анализ этой задачи позволяет легко получить достаточное условие несовместности.

Поскольку дальше речь пойдет о численных методах для решения задачи (1), то сформулируем, что будем считать ее решением, или, другими словами, определимся с критерием останова для методов. Пусть $X_\varepsilon^* = \{x : f(x) - f^* \leq \varepsilon\}$, где $\varepsilon > 0$. Точку $x_k \in X_\varepsilon^*$, т. е. для которой выполняется $f(x_k) - f^* \leq \varepsilon$, будем считать решением задачи (1) с точностью ε по функционалу (ε -решением). Входным параметром методов будет точность ε_f , с которой требуется решить задачу (1) по функционалу, и их останов будем производить, как только достигнута точка $x_k \in X_{\varepsilon_f}^*$.

О фейеровских методах. Фейеровские методы восходят к [10, 11] и являются одним из вариантов метода субградиентного спуска, который предполагает определенную регулировку шагового множителя в направлении нормированного субградиента. Для решения задачи (1) фейеровские методы представляются итеративным процессом

$$x_{k+1} = x_k - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|}, \quad h_k = \gamma \frac{(f(x_k) - f^*)}{\|\partial f(x_k)\|}, \quad (2)$$

где $\partial f(x_k)$ – субградиент $f(x)$ в точке x_k , γ – некоторый скаляр, такой, что $0 < \gamma < 2$. Такой выбор значений параметра γ гарантирует монотонное уменьшение расстояния до минимума на каждом шаге процесса (2) и обеспечивает простоту доказательства сходимости этих процессов. Шаг h_k при $\gamma = 1$ условимся называть классическим фейеровским шагом, а соответствующий ему метод – классическим фейеровским методом. Для решения задачи (1) с точностью ε_f по функционалу классический фейеровский метод достаточно конструктивен и принимает следующий вид.

В начале процесса имеем $x_0 \in R^n$, $\varepsilon_f > 0$.

Пусть на k -м шаге получена $x_k \in R^n$, $f(x_k)$ и $\partial f(x_k)$ – вычисленные в ней значения функции и субградиента. Тогда, если $f(x_k) - f^* \leq \varepsilon_f$,

то x_k — искомая точка. Иначе вычисляем очередное приближение

$$x_{k+1} = x_k - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|}, \quad h_k = \frac{(f(x_i) - f^*)}{\|\partial f(x_k)\|} \quad (3)$$

и переходим к $(k+1)$ -му шагу.

Справедлива следующая теорема.

Теорема 1. Пусть на каждом шаге метода (3) $\|\partial f(x_k)\| \leq C$. Тогда метод (3) позволяет найти ε_f -решение задачи (1) не более, чем за K -шагов, где $K = \left\lceil \left(\frac{C\|x_0 - x^*\|}{\varepsilon_f} \right)^2 \right\rceil + 1$. Здесь $\lceil a \rceil$ — целая часть от числа a .

Доказательство приводить не будем. Оно тривиально и основано на том, что на каждом шаге метода (3) расстояние до точки минимума уменьшается с точностью до величины квадрата сдвига в направлении нормированного субградиента (т. е. с точностью до h_k^2), и на том, что шаг h_k ограничен снизу величиной (ε_f / C) . Отметим лишь, что условие теоремы 1 можно и усилить, потребовав ограниченности $\|\partial f(x)\|$ в шаре радиуса $\|x_0 - x^*\|$ с центром в точке x^* . Но это принципиально ничего не меняет, а только приведет к усложнению материала данной статьи.

Несмотря на теоретическую привлекательность, метод (3) пользуется плохой репутацией в практическом отношении даже для гладких выпуклых функций, что обусловлено его медленной сходимостью для овражных функций. Так, когда поверхности уровня $f(x)$ не сильно вытянуты, его работа вполне удовлетворительна для задач даже не малых размеров ($n \sim 100$). Однако, когда направление антисубградиента в точке достаточно близко к ортогональному — к направлению на минимум, что характерно для овражных функций, метод (3) почти безнадёжен даже для небольших задач ($n \sim 10$). Для негладких выпуклых функций, где овражность скорее правило, чем исключение, метод (3) безнадёжен для задач еще меньших размеров ($n \sim 2 - 5$). Так, для минимизации функции двух переменных $f(x_1, x_2) = |x_1| + k|x_2|$ он сходится со скоростью геометрической прогрессии со знаменателем $\sqrt{1 - 1/k^2}$, который близок к единице при небольших значениях k , ($k \sim 10 - 100$).

Для ускорения сходимости, основываясь на приближении функционала вблизи минимума кусочно-линейным, в [12] предложен более общий метод:

$$x_{k+1} = P_{Q_k}(x_k), \quad Q_k = \left\{ x : f(x_i) + (\partial f(x_i), x - x_i) \leq f^*, \quad i \in I_k \right\}. \quad (4)$$

Здесь I_k – любое подмножество индексов из $0, 1, \dots, k$, содержащее обязательно k , P_{Q_k} – оператор проектирования на Q_k . Главная проблема в методе (4) – это указать конструктивный способ формирования множества I_k , для того, чтобы ограничить число хранимых субградиентов. Наиболее содержательным ответом здесь можно считать метод ортогонального спуска [13]. Он основан на идее последовательного погружения нетупоугольного конуса, содержащего X^* , в прямоугольный конус и использует не более чем $(n + 1)$ векторов. Однако, он не всегда приводит к хорошим результатам по точности решения задачи (4) по функционалу. Об этом можно судить по ряду численных экспериментов как с методом ортогонального спуска, так и с его модификацией, представленных в [14, 15]. Несмотря на то, что задачи, как правило, небольшого размера ($n \sim 10$) и не сильно овражные, тем не менее точность их решения по функционалу часто $\varepsilon_f \sim 10^{-4} - 10^{-2}$.

Еще один метод фейеровского типа предложен в [16]. Он использует всего два вектора и для построения очередного направления спуска использует линейную комбинацию направления субградиента и направления движения на предыдущем шаге (по типу сопряженных градиентов) с тем, чтобы оно составляло более острый угол с направлением на минимум. Применительно к овражным функциям он обеспечивает более «плавное» движение вдоль оврага, чем приведенные выше методы с использованием всего двух векторов, поведение которых «скачкообразно». Но он также неэффективен для овражных функций, особенно, когда овраг многомерный.

В смысле точности решения задачи (1) ситуацию в лучшую сторону для метода типа (3) способно изменить линейное преобразование пространства, направленное на выравнивание структуры поверхностей уровня функции в преобразованном пространстве. Пусть $Y = AX$ – преобразованное посредством линейного оператора A пространство аргументов. Здесь A – невырожденная матрица размера $n \times n$. Пусть B – обратный к A оператор, $B = A^{-1}$. Тогда классический фейеровский метод с преобразованием пространства в X приобретает вид

$$x_{k+1} = x_k - h_k B \frac{B^T \partial f(x_k)}{\|B^T \partial f(x_k)\|}, \quad h_k = \frac{(f(x_k) - f^*)}{\|B^T \partial f(x_k)\|}, \quad (5)$$

что соответствует классическому фейеровскому шагу в направлении антисубградиента в преобразованном пространстве аргументов Y

$$y_{k+1} = y_k - h_k \frac{\partial \varphi(y_k)}{\|\partial \varphi(y_k)\|}, \quad h_k = \frac{(\varphi(y_k) - \varphi^*)}{\|\partial \varphi(y_k)\|}. \quad (6)$$

Здесь $y = Ax$ – образы точек из X в преобразованном пространстве Y , $\varphi(y)$ – выпуклая функция, определенная в преобразованном пространстве Y : $\varphi(y) = f(A^{-1}y) = f(x)$. $\partial\varphi(y_k) = B^T \partial f(x_k)$ – субградиент в точке $y_k = Ax_k$ для функции $\varphi(y)$.

Пусть $Y_k = A_k X$ – текущее пространство аргументов. Тогда, улучшая структуру поверхностей уровня $\varphi_{k+1}(y)$ в очередном преобразованном пространстве аргументов $Y_{k+1} = T_{k+1}Y_k$, естественно от методов типа (5) ожидать более эффективной работы, чем от метода (3). Здесь T_{k+1} – невырожденная матрица размера $n \times n$, задающая преобразование из Y_k в Y_{k+1} .

Основные соотношения, которые требуются для реализации такого процесса, – это пересчет матрицы B_{k+1} согласно соотношению

$$B_{k+1} = B_k T_{k+1}^{-1} \quad (7)$$

и пересчет субградиентов для $\varphi_{k+1}(y)$, определенной в Y_{k+1} , согласно соотношению

$$\partial\varphi_{k+1}(y) = (T_{k+1}^{-1})^T \partial\varphi_k(y). \quad (8)$$

В дальнейшем часто будем использовать различные тождества, которые следуют из цепочки равенств

$$A_{k+1} = B_{k+1}^{-1} = (B_k T_{k+1}^{-1})^{-1} = T_{k+1} B_k^{-1} = T_{k+1} A_k.$$

Учитывая, что

$$\|B^T \partial f(x_k)\|^2 = (B^T \partial f(x_k), B^T \partial f(x_k)) = (\partial f(x_k), BB^T \partial f(x_k)),$$

процесс (5) можно записать в форме

$$x_{k+1} = x_k - \frac{(f(x_k) - f^*)}{(\partial f(x_k), H \partial f(x_k))} H \partial f(x_k), \quad (9)$$

где $H = BB^T$ – положительно определенная симметричная матрица размера $n \times n$. В зависимости от того, на какое из соотношений ориентироваться, имеется две возможности для практической реализации методов. Будем придерживаться для них тех названий, которые закрепились за ними для методов субградиентного типа с растяжением пространства. Метод на основе (5) принято называть методом в B -форме, или B -методом, а на основе (9) – методом в H -форме или H -методом. Преимущество H -метода состоит в том, что он требует для хранения

симметричной матрицы H почти в два раза меньше оперативной памяти ($n \times (n + 1)/2$), чем B -метод ($n \times n$ ячеек). Однако, метод в B -форме позволяет просто интерпретировать процесс в преобразованном пространстве Y . Следовательно, целесообразно строить методы в B -форме, а уже как следствие рассматривать их аналоги в H -форме, в основном с целью экономии памяти и вычислений. Поэтому будем ориентироваться на методы на основе соотношения (5).

2. Одноранговый эллипсоидальный оператор

Для решения задачи (1) в [8] рассмотрены два субградиентных метода с классическим фейеровским шагом в преобразованном пространстве аргументов. Они основаны на уменьшении объема области локализации множества экстремумов и сохраняют монотонность по оценке расстояния до оптимума в преобразованном пространстве аргументов. Используемое в них преобразование пространства по смыслу близко к применяемому в r -алгоритмах [6] и сводится к двум последовательным растяжениям пространства аргументов в ортогональных направлениях. Оператор преобразования пространства имеет вид

$$T_2(\xi, \eta) = R_{\alpha_1} \left(\frac{\xi + \eta}{\|\xi + \eta\|} \right) R_{\alpha_2} \left(\frac{\xi - \eta}{\|\xi - \eta\|} \right). \quad (10)$$

Здесь ξ, η – нормированные векторы из R^n , такие, что $(\xi, \eta)^2 \neq 1$. $\alpha_1 = 1/\beta_1$, где $\beta_1 = \sqrt{1 - (\xi, \eta)}$ и $\alpha_2 = 1/\beta_2$, где $\beta_2 = \sqrt{1 + (\xi, \eta)}$. В дальнейшем условимся оператор (10) называть двухранговым эллипсоидальным оператором.

Методы в [8] используют незначительную информацию об отсекающих множество экстремумов гиперплоскостях. Так, первый метод использует только два субградиента в последовательных точках, полученных согласно классическому фейеровскому шагу. Второй метод, кроме двух последовательных субградиентов, использует еще и третий вектор агрегатного типа, который является выпуклой комбинацией вычисленных ранее субградиентов. Несмотря на непротивительность по используемой информации, приведенные в [8] вычислительные эксперименты с этими методами для ряда как гладких, так и негладких тестовых задач оказались не такими уж и плохими. Поэтому более разумные стратегии использования вычисленных субградиентов способны привести к более эффективным методам на основе преобразования (10).

Однако, несмотря на простую геометрическую интерпретацию, двухранговое эллипсоидальное преобразование неудобно тем, что для методов в B -форме при переходе от k -го шага к $(k+1)$ -ому шагу расчет матрицы B_{k+1} требует двух одноранговых коррекций матрицы размера $n \times n$, что приводит к $4n^2$ арифметических операций. Но оказывается, что существует преобразование пространства, которое допускает одноранговую коррекцию матрицы B_{k+1} , и по смыслу не отличающееся от двухрангового эллипсоидального преобразования. Это не удивительно, так как методы типа (5) определяются матрицей $H_k = B_k B_k^T$. В силу неоднозначности разложения матрицы H_k , одному и тому же методу в H -форме могут соответствовать различные методы в B -форме.

Одноранговый эллипсоидальный оператор. Пусть $\xi, \eta \in R^n$ – векторы такие, что $\|\xi\| = 1$, $\|\eta\| = 1$, и их скалярное произведение удовлетворяет условию $(\xi, \eta)^2 \neq 1$.

Одноранговым эллипсоидальным оператором будем называть линейный оператор из R^n в R^n , который в матричной форме представим

$$T_1(\xi, \eta) = I - \frac{1}{1 - (\xi, \eta)^2} \left(\left(1 - \sqrt{1 - (\xi, \eta)^2} \right) \eta - (\xi, \eta) \xi \right) \eta^T. \quad (11)$$

Здесь I – единичная матрица размера $n \times n$. Этот оператор обладает рядом интересных свойств, что позволяет на его основе строить и обосновывать различного рода методы переменной метрики. Для него справедливы следующие свойства.

Лемма 1. Для оператора $T_1(\xi, \eta)$, заданного согласно (11), в случае, когда $(\xi, \eta)^2 \neq 1$, существует обратный $T_1^{-1}(\xi, \eta)$ и

$$T_1^{-1}(\xi, \eta) = I + \frac{1}{\sqrt{1 - (\xi, \eta)^2}} \left(\left(1 - \sqrt{1 - (\xi, \eta)^2} \right) \eta - (\xi, \eta) \xi \right) \eta^T. \quad (12)$$

Кроме того, для $T_1(\xi, \eta)$ и $T_1^{-1}(\xi, \eta)$ выполнены соотношения:

$$T_1^T(\xi, \eta) T_1(\xi, \eta) = I + \frac{(\xi, \eta)}{1 - (\xi, \eta)^2} (\xi \eta^T + \eta \xi^T), \quad (13)$$

$$\begin{aligned} T_1^{-1}(\xi, \eta) \left(T_1^{-1}(\xi, \eta) \right)^T &= I + \frac{(\xi, \eta)^2}{1 - (\xi, \eta)^2} (\xi \xi^T + \eta \eta^T) - \\ &- \frac{(\xi, \eta)}{1 - (\xi, \eta)^2} (\xi \eta^T + \eta \xi^T). \end{aligned} \quad (14)$$

Доказательство леммы 1 приводить не будем в силу его громоздкости. Отметим лишь, что в справедливости (12)–(14) легко убедиться непосредственной проверкой. Приемлемый вид соотношений (13) и (14) обусловлен тем, что при приведении подобных членов при одноранговых матрицах $\xi\xi^T$, $\xi\eta^T$, $\eta\xi^T$ и $\eta\eta^T$ члены с $\sqrt{1 - (\xi, \eta)^2}$ сокращаются.

Лемма 2. Пусть B_k – невырожденная матрица размера $n \times n$, p_1 , p_2 – n -мерные векторы, такие, что $\left(\frac{B_k^T p_1}{\|B_k^T p_1\|}, \frac{B_k^T p_2}{\|B_k^T p_2\|} \right)^2 < 1$. И пусть

$$B_{k+1} = B_k T_1^{-1}(\xi, \eta), \quad \text{где} \quad \xi = \frac{B_k^T p_1}{\|B_k^T p_1\|} \quad \text{и} \quad \eta = \frac{B_k^T p_2}{\|B_k^T p_2\|}. \quad (15)$$

Тогда матрица B_{k+1} невырождена, и, кроме того, для нее выполнены следующие свойства:

$$\text{а) } \det(B_{k+1}) = \det(B_k) \sqrt{1 - (\xi, \eta)^2} \quad \text{и} \quad \text{б) } (B_{k+1}^T p_1, B_{k+1}^T p_2) = 0.$$

Доказательство. Невырожденность матрицы B_{k+1} и свойство а) следуют из

$$\begin{aligned} \det(B_{k+1}) &= \det(B_k) \det(T_1^{-1}(\xi, \eta)) = \\ &= \det(B_k) \left(1 + \frac{1}{\sqrt{1 - (\xi, \eta)^2}} \left((1 - \sqrt{1 - (\xi, \eta)^2}) \eta - (\xi, \eta) \xi, \eta \right) \right) = \\ &= \det(B_k) \left(1 + \frac{1}{\sqrt{1 - (\xi, \eta)^2}} \left(1 - \sqrt{1 - (\xi, \eta)^2} - (\xi, \eta)^2 \right) \right) = \\ &= \det(B_k) \sqrt{1 - (\xi, \eta)^2} \neq 0. \end{aligned}$$

Для свойства б) имеем

$$\begin{aligned} (B_{k+1}^T p_1, B_{k+1}^T p_2) &= \left((T_1^{-1}(\xi, \eta))^T B_k^T p_1, (T_1^{-1}(\xi, \eta))^T B_k^T p_2 \right) = \\ &= \left(B_k^T p_1, T_1^{-1}(\xi, \eta) (T_1^{-1}(\xi, \eta))^T B_k^T p_2 \right) = \\ &= \frac{1}{\|B_k^T p_2\| \cdot \|B_k^T p_1\|} \left(\xi, T_1^{-1}(\xi, \eta) (T_1^{-1}(\xi, \eta))^T \eta \right). \end{aligned}$$

Используя соотношение (14) леммы 1, имеем

$$T_1^{-1}(\xi, \eta) (T_1^{-1}(\xi, \eta))^T \eta =$$

$$\begin{aligned}
&= \left(I + \frac{(\xi, \eta)^2}{1 - (\xi, \eta)^2} (\xi \xi^T + \eta \eta^T) - \frac{(\xi, \eta)}{1 - (\xi, \eta)^2} (\xi \eta^T + \eta \xi^T) \right) \eta = \\
&= \left(1 + \frac{(\xi, \eta)^2}{1 - (\xi, \eta)^2} - \frac{(\xi, \eta)^2}{1 - (\xi, \eta)^2} \right) \eta + \\
&\quad + (\xi, \eta) \left(\frac{(\xi, \eta)^2}{1 - (\xi, \eta)^2} - \frac{1}{1 - (\xi, \eta)^2} \right) \xi = \eta - (\xi, \eta) \xi,
\end{aligned}$$

откуда следует

$$\begin{aligned}
(B_{k+1}^T p_1, B_{k+1}^T p_2) &= \frac{1}{\|B_k^T p_2\| \cdot \|B_k^T p_1\|} (\xi, \eta - (\xi, \eta) \xi) = \\
&= \frac{1}{\|B_k^T p_2\| \cdot \|B_k^T p_1\|} ((\xi, \eta) - (\xi, \eta)) = 0.
\end{aligned}$$

Одноранговое эллипсоидальное преобразование пространства (11) имеет много общего с двухранговым эллипсоидальным преобразованием (10). Так, для них характерно одинаковое уменьшение детерминанта обратной матрицы, что при определенных условиях равносильно уменьшению объема области локализации множества экстремумов. Как первое, так и второе ортогонализуют векторы при переходе в преобразованное пространство аргументов. В дальнейшем увидим, что оба эти преобразования приводят к одному и тому же методу в H -форме. Однако, в отличие от двухрангового эллипсоидального преобразования, преобразование (11) требует в два раза меньше арифметических операций при пересчете матрицы B_k , т. е. ровно столько, сколько и растяжение пространства в некотором направлении. Поэтому его использование приводит к вычислительно более экономным методам в B -форме, чем применение двухрангового эллипсоидального преобразования (10).

Основываясь на уменьшении объема области локализации множества экстремумов, одноранговый эллипсоидальный оператор (11) позволяет просто обосновывать методы для решения задачи (1), чтобы улучшать структуру поверхностей уровня функции в очередном преобразованном пространстве. Пусть $g_k = B_k^T \partial f(x_k)$ и $g_{k+1} = B_k^T \partial f(x_{k+1})$ – субградиенты $\varphi_k(y)$ в $Y_k = A_k X$ в точках $y_k = A_k x_k$ и $y_{k+1} = A_k x_{k+1}$. Здесь y_{k+1} получена согласно классическому фейеровскому шагу в преобразованном пространстве аргументов Y_k . Тогда, если $(g_k, g_{k+1}) \geq 0$, то преобразование пространства не требуется. Пусть $(g_k, g_{k+1}) < 0$. Тогда преобразование пространства $Y_{k+1} = T_1\left(\frac{g_k}{\|g_k\|}, \frac{g_{k+1}}{\|g_{k+1}\|}\right) Y_k$ либо $Y_{k+1} = T_1\left(\frac{g_{k+1}}{\|g_{k+1}\|}, \frac{g_k}{\|g_k\|}\right) Y_k$ согласно лемме 2 позволяет ортогонализо-

вать образы субградиентов g_k и g_{k+1} в преобразованном пространстве аргументов Y_{k+1} , обеспечив при этом более предпочтительные поверхности уровня для функции $\varphi_{k+1}(y)$ в Y_{k+1} . Эти простые рассуждения приводят к следующему методу для решения задачи (1).

Метод фейеровского типа с одноранговым эллипсоидальным преобразованием пространства, использующий два последовательных субградиента. Перед началом вычислений имеем $x_0 \in R^n$, $B_0 = I$ – единичная матрица размера $n \times n$, ε_f – точность по функционалу, с которой требуется решить задачу (1), $f(x_0)$ и $g(x_0) = \partial f(x_0)$ – вычисленные в x_0 значения функции и субградиента. Тогда, если $f(x_0) - f^* \leq \varepsilon_f$, то x_0 – искомая точка и ОСТАНОВ.

Пусть на k -м шаге получены $x_k \in R^n$, $f(x_k)$, $g(x_k) = \partial f(x_k)$, B_k – матрица $n \times n$.

- 1) Вычислим очередное приближение

$$x_{k+1} = x_k - \frac{f(x_k) - f^*}{\|B_k^T g(x_k)\|} B_k \frac{B_k^T g(x_k)}{\|B_k^T g(x_k)\|}. \quad (16)$$

- 2) Вычислим $f(x_{k+1})$, $g(x_{k+1}) = \partial f(x_{k+1})$. Если $f(x_{k+1}) - f^* \leq \varepsilon_f$, то x_{k+1} – искомая точка и ОСТАНОВ. Иначе переходим к п. 3.

- 3) Положим

$$\xi_1 = \frac{B_k^T g(x_k)}{\|B_k^T g(x_k)\|}, \quad \xi_2 = \frac{B_k^T g(x_{k+1})}{\|B_k^T g(x_{k+1})\|}. \quad (17)$$

- 4) Если $(\xi_1, \xi_2) \geq 0$, полагаем $B_{k+1} = B_k$ и переходим к 5. Иначе вычисляем

$$B_{k+1} = B_k T_1^{-1}(\xi_1, \xi_2), \quad \text{либо} \quad B_{k+1} = B_k T_1^{-1}(\xi_2, \xi_1). \quad (18)$$

- 5) Переходим к очередному шагу с x_{k+1} , B_{k+1} , $g(x_{k+1})$, $f(x_{k+1})$.

Справедлива следующая теорема.

Теорема 2. Последовательность $\{x_{k+1}\}_{k=0}^\infty$, генерируемая методом (16)–(18), удовлетворяет неравенству

$$\|A_{k+1}(x_{k+1} - x^*)\|^2 \leq \|A_k(x_k - x^*)\|^2 - \frac{(f(x_k) - f^*)^2}{\|B_k^T g(x_k)\|^2}. \quad (19)$$

Здесь $A_k = B_k^{-1}$, $A_{k+1} = B_{k+1}^{-1}$, $k = 0, 1, 2, \dots$

Доказательство. Вначале покажем, что для метода (16)–(18) всегда выполнено неравенство

$$\|A_{k+1}(x_{k+1} - x^*)\|^2 \leq \|A_k(x_{k+1} - x^*)\|^2. \quad (20)$$

Если $(\xi_1, \xi_2) \geq 0$, то (20) выполняется как равенство, поскольку $A_{k+1} = A_k$.

Если $(\xi_1, \xi_2) < 0$, тогда, используя соотношение (13) леммы 1 и

$$\xi_1 = \frac{B_k^T g(x_k)}{\|B_k^T g(x_k)\|}, \quad \xi_2 = \frac{B_k^T g(x_{k+1})}{\|B_k^T g(x_{k+1})\|},$$

имеем

$$\begin{aligned} \|A_{k+1}(x_{k+1} - x^*)\|^2 &= \|T_1(\xi_1, \xi_2) A_k(x_{k+1} - x^*)\|^2 = \\ &= \left(T_1(\xi_1, \xi_2) A_k(x_{k+1} - x^*), T_1(\xi_1, \xi_2) A_k(x_{k+1} - x^*) \right) = \\ &= \left(A_k(x_{k+1} - x^*), (T_1(\xi_1, \xi_2))^T T_1(\xi_1, \xi_2) A_k(x_{k+1} - x^*) \right) = \\ &= \left(A_k(x_{k+1} - x^*), \left(I + \frac{(\xi_1, \xi_2)}{1 - (\xi_1, \xi_2)^2} (\xi_1 \xi_2^T + \xi_2 \xi_1^T) \right) A_k(x_{k+1} - x^*) \right) = \\ &= \|A_k(x_{k+1} - x^*)\|^2 + \\ &\quad + \frac{2(\xi_1, \xi_2)}{1 - (\xi_1, \xi_2)^2} \left(\xi_1, A_k(x_{k+1} - x^*) \right) \left(\xi_2, A_k(x_{k+1} - x^*) \right) = \\ &= \|A_k(x_{k+1} - x^*)\|^2 + \\ &\quad + \frac{2(\xi_1, \xi_2)}{1 - (\xi_1, \xi_2)^2} \cdot \frac{(g(x_k), x_{k+1} - x^*)}{\|B_k^T g(x_k)\|} \cdot \frac{(g(x_{k+1}), x_{k+1} - x^*)}{\|B_k^T g(x_{k+1})\|}. \end{aligned} \quad (21)$$

Причем, равенство (21) справедливо как для $B_{k+1} = B_k T_1^{-1}(\xi_1, \xi_2)$, так и для $B_{k+1} = B_k T_1^{-1}(\xi_2, \xi_1)$.

Второе слагаемое в (21) отрицательно, так как один его сомножитель

$$\frac{2(\xi_1, \xi_2)}{(1 - (\xi_1, \xi_2)^2) \cdot \|B_k^T g(x_k)\| \cdot \|B_k^T g(x_{k+1})\|} < 0$$

в силу того, что $(\xi_1, \xi_2) < 0$, $\|\xi_1\| = 1$, $\|\xi_2\| = 1$, а два других, учитывая выпуклость $f(x)$, положительны

$$(g(x_k), x_{k+1} - x^*) =$$

$$\begin{aligned}
&= \left(g(x_k), x_k - x^* - \frac{(f(x_i) - f^*)}{\|B_k^T g(x_k)\|} \cdot B_k \cdot \frac{B_k^T g(x_k)}{\|B_k^T g(x_k)\|} \right) = \\
&= \left(g(x_k), x_k - x^* \right) - \frac{(f(x_i) - f^*)}{\|B_k^T g(x_k)\|^2} \cdot \left(g(x_k), B_k B_k^T g(x_k) \right) = \\
&= \left(g(x_k), x_k - x^* \right) - \frac{(f(x_i) - f^*)}{\|B_k^T g(x_k)\|^2} \cdot \left(B_k^T g(x_k), B_k^T g(x_k) \right) = \\
&= \left(g(x_k), x_k - x^* \right) - \left(f(x_k) - f^* \right) \geq 0
\end{aligned}$$

и

$$\left(g(x_{k+1}), x_{k+1} - x^* \right) \geq f(x_{k+1}) - f^* \geq 0.$$

Следовательно, неравенство (20) справедливо и для $(\xi_1, \xi_2) < 0$.
Далее, учитывая, что в силу выпуклости $f(x)$

$$\left(x_k - x^*, g(x_k) \right) \geq f(x_k) - f^* \geq 0,$$

неравенство (20) можно записать

$$\begin{aligned}
&\left\| A_{k+1}(x_{k+1} - x^*) \right\|^2 \leq \left\| A_k(x_{k+1} - x^*) \right\|^2 = \\
&= \left\| A_k \left(x_k - x^* - \frac{(f(x_i) - f^*)}{\|B_k^T g(x_k)\|} \cdot B_k \cdot \frac{B_k^T g(x_k)}{\|B_k^T g(x_k)\|} \right) \right\|^2 = \\
&= \left\| A_k(x_k - x^*) - \frac{(f(x_i) - f^*)}{\|B_k^T g(x_k)\|^2} B_k^T g(x_k) \right\|^2 = \\
&= \left\| A_k(x_k - x^*) \right\|^2 - \\
&\quad - 2 \frac{(f(x_i) - f^*)}{\|B_k^T g(x_k)\|^2} \left(A_k(x_k - x^*), B_k^T g(x_k) \right) + \frac{(f(x_i) - f^*)^2}{\|B_k^T g(x_k)\|^2} = \\
&= \left\| A_k(x_k - x^*) \right\|^2 - 2 \frac{(f(x_i) - f^*)}{\|B_k^T g(x_k)\|^2} \left(x_k - x^*, g(x_k) \right) + \frac{(f(x_i) - f^*)^2}{\|B_k^T g(x_k)\|^2} \leq \\
&\leq \left\| A_k(x_k - x^*) \right\|^2 - \frac{(f(x_i) - f^*)^2}{\|B_k^T g(x_k)\|^2},
\end{aligned}$$

что завершает доказательство теоремы.

Для метода (16)–(18) теорема 2 обеспечивает возможность доказательства его предельной сходимости к решению задачи (1) за счет уменьшения объема эллипсоида, локализирующего x^* . При этом уменьшение объема тем лучше, чем больше классический фейеровский шаг в преобразованном пространстве и чем более тупой угол между последовательными субградиентами. Но оба эти обстоятельства зависят от конкретных свойств $f(x)$ и получить какие-либо оценки для общей выпуклой $f(x)$ не представляется возможным. Поэтому остановимся на следующем варианте доказательства сходимости метода (16)–(18).

Теорема 3. Пусть на каждом шаге метода (16)–(18) выполнены условия $\|B_k\| \leq c_1$ и $\|\partial f(x_k)\| \leq c_2$. Здесь $\|B_k\|$ – евклидова норма матрицы B_k , т. е. $\|B_k\| = \left(\sum_{i=1}^n \sum_{j=1}^n |b_{ij}^k|^2 \right)^{1/2}$. Тогда метод (16)–(18) решает задачу (1) с точностью ε_f по функционалу не более чем за K шагов, где $K = \left\lceil \left(\frac{c_1 c_2 \|x_0 - x^*\|}{\varepsilon_f} \right)^2 \right\rceil + 1$. Здесь $\lceil a \rceil$ – целая часть от числа a .

Доказательство. Из теоремы 2 следует справедливость неравенства

$$\|A_k(x_k - x^*)\|^2 \leq \|x_0 - x^*\|^2 - \sum_{i=0}^{k-1} \frac{(f(x_i) - f^*)^2}{\|B_i^T g(x_i)\|^2}.$$

Пусть метод (16)–(18) за k -шагов не решил задачу (1) с точностью ε_f по функционалу. Тогда для всех x_i , $i = 0, \dots, k$, выполнено $f(x_i) - f^* \geq \varepsilon_f$. Следовательно,

$$\sum_{i=0}^{k-1} \frac{(f(x_i) - f^*)^2}{\|B_i^T g(x_i)\|^2} \geq \sum_{i=0}^{k-1} \frac{(f(x_i) - f^*)^2}{\|B_i\|^2 \|g(x_i)\|^2} \geq \frac{1}{c_1^2 c_2^2} \sum_{i=0}^{k-1} \varepsilon_f^2 \geq \frac{k \varepsilon_f^2}{c_1^2 c_2^2},$$

откуда

$$\|A_k(x_k - x^*)\|^2 \leq \|x_0 - x^*\|^2 - \frac{k \varepsilon_f^2}{c_1^2 c_2^2}.$$

Учитывая, что $\|A_k(x_k - x^*)\|^2$ должно быть неотрицательным, получаем, что не более, чем за $K = \left\lceil \left(\frac{c_1 c_2 \|x_0 - x^*\|}{\varepsilon_f} \right)^2 \right\rceil + 1$ шагов метод (16)–(18) решит задачу (1) с точностью ε_f по функционалу.

Итак, если выполнены условия теоремы 3, метод (16)–(18) с любого начального приближения сходится к ε_f -решению задачи (1) за конечное

число итераций. Но оценка максимального числа итераций, необходимых для достижения точности ε_f , очень грубая. При реальной работе метода следует ожидать, что для овражных функций критерий останова будет срабатывать из условия

$$f(x_k) - f^* \leq \|B_k^T g(x_k)\| \sqrt{\|x_0 - x^*\|^2 - \sum_{i=0}^{k-1} \frac{(f(x_i) - f^*)^2}{\|B_i^T g(x_i)\|^2}} \quad (22)$$

в силу малости $\|B_k^T g(x_k)\|$ при больших k , так как $\det(B_k)$ стремится к нулю при $k \rightarrow \infty$. Поэтому имеет смысл усиливать преобразование пространства с тем, чтобы гарантировать более сильное стремление к нулю $\det(B_k)$ при $k \rightarrow \infty$. В частности, для этого подходит вариант метода как в [8], использующий для усиления преобразования агрегатный вектор, который является выпуклой комбинацией вычисленных ранее субградиентов. При этом метод незначительно усложнится, и для него по отношению к методу (16)–(18) будет характерно более сильное выравнивание поверхностей уровня функции в преобразованном пространстве, особенно при многомерных оврагах. Сосредоточим внимание на удобных для реализации вычислительных схемах двух этих методов. Хотя следует отметить, что преобразование (11) позволяет обосновывать целое семейство методов фейеровского типа. Дальше некоторые из них обсудим более подробно.

О практической реализации методов. Несмотря на удобства при доказательствах метод (16)–(18) неудобен для реализации на ЭВМ. Так, когда реализуется операция преобразования пространства, он требует вычислять образ субградиента $\partial f(x_{k+1})$ как в пространстве $Y_k - B_k^T \partial f(x_{k+1})$, так и в пространстве $Y_{k+1} - B_{k+1}^T \partial f(x_{k+1})$. Вычисление каждого из них требует n^2 арифметических операций умножения и столько же сложений, что приводит в общей сложности к $4n^2$ арифметических операций. Этого можно избежать, если субградиент при переходе в очередное преобразованное пространство аргументов пересчитывать согласно соотношению (8). Тем более, что в рассматриваемых методах в качестве направления движения используется именно анти-субградиент в преобразованном пространстве аргументов. Для преобразования (11) это позволяет следующая лемма.

Лемма 3. Пусть B_k – невырожденная матрица размера $n \times n$, p_k, p_{k+1} – n -мерные векторы, такие, что $\left(\frac{B_k^T p_k}{\|B_k^T p_k\|}, \frac{B_k^T p_{k+1}}{\|B_k^T p_{k+1}\|} \right)^2 < 1$.

И пусть $B_{k+1} = T_1^{-1}(\xi_k, \xi_{k+1})$, где $\xi_k = \frac{B_k^T p_k}{\|B_k^T p_k\|}$, $\xi_{k+1} = \frac{B_k^T p_{k+1}}{\|B_k^T p_{k+1}\|}$. Тогда,

$$\frac{B_{k+1}^T p_{k+1}}{\|B_{k+1}^T p_{k+1}\|} = \xi_{k+1},$$

$$\frac{B_{k+1}^T p_k}{\|B_{k+1}^T p_k\|} = \frac{1}{\sqrt{1 - (\xi_k, \xi_{k+1})^2}} \left(\xi_k - (\xi_k, \xi_{k+1}) \xi_{k+1} \right), \quad (23)$$

$$\|B_{k+1}^T p_i\| = \sqrt{1 - (\xi_k, \xi_{k+1})^2} \|B_k^T p_i\|, \quad i = k, k+1. \quad (24)$$

Доказательство. Для $B_{k+1}^T p_{k+1}$ имеем

$$\begin{aligned} B_{k+1}^T p_{k+1} &= \\ &= \left(T_1^{-1}(\xi_k, \xi_{k+1}) \right)^T B_k^T p_{k+1} = \|B_k^T p_{k+1}\| \left(T_1^{-1}(\xi_k, \xi_{k+1}) \right)^T \xi_{k+1} = \\ &= \|B_k^T p_{k+1}\| \left(I + \xi_{k+1} \left(\left(\frac{1}{\sqrt{1 - (\xi_k, \xi_{k+1})^2}} - 1 \right) \xi_{k+1} - \right. \right. \\ &\quad \left. \left. - \frac{(\xi_k, \xi_{k+1})}{\sqrt{1 - (\xi_k, \xi_{k+1})^2}} \xi_k \right)^T \right) \xi_{k+1} = \\ &= \|B_k^T p_{k+1}\| \left(1 + \frac{1}{\sqrt{1 - (\xi_k, \xi_{k+1})^2}} - 1 - \frac{(\xi_k, \xi_{k+1})^2}{\sqrt{1 - (\xi_k, \xi_{k+1})^2}} \right) \xi_{k+1} = \\ &= \|B_k^T p_{k+1}\| \sqrt{1 - (\xi_k, \xi_{k+1})^2} \xi_{k+1}, \end{aligned}$$

откуда следует справедливость первой части (23), а также справедливость (24) при $i = k+1$.

Аналогично и для $B_{k+1}^T p_k$ имеем

$$\begin{aligned} B_{k+1}^T p_k &= \\ &= \left(T_1^{-1}(\xi_k, \xi_{k+1}) \right)^T B_k^T p_k = \|B_k^T p_k\| \left(T_1^{-1}(\xi_k, \xi_{k+1}) \right)^T \xi_k = \\ &= \|B_k^T p_k\| \left(I + \xi_{k+1} \left(\left(\frac{1}{\sqrt{1 - (\xi_k, \xi_{k+1})^2}} - 1 \right) \xi_{k+1} - \right. \right. \\ &\quad \left. \left. - \frac{(\xi_k, \xi_{k+1})}{\sqrt{1 - (\xi_k, \xi_{k+1})^2}} \xi_k \right)^T \right) \xi_k = \end{aligned}$$

$$\begin{aligned}
&= \|B_k^T p_k\| \left(\xi_k + \left(\frac{(\xi_k, \xi_{k+1})}{\sqrt{1 - (\xi_k, \xi_{k+1})^2}} - (\xi_k, \xi_{k+1}) - \right. \right. \\
&\quad \left. \left. - \frac{(\xi_k, \xi_{k+1})}{\sqrt{1 - (\xi_k, \xi_{k+1})^2}} \right) \xi_{k+1} \right) = \\
&= \|B_k^T p_k\| \left(\xi_k - (\xi_k, \xi_{k+1}) \xi_{k+1} \right),
\end{aligned}$$

откуда, учитывая, что $\|\xi_k - (\xi_k, \xi_{k+1})\xi_{k+1}\| = \sqrt{1 - (\xi_k, \xi_{k+1})^2}$, следует справедливость второй части (23), а также (24) при $i = k$.

Для векторов $B_k^T p_k$ и $B_k^T p_{k+1}$ в преобразованном пространстве $Y_k = A_k X$ соотношения (23)–(24) леммы 3 обеспечивают пересчет как нормированных направлений, так и их норм при переходе в очередное преобразованное пространство аргументов, полученное в результате применения оператора $T_1\left(\frac{B_k^T p_k}{\|B_k^T p_k\|}, \frac{B_k^T p_{k+1}}{\|B_k^T p_{k+1}\|}\right)$. Однако, для B -формы методов все равно, какое из преобразований $T_1\left(\frac{B_k^T p_k}{\|B_k^T p_k\|}, \frac{B_k^T p_{k+1}}{\|B_k^T p_{k+1}\|}\right)$ или $T_1\left(\frac{B_k^T p_{k+1}}{\|B_k^T p_{k+1}\|}, \frac{B_k^T p_k}{\|B_k^T p_k\|}\right)$ выбирать. Выбор первого более рационален, поскольку при этом обеспечивается пересчет субградиентов, который позволяет не накапливать ошибок в нормированных направлениях для осуществления шага в очередном преобразованном пространстве аргументов.

Для реализации методов требуется и пересчет классического фейеровского шагового множителя в направлении нормированного субградиента при переходе в очередное преобразованное пространство аргументов. Это не представляет особых проблем, так как их пересчет связан с изменением норм субградиентов и, следовательно,

$$h_{k+1} = h_k \frac{\|B_k^T g_k\|}{\|B_{k+1}^T g_k\|}.$$

Здесь h_{k+1} и h_k – классические фейеровские шаги в направлении нормированного субградиента в пространствах $Y_{k+1} = A_{k+1} X$ и $Y_k = A_k X$.

Итак, лемма 3 позволяет избежать для B -формы метода (16)–(18) лишних вычислений и для задачи (1) он приобретает вычислительно более рациональный вид.

Перед началом вычислений имеем $\varepsilon_f > 0$, $x_0 \in R^n$. Тогда, если $f(x_0) - f^* \leq \varepsilon_f$, то x_0 – искомая точка и ОСТАНОВ. Иначе полагаем

$h_0 = \frac{f(x_0) - f^*}{\|\partial f(x_0)\|}$, $\xi_0 = \frac{\partial f(x_0)}{\|\partial f(x_0)\|} \in R^n$, $B_0 = I$ – единичная матрица размера $n \times n$.

Пусть на k -м шаге получены $x_k \in R^n$, h_k , $\xi_k \in R^n$, B_k – матрица размера $n \times n$.

1) Вычислим очередное приближение

$$x_{k+1} = x_k - h_k B_k \xi_k. \quad (25)$$

2) Вычислим $f(x_{k+1})$, $\partial f(x_{k+1})$. Тогда, если $f(x_{k+1}) - f^* \leq \varepsilon_f$, то x_{k+1} – искомая точка и ОСТАНОВ. Иначе положим

$$\xi_{k+1} = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|}, \quad h_{k+1} = \frac{f(x_{k+1}) - f^*}{\|B_k^T \partial f(x_{k+1})\|}. \quad (26)$$

3) Если $(\xi_k, \xi_{k+1}) \geq 0$, полагаем $B_{k+1} = B_k$ и переходим к п. 4. Иначе вычисляем

$$\eta = \left(\frac{1}{\sqrt{1 - (\xi_k, \xi_{k+1})^2}} - 1 \right) \xi_{k+1} - \frac{(\xi_k, \xi_{k+1})}{\sqrt{1 - (\xi_k, \xi_{k+1})^2}} \xi_k, \quad (27)$$

$$B_{k+1} = B_k \left(I + \eta \xi_{k+1}^T \right), \quad h_{k+1} = \frac{h_{k+1}}{\sqrt{1 - (\xi_k, \xi_{k+1})^2}}. \quad (28)$$

4) Переходим к очередному шагу с x_{k+1} , B_{k+1} , ξ_{k+1} , h_{k+1} .

Итак, метод (25)–(28) тождественен методу (16)–(18), и для него справедливы теоремы 2 и 3. Однако, он вычислительно более рационален, чем метод (25)–(28), и дальше именно его выберем для численных экспериментов.

Метод фейеровского типа с одноранговым эллипсоидальным преобразованием пространства, использующий два последовательных субградиента и вектор агрегатного типа. Аналогично, как и в [8], легко построить метод, использующий два последовательных субградиента и некоторый агрегатный вектор, который позволяет усилить преобразование пространства и обеспечить более быструю сходимость метода в смысле (22). Используя приведенные выше леммы 2 и 3, а также лемму 3 из [8], приходим к следующему методу для решения задачи (1).

Перед началом вычислений имеем $\varepsilon_f > 0$, $x_0 \in R^n$. Тогда, если $f(x_0) - f^* \leq \varepsilon_f$, то x_0 – искомая точка и ОСТАНОВ. Иначе полагаем

$h_0 = \frac{f(x_0) - f^*}{\|\partial f(x_0)\|}$, $\xi_0 = \frac{\partial f(x_0)}{\|\partial f(x_0)\|} \in R^n$, $p_0 = 0 \in R^n$, $B_0 = I$ – единичная матрица размера $n \times n$.

Пусть на k -м шаге получены $x_k \in R^n$, h_k , $\xi_k \in R^n$, $p_k \in R^n$, B_k – матрица $n \times n$.

1) Вычислим очередное приближение

$$x_{k+1} = x_k - h_k B_k \xi_k. \quad (29)$$

2) Вычислим $f(x_{k+1})$, $\partial f(x_{k+1})$. Тогда, если $f(x_{k+1}) - f^* \leq \varepsilon_f$, то x_{k+1} – искомая точка и ОСТАНОВ. В противном случае положим

$$\xi_{k+1} = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|}, \quad h_{k+1} = \frac{f(x_{k+1}) - f^*}{\|B_k^T \partial f(x_{k+1})\|}. \quad (30)$$

3) Вычислим

$$\lambda_1 = -\frac{(p_k, \xi_{k+1})}{\sqrt{(p_k, \xi_{k+1})^2 + (\xi_k, \xi_{k+1})^2}}, \quad \lambda_2 = -\frac{(\xi_k, \xi_{k+1})}{\sqrt{(p_k, \xi_{k+1})^2 + (\xi_k, \xi_{k+1})^2}}$$

и положим

$$p_{k+1} = \begin{cases} \lambda_1 p_k + \lambda_2 \xi_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 > 0, \\ p_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 \leq 0, \\ \xi_k, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 > 0, \\ 0, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 \leq 0. \end{cases} \quad (31)$$

4) Если $(p_{k+1}, \xi_{k+1}) \geq 0$, полагаем $B_{k+1} = B_k$ и переходим к 5. Иначе вычисляем

$$\eta = \left(\frac{1}{\sqrt{1 - (p_{k+1}, \xi_{k+1})^2}} - 1 \right) \xi_{k+1} - \frac{(p_{k+1}, \xi_{k+1})}{\sqrt{1 - (p_{k+1}, \xi_{k+1})^2}} p_{k+1}, \quad (32)$$

$$B_{k+1} = B_k \left(I + \eta \xi_{k+1}^T \right), \quad h_{k+1} = \frac{h_{k+1}}{\sqrt{1 - (p_{k+1}, \xi_{k+1})^2}}, \quad (33)$$

$$p_{k+1} = \frac{1}{\sqrt{1 - (p_{k+1}, \xi_{k+1})^2}} \left(p_{k+1} - (p_{k+1}, \xi_{k+1}) \xi_{k+1} \right). \quad (34)$$

5) Переходим к очередному шагу с x_{k+1} , B_{k+1} , ξ_{k+1} , h_{k+1} , p_{k+1} .

Для метода (29)–(34) характерно использование минимума информации о поведении функции с сохранением классического фейеровского шага в направлении последнего вычисленного субградиента. Так, кроме двух последних субградиентов, он использует еще и вектор агрегатного типа p_k , который является выпуклой комбинацией вычисленных ранее субградиентов, и задает гиперплоскость, отсекающую от образа точки x_k множество экстремумов в преобразованном пространстве аргументов. При этом проблема обновления вектора p_{k+1} решается автоматически посредством анализа углов между последними субградиентами и текущим p_k .

Для метода (29)–(34) справедливы аналоги теорем 2 и 3, как и для метода (25)–(28). В частности, справедлива следующая теорема.

Теорема 4. Пусть на каждом шаге метода (29)–(34) $\|B_k\| \leq c_1$ и $\|\partial f(x_k)\| \leq c_2$. Тогда метод (29)–(34) решает задачу (1) с точностью ε_f по функционалу не более, чем за K -шагов, где

$$K = \left\lceil \left(\frac{c_1 c_2 \|x_0 - x^*\|}{\varepsilon_f} \right)^2 \right\rceil + 1.$$

Она гарантирует его конечную сходимость к ε_f -решению задачи (1) с любого начального приближения. Для овражных функций по отношению к методу (25)–(28) он обеспечивает более быстрое стремление к нулю детерминанта матрицы B_k при $k \rightarrow \infty$. Поэтому от него следует ожидать более эффективной практической работы в смысле (22), чем от метода (25)–(28).

Об H -форме приведенных методов. Рассмотренные выше методы, использующие одноранговое эллипсоидальное преобразование пространства, близки к методам в [8] на базе двухрангового эллипсоидального преобразования. Так, первому из них соответствует метод (28)–(33), второму – метод (37)–(44) из [8]. Пусть все эти методы стартуют из одной и той же начальной точки. Тогда B -формы методов (25)–(28), (29)–(34) и B -формы соответствующих им аналогов из [8] различны в том смысле, что они генерируют различные последовательности матриц B_k . При этом естественно отличаются и направления движения в преобразованных пространствах. Однако, последовательности точек $\{x_k\}_{k=0}^{\infty}$, которые генерируют методы (25)–(28) и (29)–(34) совпадают с такими же последовательностями для их аналогов из [8]. Дело в том, что H -формы этих методов совпадают, т. е. последовательности матриц $\{H_k\}_{k=0}^{\infty}$ для них одинаковые. На самом деле H -форму приведенных выше алгоритмов определяет соотношение (14) леммы 1 для

$T_1^{-1}(\xi, \eta)(T_1^{-1}(\xi, \eta))^T$. Аналогичное произведение для $T_2(\xi, \eta)$ имеет вид

$$\begin{aligned}
& T_2^{-1}(\xi, \eta)(T_2^{-1}(\xi, \eta))^T = \\
& = R_{\beta_1}\left(\frac{\xi + \eta}{\|\xi + \eta\|}\right) R_{\beta_2}\left(\frac{\xi - \eta}{\|\xi - \eta\|}\right) R_{\beta_1}^T\left(\frac{\xi + \eta}{\|\xi + \eta\|}\right) R_{\beta_2}^T\left(\frac{\xi - \eta}{\|\xi - \eta\|}\right) = \\
& = R_{\beta_1^2}\left(\frac{\xi + \eta}{\|\xi + \eta\|}\right) R_{\beta_2^2}\left(\frac{\xi - \eta}{\|\xi - \eta\|}\right) = \left(I + (\beta_1^2 - 1)\left(\frac{\xi + \eta}{\|\xi + \eta\|}\right)\left(\frac{\xi + \eta}{\|\xi + \eta\|}\right)^T\right) \times \\
& \quad \times \left(I + (\beta_2^2 - 1)\left(\frac{\xi - \eta}{\|\xi - \eta\|}\right)\left(\frac{\xi - \eta}{\|\xi - \eta\|}\right)^T\right) = \\
& = I + (\beta_1^2 - 1)\left(\frac{\xi + \eta}{\|\xi + \eta\|}\right)\left(\frac{\xi + \eta}{\|\xi + \eta\|}\right)^T + (\beta_2^2 - 1)\left(\frac{\xi - \eta}{\|\xi - \eta\|}\right)\left(\frac{\xi - \eta}{\|\xi - \eta\|}\right)^T = \\
& = I - \frac{(\xi, \eta)}{2(1 + (\xi, \eta))}(\xi + \eta)(\xi + \eta)^T + \frac{(\xi, \eta)}{2(1 - (\xi, \eta))}(\xi - \eta)(\xi - \eta)^T = \\
& = I + \left(\frac{(\xi, \eta)}{2(1 - (\xi, \eta))} - \frac{(\xi, \eta)}{2(1 + (\xi, \eta))}\right)(\xi\xi^T + \eta\eta^T) - \\
& \quad - \left(\frac{(\xi, \eta)}{2(1 + (\xi, \eta))} + \frac{(\xi, \eta)}{2(1 - (\xi, \eta))}\right)(\xi\eta^T + \eta\xi^T) = \\
& = I + \frac{(\xi, \eta)^2}{1 - (\xi, \eta)^2}(\xi\xi^T + \eta\eta^T) - \frac{(\xi, \eta)}{1 - (\xi, \eta)^2}(\xi\eta^T + \eta\xi^T)
\end{aligned}$$

и в точности соответствует соотношению (14) для $T_1^{-1}(\xi, \eta)(T_1^{-1}(\xi, \eta))^T$.

Прямолинейное использование соотношения (14) для пересчета матрицы H_{k+1} требует четырех одноранговых коррекций матриц размера $n \times n$. Однако, сгруппировав соответствующим образом члены при одноранговых матрицах, для пересчета H_{k+1} достаточно двух одноранговых коррекций. Так, для H -формы, соответствующей методу (25)–(28), формулы для пересчета матрицы H_{k+1} принимают одну из форм

$$\begin{aligned}
H_{k+1} &= H_k - \frac{H_k g_{k+1} g_{k+1}^T H_k}{(g_{k+1}, H_k g_{k+1})} + \frac{H_k p p^T H_k}{(p, H_k p)}, \\
p &= \frac{(g_k, H_k g_{k+1})}{(g_k, H_k g_k)} g_k - g_{k+1},
\end{aligned} \tag{35}$$

либо

$$H_{k+1} = H_k - \frac{H_k g_k g_k^T H_k}{(g_k, H_k g_k)} + \frac{H_k p p^T H_k}{(p, H_k p)},$$

$$p = \frac{(g_k, H_k g_{k+1})}{(g_{k+1}, H_k g_{k+1})} g_{k+1} - g_k. \quad (36)$$

Здесь $g_k = \partial f(x_k)$ и $g_{k+1} = \partial f(x_{k+1})$ — субградиенты $f(x)$ в последовательных точках. Для H -формы, соответствующей методу (29)–(34), формулы для пересчета матрицы H_{k+1} незначительно усложнятся.

Кроме экономии памяти, требуемой для хранения матрицы, использование H -формы приводит и к вычислительно более экономным методам, чем приведенные выше B -методы. Так, двухранговая коррекция матрицы H_{k+1} , учитывая ее симметричность, требует не больше операций, чем одноранговая коррекция матрицы B_{k+1} , а взамен двух умножений матрицы на вектор, требуемых методами (25)–(28) и (29)–(34), для H -формы достаточно и одного. Вполне возможно, что и методика обоснования H -методов может быть не менее красивой, чем B -методов. Однако эти вопросы требуют специального исследования и здесь ими заниматься не будем. Отметим лишь, что формулы (35) и (36) для пересчета H_{k+1} очень напоминают пересчет аналогичных матриц в квазиньютоновских методах [5].

Численные эксперименты. В силу того, что методы (25)–(28) и (29)–(34) идентичны методам (28)–(33) и (37)–(44) из [8], то нет особой надобности в численном исследовании их поведения в зависимости от размеров задач. Можно считать, что для них справедливы результаты, приведенные в [8] (табл. 1 и 2). Хотя здесь требуется некоторая оговорка. Дело в том, что набор тестовых задач в [8] характеризуется сильной овражностью. Учитывая, что различным B -методам соответствуют разные способы накопления ошибок в матрице B_k , численные расчеты могут в точности не совпадать и возможны незначительные расхождения.

Поэтому в первой серии экспериментов ограничимся задачами небольших размеров $n \sim 5 - 10$ и проверим численную устойчивость приведенных выше методов по отношению к точности решения задач по функционалу. В качестве тестовых негладких задач ограничим выбор на двух наиболее часто упоминаемых примерах кусочно-квадратичных функций:

$$Shor(n = 5, \quad f^* = 22.6001620958) \quad \text{и}$$

$$Maxquad(n = 10, \quad f^* = -0.841408334596),$$

которые считаются плохими задачами для субградиентных методов. Столь скрупулезное указание f^* для задач *Shor* и *Maxquad* обусловлено тем, чтобы проверить работу методов при достаточно малых значениях ε_f . В качестве тестовых гладких задач ограничимся квадратичными задачами $Quad(t)$ из [8] при различной степени овражности ($Quad(10.)$ и $Quad(3.)$) и при разных размерах ($n = 5$ и $n = 10$). Здесь $f^* = 0$. Все эти задачи характеризует единственная точка минимума, но для негладких задач она определена точностью задания f^* . В качестве ε_f возьмем ε_0 и ε_0^2 . Здесь $\varepsilon_0 = 10^{-5}$ для негладких задач и равно 10^{-10} для квадратичных задач.

Таблица 1. ε_f -сходимость методов (25)-(28) и (29)-(34)

Задача	n	метод (25)–(28)		метод (29)–(34)	
		$iter(\varepsilon_0)$	$iter(\varepsilon_0^2)$	$iter(\varepsilon_0)$	$iter(\varepsilon_0^2)$
<i>Shor</i>	5	112(109)	227(224)	38(36)	70(68)
<i>Maxquad</i>	10	120(113)	293(286)	41(35)	85(79)
<i>Quad</i> (3)	5	40(11)	73(11)	40(11)	73(11)
<i>Quad</i> (3)	10	82(60)	115(74)	76(59)	109(80)
<i>Quad</i> (10)	5	60(22)	93(22)	57(21)	90(21)
<i>Quad</i> (10)	10	187(141)	220(152)	148(124)	181(141)

Результаты работы методов приведены в табл. 1. Здесь $iter(\varepsilon)$ — число вычислений значений функции и ее субградиента, необходимое для достижения точности ε по функционалу. В скобках приведено число реализуемых при этом преобразований пространства. Как видно из табл. 1, работа методов (25)–(28) и (29)–(34) не столь безнадежна, как работа классического фейеровского метода без преобразования пространства. Это в какой-то мере подтверждает, что преобразование пространства, направленное на выравнивание поверхностей уровня функции, способно значительно увеличить точность решения задач по функционалу для методов фейеровского типа.

Затраченные усилия на решение этих задач с точностью $\varepsilon_f = \varepsilon_0^2$ по функционалу часто в два раза больше, чем с точностью $\varepsilon_f = \varepsilon_0$. Однако, этот разрыв существенно сокращается при увеличении размеров задач.

Таблица 2. ε_f -сходимость метода (29)–(34)

Задача	n	$f(x_0) - f^*$	$iter(10^{-5})$	$iter(10^{-10})$	$iter(10^{-20})$
<i>Quad</i> (1.1)	50	581.955	42(32)	65(49)	102(73)
<i>Sabs</i> (1.1)	50	1163.909	176	279	347
<i>Quad</i> (1.05)	100	1305.013	51(43)	79(65)	124(97)
<i>Sabs</i> (1.05)	100	2610.025	318	424	614

Об этом свидетельствует вторая серия экспериментов с методом (29)–(34) для слабо овражных задач $Quad(t)$ и $Sabs(t)$ из [8]. Результаты приведены в табл. 2. Для задач $Sabs(t)$ преобразование пространства реализуется практически на каждом шаге метода (29)–(34). Поэтому для них в табл. 2 приведено только общее число итераций.

Некоторые выводы и замечания общего характера. Итак, одноранговое эллипсоидальное преобразование пространства – вполне достойная замена двухранговому эллипсоидальному преобразованию в том смысле, что B -методы на его основе требуют меньшего числа арифметических операций. Оно, как и двухранговое, позволяет строить различного рода методы переменной метрики, обладающие простой геометрической интерпретацией, и допускает достаточно удобный механизм доказательства методов в B -форме, основанный на внешней аппроксимации множества экстремумов монотонно уменьшающимся по объему эллипсоидом в X , который является образом шара в Y_k . Этот факт позволяет получить вполне удовлетворительные ответы на ряд вопросов в выпуклом программировании.

Так, теорема 3 указывает на принципиальную возможность построения конструктивного критерия останова для несовместных систем выпуклых неравенств посредством анализа последовательности $\sum_{k=0}^{\infty} h_k^2$. Если

$\sum_{k=0}^K h_k^2 \geq r^2$ при некотором K , то это значит, что система выпуклых неравенств не имеет допустимой точки в шаре радиуса r с центром в точке x_0 . Следовательно, приведенные выше методы за конечное число шагов либо находят допустимую точку системы выпуклых неравенств, либо получают достаточное условие ее несовместности.

При построении процессов нахождения минимума выпуклой функции $f(x)$ при неизвестном значении f^* радиус шара, локализирующего множество экстремумов в преобразованном пространстве, позволяет простой механизм уточнения нижней оценки f^* . Для получения верхней оценки f^* можно использовать как «грубое» ее уточнение путем сочетания с шагами, близкими к классическому методу эллипсоидов, так и более «точное» за счет процедур типа наискорейшего спуска. Как первый, так и второй путь позволяет для доказательства сходимости методов сохранить критерий уменьшения объема области локализации множества экстремумов. Но, если первый в ряде случаев позволяет сохранить и возможность движения по антисубградиенту и приводит к субградиентным методам фейеровского типа со сменой начального приближения, то второй потребует смены направления исследования поведения функции (например, как в ε -субградиентных методах) и приведет к методам по типу сопряженных градиентов, но в преобразованном пространстве аргументов.

В частности, управление параметром γ : $0 \leq \gamma < 1$ и для фейеровских процессов при известном f^* позволяет строить методы, близкие к методам по типу сопряженных градиентов. Например, как в [16], т. е. в качестве направления движения из точки в преобразованном пространстве используется агрегатный вектор, который является выпуклой комбинацией вычисленных ранее субградиентов и задает более предпочтительное направление на точку оптимума, чем последний субградиент. Это приводит к достаточно разумным схемам методов даже в условиях минимума используемой информации. Так, для одного из них достаточно использовать последний вычисленный субградиент и некоторый агрегатный вектор, по которому осуществляется движение из точки. Для второго достаточно двух агрегатных векторов и последнего субградиента. При этом автоматически будут обновляться векторы агрегатного типа и они будут заменяться последними вычисленными субградиентами. Учитывая, что при этом имеется еще и возможность вычислять субградиент в точке, в которую перенесены векторы агрегатного типа, то вычислительно эти методы могут оказаться предпочтительнее, чем метод (29)–(34).

Но, в любом случае, особенно для задач больших размерностей, не обойтись без некоторой процедуры накопления субградиентов, которые бы характеризовали овражность функции. Очевидно, что метод (29)–(34) можно улучшить посредством использования большого числа вычисленных ранее субградиентов с переводом их в очередное преобразованное пространство аргументов. Однако, при этом сложно строить

конструктивные правила отсева явно лишних субградиентов. Поэтому целесообразно схемы методов строить так, чтобы механизм отсева был простым. Лучше всего для этих целей подходит процедура последовательного проектирования некоторой точки в преобразованном пространстве, например центра шара, локализирующего множество экстремумов, на множество вида, как в (4), начиная с вычисленного в этой точке $\varphi(y)$ и $\partial\varphi(y)$, и до тех пор, пока задача проектирования решается аналитически. Очевидно, что поступая так, нельзя накопить больше $m \leq (n + 1)$ субградиентов, где первые $(m - 1)$ из них будут задавать конус, локализирующий множество экстремумов, а m -й субградиент будет вычислен в вершине этого конуса, находящейся на кратчайшем расстоянии от центра шара. Далее переходим в вершину конуса, и посредством преобразования (11) избавляемся от тупых углов между субградиентами. Используя в качестве новой точки вершину конуса и вычисленный в ней субградиент, указанную процедуру повторяем. Если при этом проектируемую точку заменять на вершину текущего конуса, то получим процедуру накопления субградиентов, как в методе ортогонального спуска [13], и при этом угол между любыми двумя субградиентами будет неострым. Такие схемы методов приводят к достаточно простому решению проблем проектирования в фейеровских методах, проблем отсева лишних субградиентов, и, кроме того, позволяют обеспечить более разумные поверхности уровня функции в последующих преобразованных пространствах. Достаточно устойчивая работа метода (29)–(34) дает все основания надеяться на эффективность таких методов для задач немалых размеров.

Литература

1. ШОР Н. З. Использование операций растяжения пространства в задачах минимизации выпуклых функций // Кибернетика. – 1970. – № 1. – С. 6–12.
2. DAVIDON W. C. Variable metric methods for minimization, AEC Research and Development Rept. ANL 5990 (Rev.), 1959.
3. FLETCHER R., POWELL M. J. D. A rapidly convergent descent method for minimization // Comput. J. – 1963. – 6, № 2 – P. 163–168.
4. ПШЕНИЧНЫЙ Б. Н., ДАНИЛИН Ю. М. Численные методы в экстремальных задачах – М.: Наука, 1975. – 319 с.

5. Поляк Б. Т. Введение в оптимизацию. – М.: Наука, 1983. – 384 с.
6. Шор Н. З. Методы минимизации недифференцируемых функций и их приложения. – Киев: Наук. думка, 1979. – 199 с.
7. Шор Н. З., Стеценко С. И. Квадратичные экстремальные задачи и недифференцируемая оптимизация. – Киев: Наук. думка, 1989. – 208 с.
8. Стецюк П. И. r -алгоритмы и эллипсоиды // Кибернетика и системный анализ. – 1996. – № 1. – С. 113–134.
9. Стецюк П. И. Метод центров тяжести простых тел // Кибернетика и системный анализ. – 1996. – № 5. – С. 117–138.
10. AGMON S. The relaxation method for linear inequalities // Canadian Journal of Mathematics. – 1954. – N 6. – P. 382–392.
11. MOTZKIN T. AND SCHOENBERG I. J. The relaxation method for linear inequalities // Canadian Journal of Mathematics. – 1954. – N 6. – P. 393–404.
12. Поляк Б. Т. Минимизация негладких функционалов // Вычисл. математика и мат. физика. – 1969. – 9, № 3. – С. 507–521.
13. ЩЕПАКИН М. Б. О методе ортогонального спуска // Кибернетика. – 1987. – № 1. – С. 58–62.
14. Скоков В. А., ЩЕПАКИН М. Б. Численное исследование метода ортогонального спуска // Кибернетика и системный анализ. – 1994. – № 2. – С. 139–149.
15. ЩЕПАКИН М. Б., ШУБЕНКОВА И. А. Исследование модифицированного метода ортогонального спуска для поиска нуля выпуклой функции // Кибернетика и системный анализ. – 1993. – № 4. – С. 63–72.
16. CAMERINI P., FRATTA L., MAFFIOLI F. On improving relaxation methods by modified gradient techniques // Math. Program. – 1975. – Study 3. – P. 26–34.

Ортогонализующие линейные операторы в выпуклом программировании. II¹.

П. И. Стецюк

Кибернетика и системный анализ. — 1997. — № 5. — С. 111–124.

Рассмотрены два одноранговых линейных оператора преобразования пространства. Их использования позволяет обосновать сходимость семейства методов переменной метрики для решения задачи выпуклого программирования. На основе этих операторов построены субградиентные методы фееровского типа для нахождения минимума выпуклой функции при известном оптимальном значении функции. Приведены численные эксперименты с этими методами.

В первой части статьи был рассмотрен одноранговый эллипсоидальный оператор и его основные свойства. Для нахождения точки минимума выпуклой функции при известном f^* были построены два субградиентных метода фееровского типа с преобразованием пространства на основе этого оператора.

3. Доортогонализующий одноранговый оператор

Одноранговый эллипсоидальный оператор не позволяет просто работать с числом векторов больше, чем 2 – 3. Пусть $K(g_1, g_2, \dots, g_m)$, $m \leq n$ – остроугольный конус, локализирующий множество экстремумов. Здесь g_1, g_2, \dots, g_m – линейно-независимые субградиенты и, кроме того, $(g_i, g_j) < 0$ для всех $i = 1, 2, \dots, m$, $j = 1, 2, \dots, m$, таких, что $i \neq j$. Для эффективного преобразования конуса $K(g_1, g_2, \dots, g_m)$ посредством однорангового эллипсоидального оператора требуется в каждом из преобразованных пространств решать задачи погружения такого типа конусов в более простые конусы $K(p_1, p_2)$, такие, что $(p_1, p_2) < 0$. Эти задачи не представляют особых проблем и их решение требует незначительных вычислительных затрат. Однако, этого можно избежать, если на каждом шаге метода обеспечить работу с более простым

¹Первую часть статьи см. на стр. 192–219

конусом, например, ортогональным. Такую возможность легко реализовать, и для преобразования пространства можно использовать описанный ниже линейный неортогональный оператор.

Доортогонализирующий одноранговый оператор. Пусть в R^n задан набор векторов

$$P = \{p_1, p_2, \dots, p_m\}, \quad \|p_i\| = 1, \quad i = 1, \dots, m, \quad m \leq n, \quad (37)$$

и пусть для них выполнены следующие условия:

$$(p_i, p_j) = 0, \quad i \neq j, \quad i = 1, 2, \dots, m-1, \quad j = 1, 2, \dots, m-1, \quad (38)$$

$$\left\| p_m - \sum_{i=1}^{m-1} (p_m, p_i) p_i \right\|^2 = 1 - \sum_{i=1}^{m-1} (p_m, p_i)^2 > 0. \quad (39)$$

т. е. векторы из P линейно-независимы и, кроме того, первые $(m-1)$ векторов взаимно-ортогональны, а p_m не обязательно ортогонален всем предыдущим.

Обозначим $p = \sum_{i=1}^{m-1} (p_m, p_i) p_i$. Для векторов (37), удовлетворяющим (38) и (39), выполнены следующие свойства:

- а) $(p, p_m - p) = 0$;
- б) $(p_i, p_m - p) = 0, \quad i = 1, \dots, m-1$;
- в) $(p_m, p_m - p) = \|p_m - p\|^2$.

Ими будем часто пользоваться в дальнейшем.

Рассмотрим линейный оператор из R^n в R^n , который в матричном виде представим

$$T_\lambda(p_m, p) = I + \frac{p_m - p}{\|p_m - p\|^2} \left(\frac{1}{\lambda} p_m + p \right)^T, \quad (41)$$

где I – единичная матрица размера $n \times n$, λ – некоторый скаляр, такой, что $\lambda(\lambda+1) \neq 0$. Выбор параметра λ и его смысл обсудим чуть позже. Оператор (41) условимся называть доортогонализирующим одноранговым оператором. Для него справедливы следующие утверждения.

Лемма 4. Для оператора $T_\lambda(p_m, p)$ при $\lambda(\lambda+1) \neq 0$ существует обратный $T_\lambda^{-1}(p_m, p)$ и

$$T_\lambda^{-1}(p_m, p) = I - \frac{p_m - p}{\|p_m - p\|^2} \left(\frac{1}{\lambda+1} p_m + \frac{\lambda}{\lambda+1} p \right)^T. \quad (42)$$

Кроме того,

$$\begin{aligned} T_\lambda^T(p_m, p) T_\lambda(p_m, p) &= \\ &= I + \frac{1}{\|p_m - p\|^2} \left(\left(\frac{2}{\lambda} + \frac{1}{\lambda^2} \right) p_m p_m^T - p p^T + p_m p^T + p p_m^T \right). \end{aligned} \quad (43)$$

Доказательство. В справедливости (42), (43) легко убедиться непосредственной проверкой. Так, учитывая, что в силу (40а) и (40с), имеет место $\left(\frac{1}{\lambda} p_m + p, p_m - p \right) = \frac{1}{\lambda} (p_m, p_m - p) + (p, p_m - p) = \frac{1}{\lambda} \|p_m - p\|^2$ для проверки (42) имеем

$$\begin{aligned} T_\lambda(p_m, p) T_\lambda^{-1}(p_m, p) &= \left(I + \frac{p_m - p}{\|p_m - p\|^2} \left(\frac{1}{\lambda} p_m + p \right)^T \right) \times \\ &\quad \times \left(I - \frac{p_m - p}{\|p_m - p\|^2} \left(\frac{1}{\lambda + 1} p_m + \frac{\lambda}{\lambda + 1} p \right)^T \right) = \\ &= \left(I + \frac{p_m - p}{\|p_m - p\|^2} \left(\frac{1}{\lambda} p_m + p \right)^T \right) \left(I - \frac{\lambda}{\lambda + 1} \frac{p_m - p}{\|p_m - p\|^2} \left(\frac{1}{\lambda} p_m + p \right)^T \right) = \\ &= I + \left(1 - \frac{\lambda}{\lambda + 1} - \frac{\lambda}{\lambda + 1} \frac{\left(\frac{1}{\lambda} p_m + p, p_m - p \right)}{\|p_m - p\|^2} \right) \frac{p_m - p}{\|p_m - p\|^2} \times \\ &\quad \times \left(\frac{1}{\lambda} p_m + p \right)^T = \\ &= I + \left(\frac{1}{\lambda + 1} - \frac{\lambda}{\lambda + 1} \frac{1}{\lambda} \frac{p_m - p}{\|p_m - p\|^2} \right) \frac{p_m - p}{\|p_m - p\|^2} \left(\frac{1}{\lambda} p_m + p \right)^T = I. \end{aligned}$$

Справедливость (43) следует из

$$\begin{aligned} T_\lambda^T(p_m, p) T_\lambda(p_m, p) &= \\ &= \left(I + \left(\frac{1}{\lambda} p_m + p \right) \left(\frac{p_m - p}{\|p_m - p\|^2} \right)^T \right) \left(I + \frac{p_m - p}{\|p_m - p\|^2} \left(\frac{1}{\lambda} p_m + p \right)^T \right) = \\ &= I + \frac{1}{\|p_m - p\|^2} \left(\left(\frac{1}{\lambda} p_m + p \right) (p_m - p) + (p_m - p) \left(\frac{1}{\lambda} p_m + p \right)^T + \right. \\ &\quad \left. + \left(\frac{1}{\lambda} p_m + p \right) \left(\frac{1}{\lambda} p_m + p \right)^T \right) = \\ &= I + \frac{1}{\|p_m - p\|^2} \left(\frac{2}{\lambda} + \frac{1}{\lambda^2} \right) p_m p_m^T + (-1 - 1 + 1) p p^T + \end{aligned}$$

$$\begin{aligned}
& + \left(-\frac{1}{\lambda} + \frac{1}{\lambda+1} \right) p_m p^T + \left(\frac{1}{\lambda} - \frac{1}{\lambda+1} \right) p p_m^T = \\
& = I + \frac{1}{\|p_m - p\|^2} \left(\left(\frac{2}{\lambda} + \frac{1}{\lambda^2} \right) p_m p_m^T - p p^T + p_m p^T + p p_m^T \right).
\end{aligned}$$

Лемма 5. Пусть набор векторов (37) удовлетворяет (38) и (39), и $\tilde{P} = \{\tilde{p}_1, \tilde{p}_2, \dots, \tilde{p}_m\}$ – набор векторов, которые получены из (37), как $\tilde{p}_i = (T_\lambda^{-1}(p_m, p))^T p_i$ для всех $i = 1, 2, \dots, m$ при $\lambda(\lambda+1) \neq 0$. Тогда ни один из векторов \tilde{p}_i тождественно не равен нулевому вектору, $(\tilde{p}_i, \tilde{p}_j) = 0$, $i \neq j$, $i = 1, 2, \dots, m$, $j = 1, 2, \dots, m$, и, кроме того,

$$\tilde{p}_i = p_i, \quad i = 1, 2, \dots, m-1, \quad \tilde{p}_m = \frac{\lambda}{\lambda+1} (p_m - p). \quad (44)$$

Доказательство. Пересчет векторов p_i для всех $i = 1, 2, \dots, m$ осуществляется согласно соотношению

$$\begin{aligned}
\tilde{p}_i &= (T_\lambda^{-1}(p_m, p))^T p_i = \left(I - \left(\frac{1}{\lambda+1} p_m + \frac{\lambda}{\lambda+1} p \right) \frac{(p_m - p)^T}{\|p_m - p\|^2} \right) p_i = \\
&= p_i - \left(\frac{1}{\lambda+1} p_m + \frac{\lambda}{\lambda+1} p \right) \frac{(p_m - p, p_i)}{\|p_m - p\|^2}.
\end{aligned}$$

Отсюда следует справедливость первой части (44) так, как $(p_m - p, p_i) = 0$ для всех $i = 1, 2, \dots, m-1$ в силу свойства (40b). Для \tilde{p}_m с учетом (40с) получаем

$$\tilde{p}_m = p_m - \left(\frac{1}{\lambda+1} p_m + \frac{\lambda}{\lambda+1} p \right) \frac{\|p_m - p\|^2}{\|p_m - p\|^2} = \frac{\lambda}{\lambda+1} (p_m - p).$$

Тот факт, что ни один из векторов \tilde{p}_i , $i = 1, 2, \dots, m$, тождественно не равен нулевому вектору, следует из того, что $\|\tilde{p}_i\| = \|p_i\| = 1$, $i = 1, 2, \dots, m-1$, а также из условия (39) и $\lambda(\lambda+1) \neq 0$, что равносильно $\|\tilde{p}_m\| \neq 0$.

Далее, $(\tilde{p}_i, \tilde{p}_j) = (p_i, p_j) = 0$, $i = 1, 2, \dots, m-1$, $j = 1, 2, \dots, m-1$, $i \neq j$, согласно (38) и $(\tilde{p}_m, \tilde{p}_i) = \frac{\lambda}{\lambda+1} (p_m - p, p_i) = 0$ для всех $i = 1, 2, \dots, m-1$ согласно (40b).

Итак, оператор (41) допускает одноранговую коррекцию матрицы B_{k+1} при переходе в очередное преобразованное пространство аргументов и, более того, обеспечивает достаточно удобный механизм работы с

конусом, определяемым конечным набором векторов $m \leq n$. Его использование позволяет строить конструктивные методы переменной метрики с достаточно наглядной геометрической интерпретацией в преобразованном пространстве аргументов.

1. Если в качестве векторов выбирать субградиенты, можно получить различного рода методы, типа сопряженных градиентов для гладких функций. Так, для безусловной минимизации положительно определенных квадратичных функций простая процедура последовательного накопления градиентов с их доортогонализацией приводит к методам этого типа, которые требуют не более чем $(n + 1)$ вычислений значения функции и градиента. При этом посредством выбора шагов необходимо лишь гарантировать линейную независимость векторов $y_0 - y_k$, $k = 1, 2, \dots, m$, $m \leq n$, где $y_0 = A_m x_0$, $y_k = A_m x_k$, а на последнем m -м шаге метода, когда получен линейно-зависимый градиент, для нахождения точки минимума решить в преобразованном пространстве простую систему линейных уравнений с диагональной матрицей размера $m \times m$. Управление параметром λ в операторе (41) позволяет сохранять положительную определенность матрицы $H_k = B_k B_k^T$ и управлять степенью ее вырожденности. Это делает возможным построение методов по типу сопряженных градиентов для негладких задач.

2. Использование скаляра $\left(p_m, \frac{p}{\|p\|}\right)$ задает достаточно удобный механизм для уточнения нижней оценки f^* в преобразованном пространстве аргументов. Это обстоятельство позволяет обеспечить конструктивный критерий останова в смысле приближенного выполнения условий оптимальности для задач выпуклого программирования. Его можно использовать для построения методов переменной метрики, используя процедуры по типу ε -субградиентных [17], но в преобразованном пространстве аргументов. Учитывая, что при некотором значении параметра λ сохраняется внешняя аппроксимация множества экстремумов в преобразованном пространстве шаром без увеличения радиуса, такие процедуры подходят и для реализации внутреннего алгоритма в МЦТПТ [9].

3. С помощью преобразования (41) можно попытаться добиться вычислительно устойчивых методов для решения систем линейных уравнений в рамках методов ортогонализации, которые зарекомендовали себя численно плохо устойчивыми [18]. При этом, кроме самого преобразования, имеется ряд механизмов как на уровне B -формы, так и H -формы, чтобы обеспечить устойчивый пересчет расстояний до гиперплоскостей в преобразованном пространстве аргументов.

4. Преобразование (41) позволяет просто строить методы переменной метрики, основанные на локализации множества экстремумов конусом в R^n . На одном из таких вариантов методов для решения задачи (1), который допускает классический фейеровский шаг в направлении антисубградиента, и остановимся в данном разделе. В его основе лежит простой геометрический факт. Пусть y_k – вершина ортогонального конуса $K(g_1, \dots, g_m)$, локализующего множество экстремумов, и g_k – вычисленный в ней субградиент. Пусть $K_1(g'_1, \dots, g'_{m_1}, g_k)$ – построенный на базе этой информации очередной конус с вершиной в y_k . Здесь g'_1, \dots, g'_{m_1} – те из ортогональных субградиентов конуса $K(g_1, \dots, g_m)$, для которых выполнено $(g_i, g_k) < 0$. В силу построения этот конус будет содержать множество экстремумов задачи (1). Конус $K_1(g'_1, \dots, g'_{m_1}, g_k)$ – нетупоугольный, и его ребра образуют между собой острые углы. Преобразование его в ортогональный конус естественно использовать для улучшения структуры поверхностей уровня овражных функций.

Оператор (41) и такая стратегия построения конуса от шага к шагу приводит к ряду методов переменной метрики, основанных на локализации множества экстремумов в преобразованном пространстве ортогональным конусом, определяемым вычисленными ранее субградиентами. При этом автоматически обеспечивается решение проблемы, как ограничить число хранимых субградиентов. Более того, отсев их происходит не простым «забыванием», а каждый из этих векторов вносит определенный вклад в преобразование пространства, которое направлено на расширение конуса возможных направлений убывания функции.

Для процессов с классическим фейеровским шагом в направлении антисубградиента и описанной стратегией построения конуса, локализующего множество экстремумов, имеется две возможности их реализации. Первая – сначала делать преобразование пространства, а затем в преобразованном пространстве реализовывать классический фейеровский шаг в направлении \tilde{g}_k . Здесь \tilde{g}_k – образ субградиента g_k в преобразованном пространстве аргументов. Вторая – сначала делать классический фейеровский шаг в направлении g_k , а затем преобразовывать пространство аргументов. В обоих случаях получим локализацию множества экстремумов ортогональным конусом в преобразованном пространстве аргументов. В первом случае для всех перенесенных в текущую точку вычисленных ранее субградиентов будет реализован максимальный сдвиг по выпуклости $f(x)$. Во втором – каждый из перенесенных в текущую точку субградиентов, кроме последнего, допускает положительный шаг в направлении антисубградиента с тем, чтобы не нарушить локализации множества экстремумов ортогональным конусом.

Следовательно, во втором случае метод можно улучшить в смысле локализации множества экстремумов за счет шага по выпуклой комбинации ортогональных субградиентов, не включая последнего. Это приводит к направлению движения, отличному от антисубградиента, и второй путь рассматривать не будем. Очевидно, что без шага по выпуклой комбинации предыдущих субградиентов, он явно будет проигрывать первому.

Остановимся на первом варианте и описанной стратегии построения конуса от шага к шагу. Такие процессы применительно к решению задачи (1) условимся называть методами ортогонального субградиентного спуска с классическим фейеровским шагом в силу того, что движение из точки в преобразованном пространстве определяется субградиентом в ней, который ортогонален к накопленным субградиентам.

Метод ортогонального субградиентного спуска с классическим фейеровским шагом (*ORTGF*). Прежде, чем переходить к его описанию, условимся еще о некоторых параметрах, которые имеют чисто практическое значение и конкретизируют правило формирования очередного конуса. Так, кроме точности решения задачи (1) по функционалу $-\varepsilon_f > 0$, будем использовать еще два достаточно малых положительных скаляра: ε_K и ε_R , а также параметр $m_0 \leq n - 1$. Здесь $\varepsilon_K > 0$ будет определять правило отсева субградиентов, т. е. при переходе к $(k+1)$ -му шагу будем оставлять только те из ортогональных субградиентов, для которых выполнено условие $\left(\frac{g_i}{\|g_i\|}, \frac{g_k}{\|g_k\|} \right) < -\varepsilon_K$, $\varepsilon_R > 0$ будет определять дополнительный отсев субградиентов, чтобы избежать накопления ошибок вычислений в процессе доортогонализации, т. е. те из ортогональных субградиентов, для которых $\left| \left(\frac{g_i}{\|g_i\|}, \frac{g_j}{\|g_j\|} \right) \right| > \varepsilon_R$, будем исключать. Параметр m_0 будет задавать ограничение на максимальное число хранимых субградиентов, и это будут субградиенты, которые вычислены последними. Множество хранимых субградиентов на k -м шаге будем обозначать P_k и его размер, т. е. число накопленных субградиентов — $size(P_k)$. В частности, если $P_k = \emptyset$, то $size(P_k) = 0$. Для работы с P_k будем использовать операции: суммы (\cup) и разности (\setminus).

Для упрощения изложения пойдем еще на некоторые ограничения в методе *ORTGF*. Так, хранить будем только нормированные субградиенты и, кроме того, параметр λ будем считать постоянным. Хотя следует заметить, что используя нормы субградиентов и переменный параметр λ_k , можно обеспечить более содержательные методы ортогонального субградиентного спуска. Например, посредством использования оператора растяжения пространства и оператора (41) можно обеспечить преобразование пространства таким образом, чтобы направление

движения в Y_k совпадало с кратчайшим вектором к выпуклой оболочке накопленных субградиентов. Такое направление движения используется в ε -субградиентных методах и считается удачной заменой ньютоновскому направлению.

Согласно сказанному выше, $ORTGF(x_0, \varepsilon_f, \lambda, \varepsilon_K, \varepsilon_R, m_0)$ – метод ортогонального субградиентного спуска с классическим фейеровским шагом – принимает следующий вид.

Инициализация: $B_0 = I_n$, $P_0 = \emptyset$.

Пусть на k -м шаге получены x_k , B_k и P_k . Вычислим $f(x_k)$ и $\partial f(x_k)$. Если $f(x_k) - f^* \leq \varepsilon_f$, то x_k – искомая точка и ОСТАНОВ. Иначе переходим к $(k+1)$ -му шагу.

- 1). Полагаем $\xi_k = \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}$, $h_k = \frac{f(x_k) - f^*}{\|B_k^T \partial f(x_k)\|}$.
- 2). Сформируем множество $\tilde{P}_k = \{p_i \in P_k : (p_i, \xi_k) < -\varepsilon_K\}$, соблюдая порядок следования p_i в P_k .
- 3). Если $size(P_k) = 0$, то полагаем $B_{k+1} = B_k$, $\xi_{k+1} = \xi_k$, $h_{k+1} = h_k$ и переходим к 4. Иначе вычисляем вектор $\tilde{p}_k = \sum_{p_i \in \tilde{P}_k} (p_i, \xi_k) p_i$ и пересчитываем параметры: $B_{k+1} = B_k T_\lambda^{-1}(\xi_k, \tilde{p}_k)$,

$$\xi_{k+1} = \frac{B_{k+1}^T \partial f(x_k)}{\|B_{k+1}^T \partial f(x_k)\|}, \quad h_{k+1} = \frac{f(x_k) - f^*}{\|B_{k+1}^T \partial f(x_k)\|}.$$

Вычислительно это равносильно $B_{k+1} = B_k (I - \eta_1 \eta_2^T)$, где

$$\eta_1 = \frac{\xi_k - \tilde{p}_k}{\|\xi_k - \tilde{p}_k\|^2}, \quad \eta_2 = \frac{1}{\lambda + 1} \xi_k + \frac{\lambda}{\lambda + 1} \tilde{p}_k.$$

$$\xi_{k+1} = \frac{\frac{\lambda}{\lambda + 1} (\xi_k - \tilde{p}_k)}{\|\frac{\lambda}{\lambda + 1} (\xi_k - \tilde{p}_k)\|}, \quad h_{k+1} = \frac{h_k}{\|\frac{\lambda}{\lambda + 1} (\xi_k - \tilde{p}_k)\|}.$$

- 4). Вычислим очередное приближение $x_{k+1} = x_k - h_{k+1} B_{k+1} \xi_{k+1}$.
- 5). Сформируем очередное множество

$$P_{k+1} = \{p_i \in \tilde{P}_k : |(p_i, \xi_{k+1})| < \varepsilon_R\} \cup \xi_{k+1},$$

соблюдая порядок следования p_i в \tilde{P}_k , ξ_{k+1} будет последним вектором в P_{k+1} . Если $size(P_{k+1}) > m_0$, то $P_{k+1} = P_{k+1} \setminus \{p_1 \in P_{k+1}\}$, где p_1 – первый вектор в P_{k+1} .

6). Переходим к очередному шагу с x_{k+1} , B_{k+1} , P_{k+1} .

Теорема 5. Для k -го шага метода *ORTGF* выполнено

$$(A_k(x_k - x^*), \xi_k) \geq \frac{f(x_k) - f^*}{\|B_k^T \partial f(x_k)\|} \geq 0, \quad (45)$$

где $A_k = B_k^{-1}$. Когда $P_k \neq \mathcal{O}$, то $\|p_i\| = 1$ для всех $p_i \in P_k$; $(p_i, p_j) = 0$ для $p_i, p_j \in P_k$ таких, что $i \neq j$; и, кроме того, справедливы неравенства

$$(A_k(x_k - x^*), p_i) \geq 0, \quad p_i \in P_k, \quad (46)$$

$$(A_k(x_k - x^*), \tilde{p}_k) \leq 0. \quad (47)$$

Доказательство. Справедливость (45) следует из

$$\begin{aligned} (A_k(x_k - x^*), \xi_k) &= \left(A_k(x_k - x^*), \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|} \right) = \\ &= \frac{(x_k - x^*, \partial f(x_k))}{\|B_k^T \partial f(x_k)\|} \geq \frac{f(x_k) - f^*}{\|B_k^T \partial f(x_k)\|} \end{aligned}$$

в силу выпуклости $f(x)$.

Ортогональность и нормированность векторов в $P_k = \tilde{P}_{k-1} \cup \xi_k$ обеспечивает оператор $T_\lambda(\xi_{k-1}, \tilde{p}_{k-1})$ согласно лемме 5, так как векторы в \tilde{P}_{k-1} ортогональны и нормированы, а $\xi_k = \frac{(T_\lambda^{-1}(\xi_{k-1}, \tilde{p}_{k-1}))^T \xi_{k-1}}{\|(T_\lambda^{-1}(\xi_{k-1}, \tilde{p}_{k-1}))^T \xi_{k-1}\|}$.

Для той части P_k , которая сформирована из \tilde{P}_{k-1} , справедливость (46) следует по индукции. На самом деле пусть

$$(A_{k-1}(x_{k-1} - x^*), p_i) \geq 0, \quad p_i \in P_{k-1}.$$

Учитывая, что согласно лемме 5 $(T_\lambda^{-1}(\xi_{k-1}, \tilde{p}_{k-1}))^T p_i = p_i$ и $(\xi_k, p_i) = 0$ для всех $p_i \in \tilde{P}_{k-1}$, имеем

$$\begin{aligned} (A_k(x_k - x^*), p_i) &= (A_k(x_{k-1} - x^* - h_k B_k \xi_k), p_i) = \\ &= (A_k(x_{k-1} - x^*) - h_k \xi_k, p_i) = (A_k(x_{k-1} - x^*), p_i) - h_k (\xi_k, p_i) = \\ &= (A_k(x_{k-1} - x^*), p_i) = \left(A_k(x_{k-1} - x^*), (T_\lambda^{-1}(\xi_{k-1}, \tilde{p}_{k-1}))^T p_i \right) = \\ &= (T_\lambda^{-1}(\xi_{k-1}, \tilde{p}_{k-1}) A_k(x_{k-1} - x^*), p_i) = (A_{k-1}(x_{k-1} - x^*), p_i), \end{aligned}$$

откуда следует справедливость (46) для той части P_k , которая сформирована из \tilde{P}_{k-1} .

Для ξ_k , так как $\xi_k = \frac{B_k^T \partial f(x_{k-1})}{\|B_k^T \partial f(x_{k-1})\|}$ и $h_k = \frac{f(x_{k-1}) - f^*}{\|B_k^T \partial f(x_{k-1})\|}$, справедливость (46) следует из

$$\begin{aligned} & \left(A_k(x_k - x^*), \xi_k \right) = \left(A_k(x_{k-1} - x^* - h_k B_k \xi_k), \xi_k \right) = \\ & = \left(A_k(x_{k-1} - x^*) - h_k \xi_k, \xi_k \right) = \left(A_k(x_{k-1} - x^*), \xi_k \right) - h_k = \\ & = \left(A_k(x_{k-1} - x^*), \frac{B_k^T \partial f(x_{k-1})}{\|B_k^T \partial f(x_{k-1})\|} \right) - \frac{f(x_{k-1}) - f^*}{\|B_k^T \partial f(x_{k-1})\|} = \\ & = \frac{(x_{k-1} - x^*, \partial f(x_{k-1})) - (f(x_{k-1}) - f^*)}{\|B_k^T \partial f(x_{k-1})\|} \geq 0 \end{aligned}$$

в силу выпуклости $f(x)$.

И, наконец, справедливость (47) обеспечивают неравенства (46) и способ формирования множества \tilde{P}_k , который для вектора \tilde{p}_k гарантирует выбор отрицательных коэффициентов при соответствующих p_i .

Теорема 6. Последовательность $\{x_{k+1}\}_{k=0}^\infty$, генерируемая алгоритмом ORTGF при $\lambda = -1/2$, удовлетворяет неравенству

$$\|A_{k+1}(x_{k+1} - x^*)\|^2 \leq \|A_k(x_k - x^*)\|^2 - \frac{(f(x_k) - f^*)^2}{\|B_k^T \partial f(x_k)\|^2}. \quad (48)$$

Здесь $A_k = B_k^{-1}$, $A_{k+1} = B_{k+1}^{-1}$, $k = 0, 1, 2, \dots$

Доказательство. Если $P_k = \mathcal{O}$, то $\xi_{k+1} = \xi_k = \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}$, $B_{k+1} = B_k$, $h_{k+1} = h_k = \frac{f(x_k) - f^*}{\|B_k^T \partial f(x_k)\|}$, и неравенство (48) следует из

$$\begin{aligned} & \|A_{k+1}(x_{k+1} - x^*)\|^2 = \\ & = \|A_k(x_k - x^* - h_k B_k \xi_k)\|^2 = \|A_k(x_k - x^*) - h_k \xi_k\|^2 = \\ & = \|A_k(x_k - x^*)\|^2 - 2h_k (A_k(x_k - x^*), \xi_k) + h_k^2 = \\ & = \|A_k(x_k - x^*)\|^2 - 2 \frac{f(x_k) - f^*}{\|B_k^T \partial f(x_k)\|} (A_k(x_k - x^*), \xi_k) + \frac{(f(x_k) - f^*)^2}{\|B_k^T \partial f(x_k)\|^2} \leq \\ & \leq \|A_k(x_k - x^*)\|^2 - \frac{(f(x_k) - f^*)^2}{\|B_k^T \partial f(x_k)\|^2} \end{aligned}$$

в силу соотношения (45) теоремы 5.

Когда $P_k \neq \emptyset$, в силу выпуклости $f(x)$ имеем

$$\begin{aligned}
& \|A_{k+1}(x_{k+1} - x^*)\|^2 = \\
& = \left\| A_{k+1}(x_k - x^*) - \frac{f(x_k) - f^*}{\|B_{k+1}^T \partial f(x_k)\|} \cdot \frac{B_{k+1}^T \partial f(x_k)}{\|B_{k+1}^T \partial f(x_k)\|} \right\|^2 = \\
& = \|A_{k+1}(x_k - x^*)\|^2 - 2 \frac{f(x_k) - f^*}{\|B_{k+1}^T \partial f(x_k)\|^2} (x_k - x^*, \partial f(x_k)) + \\
& + \frac{(f(x_k) - f^*)^2}{\|B_{k+1}^T \partial f(x_k)\|^2} \leq \|A_{k+1}(x_k - x^*)\|^2 - \frac{(f(x_k) - f^*)^2}{\|B_{k+1}^T \partial f(x_k)\|^2}.
\end{aligned}$$

Оценим оба слагаемых в (49). Для первого, используя соотношение (43) леммы 4 при $\lambda = -1/2$, имеем

$$\begin{aligned}
& \|A_{k+1}(x_k - x^*)\|^2 = \|T_{-1/2}(\xi_k, \tilde{p}_k) A_k(x_k - x^*)\|^2 = \\
& = \left(A_k(x_k - x^*), T_{-1/2}(\xi_k, \tilde{p}_k) T_{-1/2}(\xi_k, \tilde{p}_k) A_k(x_k - x^*) \right) = \\
& = \left(A_k(x_k - x^*), \left(I + \frac{(\tilde{p}_k \xi_k^T + \xi_k \tilde{p}_k^T - \tilde{p}_k \tilde{p}_k^T)}{\|\xi_k - \tilde{p}_k\|^2} \right) A_k(x_k - x^*) \right) = \\
& = \|A_k(x_k - x^*)\|^2 + \\
& + \frac{2(A_k(x_k - x^*), \xi_k)(A_k(x_k - x^*), \tilde{p}_k)}{\|\xi_k - \tilde{p}_k\|^2} - \frac{(A_k(x_k - x^*), \tilde{p}_k)^2}{\|\xi_k - \tilde{p}_k\|^2} \leq \\
& \leq \|A_k(x_k - x^*)\|^2 + \\
& + \frac{2(A_k(x_k - x^*), \xi_k)(A_k(x_k - x^*), \tilde{p}_k)}{\|\xi_k - \tilde{p}_k\|^2} \leq \|A_k(x_k - x^*)\|^2. \quad (49)
\end{aligned}$$

Последнее неравенство в (49) следует в силу соотношений (45) и (47) теоремы 5, поскольку $(A_k(x_k - x^*), \tilde{p}_k) \leq 0$, а $(A_k(x_k - x^*), \xi_k) \geq 0$.

Кроме того,

$$\begin{aligned}
\frac{(f(x_k) - f^*)^2}{\|B_{k+1}^T \partial f(x_k)\|^2} &= \frac{\|B_k^T \partial f(x_k)\|^2}{\|B_{k+1}^T \partial f(x_k)\|^2} \cdot \frac{(f(x_k) - f^*)^2}{\|B_k^T \partial f(x_k)\|^2} = \\
&= \frac{1}{\|\xi_k - \tilde{p}_k\|^2} \cdot \frac{(f(x_k) - f^*)^2}{\|B_k^T \partial f(x_k)\|^2} \geq \frac{(f(x_k) - f^*)^2}{\|B_k^T \partial f(x_k)\|^2}, \quad (50)
\end{aligned}$$

так как $\|\xi_k - \tilde{p}_k\|^2 \leq 1$. Учитывая (49), (50), из (49) следует неравенство (48) и в том случае, когда $P_k \neq \emptyset$, что завершает доказательство теоремы.

Теорема 6 означает, что выбор параметра $\lambda = -1/2$ для метода *ORTGF* позволяет сохранить внешнюю аппроксимацию множества экстремумов шаром с уменьшением радиуса при переходе в преобразованное пространство аргументов. Учитывая, что $|\det(B_k)| = 1$, такой выбор параметра λ позволяет обосновывать сходимость метода *ORTGF* в смысле внешней локализации множества экстремумов эллипсоидом с уменьшением объема. Однако, уменьшение объема здесь будет небольшим, так как его обеспечивает только классический фейеровский шаг из центра шара. Тем не менее, оператор (41) при $\lambda = -1/2$ (к сожалению он единственный) позволяет улучшать структуру поверхностей уровня функции, не увеличивая объема области локализации множества экстремумов. Этот факт позволяет обосновывать методы переменной метрики и при любых других стратегиях построения нетупоугольного конуса, локализирующего множество экстремумов.

При этом преобразование (41) принимает вид

$$\begin{aligned} T_{-1/2}(p_m, p) &= T_{-1/2}^{-1}(p_m, p) = I - \frac{p_m - p}{\|p_m - p\|^2} (2p_m - p)^T = \\ &= \left(I - 2 \frac{p_m - p}{\|p_m - p\|} \left(\frac{p_m - p}{\|p_m - p\|} \right)^T \right) \left(I + \frac{p_m - p}{\|p_m - p\|^2} p^T \right) \end{aligned}$$

и содержит как «чистое» отражение, определяемое ортогональной матрицей, так и преобразование, действующее по типу «растяжения». Относительно отражения субградиентные методы, а также методы типа отсечений инвариантны, а преобразование по типу «растяжения» способно улучшать структуру поверхностей уровня оптимизируемой функции. Пожалуй, недостаток оператора (41) при $\lambda = -1/2$ состоит в том, что он не является сжимающим для пространства субградиентов, т. е. $\det(B_k B_k^T) = 1$. Это способствует накоплению ошибок при вычислении нормированных направлений и может привести к неустойчивой работе методов для сильно овражных функций.

Из теоремы (6) следует теорема (7).

Теорема 7. Пусть на каждом шаге метода *ORTGF* при $\lambda = -1/2$ $\|B_k\| \leq c_1$ и $\|\partial f(x_k)\| \leq c_2$. Тогда метод *ORTGF* решает задачу (1) с точностью ε_f не более, чем за K шагов, где $K = \left\lceil \left(\frac{c_1 c_2 \|x_0 - x^*\|}{\varepsilon_f} \right)^2 \right\rceil + 1$.

Ее доказательство аналогично доказательству теоремы 3.

Однако, параметр $\lambda = -1/2$ единственный, который позволяет, основываясь на расходимости суммы ряда $\sum_{i=0}^k \frac{(f(x_i) - f^*)^2}{\|B_i^T \partial f(x_i)\|^2}$, об-

основать сходимость для метода $ORTGF$ в смысле ε_f -сходимости по функционалу. Для обоснования сходимости $ORTGF$ при других значениях параметра λ , в том числе и переменных, требуется привлечение другого аппарата. Более всего здесь подходит интерпретация этих методов как методов типа сопряженных направлений [4]. Однако этот вопрос требует отдельного обсуждения. Здесь ограничимся только численной проверкой метода $ORTGF$ при двух значениях параметра λ : $\lambda = -1/2 - ORTGF(-0.5)$ и $\lambda = 1 - ORTGF(1.)$. Однако, если для задачи (1) сходимость метода $ORTGF(-0.5)$ к ε_f -решению обоснована, то метод $ORTGF(1.)$ требует более четкого обоснования.

Оправданием второму выбору параметра λ может служить следующее геометрическое объяснение. Пусть в точке y_k на k -м шаге метода $ORTGF(-0.5)$ $\tilde{P}_k \neq \emptyset$. Тогда, прежде чем преобразовывать конус, локализирующий множество экстремумов, с помощью оператора $T_{-1/2}(\xi_k, \tilde{p}_k)$, зададимся целью уменьшить объем области локализации множества экстремумов, не испортив при этом структуры конуса. Это просто сделать в рамках эллипсоида по типу, как в [8], где одна из осей совпадает с направлением $(\xi_k - \tilde{p}_k)$, а вторая — с тем из $p_i \in \tilde{P}_k$, для которого $(\xi_k, p_i) \leq -1/\sqrt{2}$. Это, как правило, будет иметь место для сильно овражных функций. Преобразование такого эллипсоида в шар не изменяет структуры локализирующего множество экстремумов конуса. Оно требует растяжения пространства в направлении $(\xi_k - \tilde{p}_k)$ и сжатия в ортогональном ему направлении p_i и приводит к уменьшению нормы последнего субградиента в преобразованном пространстве аргументов. Выбор параметра $\lambda = 1$ фактически придает методу этот смысл, но без дополнительных преобразований пространства. При этом норма последнего субградиента будет уменьшаться в два раза по отношению к тому, что обеспечивал бы оператор $T_{-1/2}(\xi_k, \tilde{p}_k)$.

Численные эксперименты. В первой серии экспериментов проверим численную устойчивость методов $ORTGF(-0.5)$ и $ORTGF(1.)$ для того же набора тестовых задач, что и в разд. 2, и для достижения той же точности ε_f по функционалу. В качестве параметров формирования конуса выберем: $\varepsilon_K = 10^{-4}$, $\varepsilon_R = 10^{-8}$, $m_0 = n - 1$. Результаты приведены в табл. 3. Здесь в скобках первым указано число преобразований, а вторым — максимальное число накопленных субградиентов. Как видно из таблицы, сходимость методов к ε_f -решению достаточно устойчива и практически такая же как и для метода (29)–(34).

Хотя метод $ORTGF$ построен с учетом выравнивания структуры поверхностей уровня $f(x)$, чтобы преодолеть ее овражность, тем не менее он не всегда будет достигать этой цели. Дело в том, что овраг $\varphi(y)$

Таблица 3. ε_f -сходимость методов $ORTGF(-0.5)$ и $ORTGF(1.)$.

Задача	n	$ORTGF(-0.5)$		$ORTGF(1.)$	
		$iter(\varepsilon_0)$	$iter(\varepsilon_0^2)$	$iter(\varepsilon_0)$	$iter(\varepsilon_0^2)$
<i>Shor</i>	5	33(30,4)	59(56,4)	33(30,4)	69(66,4)
<i>Maxquad</i>	10	45(37,5)	95(87,5)	42(35,5)	88(79,5)
<i>Quad(3.)</i>	5	40(9,2)	71(9,2)	52(30,3)	96(58,3)
<i>Quad(3.)</i>	10	80(61,5)	113(62,5)	86(68,3)	141(109,3)
<i>Quad(10.)</i>	5	57(26,3)	90(26,3)	50(22,3)	74(22,3)
<i>Quad(10.)</i>	10	156(123,8)	189(128,8)	131(109,4)	193(161,4)

в точке y_k характеризует локальное поведение $\varphi(y)$ вблизи y_k . Однако при использовании классического фейеровского шага и принятого способа накопления субградиентов существует опасность, что накопленные субградиенты, особенно самые «старые», только портят картину овражности в текущей точке. Для гладких функций это, как правило, срабатывать не будет, так как сходимости метода помогает стремление к нулю нормы субградиента. Это будет обеспечивать и накопление субградиентов, которые разумно характеризуют овраг. Однако для кусочно-линейных функций это не так. Здесь сходимость, скорее, будет обеспечиваться за счет вычисления вершины конуса, определяемого числом кусков, близким к n .

Следовательно, для кусочно-линейных функций не следует ожидать хорошей практической работы от $ORTGF(-0.5)$ при $m_0 \ll n$ в силу сказанного выше. Для сильно овражных функций ситуация еще и ухудшится за счет накопления ошибок в матрице B_k . Конечно, положение здесь можно изменить в лучшую сторону как за счет дополнительных растяжений пространства, направленных на уменьшение объема, так и за счет процедуры «восстановления» и усовершенствования способа накопления субградиентов. Для $ORTGF(1.)$ преобразование направлено на более сильное растяжение конуса, и у него значительно больше шансов на хорошую практическую работу при небольших значениях m_0 .

Во второй серии экспериментов проверим работу методов применительно к минимизации выпуклой кусочно-линейной функции с очень большим числом кусков. В качестве таковой выберем «плохую»

для субградиентных методов задачу $TR48$ [$n = 48$, $f^* = -683565$.] из [19, с. 161]. Результаты для двух разных начальных точек из [19] приведены в табл. 4. Здесь в скобках приведено максимальное число накопленных субградиентов. ε_K и ε_R , такие как и раньше. Параметр $\varepsilon_f = 50$ для первой начальной точки выбран так, чтобы число вычислений $f(x)$ и $\partial f(x)$ можно было сравнить с результатами, представленными в [20], где в качестве приемлемого решения считались точки, где $f(x) \leq -683500$. Как видно из табл. 4, здесь также характерна устойчивость методов в смысле ε_f -сходимости по функционалу. И, наконец,

Таблица 4. $ORTGF(-0.5)$ и $ORTGF(1.)$ для задачи $TR48$

Метод	m_0	$f(x_0) = -464816.$		$f(x_0) = -638524.94$	
		$iter(50.)$	$iter(10^{-5})$	$iter(1.)$	$iter(10^{-5})$
$ORTGF(-0.5)$	n-1	139(28)	222(31)	72(34)	151(34)
$ORTGF(1.)$	n-1	170(24)	344(24)	97(30)	248(30)
$ORTGF(1.)$	20	172	358	162	303
$ORTGF(1.)$	10	166	345	200	340
$ORTGF(1.)$	5	199	412	207	357

в последней третьей серии экспериментов проверим численную устойчивость метода $ORTGF(1.)$ для сильно овражных задач средних размеров ($n \sim 30 - 100$). В качестве тестовых задач остановимся на $Quad(t)$ и $Sabs(t)$. Результаты приведены в табл. 5. Здесь $iter(\varepsilon)$ – число вычислений $f(x)$ и $\partial f(x)$. В скобках приведено число преобразований, а для $ORTGF(1.)$ при $m_0 = n - 1$ – и максимальное число накопленных субградиентов. Если считать одной большой итерацией метода $ORTGF(1.)$ n вычислений $f(x)$ и $\partial f(x)$, что по трудоемкости равносильно одной итерации метода Ньютона, то, как видно из табл. 5, число больших итераций не такое уж и большое и составляет ~ 10 .

Общие замечания. Итак, на основе оператора (41) также легко строить методы переменной метрики с простой геометрической интерпретацией в преобразованном пространстве аргументов. При этом для общей задачи выпуклого программирования вполне возможно создание практически эффективных методов со строгим обоснованием их сходи-

Таблица 5. ε_f -сходимость метода $ORTGF(1.)$ для сильно овражных задач

Задача	n	$ORTGF(1.), m_0 = n - 1$		$ORTGF(1.), m_0 = 10$	
		$iter(10^{-10})$	$iter(10^{-20})$	$iter(10^{-10})$	$iter(10^{-20})$
$Quad(2.)$	30	236(229,5)	332(315,5)	236(229)	332(315)
$Sabs(2.)$	30	476(464,12)	527(524,12)	462(459)	523(520)
$Quad(1.2)$	60	188(183,5)	277(259,5)	188(183)	277(259)
$Sabs(1.2)$	60	464(462,21)	541(539,21)	469(467)	544(542)
$Quad(1.2)$	100	428(422,8)	542(534,8)	428(422)	542(534)
$Sabs(1.2)$	100	1480(1475,19)	1564(1559,19)	1293(1291)	1375(1373)

мости, используя уменьшение объема области локализации множества экстремумов.

Однако, само преобразование – только одна составляющая методов. Вторая требует выбора направления исследования функции и способа регулировки шагового множителя. Используемые в $ORTGF$ вряд ли следует признать очень удачными, так как обязательное движение из точки по антисубградиенту сильно затрудняет анализ линий уровня $\varphi(y)$. Поэтому, более рациональными представляются схемы алгоритмов, где линии уровня $\varphi(y)$ анализируются или уточняются относительно фиксированной на некоторое время точки. При этом для вычисления $f(x)$ и $\partial f(x)$ можно обеспечить простые процедуры типа ε -наискорейшего спуска в преобразованном пространстве аргументов, используя ортогональность хранимых субградиентов или их выпуклых комбинаций. Просто решаются вопросы отсева лишних субградиентов, легко уточняются верхняя и нижняя оценка f^* и т. д.

Преобразование пространства естественно направить на то, чтобы направление исследования функции $f(x)$ из точки приближалось к направлению на оптимум. Тогда, изменяя по некоторому правилу фиксированную точку (например, на точку рекорда $f(x)$, или проекцию фиксированной точки на множество

$$Q_k = \left\{ y : \varphi(y_i) + (\partial\varphi(y_i), y - y_i) \leq f^{record}, \quad i \in I_k \right\}$$

и, повторяя такую же процедуру относительно новой точки, приходим к методам по типу ньютоновских (квазиньютоновских), которые будут применимы как для гладких, так и для негладких функций.

Заключение

Итак, применение описанных преобразований пространства, направленное на выравнивание структуры поверхностей уровня функции, позволяет существенно улучшить работоспособность субградиентных методов. Для задачи выпуклого программирования использование операторов (11) и (41) в сочетании с оператором растяжения пространства позволяет построить целую серию эффективных методов переменной метрики. При этом для методов в B -форме можно дать простую геометрическую интерпретацию процесса в преобразованном пространстве аргументов.

Рассмотренные преобразования далеко не исчерпывают перечень эффективных одноранговых преобразований пространства. Так, интерес представляет оптимальное одноранговое преобразование по типу (11), которое позволило бы улучшить сходимость в смысле уменьшения объема области локализации множества экстремумов. В рамках доортогонализирующих одноранговых преобразований перспективным представляется преобразование по типу симплексного, которое позволяет строить методы для выпуклых задач достаточно больших размеров, обходясь при этом хранением только конечного набора векторов, а не полной матрицы размера $n \times n$. Кроме того, можно построить и ряд других интересных линейных операторов ортогонализирующего типа.

Автор признателен профессору Н. З. Шору за конструктивное обсуждение данной работы.

Литература

1. ШОР Н. З. Использование операций растяжения пространства в задачах минимизации выпуклых функций // Кибернетика. – 1970. – № 1. – С. 6–12.
2. DAVIDON W. C. Variable metric methods for minimization, AEC Research and Development Rept. ANL 5990 (Rev.), 1959.
3. FLETCHER R., POWELL M. J. D. A rapidly convergent descent method for minimization // Comput. J. – 1963. – 6, № 2. – P. 163–168.

4. ПШЕНИЧНЫЙ Б. Н., ДАНИЛИН Ю. М. Численные методы в экстремальных задачах. – М.: Наука, 1975. – 319 с.
5. Поляк Б. Т. Введение в оптимизацию. – М.: Наука, 1983. – 384 с.
6. ШОР Н. З. Методы минимизации недифференцируемых функций и их приложения. – Киев: Наук. думка, 1979. – 199 с.
7. ШОР Н. З., СТЕЦЕНКО С. И. Квадратичные экстремальные задачи и недифференцируемая оптимизация. – Киев: Наук. думка, 1989. – 208 с.
8. СТЕЦЮК П. И. r -алгоритмы и эллипсоиды // Кибернетика и системный анализ. – 1996. – № 1. – С. 113–134.
9. СТЕЦЮК П. И. Метод центров тяжести простых тел // Там же. – 1996. – № 5. – С. 117–138.
10. S. AGMON The relaxation method for linear inequalities // Canadien J. of Mathematics. – 1954. – 6. – P. 382–392.
11. T. MOTZKIN AND I. J. SCHOENBERG The relaxation method for linear inequalities // Canadien J. of Mathematics. – 1954. – 6. – P. 393–404.
12. Поляк Б. Т. Минимизация негладких функционалов // Журн. вычислит. математики и матем. физики. – 1969. – 9, № 3. – С. 507–521.
13. ЩЕПАКИН М. Б. О методе ортогонального спуска // Кибернетика. – 1987. – № 1. – С. 58–62.
14. СКОКОВ В. А., ЩЕПАКИН М. Б. Численное исследование метода ортогонального спуска // Кибернетика и системный анализ. – 1994. – № 2. – С. 139–149.
15. ЩЕПАКИН М. Б., ШУБЕНКОВА И. А. Исследование модифицированного метода ортогонального спуска для поиска нуля выпуклой функции // Там же. – 1993. – № 4. – С. 63–72.
16. CAMERINI P., FRATTA L., MAFFIOLI F. On improving relaxation methods by modified gradient techniques // Math. Program. – 1975. – Study 3. – P. 26–34.
17. HIRIART-URRUTY J. B., LEMARÉCHAL C. Convex analysis and minimization Algorithms. – Berlin: Springer-Verlag, 1994. – Vol I-II.

18. ВОЕВОДИН В. В. Численные методы алгебры. – М.: Наука, 1966. – 248 с.
19. Nonsmooth optimization / Eds. Lemaréchal C., Mifflin R. – Oxford: Pergamon Press, 1978. – 186 p.
20. LEMARÉCHAL C. Numerical experiments in nonsmooth optimization. In: Progress in nondifferentiable optimization / Ed. Nurminski E. A. CP-82-S2, International Institute for Applied System Analysis: Laxenburg, Austria, 1982. – P. 61–84.

Классические фейеровские методы с преобразованием пространства

П. И. Стецюк

Методы решения экстремальных задач. – 1996. – С. 3–9.

Описаны два субградиентных метода с преобразованием пространства, которые используют классическую фейеровскую регулировку шагового множителя в направлении субградиента в преобразованном пространстве аргументов. Приведены численные эксперименты, показывающие эффективность этих методов.

Постановка задачи. Пусть имеется задача безусловной минимизации:

$$\min f(x), \quad (1)$$

где $f(x)$ – выпуклая функция векторного аргумента $x \in R^n$; R^n – евклидово пространство размерностью n со скалярным произведением (x, y) . Пусть множество экстремумов задачи (1) – X^* непусто и известно значение минимума $f(x)$: $f^* = f(x)$, $x \in X^*$. Не ограничивая общности, будем считать, что множество X^* состоит из единственной точки x^* .

Для $\varepsilon > 0$ обозначим $X_\varepsilon^* = \{x : f(x) - f^* \leq \varepsilon\}$. Точку $x_k \in X_\varepsilon^*$ будем считать решением задачи (1) с точностью ε по функционалу (ε -решением).

Входными параметрами рассматриваемых методов будут x_0 и ε , где x_0 – начальное приближение, $x_0 \in R^n$; ε – точность, с которой требуется решить задачу (1) по функционалу, $\varepsilon > 0$. Процесс вычислений будем останавливать, как только будет достигнута точка $x_k \in X_\varepsilon^*$.

Для решения задачи (1) с точностью ε по функционалу конструктивным является известный субградиентный метод, который использует классическую фейеровскую регулировку шагового множителя в направлении субградиента [1].

$Fejer(x_0, \varepsilon)$ – «классический» фейеровский метод. В начале процесса имеем $x_0 \in R^n$, $\varepsilon > 0$. Пусть на k -ом шаге получена $x_k \in R^n$, $f(x_k)$ и $\partial f(x_k)$ – вычисленные в ней значения функции и субградиента. Тогда, если $f(x_k) - f^* \leq \varepsilon$, то x_k – искомая точка и ОСТАНОВ. Иначе вычисляем очередное приближение:

$$x_{k+1} = x_k - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|}, \quad h_k = \frac{(f(x_k) - f^*)}{\|\partial f(x_k)\|}$$

и переходим к $(k + 1)$ -му шагу. Справедлива следующая теорема.

Теорема 1. Пусть на некотором шаге метода $Fejer(x_0, \varepsilon)$ выполнено $\|\partial f(x_k)\| \leq C$. Тогда он находит (ε -решение) задачи (1) не более, чем за K шагов, где $K = \left\lceil \left(\frac{C\|x_0 - x^*\|}{\varepsilon} \right)^2 \left[+1. \text{ Здесь } \right] a \left[- \text{целая часть действительного числа } a. \right. \right.$

Несмотря на теоретическую привлекательность, метод $Fejer(x_0, \varepsilon)$ при достаточно малых ε пользуется плохой репутацией в практическом отношении, даже для гладких выпуклых функций. Так, когда направление антисубградиента в точке достаточно близко к ортогональному направлению на минимум, что характерно для овражных функций, он неэффективен даже для небольших задач ($n \sim 10$). Для негладких выпуклых функций, где овражность является скорее правилом, чем исключением, метод $Fejer(x_0, \varepsilon)$ неэффективен для задач еще меньших размеров ($n \sim 2 - 5$). Так, для минимизации функции двух переменных $f(x_1, x_2) = |x_1| + k|x_2|$ он сходится со скоростью геометрической прогрессии со знаменателем $\sqrt{1 - \frac{1}{k^2}}$, который близок к единице даже при сравнительно небольших значениях k .

В смысле точности решения задачи (1) по функционалу ситуацию в лучшую сторону для методов с классическим фейеровским шагом способно изменить линейное преобразование пространства, направленное на выравнивание структуры поверхностей уровня функции в преобразованном пространстве (например, как это сделано в r -алгоритмах [2]). Практическую эффективность этих методов способно улучшить любое из линейных преобразований, направленное на уменьшение угла между двумя последовательными субградиентами, которые получены согласно классическому фейеровскому шагу в преобразованном пространстве аргументов. Однако не для любого из таких преобразований можно обосновать сходимость соответствующих им методов. В частности, простое обоснование субградиентных методов с преобразованием пространства, использующих классический фейеровский шаг в направлении субградиента, допускают те преобразования, которые не увеличивают оценку расстояния до множества экстремумов в преобразованном пространстве аргументов.

Ниже опишем два субградиентных метода с преобразованием пространства, которые используют классический фейеровский шаг в преобразованном пространстве аргументов. Они основаны на простых преобразованиях пространства, которые допускают одноранговый перерасчет матрицы при переходе в очередное преобразованное пространство

аргументов. Кроме того, эти преобразования не увеличивают оценку расстояния до множества экстремумов в преобразованном пространстве аргументов, что позволяет обосновать для этих методов конечную ε -сходимость, как для метода $Fejer(x_0, \varepsilon)$.

Метод $Fellg2p1(x_0, \varepsilon)$ – классический фейеровский метод с преобразованием пространства, использующим два последовательных субградиента и вектор агрегатного типа, который является выпуклой комбинацией вычисленных ранее субградиентов.

Перед началом вычислений имеем $\varepsilon > 0$, $x_0 \in R^n$. Тогда, если $f(x_0) - f^* \leq \varepsilon$, то x_0 – искомая точка и ОСТАНОВ. Иначе полагаем

$$h_0 = \frac{(f(x_0) - f^*)}{\|\partial f(x_0)\|}, \quad \xi_0 = \frac{\partial f(x_0)}{\|\partial f(x_0)\|} \in R^n,$$

$p_0 = 0 \in R^n$, $B_0 = I_n$ – единичная матрица размером $n \times n$.

Пусть на k -м шаге получены $x_k \in R^n$, h_k , $\xi_k \in R^n$, $p_k \in R^n$, B_k – матрица $n \times n$.

1. Вычислим очередное приближение:

$$x_{k+1} = x_k - h_k B_k \xi_k.$$

2. Вычислим $f(x_{k+1})$, $\partial f(x_{k+1})$. Тогда, если $f(x_{k+1}) - f^* \leq \varepsilon$, то x_{k+1} – искомая точка и ОСТАНОВ. В противном случае положим

$$\xi_{k+1} = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|}, \quad h_{k+1} = \frac{f(x_{k+1}) - f^*}{\|B_k^T \partial f(x_{k+1})\|}.$$

3. Вычислим

$$\lambda_1 = -\frac{(p_k, \xi_{k+1})}{\sqrt{(p_k, \xi_{k+1})^2 + (\xi_k, \xi_{k+1})^2}}, \quad \lambda_2 = -\frac{(\xi_k, \xi_{k+1})}{\sqrt{(p_k, \xi_{k+1})^2 + (\xi_k, \xi_{k+1})^2}}$$

и положим

$$p_{k+1} = \begin{cases} \lambda_1 p_k + \lambda_2 \xi_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 > 0, \\ p_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 \leq 0, \\ \xi_k, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 \leq 0, \\ 0, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 > 0. \end{cases}$$

4. Если $(p_{k+1}, \xi_{k+1}) \geq 0$, полагаем $B_{k+1} = B_k$ и переходим к п. 5.

Иначе вычисляем

$$\eta = \left(\frac{1}{\sqrt{1 - (p_{k+1}, \xi_{k+1})^2}} - 1 \right) \xi_{k+1} - \frac{(p_{k+1}, \xi_{k+1})}{\sqrt{1 - (p_{k+1}, \xi_{k+1})^2}} p_{k+1},$$

$$B_{k+1} = B_k (I + \eta \xi_{k+1}^T), \quad h_{k+1} = \frac{h_{k+1}}{\sqrt{1 - (p_{k+1}, \xi_{k+1})^2}},$$

$$p_{k+1} = \frac{1}{\sqrt{1 - (p_{k+1}, \xi_{k+1})^2}} (p_{k+1} - (p_{k+1}, \xi_{k+1}) \xi_{k+1}).$$

5. Переходим к очередному шагу с x_{k+1} , B_{k+1} , ξ_{k+1} , h_{k+1} , p_{k+1} .

Метод $Fortgn(x_0, \varepsilon)$ – метод ортогонального субградиентного спуска с классическим фейеровским шагом в направлении субградиента в преобразованном пространстве аргументов.

Пусть $B_0 = I_n$, $P_0 = \emptyset$. Здесь P_k , $k = 0, 1, \dots$ – множество хранимых субградиентов на k -м шаге.

Пусть на k -м шаге получены x_k , B_k и P_k . Вычислим $f(x_k)$ и $\partial f(x_k)$. Если $f(x_k) - f^* \leq \varepsilon$, то x_k – искомая точка и ОСТАНОВ. Иначе переходим к $(k+1)$ -му шагу.

1. Полагаем

$$\xi_k = \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}, \quad h_k = \frac{f(x_k) - f^*}{\|B_k^T \partial f(x_k)\|}.$$

2. Сформируем множество:

$$\tilde{P}_k = (p_i \in P_k : (p_i, \xi_k) < 0).$$

3. Если $\tilde{P}_k = \emptyset$, то полагаем

$$B_{k+1} = B_k, \quad \xi_{k+1} = \xi_k, \quad h_{k+1} = h_k$$

и переходим к п. 4. Иначе вычисляем вектор

$$\tilde{p}_k = \sum_{p_i \in P_k} (p_i, \xi_k) p_i$$

и пересчитываем параметры $B_{k+1} = B_k (I - \eta_1 \eta_2^T)$, где

$$\eta_1 = \frac{\xi_k - \tilde{p}_k}{\|\xi_k - \tilde{p}_k\|^2}, \quad \eta_2 = 2\xi_k - \tilde{p}_k,$$

$$\xi_{k+1} = \frac{\xi_k - \tilde{p}_k}{\|\xi_k - \tilde{p}_k\|}, \quad h_{k+1} = \frac{h_k}{\|\xi_k - \tilde{p}_k\|}.$$

4. Вычислим очередное приближение:

$$x_{k+1} = x_k - h_{k+1} B_{k+1} \xi_{k+1}.$$

5. Сформируем очередное множество

$$P_{k+1} = \left\{ p_i \in \tilde{P}_k : |(p_i, \xi_{k+1})| < \varepsilon_0 \right\} \cup \xi_{k+1},$$

где ε_0 — достаточно малый параметр, чтобы избежать накопления ошибок вычислений в процессе доортогонализации векторов. При вычислениях будем использовать $\varepsilon_0 = 10^{-6}$.

6. Переходим к очередному шагу с x_{k+1} , B_{k+1} , P_{k+1} .

Справедлива следующая теорема.

Теорема 2. Последовательность $(x_{k+1})_{k=0}^{\infty}$, генерируемая каким-либо из методов $Fellg2p1(x_0, \varepsilon)$ или $Fortgn(x_0, \varepsilon)$, удовлетворяет неравенству

$$\|A_{k+1}(x_{k+1} - x^*)\|^2 \leq \|A_k(x_k - x^*)\|^2 - \frac{(f(x_k) - f^*)^2}{\|B_k^T \partial f(x_k)\|^2}, \quad k = 0, 1, 2, \dots$$

Здесь $A_k = B_k^{-1}$, $A_{k+1} = B_{k+1}^{-1}$.

Из теоремы (2) следует

Теорема 3. Пусть на каждом шаге одного из методов $Fellg2p1(x_0, \varepsilon)$ или $Fortgn(x_0, \varepsilon)$ выполнены условия $\|B_k\| \leq c_1$ и $\|\partial f(x_k)\| \leq c_2$. Тогда он получает ε -решение задачи (1) не более, чем за K -шагов, где $K = \left\lceil \frac{c_1 c_2 \|x_0 - x^*\|}{\varepsilon} \right\rceil + 1$.

Численные эксперименты. Проверим численную устойчивость приведенных выше методов по отношению к точности решения задач по функционалу. Основным показателем их работы будем считать число итераций, т.е. число вычислений $f(x)$ и $\partial f(x)$. В качестве тестовых задач выберем два наиболее часто упоминаемых примера кусочно-квадратичных функций:

$$Shor(n = 5, f^* = 22.6001620958) \quad [2],$$

$$Maxquad(n = 10, f^* = -0.841408334596) \quad [3],$$

которые считаются плохими задачами для субградиентных методов. Столь скрупулезное указание f^* для задач *Shor* и *Maxquad* обусловлено тем, чтобы проверить работу методов при достаточно малых значениях ε . Результаты работы для всех трех методов при разных значениях ε приведены в таблице. Здесь «прочерк», означает, что метод не решил задачу с требуемой точностью за 10 000 итераций.

ε	<i>Shor</i> ($f(x_0) = 80$)			<i>Maxquad</i> ($f(x_0) = 5337.07$)		
	<i>Fejer</i>	<i>Fellg2pl</i>	<i>Fortgn</i>	<i>Fejer</i>	<i>Fellg2pl</i>	<i>Fortgn</i>
10.	4	3	4	8	5	7
1.	23	8	9	32	10	13
0.1	183	14	14	108	17	16
0.01	1713	20	18	684	22	22
0.001	—	28	22	6556	29	29
10^{-6}	—	38	33	—	41	45
10^{-8}	—	56	48	—	64	75
10^{-10}	—	70	59	—	85	95

Как видно из таблицы, методы с преобразованием пространства гораздо более эффективны, чем классический фейеровский метод без преобразования пространства. Число итераций, необходимых для получения ε -решения задач, здесь незначительно увеличивается при значительном уменьшении ε . Это подтверждает, что преобразования пространства, направленные на уменьшение овражности поверхностей уровня функции, способны для овражных функций значительно увеличить точность решения задач по функционалу для фейеровских методов.

Литература

1. Поляк Б. Т. Минимизация негладких функционалов // ЖВМ и МФ. — 1965. — **9**, № 3. — С. 507–521.
2. ШОР Н. З. Метод минимизации недифференцируемых функций и их приложения. — К.: Наук. думка. — 1979. — 199 с.
3. LEMARÉCHAL C., MIFFLIN R. Nonsmooth optimization // Oxford.: Pergamon Press. — 1978. — 186 p.

Об одном способе ускорения метода ортогонального субградиентного спуска

П. И. Стецюк

Оптимизация и её приложения. – 1997. – С. 42–49.

Получены условия, которые при минимизации существенно овражных кусочно-линейных выпуклых функций позволяют ускорить эффективность метода ортогонального субградиентного спуска, сохранив при этом монотонность по максимальному расстоянию до множества экстремумов.

В [1] рассмотрен доортогонализирующий одноранговый оператор преобразования пространства – линейный оператор из R^n в R^n (R^n – евклидово пространство размерностью n), который в матричной форме представим в следующем виде:

$$T_\lambda(p_m, p) = I + \frac{p_m - p}{\|p_m - p\|^2} \left(\frac{1}{\lambda} p_m + p \right)^T, \quad (1)$$

где I – единичная матрица размером $n \times n$, λ – скаляр, такой, что $\lambda(\lambda + 1) \neq 0$, $p = \sum_{i=1}^{m-1} (p_m, p_i) p_i$, где p_1, p_2, \dots, p_m ($m < n$) – нормированные векторы из R^n , которые удовлетворяют условиям

$$(p_i, p_j) = 0, \quad 1 < i \neq j < m - 1, \quad \|p_m - p\|^2 > 0.$$

Оператор (1) имеет простой геометрический смысл. Пусть в $X = R^n$ имеется $P = \{p_1, p_2, \dots, p_m\}$ – система m ($m < n$) линейно-независимых нормированных векторов, из которых первые $(m - 1)$ векторов взаимно ортогональны, а последний неортогонален хотя бы одному из предыдущих. Тогда в преобразованном оператором (1) пространстве $Y = T_\lambda(p_m, p)X$ образ системы P , т. е. $\tilde{P} = \{\tilde{p}_1, \tilde{p}_2, \dots, \tilde{p}_m\}$ будет образовывать систему m взаимно ортогональных векторов. При этом $\tilde{p}_i = p_i$, $i = 1, 2, \dots, m - 1$ (и, следовательно, они будут и нормированными), а $\tilde{p}_m = \frac{\lambda}{\lambda + 1} (p_m - p)$.

Для задач минимизации выпуклых функций оператор (1) позволяет строить методы, с преобразованием пространства переменных (методы

переменной метрики), реализуя при этом одноранговую коррекцию матрицы обратного преобразования B_k , как и при использовании оператора растяжения пространства [2]. Кроме того, он обеспечивает достаточно удобный механизм работы с ортогональным набором векторов в преобразованном пространстве переменных, который значительно упрощает решение внутренних подзадач.

В зависимости от того, какие векторы выбирать для процесса ортогонализации, можно получить ряд вычислительных схем методов с преобразованием пространства на основе оператора (1). Используя локализацию множества экстремумов некоторым специальным конусом (о нем речь идет ниже), можно строить методы минимизации овражных выпуклых функций, для которых преобразование пространства направлено на уменьшение степени овражности; как и в r -алгоритмах [2] «расширяется» конус возможных направлений убывания функции в преобразованном пространстве переменных. Используя это свойство, в [1] для нахождения точки минимума выпуклой функции $f(x)$ при априорном знании значения функции в точке минимума (f^*) построено семейство методов ортогонального субградиентного спуска с классическим фейеровским шагом в направлении антисубградиента $ORTGF(\lambda)$. Один из представителей этого класса методов при значении параметра $\lambda = -1/2$ (метод $Fortgn(x_0, \varepsilon)$ [3]) обладает тем свойством, что на каждом его шаге гарантировано монотонное уменьшение радиуса шара, локализующего множество экстремумов в преобразованном пространстве переменных.

Пусть в пространстве $X = R^n$ (пространство переменных) имеется некоторый конус $K_m(x_k, G_k)$:

$$\begin{aligned} K_m(x_k, G_k) &= K_m(x_k, g_1, \dots, g_m) = \\ &= \{x \in X : (x_k - x, g_i) > 0, \quad i = \overline{1, m}\}, \end{aligned} \quad (2)$$

где векторы $g_i \in G_k$, $i = \overline{1, m}$ — нормированные и линейно-независимые, $G_k \in G$, G — пространство субградиентов. И пусть этот конус содержит некоторое замкнутое ограниченное множество X^* (множество экстремумов), $X^* \in K_m(x_k, G_k)$, т. е. для любого $x^* \in X^*$ выполняются неравенства $(x_k - x^*, g_i) > 0$, $i = \overline{1, m}$.

Кроме того, пусть векторы из G_k удовлетворяют следующим условиям:

$$(g_i, g_j) = 0, \quad 1 < i \neq j < m-1, \quad (g_m, g_i) < 0, \quad 1 < i < m-1. \quad (3)$$

Конус (2) при выполнении условий (3) условимся называть специальным конусом $K_m(x_k, G_k)$.

Специальный конус $K_m(x_k, G_k)$ в X с помощью доортогонализирующего однорангового оператора $T_\lambda(g_m, g)$ можно преобразовать в ортогональный конус $K_m(y_k, \tilde{G}_k)$ в $Y = T_\lambda(g_m, g)X$ (преобразованное пространство переменных). Здесь $g = \sum_{i=1}^{m-1} (g_m, g_i)g_i$, $y_k = T_\lambda(g_m, g)x_k$ — образ точки x_k в преобразованном пространстве переменных, \tilde{G}_k — образ G_k в преобразованном пространстве субградиентов: $\tilde{g}_i = g_i$, $i = \overline{1, m-1}$, $\tilde{g}_m = \frac{\lambda}{\lambda+1}(g_m - g)$. Учитывая, что при $\lambda = -1/2$ справедливо

$$T_{-1/2}^T(g_m, g)T_{-1/2}(g_m, g) = I + \frac{1}{\|g_m - g\|^2} (-gg^T + g_m g^T + g g_m^T),$$

имеем

$$\begin{aligned} & \|T_{-1/2}(g_m, g)(x_k - x^*)\|^2 = \\ &= \left(x_k - x^*, T_{-1/2}^T(g_m, g)T_{-1/2}(g_m, g)(x_k - x^*) \right) = \\ &= \left((x_k - x^*), \left(I + \frac{1}{\|g_m - g\|^2} (-gg^T + g_m g^T + g g_m^T) \right) (x_k - x^*) \right) = \\ &= \|x_k - x^*\|^2 + 2 \frac{(g, x_k - x^*)(g_m, x_k - x^*)}{\|g_m - g\|^2} - \frac{(g, x_k - x^*)^2}{\|g_m - g\|^2} < \\ &< \|x_k - x^*\|^2. \end{aligned} \quad (4)$$

Здесь x^* — точка, принадлежащая X^* и максимально удаленная от точки x_k . Неравенство в (4) следует из того, что $(g_m, x_k - x^*) > 0$ и

$$(g, x_k - x^*) = \sum_{i=1}^{m-1} (g_m, g_i)(g_i, x_k - x^*) < 0.$$

Неравенство (4) означает, что оператор преобразования пространства $T_{-1/2}(g_m, g)$ преобразует специальный конус $K_m(x_k, G_k) \in X$ в ортогональный конус $K_m(y_k, \tilde{G}_k) \in Y$, не увеличивая максимальной оценки расстояния от точки $y_k = T_{-1/2}(g_m, g)x_k$ до множества экстремумов в Y , или, другими словами, в преобразованном пространстве переменных множество экстремумов остается в шаре, радиус которого не больше, чем радиус шара, локализирующего множество экстремумов в пространстве X .

Основываясь на этом принципе, для нахождения точки минимума (с точностью ε по функционалу) выпуклой функции $f(x)$ при априор-

ном знании f^* построен метод $Fortgn(x_0, \varepsilon)$ [3]. Он использует классический фейеровский шаг в направлении последнего вычисленного субградиента. Механизм отсева лишних субградиентов основан на том, чтобы множество экстремумов можно было погрузить в специальный конус $K_m(y_k, g_1, \dots, g_m)$, где g_1, \dots, g_m – субградиенты, вычисленные на предыдущих шагах метода.

Метод $Fortgn(x_0, \varepsilon)$ оказался на практике значительно более эффективным, чем классический фейеровский метод без преобразования пространства. Однако, хотя он и построен с целью улучшения структуры поверхностей овражных функций, он не всегда достигает этой цели. Особенно это характерно для существенно овражных кусочно-линейных выпуклых функций. Одна из причин состоит в том, что оператор $T_{-1/2}(g_m, g)$ не является «сжимающим» пространство субградиентов (т. е. $|\det(T_{-1/2}(g_m, g))| = 1$). Это способствует накоплению ошибок в матрице обратного преобразования B_k , особенно для существенно овражных функций. И, если на гладких овражных функциях это сильно не сказывается, так как в этом случае сходимости метода сопутствует стремление к нулю нормы текущего субградиента, что обеспечивает и достаточно хороший механизм отсева лишних субградиентов. Для кусочно-линейных функций это не так. Здесь сходимость метода обеспечивается за счет накопления числа субградиентов, образующих ортогональный конус в преобразованном пространстве переменных, которое близко к n (n – размерность пространства переменных), и при существенной овражности процессу ортогонализации субградиентов сопутствует накопление ошибок.

В силу сказанного можно считать, что для схемы методов, построенных на локализации множества экстремумов специальным конусом $K_m(x_k, G_k)$ с сохранением монотонности по максимальному расстоянию до множества экстремумов, резерв практической эффективности скрыт в монотонном уменьшении $\det(B_k)$, где B_k – матрица обратного преобразования. Это в какой-то мере подтверждают численные эксперименты с методом $ORTGF(\lambda)$ при $\lambda = 1$ (в этом случае, детерминант уменьшается в два раза), который оказывается эффективным для существенно овражных кусочно-линейных функций даже при небольшом числе хранимых субградиентов. Однако, метод $ORTGF(1.)$ не сохраняет монотонность по максимальному расстоянию до множества экстремумов в преобразованном пространстве переменных. Поэтому, интерес представляют схемы методов по типу методов ортогонального субградиентного спуска, которые сохраняют монотонность по максимальному расстоянию до множества экстремумов в преобразованном простран-

стве и при этом уменьшают $\det(B_k)$, или, что то же самое, уменьшают объем эллипсоида, локализирующего множество экстремумов.

Очевидно, что это можно сделать, так как конус $K_m(x_k, G_k)$ в X можно погружать в более простые конусы $K_2(x_k, p_1, p_2)$, где $(p_1, p_2) < 0$. А уже для тела, полученного пересечением конуса $K_2(x_k, p_1, p_2)$ и шара $S(x_k, \|x_k - x^*\|)$, согласно [4], существует эллипсоид

$$\text{ell}^*(x_k, a, b, \|x_k - x^*\|),$$

содержащий это тело и объем, меньший объема шара $S(x_k, \|x_k - x^*\|)$. Преобразование $\text{ell}^*(x_k, a, b, \|x_k - x^*\|)$ в шар $S(y_k, \|x_k - x^*\|)$ можно реализовать либо последовательным применением двух операций растяжения пространства [4], либо используя одноранговый эллипсоидальный оператор [1]. Учитывая, что преобразование этого эллипсоида в шар также не увеличивает максимальной оценки расстояния до множества экстремумов в преобразованном пространстве переменных, то сочетания трех указанных операторов позволяет создавать различные комбинированные схемы методов ортогонального субградиентного спуска, ускоренные в смысле уменьшения $\det(B_k)$. При этом естественно ожидать более эффективной работы от таких методов.

Одна из таких комбинированных схем позволяет улучшить метод $\text{Fortgn}(x_0, \varepsilon)$, сохранив при этом все накопленные к k -му шагу субградиенты. В ее основе лежит следующая простая задача: как и при каких условиях множество экстремумов, локализованное специальным конусом $K_m(x_k, G_k)$, можно аппроксимировать с помощью эллипсоида $\text{ell}^*(x_k, a, b, \|x_k - x^*\|)$, чтобы после преобразований этого эллипсоида в шар $S(y_k, \|x_k - x^*\|)$ не изменить структуру специального конуса, т. е., чтобы в преобразованном пространстве аргументов образом специального конуса $K_m(x_k, G_k)$ был конус $K_m(y_k, \tilde{G}_k)$, где векторы из \tilde{G}_k удовлетворяют условиям (3).

Легко видеть, что операция растяжения пространства переменных с произвольным положительным коэффициентом как в направлении $\frac{g_m - g}{\|g_m - g\|}$, так и в направлениях g_i , $1 < i < m - 1$, не изменит структуры специального конуса. На самом деле, при растяжении пространства переменных с коэффициентом α в направлении g_{i^*} , где $1 < i^* < m - 1$ пространство субградиентов преобразуется согласно формуле $\tilde{G} = R_\beta(g_{i^*})G$, где $\beta = 1/\alpha$, т. е.

$$\tilde{G}_k = \begin{cases} \tilde{g}_m = g_m + (\beta - 1)(g_m, g_{i^*})g_{i^*}, \\ \tilde{g}_i = g_i, & 1 < i \neq i^* < (m - 1), \\ \tilde{g}_{i^*} = \beta g_{i^*}, \end{cases}$$

откуда следует, что $(\tilde{g}_i, \tilde{g}_j) = 0$, $1 < i \neq j < m-1$, и, кроме того,

$$(\tilde{g}_m, \tilde{g}_i) = (g_m, g_i) + (\beta - 1)(g_m, g_{i^*})(g_i, g_{i^*}) = (g_m, g_i) < 0,$$

$$1 < i \neq i^* < m-1,$$

и

$$(\tilde{g}_m, \tilde{g}_{i^*}) = (g_m, g_{i^*}) + (\beta - 1)(g_m, g_{i^*}) = \beta(g_m, g_{i^*}) < 0,$$

так как $\beta = 1/\alpha > 0$. При растяжении пространства переменных с коэффициентом α в направлении $\frac{g_m - g}{\|g_m - g\|}$ имеем

$$\tilde{G}_k = \begin{cases} \tilde{g}_m &= g_m + (\beta - 1)(g_m - g); \\ \tilde{g}_i &= g_i, \quad 1 < i < (m-1), \end{cases}$$

откуда следует, что $(\tilde{g}_i, \tilde{g}_j) = (g_i, g_j) = 0$, $1 < i \neq j < m-1$, и, кроме того, $(\tilde{g}_m, \tilde{g}_i) = (g_m, g_i) < 0$, $1 < i < m-1$. Из того, что рассмотренные направления растяжения взаимно ортогональны, следует, что структура конуса $K_m(y_k, \tilde{G}_k)$ не изменится, если отнормировать образы векторов в преобразованном пространстве субградиентов, даже в том случае, если выполнить m растяжений пространства, используя различные коэффициенты в указанных направлениях.

Преобразование эллипсоида $\text{ell}^*(x_k, a, b, \|x_k - x^*\|)$, содержащего пересечение конуса $K_2(x_k, p_1, p_2)$ и шара $S(x_k, \|x_k - x^*\|)$, в новый шар $S(y_k, \|x_k - x^*\|)$, требует последовательного применения двух операторов растяжения пространства в ортогональных направлениях — в направлении $\xi = \frac{p_2 - p_1}{\|p_2 - p_1\|}$ с коэффициентом $\alpha = 1/\sqrt{1 + (p_1, p_2)}$ и в направлении $\eta = \frac{p_2 + p_1}{\|p_2 + p_1\|}$ с коэффициентом $\alpha = 1/\sqrt{1 - (p_1, p_2)}$. Чтобы при произвольном i^* , $1 < i^* < m-1$, эти направления не изменяли структуры специального конуса в преобразованном пространстве переменных, достаточно в качестве образующих p_1 и p_2 для конуса $K_2\left(y_k, \frac{p_1}{\|p_1\|}, \frac{p_2}{\|p_2\|}\right)$ выбрать следующие линейные комбинации (с неотрицательными коэффициентами) векторов из G_k :

$$p_1 = g_m - g + (g_m, g_{i^*})g_{i^*} = g_m - \sum_{i=1, i \neq i^*}^{m-1} (g_m, g_i)g_i, \quad (5)$$

$$p_2 = g_m - g - (g_m, g_{i^*})g_{i^*} = g_m - \sum_{i=1, i \neq i^*}^{m-1} (g_m, g_i)g_i - 2(g_m, g_{i^*})g_{i^*}. \quad (6)$$

На самом деле, учитывая, что $\|p_1\|^2 = \|p_2\|^2 = \|p_m - p\|^2 + (g_m, g_{i^*})^2$, направления $\xi = \frac{p_2 - p_1}{\|p_2 - p_1\|} = g_{i^*}$ и $\eta = \frac{p_2 + p_1}{\|p_2 + p_1\|} = \frac{g_m - g}{\|g_m - g\|}$ будут такими, растяжение по которым не изменяет структуру специального конуса.

И, наконец, для того, чтобы имела место внешняя аппроксимация множества экстремумов эллипсоидом $\text{ell}^*(x_k, a, b, \|x_k - x^*\|)$, угол между векторами p_1 и p_2 , вычисленными согласно (5)–(6), должен быть тупым. Учитывая, что

$$\cos \varphi = \left(\frac{p_1}{\|p_1\|}, \frac{p_2}{\|p_2\|} \right) = \frac{\|g_m - g\|^2 - (g_m, g_{i^*})^2}{\|g_m - g\|^2 + (g_m, g_{i^*})^2},$$

угол между векторами p_1 и p_2 будет тупым, если

$$(g_m, g_{i^*})^2 > \|g_m - g\|^2 = 1 - \sum_{i=1}^{m-1} (g_m, g_i)^2. \quad (7)$$

Выполнение условия (7) достаточно для того, чтобы ускорить метод $\text{Fortgn}(x_0, \varepsilon)$ (в смысле уменьшения $\det(B_k)$) за счет вспомогательных шагов, основанных на погружении конуса специального типа в более простой конус, пересечение которого с шаром можно аппроксимировать эллипсоидом меньшего объема. При этом константа уменьшения объема (уменьшает $\det(B_k)$) связана со структурой специального конуса $K_m(x_k, G_k)$ и будет равна

$$q = \sqrt{1 - \cos^2 \varphi} = \frac{2|(g_m, g_{i^*})| \cdot \|g_m - g\|}{\|g_m - g\|^2 + (g_m, g_{i^*})^2} = \frac{2}{\frac{\|g_m - g\|}{|(g_m, g_{i^*})|} + \frac{|(g_m, g_{i^*})|}{\|g_m - g\|}} < 1.$$

Вспомогательный шаг будет означать либо последовательное применение двух операторов растяжения пространства, либо применение однорангового эллипсоидального оператора [1]. При этом таких вспомогательных шагов будет не более $m - 1$.

Естественно, что для существенно овражных кусочно-линейных выпуклых функций такое улучшение должно привести к более устойчивой работе методов по типу ортогонального субградиентного спуска.

Это показали численные эксперименты с ускоренным вариантом метода ортогонального субградиентного спуска с использованием на вспомогательных шагах оператора растяжения пространства.

Литература

1. СТЕЦЮК П. І. Субградієнтні методи з перетворенням простору для мінімізації негладких опуклих функцій: Автореф. дис. канд. фіз.-мат. наук. — Київ, 1996. — 16 с.
2. ШОР Н. З. Методы минимизации недифференцируемых функций и их приложения. — Киев: Наук. думка, 1979. — 199 с.
3. СТЕЦЮК П. И. Классические фейеровские методы с преобразованием пространства // Методы решения экстремальных задач. — Киев: Ин-т кибернетики им. В. М. Глушкова НАН Украины, 1996. — С. 3–9.
4. СТЕЦЮК П. И. r -алгоритмы и эллипсоиды // Кибернетика и систем. анализ. — 1996. — № 1. — С. 113–134.

Об одном методе для нахождения допустимой точки выпуклого неравенства

П. И. Стецюк

Теория оптимальных решений. – 2000. – С. 3–10.

Предложен метод фейеровского типа с преобразованием пространства для нахождения допустимой точки выпуклого неравенства. Приведены: вычислительная схема метода, обоснование его сходимости, тестовые расчеты для существенно овражных функций.

1. Постановка задачи. Пусть имеется неравенство:

$$f(x) \leq 0, \quad (1)$$

где $f(x)$ – выпуклая функция векторного аргумента $x \in R^n$. Здесь R^n – евклидово пространство размерности n со скалярным произведением (x, y) , $x \in R^n$, $y \in R^n$. Неравенство (1) называется выпуклым неравенством.

Пусть X^* – множество допустимых точек (решений) неравенства (1), т. е.

$$X^* = \{x : f(x) \leq 0\}. \quad (2)$$

Какова структура множества X^* и какое влияние она оказывает на методы для нахождения решений неравенства (1)? Выделим три случая, каждый из которых в значительной степени определяет требования к методам:

$X^* = \emptyset$ – множество X^* может быть пустым, означает, что неравенство (1) не имеет решений. В этом случае требуется обеспечить выполнение некоторых достаточных условий, которые дают основания утверждать что $X^* = \emptyset$.

$X^* \equiv x^*$ – множество X^* может содержать единственную точку x^* , означает, что неравенство (1) имеет единственное решение. В этом случае в окрестности x^* функция, как правило, имеет овражный характер.

$X^* \equiv M$ (множество M имеет непустую внутренность), означает, что неравенство (1) имеет сколь-угодно много решений. Здесь множество M – выпуклое замкнутое подмножество из R^n и выпуклое замкнутое ограниченное подмножество из R^n , когда кроме выпуклости функция

$f(x)$ удовлетворяет следующему свойству: $\lim_{\|x\| \rightarrow +\infty} f(x) = +\infty$. Этот случай часто проще для методов, чем предыдущий, так как речь идет о нахождении некоторой точки из множества, имеющего внутренние точки.

Задача нахождения допустимой точки неравенства (1) эквивалентна следующей задаче оптимизации:

$$\min_{x \in R^n} F(x), \quad \text{где} \quad F(x) = \max\{f(x), 0\}. \quad (3)$$

Здесь $F(x)$ – выпуклая функция векторного аргумента $x \in R^n$, как результат взятия функции максимума от двух выпуклых функций. Пусть F^* – минимальное значение функции $F(x)$. Тогда $F^* > 0$ – равносильно случаю $X^* = \emptyset$, а $F^* = 0$ – задает два других случая и при этом множество минимумов функции $F(x)$ совпадает с X^* . Для нахождения $x^* \in X^*$ можно использовать методы безусловной минимизации выпуклых функций.

В силу большого количества существующих методов безусловной минимизации выпуклых функций, казалось бы нет никаких проблем для нахождения допустимой точки неравенства (1), и вопрос нахождения $x^* \in X^*$ связан со сходимостью того или иного метода применительно к задаче (3). Как правило, значительную проблему для методов минимизации выпуклых функций представляют существенно овражные функции $f(x)$, т. е. когда поверхности уровня функции $f(x)$ имеют сильно выраженную овражную структуру. Поэтому интерес представляет разработка методов для нахождения решений выпуклого неравенства, которые бы устойчиво работали для негладких выпуклых функций в том числе и существенно овражной структуры. В какой-то мере этому требованию удовлетворяет предлагаемый метод FEG2P1.

2. Назначение метода FEG2P1. Очевидно, что когда $X^* \equiv x^*$ и функция $f(x)$ имеет достаточно сложный вид (например, $f(x)$ – кусочно-квадратичная), точное нахождение самой точки x^* не всегда корректная задача. Поэтому вопрос о нахождении допустимой точки выпуклого неравенства поставим в зависимость от точности, с которой обеспечим выполнение неравенства (1).

Пусть множество X^* не пустое. Для $\varepsilon_f > 0$, обозначим

$$X_{\varepsilon_f}^* = \{x : f(x) \leq \varepsilon_f\}. \quad (4)$$

Точку $\bar{x} \in X_{\varepsilon_f}^*$ будем считать ε_f -допустимым по функционалу решением (ε_f -решением) неравенства (1), т. е. в любой точке из $X_{\varepsilon_f}^*$ неравенство (1) нарушается не больше чем на величину равную ε_f .

Метод FEG2P1 предназначен для нахождения ε_f -решения неравенства (1) и обеспечивает хорошую скорость сходимости при малых значениях параметра ε_f , в том числе и для существенно овражных функций. Он дает вполне конкретные ответы к какому бы из вышеприведенных случаев множество X^* не относилось. Случай $X^* = \emptyset$ для метода FEG2P1 реализован посредством получения достаточного условия, что в шаре радиуса r_0 с центром в стартовой точке x_0 нет ни одной точки из множества X^* . Когда $X^* \equiv M$ метод находит точку, в которой значение $f(x)$ в точности равно нулю. В обоих случаях параметр ε_f большой роли не играет. Только когда $X^* = x^*$ точность нахождения решения зависит от использования параметра ε_f . Для метода FEG2P1 без особой боязни можно использовать достаточно малые значения параметра ε_f , скажем $\varepsilon_f \sim 10^{-12} - 10^{-25}$. Более того можно использовать и значения $\varepsilon_f \sim 10^{-25} - 10^{-40}$ и тот факт, что при таких достаточно малых по отношению к ограниченной по разрядности компьютерной арифметике значениях параметра ε_f метод ведет себя устойчиво даже для функций овражной структуры, самая большая загадка для автора.

3. Кратко о методе FEG2P1. Он является итерационным методом первого порядка (использует значение функции $f(x)$ и субградиент $\partial f(x)$ в точке) и при обосновании сходимости использует следующее свойство субградиента $\partial f(x)$ выпуклой функции $f(x)$:

$$(x - x^*, \partial f(x)) \geq f(x), \quad (5)$$

которое справедливо для $\forall x \in R^n$ и для $\forall x^* \in X^*$.

Метод FEG2P1 относится к классу субградиентных методов в преобразованном линейном оператором пространстве переменных. В названии метода автор пытался передать содержательный смысл итерационного процесса, который поставлен в основу метода, в силу чего аббревиатура «FEG2P1» была образована следующим принципом:

- F** – метод использует фейеровский (Fejer-type) шаг в направлении субградиента в преобразованном пространстве переменных на основе соотношения по типу (5);
- E** – метод основан на внешней аппроксимации множества допустимых точек монотонно уменьшающимся по объему эллипсоидом (Ellipsoid), что реализовано с помощью использования «однорангового эллипсоидального оператора» [1];

G2 – метод использует два (2) субградиента (subGradient), которые вычислены в последовательных точках процесса. Если угол между ними тупой, то эти субградиенты участвуют в преобразовании пространства переменных;

P1 – для усиления преобразования (более сильного уменьшения объема эллипсоида, локализирующего множество допустимых точек) используется один (1) вектор агрегатного типа, который является линейной комбинацией с положительными (Positive) коэффициентами вычисленных ранее субградиентов.

Преобразование пространства в методе FEG2P1 направлено на улучшение структуры поверхностей уровня овражных функций в очередном преобразованном пространстве переменных (по типу r -алгоритмов [2]), что обеспечивает для него вполне удовлетворительную скорость сходимости (по числу итераций) при работе с функциями существенно овражной структуры.

4. Вычислительная схема метода FEG2P1. Пусть для $\forall x \in R^n$ и для $\forall x^* \in X^*$ значение функции $f(x)$ и ее субградиента $\partial f(x)$ удовлетворяет следующему свойству:

$$(x - x^*, \partial f(x)) \geq \gamma f(x) \geq f(x), \quad (6)$$

где γ – некоторый скаляр ($\gamma \geq 1$), который условимся называть показателем максимального сдвига по выпуклости функции $f(x)$. Параметр γ введен с той целью, чтобы метод можно было использовать для более узкого класса функций, чем просто выпуклые, так, например, квадратичная выпуклая функция удовлетворяет свойству (6) с параметром $\gamma = 2$.

Входными параметрами метода будут:

γ – показатель максимального сдвига по выпуклости функции $f(x)$;

x_0 – начальная стартовая точка для процесса ($x_0 \in R^n$);

r_0 – радиус шара $S(x_0, r_0)$ с центром в точке x_0 , в котором по предположению существует хотя бы одна точка из X^* ;

ε_f – точность, с которой будет отыскиваться допустимая точка неравенства (1) по функционалу, $\varepsilon_f > 0$.

Процесс вычислений будем останавливать, как только найдем ε_f -решение неравенства (1) (т. е. получена точка $x_k \in X_{\varepsilon_f}^*$), либо когда получено достаточное условие того, что нет допустимой точки в шаре $S(x_0, r_0)$. Первый случай будем идентифицировать как **ОСТАНОВ** ($x_k \in X_{\varepsilon_f}^*$), а второй – как **ОСТАНОВ** ($X^* \cap S(x_0, r_0) \equiv \emptyset$).

При описании вычислительной схемы будем использовать:

знак « $:=$ » – для присвоения значений векторам и матрицам;

знак « $=$ » – для присвоения значений скалярным переменным.

Кроме того будем придерживаться стандартных математических обозначений: (\cdot, \cdot) – для скалярного произведения; $(\cdot)^T$ – для транспонирования векторов и матриц; $\|\cdot\|$ – для норм векторов. Для обозначения векторов и матриц в ходе итерационного процесса использованы символы: x – вектор из R^n (используется для текущего приближения ε_f -решения); ξ , η , p – векторы из R^n (используются для нормированного субградиента в преобразованном пространстве, направления для преобразования пространства и нормированного агрегатного вектора); B – матрица размером $n \times n$ (задает оператор обратного преобразования пространства переменных); I_n – единичная матрица размера $n \times n$.

Пусть заданы: γ , x_0 , r_0^2 , ε_f и пусть имеется процедура, которая в конкретной точке $\bar{x} \in R^n$ вычисляет значение функции $f(\bar{x}) \in R^1$ и компоненты ее субградиента $\partial f(\bar{x}) \in R^n$. Тогда вычислительная схема метода $FEFG2P1(\gamma, x_0, r_0, \varepsilon_f)$ принимает следующий вид.

Итерация 0 (Инициализация процесса). Вычислим $f(x_0)$ и $\partial f(x_0)$.

Если $f(x_0) \leq \varepsilon_f$, то ОСТАНОВ ($x_0 \in X_{\varepsilon_f}^*$).

Если $\gamma^2(f(x_0))^2 > r_0^2 \|\partial f(x_0)\|^2$, то ОСТАНОВ ($X^* \cap S(x_0, r_0) \equiv \emptyset$).

В противном случае установим

$$h_0 = \frac{\gamma f(x_0)}{\|\partial f(x_0)\|}, \quad \xi_0 := \frac{\partial f(x_0)}{\|\partial f(x_0)\|}, \quad p_0 := 0, \quad B_0 := I_n$$

и перейдем к очередной итерации.

($k+1$)-я итерация. Пусть на k -ой итерации получены $x_k \in R^n$, r_k^2 , h_k , $\xi_k \in R^n$, $p_k \in R^n$, B_k – матрица размера $n \times n$. Тогда выполним следующую последовательность операций:

Шаг 1. Вычислим очередное приближение

$$x_{k+1} := x_k - h_k B_k \xi_k$$

и квадрат радиуса шара в преобразованном пространстве переменных

$$r_{k+1}^2 = r_k^2 - h_k^2.$$

Шаг 2. Вычислим $f(x_{k+1})$ и $\partial f(x_{k+1})$. Если $f(x_{k+1}) \leq \varepsilon_f$, то ОСТАНОВ ($x_{k+1} \in X_{\varepsilon_f}^*$). Если $\gamma^2 f^2(x_{k+1}) > r_{k+1}^2 \|B_k^T \partial f(x_{k+1})\|^2$, то ОСТАНОВ ($X^* \cap S(x_0, r_0) \equiv \emptyset$).

В противном случае положим

$$\xi_{k+1} := \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|}, \quad h = \frac{\gamma f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|}.$$

Шаг 3. Вычислим

$$\lambda_1 = -\frac{(p_k, \xi_{k+1})}{\sqrt{(p_k, \xi_{k+1})^2 + (\xi_k, \xi_{k+1})^2}}, \quad \lambda_2 = -\frac{(\xi_k, \xi_{k+1})}{\sqrt{(p_k, \xi_{k+1})^2 + (\xi_k, \xi_{k+1})^2}}$$

и положим

$$p := \begin{cases} \lambda_1 p_k + \lambda_2 \xi_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 > 0; \\ p_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 \leq 0; \\ \xi_k, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 > 0; \\ 0, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 \leq 0. \end{cases}$$

Шаг 4. Если $(p, \xi_{k+1}) \geq 0$, полагаем

$$B_{k+1} := B_k; \quad h_{k+1} = h; \quad p_{k+1} := 0$$

и переходим к шагу 5. Иначе вычисляем

$$\begin{aligned} \eta &:= \left(\frac{1}{\sqrt{1 - (p, \xi_{k+1})^2}} - 1 \right) \xi_{k+1} - \frac{(p, \xi_{k+1})}{\sqrt{1 - (p, \xi_{k+1})^2}} p, \\ B_{k+1} &:= B_k (I_n + \eta \xi_{k+1}^T), \quad h_{k+1} = \frac{h}{\sqrt{1 - (p, \xi_{k+1})^2}}, \\ p_{k+1} &:= \frac{1}{\sqrt{1 - (p, \xi_{k+1})^2}} (p - (p, \xi_{k+1}) \xi_{k+1}). \end{aligned}$$

Шаг 5. Переходим к очередной итерации с x_{k+1} , r_{k+1}^2 , h_{k+1} , ξ_{k+1} , p_{k+1} и B_{k+1} .

Что можно сказать о сходимости метода? Пусть $X_1^* = X^* \cap S(x_0, r_0)$. Тогда для метода FEG2P1 справедливы следующие теоремы.

Теорема 1. Пусть $X_1^* \neq \emptyset$ и выполнено условие (6). Тогда, генерируемая методом FEG2P1($\gamma, x_0, r_0, \varepsilon_f$) последовательность $\{x_{k+1}\}_{k=0}^\infty$ удовлетворяет неравенству

$$\|B_{k+1}^{-1}(x_{k+1} - x^*)\|^2 \leq \|B_k^{-1}(x_k - x^*)\|^2 - \left(\frac{\gamma f(x_k)}{\|B_k^T \partial f(x_k)\|} \right)^2,$$

$k = 0, 1, 2, \dots$, для любой точки $x^* \in X_1^*$.

Теорема 2. Пусть на каждом шаге метода $FEG2P1(\gamma, x_0, r_0, \varepsilon_f)$ выполнены условия $\|B_k\| \leq c_1$ и $\|\partial f(x_k)\| \leq c_2$. Тогда не более чем за K -шагов, где $K = \left\lceil \left(\frac{c_1 c_2 r_0}{\gamma \varepsilon_f} \right)^2 \right\rceil + 1$, метод $FEG2P1(\gamma, x_0, r_0, \varepsilon_f)$ заканчивает работу выполнением одного из условий:

(i) найдено ε_f -решение неравенства (1);

(ii) найдено, что $X_1^* = \emptyset$, когда $X^* = \emptyset$.

5. Вычислительные эксперименты. Метод зарекомендовал себя эффективным (по числу итераций) при работе с овражными функциями, в том числе и негладкими, подтверждение [1, 3]. Здесь приведем тестовые расчеты для существенно овражных функций.

Тестовый пример построен следующим образом. Задача нахождения решения совместной системы линейных уравнений $Ax = b$ сводилась к задаче нахождения допустимой точки выпуклого неравенства тремя различными способами:

$$Quad(f_1(x) = \|Ax - b\|^2),$$

$$Sabs(f_2(x) = \sum_{i=1}^n \left| \sum_{j=1}^n a_{ij}x_j - b_i \right|),$$

$$Mabs(f_3(x) = \max_{i=1}^n \left| \sum_{j=1}^n a_{ij}x_j - b_i \right|).$$

В качестве матрицы A использовалась плохо обусловленная симметричная матрица Гильберта с коэффициентами $a_{ij} = 1 / (i + j - 1)$, а в качестве вектора b – вектор равный Ae , где $e = (1, \dots, 1)$. При этом все три выпуклых неравенства имеют единственную допустимую точку $x^* = (1, \dots, 1)$. При $n > 10$ нахождение этой точки сопряжено с значительными трудностями из-за существенной овражности функций $f_1(x)$, $f_2(x)$ и $f_3(x)$. Кроме того две последних еще и негладкие.

Результаты численных экспериментов приведены в таблице. Здесь $iter$ – число итераций, которые затрачены на поиск приближения к x^* методом $FEG2P1$, начиная с начальной стартовой точки $x_0 = (0, 0, \dots, 0)$.

Для задачи $Quad$ использовалось значение $\gamma = 2$ в силу квадратичности функции $f_1(x)$, а для задач $Sabs$ и $Mabs$ – $\gamma = 1$. Как видно из таблицы метод $FEG2P1$ – достаточно устойчив по нахождению приближения к x^* для всех трех функций при большом числе переменных.

	$Quad(\varepsilon_f = 10^{-24})$		$Sabs(\varepsilon_f = 10^{-12})$		$Mabs(\varepsilon_f = 10^{-12})$	
n	iter	$\ x_{iter}^1 - x^*\ $	iter	$\ x_{iter}^2 - x^*\ $	iter	$\ x_{iter}^3 - x^*\ $
2	2	8.08255E-16	2	1.42178E-15	4	.00000
3	3	1.30422E-15	4	6.62358E-12	36	1.98696E-14
4	7	9.16107E-13	19	2.07360E-09	89	3.59747E-09
5	8	1.42891E-11	41	2.38243E-09	134	1.12863E-07
6	41	9.90435E-10	69	3.70066E-07	184	8.98021E-06
7	31	1.82534E-04	109	9.85769E-06	181	1.85695E-04
8	57	3.49176E-05	95	3.63062E-05	250	9.13811E-05
9	50	8.68367E-05	100	8.57339E-05	262	9.52865E-05
10	67	1.63117E-04	127	1.92725E-05	272	1.72554E-04
15	46	1.73133E-04	155	2.80517E-05	358	1.91924E-04
20	62	1.12086E-04	156	1.07970E-04	437	2.09268E-04
25	98	2.64809E-04	178	6.10850E-05	424	3.02368E-04
30	104	1.28632E-04	188	1.21749E-04	503	3.49949E-04
35	76	2.14721E-04	222	5.74886E-05	546	2.81749E-04
40	124	1.07358E-04	211	9.40017E-05	564	4.06881E-04
45	116	1.43121E-04	211	1.19156E-04	584	5.67897E-04
50	124	1.95499E-04	255	5.23825E-05	680	4.08678E-04

Литература

1. СТЕЦЮК П. И. Ортогонализирующие линейные операторы в выпуклом программировании // Кибернетика и системный анализ. 1997. – №3. – Ч. I. – С.97-119. – 1997. – №5. – Ч. II. – С. 111–124.
2. ШОР Н. З. Методы минимизации недифференцируемых функций и их приложения. – К.: Наук. думка, 1979. – 199 с.
3. СТЕЦЮК П. И. Классические фейеровские методы с преобразованием пространства // Методы решения экстремальных задач. – К.: Ин-т кибернетики им В. М. Глушкова НАН Украины, 1996. – С. 3–9.

К методам решения плохообусловленных систем линейных уравнений

П. И. Стецюк

Теория оптимальных решений. – 2001. – С. 9–15.

Предложены новые итерационные методы для решения системы линейных алгебраических уравнений. Эффективность методов подтверждена численными экспериментами при решении плохо обусловленных систем линейных уравнений с матрицей Гильберта.

1. В работе [1] предложен итерационный субградиентный метод с преобразованием пространства переменных (метод FEG2P1) для нахождения ε -решений выпуклого неравенства

$$f(x) \leq 0, \quad (1)$$

где $f(x)$ – выпуклая функция векторного аргумента $x \in R^n$ (функция $f(x)$ может быть как гладкой, так и негладкой). Здесь R^n – евклидово пространство размерностью n со скалярным произведением (x, y) , $x \in R^n$, $y \in R^n$.

Под ε -решением неравенства (1) понимается такая точка $x_\varepsilon^* \in R^n$, для которой справедливо неравенство

$$f(x_\varepsilon^*) \leq \varepsilon, \quad (2)$$

где $\varepsilon > 0$. Естественно, что вычислительный процесс, который позволяет находить ε -решения неравенства (1) при достаточно малых значениях параметра ε ($\varepsilon \rightarrow +0$), будет давать достаточно хорошие приближения к $x^* \in X^*$, где X^* – множество допустимых точек (решений) выпуклого неравенства (1):

$$X^* = \{x : f(x) \leq 0\}. \quad (3)$$

Для того чтобы на практике обеспечить высокую точность таких приближений (в пределах машинной точности компьютера), вполне достаточно обеспечить хорошую работу вычислительного процесса при значениях $\varepsilon \sim 10^{-20} - 10^{-15}$, независимо от структуры поверхностей уровня функции $f(x)$. Метод FEG2P1, как правило, позволяет это сделать, гарантируя хорошую скорость сходимости, в том числе и при

самой плохой (существенно овражной) структуре поверхностей уровня функций. Это подтверждено результатами численных экспериментов для существенно овражных выпуклых функций, как гладких, так и негладких [1].

Метод FEG2P1 вполне корректно заканчивает свою работу, каким бы не было множество X^* :

а) множество X^* может быть пустым ($X^* = \emptyset$). Этот случай реализован посредством получения достаточного условия, что в шаре радиусом r_0 с центром в стартовой точке x_0 нет ни одной точки из множества X^* . Здесь значение параметра ε не столь существенно, и его можно использовать сколь угодно малым, или даже равным нулю;

б) множество X^* может содержать единственную точку x^* ($X^* \equiv x^*$). В этом случае значительную роль имеет использование параметра ε , так как точно найти значение x^* можно только для простых функций $f(x)$. Для метода FEG2P1, как правило, можно использовать достаточно малые значения параметра ε (в том числе для существенно овражных функций $\varepsilon \sim 10^{-20} - 10^{-15}$), что дает возможность находить достаточно точные приближения к x^* (в том числе и в самых плохих случаях);

в) множество X^* может содержать бесконечное число точек ($X^* \equiv M$), где M – выпуклое множество, которое имеет непустую внутренность. В этом случае метод находит точку, в которой значение $f(x)$ равно нулю, и само значение параметра ε большой роли не играет. Как и в случае а), его можно указывать сколь-угодно близким к нулю или тождественно равным нулю.

Метод FEG2P1 на каждом шаге требует вычисления значения функции $f(x)$ и ее субградиента $g_f(x) = \partial f(x)$. В соответствии с вычисленными в текущей точке x_k значениями $f(x_k)$ и $g_f(x_k)$ осуществляется переход в следующую точку согласно правилу:

$$x_{k+1} = x_k - h_k B_k \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|}, \quad h_k = \gamma \frac{f(x_k)}{\|B_k^T g_f(x_k)\|}. \quad (4)$$

Здесь B_k – матрица обратного преобразования пространства переменных, γ – некоторый скалярный параметр ($\gamma \geq 1$), который позволяет настроить метод для различных классов выпуклых функций. Так, например, если $f(x)$ – выпуклая функция, то следует использовать $\gamma = 1$, а если $f(x)$ – выпуклая квадратичная функция, – $\gamma = 2$.

Вышеизложенное дает возможность использовать метод FEG2P1 для частных случаев нахождения допустимых точек выпуклого неравенства, к которым можно свести задачу нахождения решения систе-

мы линейных уравнений. Требуемые для метода вычисления значения функции и ее субградиента здесь можно сделать аналитически, что упрощает и сам метод в силу конкретизации вычисления шагового множителя h_k .

2. Пусть имеется система линейных уравнений

$$Ax = b, \quad (5)$$

где

A – заданная вещественная матрица размером $n \times m$,

$b \in R^m$ – заданный вектор правых частей,

$x \in R^n$ – вектор переменных, который требуется найти.

Множество решений системы (5) будем обозначать X^* . В частности, оно может быть и пустым ($X^* = \emptyset$), когда система (5) не имеет решений (несовместна). Когда $m = n$ и A – невырожденная матрица размером $n \times n$, тогда система (5) имеет единственное решение $x^* = A^{-1}b$, где A^{-1} – обратная к A матрица. В случае, если матрица A плохо обусловлена, то нахождение точки x^* , как правило, достаточно трудная задача.

Для удобства дальнейшего изложения систему линейных уравнений (5) будем рассматривать в следующем (тождественном) виде:

$$(a_i, x) = b_i, \quad i = 1, \dots, m, \quad (6)$$

где $a_i \in R^n$ – вектор-столбец, который получен транспонированием i -й строки матрицы A .

Рассмотрим три способа сведения задачи нахождения решения системы линейных уравнений (5) к задаче нахождения допустимой точки выпуклого неравенства:

1) найти допустимую точку следующего выпуклого неравенства:

$$f_1(x) = \|Ax - b\|^2 \leq 0. \quad (7)$$

Здесь $f_1(x)$ – гладкая выпуклая квадратичная функция и для нее можно использовать значение параметра $\gamma = \gamma_1 = 2$. Субградиент $g_{f_1}(x)$ функции $f_1(x)$ вычисляется по формуле

$$g_{f_1}(x) = 2A^T(Ax - b); \quad (8)$$

2) найти допустимую точку следующего выпуклого неравенства:

$$f_2(x) = \sum_{i=1}^m |(a_i, x) - b_i| \leq 0, \quad (9)$$

где $|\cdot|$ – модуль (абсолютная величина) числа. Функция $f_2(x)$ – негладкая выпуклая функция и для нее следует использовать значение параметра $\gamma = \gamma_2 = 1$. Субградиент $g_{f_2}(x)$ функции $f_2(x)$ вычисляется согласно следующей формулы:

$$g_{f_2}(x) = \sum_{i=1}^m \delta_i a_i, \quad \text{где} \quad \delta_i = \begin{cases} 1, & \text{если } (a_i, x) - b_i \geq 0; \\ -1, & \text{если } (a_i, x) - b_i < 0; \end{cases} \quad (10)$$

3) найти допустимую точку следующего выпуклого неравенства:

$$f_3(x) = \max_{i=1, \dots, m} |(a_i, x) - b_i| \leq 0, \quad (11)$$

где $f_3(x)$ – также негладкая выпуклая функция, для которой следует использовать $\gamma = \gamma_3 = 1$. Субградиент $g_{f_3}(x)$ функции $f_3(x)$ вычисляется по формуле

$$g_{f_3}(x) = \delta_{i^*} a_{i^*}, \quad \text{где} \quad \delta_{i^*} = \begin{cases} 1, & \text{если } (a_{i^*}, x) - b_{i^*} \geq 0; \\ -1, & \text{если } (a_{i^*}, x) - b_{i^*} < 0. \end{cases} \quad (12)$$

Здесь i^* – такой индекс из $i \in \{1, \dots, m\}$, при котором реализуется максимум модуля линейной функции $(a_i, x) - b_i$. Если такой максимум неоднозначен, то можно выбрать любой из тех индексов, при котором он достигается (например, при программировании удобно первый или последний).

Содержательный смысл указанных сведений достаточно прозрачен. Так, например, функция $f_1(x)$ задает сумму квадратов невязок системы уравнений (5). Нахождение ε -решения выпуклого неравенства (7) равносильно требованию, чтобы сумма квадратов невязок была равной нулю или достаточно близка к нулю. Эта близость определяется самим значением параметра ε . Аналогично, функция $f_2(x)$ задает сумму модулей невязок, а функция $f_3(x)$ – максимум из модуля невязок. Естественно, что использование ε будет иметь различный смысл для каждой из указанных функций.

Заметим, что задачу нахождения решения системы линейных уравнений можно свести к задаче нахождения допустимой точки выпуклого

неравенства и рядом других способов. Например, рассматривая в качестве функции линейную комбинацию с положительными коэффициентами квадратов невязок либо модулей невязок, или же функцию максимума от квадратов или модулей невязок, умноженных на положительные числа. Это позволяет построить и ряд других методов для решения системы линейных уравнений, которые могут иметь смысл, когда поразному требуется выдержать точность выполнения уравнений в (6)). В данной работе ограничимся только вышеуказанными способами.

Если $X^* \neq \emptyset$, то каждое из выпуклых неравенств ((7), (9) или (11)) имеет допустимую точку, которая совпадает с решением системы линейных уравнений (5), когда оно единственно, или одним из решений, когда система (5) имеет бесконечное число решений. Для нахождения ε -приближений к допустимым точкам этих неравенств можно использовать метод FEG2P1 с теми значениями параметра γ , которые даны выше. В силу того, какую из функций мы выберем, получим свой метод для нахождения решения системы линейных уравнений. Условимся эти методы называть по типу функции, т. е. методу(1) будет соответствовать выбор функции $f_1(x)$, методу(2) – функции $f_2(x)$ и методу(3) – функции $f_3(x)$.

Для реализации указанных методов необходимо выбрать начальную стартовую точку x_0 и радиус r_0 . При выборе r_0 следует учитывать тот факт, чтобы можно было гарантировать отсутствие решения системы линейных уравнений, когда система (5) несовместна. В качестве начальной стартовой точки выберем $x_0 = (0, \dots, 0)$, тогда радиусу r_0 можно придать содержательный смысл, т. е. будем отыскивать такое решение системы линейных уравнений, чтобы его норма была меньше, чем r_0 ($\|x^*\| \leq r_0$). В том случае, когда система (5) несовместна, получим достаточное условие, что решения с такой нормой нет.

3. Входными параметрами методов(1)–(3) будут: r_0 – радиус шара с центром в точке $(0, \dots, 0)$, в котором по предположению существует хотя бы одно из решений системы (5), если она совместна; ε ($\varepsilon > 0$) – достаточно малое число, которое задает точность, с которой будет отыскиваться такое решение.

Заметим, что здесь ε определяет точность, с которой будет находиться ε -решение выпуклого неравенства по функционалу и, следовательно, оно будет иметь различный смысл, в зависимости от вида функции. Так метод(1) будет останавливаться как только найдена точка в которой, сумма квадратов невязок меньше либо равна ε , метод(2) будет останавливаться, как только найдена точка, в которой сумма модулей невязок меньше либо равна ε , метод(3) будет останавливаться как только найде-

на точка, в которой максимальный модуль невязки меньше либо равен ε . Естественно, что при одном и том же ε метод(2) будет давать решение более точное, чем метод(3). Для того, чтобы для метода(1) обеспечить приблизительно такую же точность, какую дает метод(2) при ε_0 , для него следует использовать значение $\varepsilon = \varepsilon_0^2$.

При описании вычислительной схемы методов будем использовать следующее:

« $:=$ » – для присвоения значений векторам и матрицам;

« $=$ » – для присвоения значений скалярным переменным;

(\cdot, \cdot) – для скалярного произведения;

$(\cdot)^T$ – для транспонирования векторов и матриц;

$\|\cdot\|$ – для норм векторов.

Для обозначения векторов и матриц в ходе итерационного процесса использованы такие символы:

x – вектор из R^n (используется для текущего приближения ε -решения);

ξ, η, p – векторы из R^n (используются для нормированного субградиента в преобразованном пространстве, направления для преобразования пространства и нормированного агрегатного вектора);

B – матрица размером $n \times n$ (задает оператор обратного преобразования пространства переменных);

I_n – единичная матрица размером $n \times n$.

Пусть заданы i, r_0^2, ε . Здесь $i \in \{1, 2, 3\}$. Тогда вычислительная схема метода (i) принимает следующий вид.

Итерация 0. Установим $x_0 := 0$. Вычислим $f_i(x_0)$ и $g_{f_i}(x_0)$.

Если $f_i(x_0) \leq \varepsilon$, то ОСТАНОВ « x_0 – решение системы (5)».

Если $\gamma^2(f_i(x_0))^2 > r_0^2 \|g_{f_i}(x_0)\|^2$, то ОСТАНОВ «система (5) не имеет решения с нормой r_0 ». Иначе установим

$$h_0 = \frac{\gamma_i f_i(x_0)}{\|g_{f_i}(x_0)\|}, \quad \xi_0 := \frac{g_{f_i}(x_0)}{\|g_{f_i}(x_0)\|}, \quad p_0 := 0, \quad B_0 := I_n$$

и перейдем к очередной итерации.

(k+1)-я итерация. Пусть на k -й итерации получены $x_k \in R^n$, r_k^2 , h_k , $\xi_k \in R^n$, $p_k \in R^n$, B_k – матрица размером $n \times n$. Тогда выполним следующую последовательность операций.

Шаг 1. Вычислим $x_{k+1} := x_k - h_k B_k \xi_k$ и $r_{k+1}^2 = r_k^2 - h_k^2$.

Шаг 2. Вычислим $f_i(x_{k+1})$ и $g_{f_i}(x_{k+1})$.

Если $f_i(x_{k+1}) \leq \varepsilon$, то ОСТАНОВ « x_{k+1} – решение системы (5)».

Если $\gamma_i^2 f_i^2(x_{k+1}) > r_{k+1}^2 \|B_k^T g_{f_i}(x_{k+1})\|^2$, то ОСТАНОВ «система (5) не имеет решения с нормой r_0 ».

Иначе положим

$$\xi_{k+1} := \frac{B_k^T g_{f_i}(x_{k+1})}{\|B_k^T g_{f_i}(x_{k+1})\|}, \quad h = \frac{\gamma_i f_i(x_{k+1})}{\|B_k^T g_{f_i}(x_{k+1})\|}.$$

Шаг 3. Вычислим

$$\lambda_1 = -\frac{(p_k, \xi_{k+1})}{\sqrt{(p_k, \xi_{k+1})^2 + (\xi_k, \xi_{k+1})^2}}, \quad \lambda_2 = -\frac{(\xi_k, \xi_{k+1})}{\sqrt{(p_k, \xi_{k+1})^2 + (\xi_k, \xi_{k+1})^2}}$$

и положим

$$p := \begin{cases} \lambda_1 p_k + \lambda_2 \xi_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 > 0; \\ p_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 \leq 0; \\ \xi_k, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 > 0; \\ 0, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 \leq 0. \end{cases}$$

Шаг 4. Если $(p, \xi_{k+1}) \geq 0$, полагаем

$$B_{k+1} := B_k; \quad h_{k+1} = h; \quad p_{k+1} := 0$$

и переходим к шагу 5. Иначе вычисляем

$$\begin{aligned} \eta &:= \left(\frac{1}{\sqrt{1 - (p, \xi_{k+1})^2}} - 1 \right) \xi_{k+1} - \frac{(p, \xi_{k+1})}{\sqrt{1 - (p, \xi_{k+1})^2}} p, \\ B_{k+1} &:= B_k (I_n + \eta \xi_{k+1}^T), \quad h_{k+1} = \frac{h}{\sqrt{1 - (p, \xi_{k+1})^2}}, \\ p_{k+1} &:= \frac{1}{\sqrt{1 - (p, \xi_{k+1})^2}} (p - (p, \xi_{k+1}) \xi_{k+1}). \end{aligned}$$

Шаг 5. Переходим к очередной итерации с x_{k+1} , r_{k+1}^2 , h_{k+1} , ξ_{k+1} , p_{k+1} и B_{k+1} .

Сходимость метода (i) $i \in \{1, 2, 3\}$ следует из теоремы 2 [1].

4. Вычислительные эксперименты дадим для самого наихудшего случая, т. е. когда система линейных уравнений (5) имеет единственное решение и матрица A плохо обусловлена. В качестве таковой возьмем симметричную матрицу A размером $n \times n$ с коэффициентами $a_{ij} = 1/(i + j - 1)$, $i = 1, \dots, n$, $j = 1, \dots, n$ (такая матрица известна как матрица Гильберта). Степень обусловленности матрицы A $\left(\sigma(A) = \lambda_{\max}(A) / \lambda_{\min}(A)\right)$ очень быстро растет с ростом n . Так, например, $\sigma(A) = 1.60246928E + 13$ при $n = 10$, $\sigma(A) = 2.55784886E + 18$ при $n = 13$ (указанные значения найдены с помощью фортрановской подпрограммы EIGEN из [2]). Решение системы линейных уравнений (с матрицей Гильберта) прямыми методами (метод Гаусса, метод квадратных корней) при $n = 13$ проблематично, а при $n = 14$ и т. д. – невозможно.

Вышерассмотренные методы оказались достаточно устойчивыми по нахождению решений систем линейных уравнений с матрицей Гильберта при значениях n , для которых уже невозможно вычислить и степень обусловленности матрицы A . В таблице приведены результаты экспериментов для $n = 10, 15, \dots, 50$. Здесь в качестве вектора b выбирался вектор, равный Ae , где $e = (1, \dots, 1)^T$. При этом система (5) имеет единственное решение $x^* = e = (1, \dots, 1)^T$. Для каждого метода (i) ($i \in \{1, 2, 3\}$) при указанном значении ε в таблице даны:

$iter$ – число итераций, которые затрачены на поиск приближения к x^* ;

$\|x_{iter}^i - x^*\|$ – норма отклонения найденного решения от точного решения.

Из таблицы видно, что все три метода устойчивы по нахождению достаточно точного приближения к x^* . При этом следует заметить, что метод (1) работает в условиях даже худшей обусловленности, чем у матрицы Гильберта. Так как он имеет дело с квадратичной функцией, которая определяется матрицей $H = A^T A$, то речь идет об обусловленности $\sigma(H) = \sigma^2(A)$.

В заключение отметим, что на основе приведенных в статье методов можно разработать алгоритмические программы для решения системы линейных уравнений, которые будут численно устойчивыми как при решении «хороших», так и при решении плохо обусловленных систем. В пользу этого свидетельствует практический опыт их тестирования.

n	Метод(1), $\varepsilon = 10^{-24}$		Метод(2), $\varepsilon = 10^{-12}$		Метод(3), $\varepsilon = 10^{-12}$	
	iter	$\ x_{iter}^1 - x^*\ $	iter	$\ x_{iter}^2 - x^*\ $	iter	$\ x_{iter}^3 - x^*\ $
10	67	1.63117E-04	127	1.92725E-05	272	1.72554E-04
15	46	1.73133E-04	155	2.80517E-05	358	1.91924E-04
20	62	1.12086E-04	156	1.07970E-04	437	2.09268E-04
25	98	2.64809E-04	178	6.10850E-05	424	3.02368E-04
30	104	1.28632E-04	188	1.21749E-04	503	3.49949E-04
35	76	2.14721E-04	222	5.74886E-05	546	2.81749E-04
40	124	1.07358E-04	211	9.40017E-05	564	4.06881E-04
45	116	1.43121E-04	211	1.19156E-04	584	5.67897E-04
50	124	1.95499E-04	255	5.23825E-05	680	4.08678E-04

Литература

1. СТЕЦЮК П. И. Об одном методе для нахождения допустимой точки выпуклого неравенства // Теория оптимальных решений. – 2000. – С. 3–10.
2. СБОРНИК НАУЧНЫХ ПРОГРАММ НА ФОРТРАНЕ. Руководство программиста. Матричная алгебра и линейная алгебра. – М.: Статистика, 1974. – Вып. 2. – 223 с.

Р А З Д Е Л V

Одноранговый эллипсоидальный оператор в фейеровских процессах

В раздел включены статьи:

- 5.1. Субградиентные методы переменной метрики, использующие
шаг Агмона–Мощкина и одноранговый эллипсоидальный
оператор (*Стецюк, 2009*) 272
- 5.2. Метод `amsgr2p` для овражных выпуклых функций
(*Стецюк, 2011*) 282
- 5.3. Релаксационный субградиентный метод минимизации овражных . .
выпуклых функций (*Стецюк, 2011*) 285
- 5.4. Ускоренные по Шору модификации субградиентного метода
Поляка (*Стецюк, 2012*) 290
- 5.5. Ускорение субградиентного метода Поляка (*Стецюк, 2012*) 299
- 5.6. Shor’s acceleration for Polyak’s subgradient method (in Russian)¹
(*Stetsyuk, 2012*) 309
- 5.7. Ускоренные модификации субградиентного метода Поляка
для овражных выпуклых функций (*Стецюк, 2012*) 313
- 5.8. Ускоренный субградиентный метод Поляка
(*Стецюк, Кошлай, 2012*) 337
- 5.9. Ускоренные по Шору модификации метода Поляка для овражных
функций (*Стецюк, 2013*) 340

¹Ускоренный по Шору субградиентный метод Поляка

Субградиентные методы переменной метрики, использующие шаг Агмона–Моцкина и одноранговый эллипсоидальный оператор¹

П. И. Стецюк

Труды АТИК 2007–2008. – Кишинэу, Эврика 2009. – С. 16–25.

В работе приведены два субградиентных метода переменной метрики для нахождения точки минимума выпуклой функции при известном ее оптимальном значении. Оба метода используют регулировку шага, известную как шаг Агмона–Моцкина, и преобразуют пространство переменных с помощью однорангового эллипсоидального оператора. Первый метод преобразует пространство по двум последовательным субградиентам, если угол между ними тупой. Второй метод использует еще и агрегатный вектор, который является выпуклой комбинацией вычисленных ранее субградиентов. В том случае, когда с помощью агрегатного вектора нельзя построить тупой угол, этот вектор автоматически обновляется. Для второго метода даны результаты тестовых экспериментов в сравнении с r -алгоритмом Шора–Журбенко.

Постановка задачи и субградиентный метод с шагом Агмона–Моцкина. Рассмотрим задачу

$$x^* = \arg \min_{x \in R^n} f(x), \quad (1)$$

где $f(x)$ – выпуклая функция векторного аргумента $x \in R^n$, а ее субградиент $\partial f(x)$ удовлетворяет следующему свойству:

$$(x - x^*, \partial f(x)) \geq m(f(x) - f^*), \quad \text{если } m \geq 1 \quad (2)$$

для $\forall x \in R^n$ и для $\forall x^* \in X^*$. Здесь R^n – евклидово пространство размерности n со скалярным произведением (x, y) ; X^* – множество минимумов функции $f(x)$; f^* – значение минимума $f(x)$: $f^* = f(x^*)$,

¹Работа выполнена при частичной финансовой поддержке совместного украинско- российского проекта ДФФД-Ф28.1/005 и РФФИ-09-01-90413-Укр_ф_а «Субградиентные методы ускоренной сходимости в задачах выпуклой оптимизации».

$x^* \in X^*$. Отметим, что произвольная выпуклая функция удовлетворяет (2) с параметром $m = 1$, но есть классы выпуклых функций, для которых параметр m больше единицы. Так, например, квадратичная выпуклая функция удовлетворяет (2) с параметром $m = 2$.

Рассмотрим задачу (1), когда известно f^* . Не ограничивая общности будем считать, что множество X^* состоит из единственной точки x^* . Для нахождения можно использовать субградиентный метод [1, 2]

$$x_{k+1} = x_k - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|}, \quad h_k = \frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|}, \quad k = 0, 1, 2, \dots \quad (3)$$

Метод (3) использует регулировку шага в направлении нормированного субградиента, известную как шаг Агмона-Моцкина. Впервые такой шаг был использован в [3, 4] в релаксационном методе для нахождения хотя бы одного решения совместной системы линейных неравенств. Метод (3) еще называют фейеровским.

Обоснование сходимости метода (3) очень простое, так как для всех точек итерационного процесса справедливы неравенства

$$\|x_{k+1} - x^*\|^2 \leq \|x_k - x^*\|^2 - \left(\frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|} \right)^2, \quad k = 0, 1, \dots$$

Отсюда для каждой итерации $K > 1$ имеем неравенство

$$\|x_K - x^*\|^2 \leq \|x_0 - x^*\|^2 - \sum_{k=0}^{K-1} \left(\frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|} \right)^2,$$

из которого, используя расходимость ряда, легко придти к противоречивому утверждению, что $\|x_{\overline{K}} - x^*\|^2 < 0$ для некоторого \overline{K} .

Метод (3) пользуется плохой репутацией на практике и причиной этому являются овражные выпуклые функции. Для них характерно «зигзаго-подобное» поведение метода и оно объясняется тем, что направление антисубградиента в точках «оврага» достаточно близко к направлению, ортогональному направлению на точку минимума. Так, например, для негладкой функции двух переменных $f(x_1, x_2) = |x_1| + k|x_2|$ скорость сходимости метода (3) определяется геометрической прогрессией со знаменателем $\sqrt{1 - 1/k^2}$, который близок к 1 даже при сравнительно небольших значениях k .

Медленную сходимость метода (3) для овражных функций определяет тупой угол между двумя последовательными субградиентами. Чем ближе этот угол к 180 градусам, тем более медленной будет сходимость

метода (3). Понятно, что если пространство переменных X преобразовать так, чтобы тупой угол между двумя последовательными субградиентами уменьшался, то в преобразованном пространстве переменных мы «расширим» конус подходящих направлений убывания $f(x)$. И тем самым для овражных функций можно ускорить сходимость субградиентного процесса с шагом Агмона-Моцкина в преобразованном пространстве переменных.

Субградиентный метод переменной метрики, использующий шаг Агмона-Моцкина и одноранговый эллипсоидальный оператор для двух последовательных субградиентов.

Тупой угол между двумя нормированными векторами ξ и η из R^n можно преобразовать в прямой с помощью линейного оператора из R^n в R^n , который в матричной форме представим

$$T_1(\xi, \eta) = I - \frac{1}{1 - (\xi, \eta)^2} \left((1 - \sqrt{1 - (\xi, \eta)^2}) \eta - (\xi, \eta) \xi \right) \eta^T. \quad (4)$$

Здесь $\xi, \eta \in R^n$ – векторы, такие, что $\|\xi\| = 1$, $\|\eta\| = 1$ и их скалярное произведение удовлетворяет условию $(\xi, \eta)^2 \neq 1$, I – единичная матрица размера $n \times n$. Для оператора $T_1(\xi, \eta)$ существует обратный $T_1^{-1}(\xi, \eta)$

$$T_1^{-1}(\xi, \eta) = I + \frac{1}{\sqrt{1 - (\xi, \eta)^2}} \left((1 - \sqrt{1 - (\xi, \eta)^2}) \eta - (\xi, \eta) \xi \right) \eta^T. \quad (5)$$

Оператор $T_1(\xi, \eta)$ введен в [5] под названием «одноранговый эллипсоидальный оператор». Он связан с преобразованием в шар специального эллипсоида, описанного вокруг тела W (см. рис. 1), которое получено в результате пересечения шара и двух полупространств, проходящих через центр шара. В случае тупого угла между нормальными полупространств этот эллипсоид содержит тело W и является минимальным по объему в рамках семейства эллипсоидов, центр которых совпадает с центром шара (см. рис. 2). Объем эллипсоида меньше, чем объем шара, и это уменьшение определяется величиной $\sqrt{1 - (\xi, \eta)^2}$. Преобразование эллипсоида в шар требует растяжения пространства в направлении $\frac{\xi - \eta}{\|\xi - \eta\|}$ с коэффициентом $\alpha_1 = \frac{1}{\sqrt{1 + (\xi, \eta)}}$ и «сжатия»,

пространства в ортогональном направлении $\frac{\xi + \eta}{\|\xi + \eta\|}$ с коэффициентом

$$\alpha_2 = \frac{1}{\sqrt{1 - (\xi, \eta)}}. \quad \text{В преобразованном пространстве эллипсоид станет шаром, а образы векторов } \xi \text{ и } \eta \text{ будут ортогональными (см. рис. 3).}$$

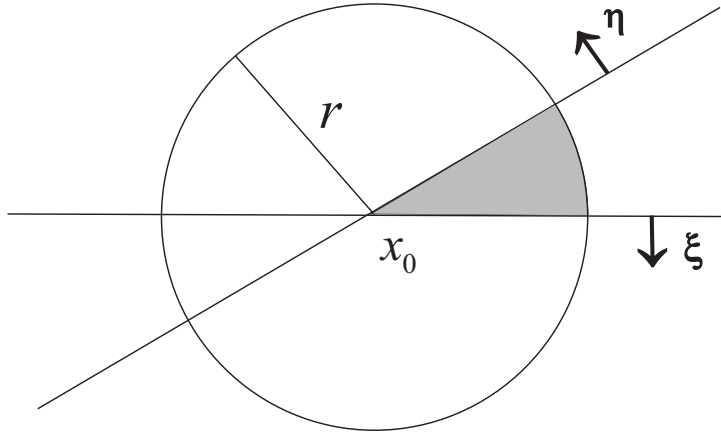


Рис. 1.

С помощью оператора (4) легко построить субградиентный метод, который использует шаг Агмона-Мозкина в преобразованном пространстве переменных, если преобразование пространства делать только тогда, когда два последовательных субградиента образуют тупой угол.

Такой метод имеет следующий вид:

$$x_{k+1} = x_k - h_k B_k \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}, \quad h_k = \frac{m(f(x_k) - f^*)}{\|B_k^T \partial f(x_k)\|},$$

$$k = 0, 1, 2, \dots, \quad (6)$$

где матрица $B_0 = I$, а матрица B_{k+1} пересчитывается с учетом векторов $\xi = \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}$ и $\eta = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|}$ по следующему правилу: если $(\xi, \eta) \geq 0$, то $B_{k+1} = B_k$, а если $(\xi, \eta) < 0$, то $B_{k+1} = B_k T^{-1}(\xi, \eta)$.

Для каждой итерации $K > 1$ метода (6) справедливо

$$\|B_K^{-1}(x_K - x^*)\|^2 \leq \|x_0 - x^*\|^2 - \sum_{k=0}^{K-1} \left(\frac{m(f(x_k) - f^*)}{\|B_k \partial f(x_k)\|} \right)^2.$$

Если ограничить евклидову норму матрицы B_k , то обоснование сходимости метода (6) аналогично обоснованию метода (3). Учитывая, что

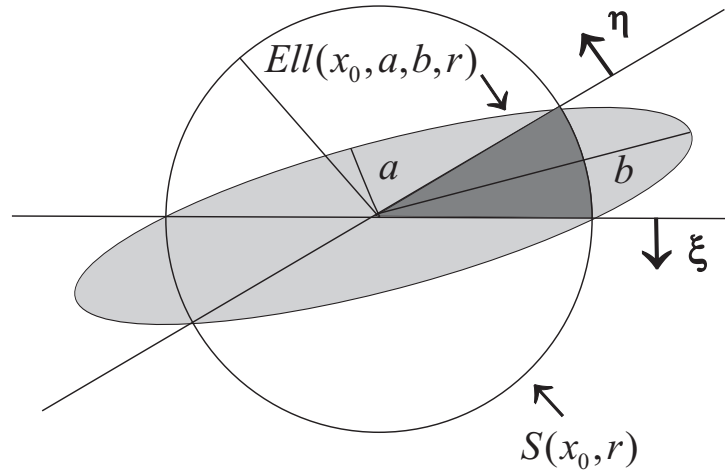


Рис. 2.

при пересчетах матрицы $\det(B_k)$ стремится к нулю, то расходимость ряда будет более быстрой, чем в методе (3). Для овражных функций это обеспечивает ускоренную сходимость в сравнении с методом (3). Так, например, для овражной функции двух переменных $f(x_1, x_2) = |x_1| + k|x_2|$ метод будет находить точку x^* за две итерации независимо от значения k .

Отметим, что метод (6) обладает двумя характерными чертами r -алгоритмов [6, 7]. Во-первых, движение из точки осуществляется в направлении антисубградиента и, во-вторых, растяжение пространства производится в направлении разности нормированных последовательных субградиентов, если угол между ними тупой. Если нормы субградиентов одинаковы, то это направление будет совпадать с разностью двух последовательных субградиентов, но в отличие от r -алгоритмов второй субградиент получен не согласно шагу наискорейшего спуска, а согласно шагу Агмона-Моцкина.

Субградиентный метод переменной метрики, использующий шаг Агмона-Моцкина и одноранговый эллипсоидальный оператор для двух последовательных субградиентов и агрегатного вектора.

Метод (6) можно усилить более радикальным уменьшением объема эллипсоида, локализирующего точку x^* . Если на k -итерации была реали-

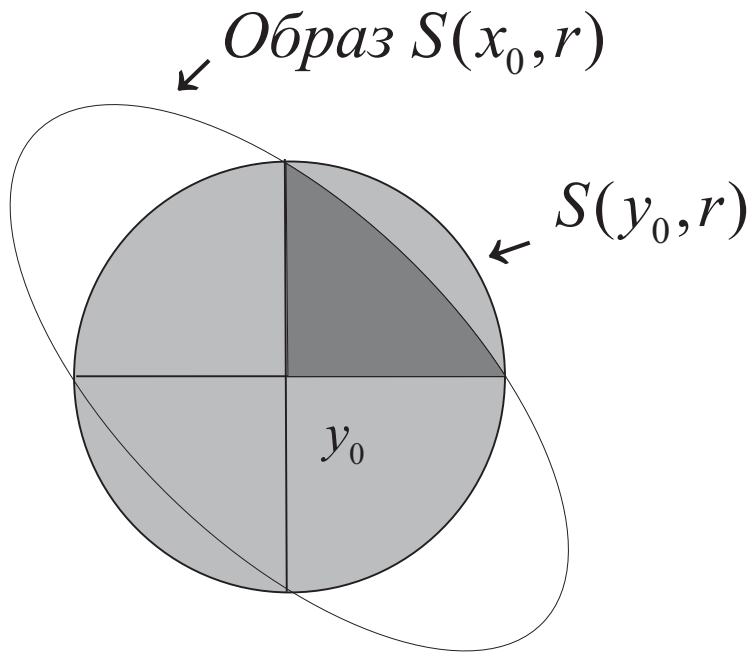


Рис. 3.

зована операция преобразования пространства, то в преобразованном пространстве $Y = A_{k+1}X = B_{k+1}^{-1}X$ для функции $\varphi_{k+1}(y) = f(B_{k+1}y)$ субградиенты будут ортогональны и в силу шага Агмона-Мощкина полупространства, определяемые ими в точке $y_{k+2} = A_{k+1}x_{k+2}$, не отсекают точку $y^* = A_{k+1}x^*$. Очевидно, что, если вычисленный в точке y_{k+2} субградиент образует тупой угол с двумя предыдущими, то можно уменьшить объем области локализации x^* . Для этого достаточно выбрать в качестве векторов, определяющих оператор $T_1(\xi, \eta)$, субградиент в точке y_{k+2} и вектор, являющийся выпуклой комбинацией первых двух, так, чтобы угол между ними был максимально тупым. Для очередной итерации (если реализуется преобразование пространства) эта ситуация повторяется, только в качестве первого вектора уже будет использоваться не субградиент, а вектор, являющийся выпуклой комбинацией первых двух предыдущих субградиентов в очередном преобразованном пространстве. Это позволяет замкнуть цикл вычислений и построить конструктивное правило использования в качестве одного из

векторов, определяющих оператор $T_1(\xi, \eta)$, агрегированного вектора.

Метод, построенный на этом принципе, условимся называть $feg2p1(x_0, \varepsilon_f)$, где f – вклад от слова «фейеровский», e – от «эллипсоид», а $g2p1$ означает использование двух последовательных субградиентов (от символа « g ») и одного вектора агрегатного типа, который далее будем обозначать символом p . Входными параметрами метода будут начальная стартовая точка x_0 и точность ε_f , с которой будем искать ε_f -решение задачи (1). Таким решением будем считать точку, которая принадлежит множеству $X_{\varepsilon_f}^* = \{x : f(x) - f^* \leq \varepsilon_f(|f^*| + 1)\}$. Содержательно ε_f -решение означает такое приближение к x^* , для которого относительная погрешность по функции не превосходит ε_f . Здесь к $|f^*|$ прибавлена единица, чтобы корректным было вычисление относительной погрешности при $f^* = 0$.

Вычислительная схема метода $feg2p1(x_0, \varepsilon_f)$ имеет следующий вид.

Перед началом вычислений заданы $\varepsilon_f > 0$, $x_0 \in R^n$. Если $f(x_0) - f^* \leq \varepsilon_f(1 + |f^*|)$, то x_0 – искомая точка и ОСТАНОВ. Иначе полагаем $h_0 = m(f(x_0) - f^*) \|\partial f(x_0)\|$, $\xi_0 = \partial f(x_0) \|\partial f(x_0)\| \in R^n$, $p_0 = 0 \in R^n$, $B_0 = I_n$ – единичная матрица размера $n \times n$.

Пусть на k -ой итерации получены $x_k \in R^n$, $h_k, \xi_k \in R^n$, $p_k \in R^n$, B_k – матрица $n \times n$.

1) Вычислим очередное приближение

$$x_{k+1} = x_k - h_k B_k \xi_k.$$

2) Вычислим $f(x_{k+1})$, $\partial f(x_{k+1})$. Если $f(x_{k+1}) - f^* \leq \varepsilon_f(1 + |f^*|)$, то x_{k+1} – искомая точка и ОСТАНОВ. Иначе вычислим

$$\xi_{k+1} = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|}, \quad h_{k+1} = \frac{m(f(x_{k+1}) - f^*)}{\|B_k^T \partial f(x_{k+1})\|}.$$

3) Вычислим

$$\lambda_1 = -\frac{(p_k, \xi_{k+1})}{\sqrt{(p_k, \xi_{k+1})^2 + (\xi_k, \xi_{k+1})^2}}, \quad \lambda_2 = -\frac{(\xi_k, \xi_{k+1})}{\sqrt{(p_k, \xi_{k+1})^2 + (\xi_k, \xi_{k+1})^2}}$$

и

$$p_{k+1} = \begin{cases} \lambda_1 p_k + \lambda_2 \xi_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 > 0, \\ p_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 \leq 0, \\ \xi_k, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 > 0, \\ 0, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 \leq 0. \end{cases}$$

- 4) Если $(p_{k+1}, \xi_{k+1}) \geq 0$, то полагаем $B_{k+1} = B_k$ и переходим к 5. Иначе вычисляем

$$\eta = \left(\frac{1}{\sqrt{1 - (p_{k+1}, \xi_{k+1})^2}} - 1 \right) \xi_{k+1} - \frac{(p_{k+1}, \xi_{k+1})}{\sqrt{1 - (p_{k+1}, \xi_{k+1})^2}} p_{k+1},$$

$$B_{k+1} = B_k (I + \eta \xi_{k+1}^T), \quad h_{k+1} = \frac{h_{k+1}}{\sqrt{1 - (p_{k+1}, \xi_{k+1})^2}},$$

$$p_{k+1} = \frac{1}{\sqrt{1 - (p_{k+1}, \xi_{k+1})^2}} (p_{k+1} - (p_{k+1}, \xi_{k+1}) \xi_{k+1}).$$

- 5) Переходим к очередной итерации с x_{k+1} , B_{k+1} , ξ_{k+1} , h_{k+1} , p_{k+1} .

Справедлива следующая теорема.

Теорема 1. Последовательность $\{x_{k+1}\}_{k=0}^{\infty}$, генерируемая методом $feq2p1(x_0, \varepsilon_f)$, удовлетворяет неравенству

$$\|A_{k+1}(x_{k+1} - x^*)\|^2 \leq \|A_k(x_k - x^*)\|^2 - \frac{m^2(f(x_k) - f^*)^2}{\|B_k^T \partial f(x_k)\|^2}, \quad k = 0, 1, 2, \dots$$

Здесь $A_k = B_k^{-1}$, $A_{k+1} = B_{k+1}^{-1}$.

Эта теорема обеспечивает обоснование сходимости метода $feq2p1(x_0, \varepsilon_f)$ аналогично обоснованию для метода (6). Но более сильное уменьшение объема эллипсоида, локализующего точку x^* , обеспечивает для него ускоренную сходимость по сравнению с методом (6). По количеству итераций метод $feq2p1(x_0, \varepsilon_f)$ сравним с r -алгоритмом, а в ряде случаев и превосходит его. Это подтверждают результаты тестовых экспериментов на рис. 4. Рассматривались 7 известных тестовых задач безусловной минимизации гладких и негладких выпуклых функций [8] (стр. 279-282) и кусочно-квадратичная функция [7] (стр. 176). Количество переменных в тестовых примерах было от 5 до 50. Все примеры решались r -алгоритмом с критериями останова $\varepsilon_x = 10^{-6}$ и $\varepsilon_g = 10^{-6}$ и методом $feq2p1(x_0, \varepsilon_f)$ при достаточно малых $\varepsilon_f = 10^{-8}$ и $\varepsilon_f = 10^{-12}$. Число затраченных методами итераций дано на рис. 4. Из него видно, что только в одном случае метод $feq2p1(x_0, \varepsilon_f)$ уступил r -алгоритму (пример N2).

В заключении отметим, что ускоренные варианты субградиентных методов на основе однорангового эллипсоидального оператора можно построить и для других способов регулировки шага. Замечательной чертой таких методов есть полная определенность в параметрах преобразования пространства.

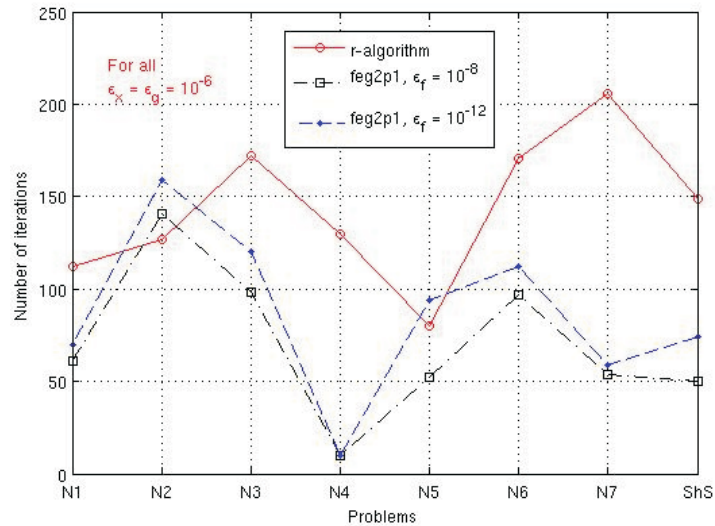


Рис. 4.

Литература

1. ЕРЕМИН А. И. И. Релаксационный метод решения систем неравенств с выпуклыми функциями// Докл. АН СССР. – 1965. – Т.160, – № 5. – С. 994-996.
2. ПОЛЯК Б. Т. Оптимизация негладких функционалов// Журн. вычислит. математики и матем. физики. – 1969, – Т. 9, – № 3., – С. 507-521.
3. AGMON S. The relaxation method for liner inequalities// Canadien Journal of Mathematics. – 1954. – 6. – P. 382-392.
4. MOTZKIN T., SCHOENBERG I. J. The relaxation method for liner inequalities// Canadien Journal of Mathematics. – 1954. – 6. – P. 393-404.
5. СТЕЦЮК П. И. Ортогонализующие линейные операторы в выпуклом программировании (Часть I)// Кибернетика. – 1971. – № 3. – С. 51-59.

6. ШОР Н. З., ЖУРБЕНКО Н. Г. Метод минимизации, использующий операцию растяжения пространства в направлении разности двух последовательных градиентов //Кибернетика. – 1971. – № 3. – С. 51-59.
7. ШОР Н. З. Методы минимизации недифференцируемых функций и их приложения //Киев: Наукова думка, 1979. – 199 с.
8. РЖЕВСКИЙ С. В. Монотонные методы выпуклого программирования//Киев: Наукова думка, 1993. – 324 с.

Метод `amsg2p` для овражных выпуклых функций

П. И. Стецюк

*Информационный бюллетень Ассоциации математического
программирования. №12. Научное издание. – Екатеринбург: УрО
РАН, 2011. – С. 57–58.*

Обсуждается субградиентный метод с преобразованием пространства для нахождения точки минимума выпуклой функции при известном минимальном значении функции.

Пусть $f(x)$ – выпуклая функция векторного аргумента $x \in R^n$ и ее субградиент $\partial f(x)$ удовлетворяет условию:

$$(x - x^*, \partial f(x)) \geq \gamma(f(x) - f^*), \quad \text{где } \gamma \geq 1, \quad (1)$$

для $\forall x \in R^n$ и $\forall x^* \in X^*$. Здесь R^n – евклидово пространство размерности n со скалярным произведением $x^T y$; X^* – множество точек минимума функции $f(x)$; f^* – минимальное значение функции $f(x)$: $f^* = f(x^*)$, $x^* \in X^*$. Параметр γ задает величину максимального сдвига по выпуклости функции $f(x)$. Для кусочно-линейной негладкой функции $\gamma=1$, для квадратичной гладкой функции $\gamma=2$.

В сообщении обсудим субградиентный метод с преобразованием пространства¹ для нахождения приближения к x^* при известном f^* , который назовем методом `amsg2p`. Здесь „ams“ указывает на способ регуляции шага в субградиентном методе

$$x_{k+1} = x_k - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|}, \quad h_k = \frac{\gamma(f(x_k) - f^*)}{\|\partial f(x_k)\|}, \quad k = 0, 1, 2, \dots \quad (2)$$

который известен как метод Б.Т.Поляка (1969). AMS-шаг h_k называют еще шагом Поляка или шагом Агмона-Моткина. Впервые его использовали S.Agmon и T.S.Motzkin, I.I.Schoenberg в 1954 году в релаксационном методе для нахождения хотя бы одного из решений совместной системы линейных неравенств. И.И.Еремин (1965) обобщил этот метод для систем выпуклых неравенств.

¹ *Стецюк П.И.* Субградиентные методы переменной метрики, использующие шаг Агмона-Моткина и одноранговый эллипсоидальный оператор // Труды АТИК - 2007–2008, Кишинэу, Эврика 2009, том. I (XII), С. 16–25.

Вторая часть в названии метода – „g2p“ указывает на то, что AMS-шаг используется в пространстве переменных, преобразованном с помощью двух последних субградиентов (g2) и агрегатного вектора (p). Для овражных функций метод amsg2p обладает ускоренной сходимостью в сравнении с методом (2), благодаря антиовражной технике, подобно тому, как это сделано в r -алгоритмах Н.З.Шора. При этом обеспечивается уменьшение расстояния до множества минимумов в преобразованном пространстве переменных.

Заданы: начальное приближение $x_0 \in R^n$; достаточно малое $\varepsilon > 0$; f^* – минимальное значение $f(x)$; γ – параметр в условии (1) для $f(x)$ и $\partial f(x)$. Метод amsg2p находит точку $x_\varepsilon^* \in X_\varepsilon^* = \{x : f(x) - f^* \leq \varepsilon\}$ и соответствующий ей номер итерации k_ε^* . Метод реализует

$$\text{Алгоритм-функция amsg2p: } (x_\varepsilon^*, k_\varepsilon^*) = \text{amsg2p}(x_0, \varepsilon, f^*, \gamma)$$

На итерации $k=0$ имеем $x_0 \in R^n$. Вычислим $f(x_0)$ и $\partial f(x_0)$. Если $f(x_0) - f^* \leq \varepsilon$, то $x_\varepsilon^* = x_0$, $k_\varepsilon^* = 0$ и окончание работы алгоритма. Иначе положим $h_0 = \frac{\gamma(f(x_0) - f^*)}{\|\partial f(x_0)\|}$, $\xi_0 = \frac{\partial f(x_0)}{\|\partial f(x_0)\|} \in R^n$, $p_0 = 0 \in R^n$, $B_0 = I_n$ – единичная матрица размера $n \times n$.

Пусть на k -й итерации получены $x_k \in R^n$, h_k , $\xi_k \in R^n$, $p_k \in R^n$, B_k – матрица $n \times n$. Для $(k+1)$ -й итерации выполним пп. 1–5.

1) Вычислим очередное приближение

$$x_{k+1} = x_k - h_k B_k \xi_k.$$

2) Вычислим $f(x_{k+1})$ и $\partial f(x_{k+1})$. Если $f(x_{k+1}) - f^* \leq \varepsilon$, то $x_\varepsilon^* = x_{k+1}$, $k_\varepsilon^* = k+1$ и окончание алгоритма. Иначе положим

$$\xi_{k+1} = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|}, \quad h_{k+1} = \frac{\gamma(f(x_{k+1}) - f^*)}{\|B_k^T \partial f(x_{k+1})\|}.$$

3) Вычислим $\lambda_1 = -p_k^T \xi_{k+1}$ и $\lambda_2 = -\xi_k^T \xi_{k+1}$. Положим

$$p_{k+1} = \begin{cases} \frac{\lambda_1}{\sqrt{\lambda_1^2 + \lambda_2^2}} p_k + \frac{\lambda_2}{\sqrt{\lambda_1^2 + \lambda_2^2}} \xi_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 > 0, \\ p_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 \leq 0, \\ \xi_k, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 > 0, \\ 0, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 \leq 0. \end{cases}$$

- 4) Вычислим $\mu_k = p_{k+1}^T \xi_{k+1}$. Если $-1 < \mu_k < 0$, то пересчитаем

$$h_{k+1} = \frac{h_{k+1}}{\sqrt{1 - \mu_k^2}}, \quad p_{k+1} = \frac{1}{\sqrt{1 - \mu_k^2}}(p_{k+1} - \mu_k \xi_{k+1})$$

и вычислим

$$B_{k+1} = B_k + (B_k \eta) \xi_{k+1}^T, \text{ где } \eta = \left(\frac{1}{\sqrt{1 - \mu_k^2}} - 1 \right) \xi_{k+1} - \frac{\mu_k}{\sqrt{1 - \mu_k^2}} p_{k+1}.$$

Иначе положим $B_{k+1} = B_k$ и $p_{k+1} = 0$.

- 5) Перейдем к следующей итерации с x_{k+1} , h_{k+1} , ξ_{k+1} , p_{k+1} , B_{k+1} .

Релаксационный субградиентный метод минимизации овражных выпуклых функций

П. И. Стецюк

Проблемы теоретической кибернетики. Материалы XVI Международной конференции (Нижний Новгород, 20–25 июня 2011 г.) / Под ред. Ю.И. Журавлева. Нижний Новгород: Изд-во Нижегородского государственного университета, 2011. – С. 449–453.

Предложен субградиентный метод с преобразованием пространства для уменьшения степени вытянутости овражных функций в преобразованном пространстве переменных. Метод использует релаксационный шаг Агмона-Моцкина-Шонберга с последовательным уточнением границ на минимальное значение функции. Приведены результаты тестовых экспериментов.

Рассмотрим задачу

$$\text{найти } x^* = \arg \min_{x \in R^n} f(x), \quad (1)$$

где $f(x)$ – выпуклая функция и ее субградиент $\partial f(x)$ для $\forall x \in R^n$ и $\forall x^* \in X^*$ удовлетворяет следующему неравенству:

$$(x - x^*, \partial f(x)) \geq \gamma(f(x) - f^*), \quad \text{где } \gamma \geq 1. \quad (2)$$

Здесь X^* – ограниченное множество точек минимума функции $f(x)$, f^* – минимальное значение функции $f(x)$: $f^* = f(x^*)$, $x^* \in X^*$. Параметр γ определяет величину максимального сдвига по выпуклости функции $f(x)$ в неравенстве (2) и может быть использован для специальных классов выпуклых функций, для которых $\gamma > 1$. Так, например, градиент выпуклой квадратичной функции удовлетворяет неравенству (2) с параметром $\gamma = 2$.

Если f^* известно, то для решения задачи (1)–(2) можно использовать метод `amsg2p` [1]. В его основу положен второй из субградиентных методов с преобразованием пространства и регулировкой шага Агмона-Моцкина-Шонберга [2]. Преобразование пространства в методе `amsg2p` реализуется с помощью однорангового эллипсоидального оператора [3] и только на тех итерациях метода, когда тупым является хотя бы один

из углов – угол между двумя последовательными субградиентами, либо угол между последним субградиентом и агрегатным вектором, который есть выпуклой комбинацией вычисленных на предыдущих итерациях субградиентов. Отсюда и название метода, где „ams“ указывает на способ регулировки шага в направлении нормированного антисубградиента, а „g2p“ указывает, что аms-шаг используется в пространстве переменных, преобразованном с помощью двух последних субградиентов (g2) и агрегатного вектора (p).

Если в задаче (1) структура поверхностей уровня функции $f(x)$ такова, что преобразования пространства не нужны, то метод **amsg2p** равносителен известному субградиентному методу Б.Т. Поляка [4]

$$x_{k+1} = x_k - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|}, \quad h_k = \frac{\gamma(f(x_k) - f^*)}{\|\partial f(x_k)\|}, \quad k = 0, 1, 2, \dots, \quad (3)$$

с тем лишь отличием, что неравенство (2) для специальных классов функций реализует более сильные аms-шаги h_k , чем это делается для выпуклой функции при $\gamma = 1$. Отметим, что аms-шаг h_k еще называют шагом Поляка, или шагом Агмона-Моцкина.

В настоящей заметке дополним метод **amsg2p** возможностью либо находить такую точку, где выпуклая функция $f(x)$ принимает значение, равное f_{min} , либо гарантировать достаточное условие, что такой точки не существует.

Метод **amsg2p** находит точку $x_\varepsilon^* \in \{x : f(x) - f_{min} \leq \varepsilon\}$ и соответствующий ей неотрицательный номер итерации k_ε^* . Отрицательный номер k_ε^* означает, что на итерации $|k_\varepsilon^*|$ получено достаточное условие отсутствия точки x_ε^* .

Алгоритм-функция **amsg2p**: $(x_\varepsilon^*, k_\varepsilon^*) = \text{amsg2p}(x_0, \varepsilon, r_0, f_{min}, \gamma)$

На итерации $k = 0$ заданы – начальное приближение $x_0 \in R^n$; начальный радиус r_0 такой, что $\|x_0 - x^*\| \leq r_0$ для $\forall x^* \in X^*$; достаточно малое $\varepsilon > 0$. Вычислим $f(x_0)$ и $\partial f(x_0)$. Если $f(x_0) - f_{min} \leq \varepsilon$, то $x_\varepsilon^* = x_0$, $k_\varepsilon^* = 0$ и окончание алгоритма. Иначе положим $h_0 = \frac{\gamma(f(x_0) - f_{min})}{\|\partial f(x_0)\|}$,

$\xi_0 = \frac{\partial f(x_0)}{\|\partial f(x_0)\|} \in R^n$, $p_0 = 0 \in R^n$, $B_0 = I$ – единичная $n \times n$ -матрица.

Перейдем к следующей итерации.

Пусть на k -й итерации получены $x_k \in R^n$, h_k , r_k , $\xi_k \in R^n$, $p_k \in R^n$, B_k – матрица $n \times n$. Для $(k + 1)$ -й итерации выполним пп. 1–5.

- 1) Вычислим $t_k = h_k/r_k$. Если $t_k > 1$, то $k_\varepsilon^* = -(k+1)$ и окончание алгоритма. Иначе положим $r_{k+1} = r_k \sqrt{1-t_k^2}$ и вычислим очередное приближение

$$x_{k+1} = x_k - h_k B_k \xi_k.$$

- 2) Вычислим $f(x_{k+1})$ и $\partial f(x_{k+1})$. Если $f(x_{k+1}) - f_{min} \leq \varepsilon$, то $x_\varepsilon^* = x_{k+1}$, $k_\varepsilon^* = k+1$ и окончание алгоритма. Иначе положим

$$\xi_{k+1} = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|}, \quad h_{k+1} = \frac{\gamma(f(x_{k+1}) - f_{min})}{\|B_k^T \partial f(x_{k+1})\|}.$$

- 3) Вычислим $\lambda_1 = -p_k^T \xi_{k+1}$ и $\lambda_2 = -\xi_k^T \xi_{k+1}$. Положим

$$p_{k+1} = \begin{cases} \frac{\lambda_1}{\sqrt{\lambda_1^2 + \lambda_2^2}} p_k + \frac{\lambda_2}{\sqrt{\lambda_1^2 + \lambda_2^2}} \xi_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 > 0, \\ p_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 \leq 0, \\ \xi_k, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 > 0, \\ 0, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 \leq 0. \end{cases}$$

- 4) Вычислим $\mu_k = p_{k+1}^T \xi_{k+1}$. Если $-1 < \mu_k < 0$, то вычислим

$$B_{k+1} = B_k + (B_k \eta) \xi_{k+1}^T, \quad \text{где } \eta = \left(\frac{1}{\sqrt{1-\mu_k^2}} - 1 \right) \xi_{k+1} - \frac{\mu_k}{\sqrt{1-\mu_k^2}} p_{k+1},$$

и пересчитаем

$$h_{k+1} = \frac{h_{k+1}}{\sqrt{1-\mu_k^2}}, \quad p_{k+1} = \frac{1}{\sqrt{1-\mu_k^2}} (p_{k+1} - \mu_k \xi_{k+1}).$$

Иначе положим $B_{k+1} = B_k$ и $p_{k+1} = 0$.

- 5) Перейдем к следующей итерации с x_{k+1} , h_{k+1} , r_{k+1} , ξ_{k+1} , p_{k+1} , B_{k+1} .

Теорема. Если $f_{min} \geq f^*$, то для $\forall x^* \in X^*$ справедливы неравенства

$$\|A_{k+1}(x_{k+1} - x^*)\|^2 \leq \|A_k(x_k - x^*)\|^2 - \left(\frac{\gamma(f(x_k) - f_{min})}{\|B_k^T \partial f(x_k)\|} \right)^2, \quad k = 0, 1, \dots,$$

Здесь $A_k = B_k^{-1}$, $A_{k+1} = B_{k+1}^{-1}$.

Теорема 1 означает, что в методе `amsg2p` преобразование пространства таково, что в каждом очередном преобразованном пространстве переменных гарантируется уменьшение расстояния до множества точек минимума. Благодаря этому для каждой итерации $k > 1$ имеем неравенство

$$\|A_k(x_k - x^*)\|^2 \leq \|x_0 - x^*\|^2 - \sum_{i=0}^{k-1} \left(\frac{\gamma(f(x_i) - f_{min})}{\|B_i^T \partial f(x_i)\|} \right)^2 = r_0^2 - \sum_{i=0}^{k-1} h_i^2 = r_k^2,$$

с помощью которого обеспечивается достаточное условие отсутствия точки x_ε^* при $f_{min} < f^* - \varepsilon$ (реализовано в п. 1 алгоритм-функции `amsg2p`).

Антиовражная техника в методе `amsg2p` направлена на уменьшение степени овражности поверхностей уровня выпуклых функций подобно тому, как это сделано в r -алгоритмах [5]. Детерминант матрицы B_k стремится к нулю, так как, если на k -м шаге реализуется преобразование пространства, то

$$\det(B_{k+1}) = \det(B_k) \det(I + \eta \xi_{k+1}^T) = \det(B_k) \sqrt{1 - \mu_k^2}.$$

Для овражных функций это обеспечивает ускоренную сходимость метода `amsg2p` при произвольной начальной стартовой точке x_0 и достаточно малых значениях параметра ε ($\varepsilon \sim 10^{-10} - 10^{-14}$).

Последовательное уточнение границ на минимальное значение функции f^* , например, по методу дихотомии позволяет использовать метод `amsg2p` для нахождения достаточно точного приближения к единственной точке минимума овражных функций. В докладе проиллюстрируем это на примере известной тестовой задачи `maxquad` [6], которая связана с минимизацией существенно овражной выпуклой кусочно-квадратичной функции $f(x) = \max_{1 \leq k \leq 5} \varphi_k(x)$, $x \in R^{10}$. Здесь $\varphi_k(x) = x^T H_k x - b_k^T x$, H_k — симметричные 10×10 -матрицы, такие что $H_{kij} = e^{i/j} \cos(ij) \sin k$, если $i < j$, и $H_{kii} = i |\sin k| / 10 + \sum_{j \neq i} |H_{kij}|$, а компоненты векторов b_k определяются $b_{ki} = e^{i/k} \sin(ik)$.

Литература

1. Стецюк П. И. Метод `amsg2p` для овражных выпуклых функций // Информационный бюллетень Ассоциации математического программирования. № 12. Научное издание. — Екатеринбург: УрО РАН, 2011. — С. 57–58.

2. Стецюк П. И. Субградиентные методы переменной метрики, использующие шаг Агмона-Мощкина и одноранговый эллипсоидальный оператор // Труды АТИК – 2007–2008. Кишинэу: Эврика. – 2009. – Т. I (XII). – С. 16–25.
3. Стецюк П. И. Ортогонализующие линейные операторы в выпуклом программировании // Кибернетика и системный анализ. – 1997. – № 3. – С. 97–119.
4. Поляк Б. Т. Введение в оптимизацию. – М.: Наука, 1983. – 384 с.
5. Шор Н. З. Методы минимизации недифференцируемых функций и их приложения. – Киев: Наукова думка, 1979. – 199 с.
6. Lemarechal C. Numerical experiments in nonsmooth optimization // In: Progress in nondifferentiable optimization /Ed. E. A. Nurminski. CP-82-58. – Laxenburg: International Institute for Applied System Analysis, 1982. – P. 61–68.

Ускоренные по Шору модификации субградиентного метода Поляка

П. И. Стецюк

Математическое моделирование, оптимизация и информационные технологии: материалы 3-й Междунар. науч. конф. (Кишинэу, 19-23 марта 2012 г.): Кишинэу: Эврика, 2012. – С. 509–519.

Обсуждаются два субградиентных метода с преобразованием пространства для нахождения точки минимума выпуклой функции при известном ее минимальном значении. В методах гарантируется монотонное уменьшение расстояния до точки минимума в последовательно преобразованных пространствах переменных. Преобразование пространства направлено на уменьшение степени овражности поверхностей уровня выпуклых функций подобно тому, как это сделано в r -алгоритмах.

Ключевые слова: субградиентный метод, шаг Поляка, шаг Агмона-Моцкина-Шенберга, овражная функция, преобразование пространства, одноранговый эллипсоидальный оператор.

Пусть $f(x)$ – выпуклая функция векторного аргумента $x \in R^n$ и ее субградиент $\partial f(x)$ удовлетворяет условию:

$$(x - x^*, \partial f(x)) \geq m(f(x) - f^*), \quad \text{где } m \geq 1, \quad (1)$$

для $\forall x \in R^n$ и $x^* \in R^n$. Здесь R^n – евклидово пространство размерности n со скалярным произведением (x, y) ; x^* – единственная точка минимума функции $f(x)$; f^* – минимальное значение функции $f(x)$: $f^* = f(x^*)$, $x^* \in R^n$. В неравенстве (1) параметр m введен для учета специальных классов выпуклых функций: для кусочно-линейной негладкой функции $m = 1$, для квадратичной гладкой функции $m = 2$.

Метод Поляка и проблема овражности. Пусть значение f^* известно. Для нахождения x^* можно использовать субградиентный метод Поляка [1]:

$$x_{k+1} = x_k - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|}, \quad h_k = \frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|}, \quad k = 0, 1, 2, \dots \quad (2)$$

Здесь шаг h_k задает величину максимального сдвига в направлении нормированного антисубградиента, при котором для выпуклой функции $f(x)$ условие (1) гарантирует, что угол между антисубградиентом

и направлением из точки x_{k+1} на точку минимума x^* будет нетупым. Величину h_k называют шагом Поляка или шагом Агмона-Моцкина-Шенберга.

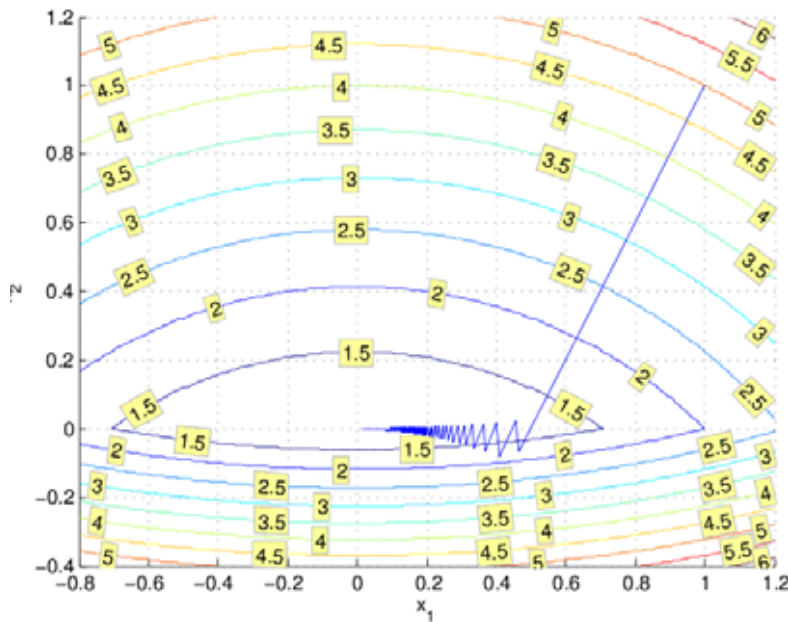


Рис.1

Метод Поляка медленно сходится для овражных и существенно овражных функций. Так, например, для негладкой функции двух переменных $f_1(x_1, x_2) = |x_1| + k|x_2|$ скорость сходимости метода (2) определяется геометрической прогрессией со знаменателем $\sqrt{1 - 1/k^2}$, который близок к единице даже при сравнительно небольших значениях k . Для существенно овражной кусочно-квадратичной функции

$$f_2(x_1, x_2) = \max \{x_1^2 + (2x_2 - 2)^2 - 3, x_1^2 + (x_2 + 1)^2\},$$

зигзагообразная траектория последовательных приближений, полученных методом Поляка, изображена на рис. 1. Здесь наблюдается вырождение в точке минимума $x^* = (0, 0)$, за 10000 итераций этим методом не удастся найти приближение, где бы значение функции $f_2(x_1, x_2)$ отличалось от минимального $f_2^* = 1$ не более, чем на 0.0001.

Ускоренный по Шору метод Поляка. Медленную сходимость метода Поляка для овражных функций определяет тупой угол между двумя последовательными субградиентами. Чем ближе угол к 180 градусам, тем более медленной будет сходимость. Ускорить метод (2) можно, если пространство переменных преобразовать так, чтобы уменьшить тупой угол между двумя последовательными субградиентами. Тупой угол между двумя нормированными векторами ξ и η из R^n можно преобразовать в прямой с помощью однорангового эллипсоидального оператора [2] – линейного оператора из R^n в R^n , который в матричной форме имеет вид:

$$T_1(\xi, \eta) = I - \frac{1}{1 - (\xi, \eta)^2} \left(\left(1 - \sqrt{1 - (\xi, \eta)^2} \right) \eta - (\xi, \eta) \xi \right) \eta^T. \quad (3)$$

Здесь $\xi, \eta \in R^n$ – векторы, $\|\xi\| = 1$, $\|\eta\| = 1$, а их скалярное произведение удовлетворяет условию $(\xi, \eta)^2 \neq 1$, I – единичная матрица размера $n \times n$. Для оператора $T_1(\xi, \eta)$ существует обратный

$$T_1^{-1}(\xi, \eta) = I + \frac{1}{\sqrt{1 - (\xi, \eta)^2}} \left(\left(1 - \sqrt{1 - (\xi, \eta)^2} \right) \eta - (\xi, \eta) \xi \right) \eta^T. \quad (4)$$

Ускоренный за счет преобразования пространства на основе оператора $T_1(\xi, \eta)$ субградиентный метод Поляка имеет следующий вид:

$$x_{k+1} = x_k - h_k B_k \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}, \quad h_k = \frac{m(f(x_k) - f^*)}{\|B_k^T \partial f(x_k)\|}, \quad k = 0, 1, 2, \dots, \quad (5)$$

где матрица $B_0 = I$, а матрица B_{k+1} пересчитывается по следующему правилу:

$$B_{k+1} = \begin{cases} B_k, & \text{если } (\xi, \eta) \geq 0, \\ B_k T^{-1}(\xi, \eta) & \text{иначе,} \end{cases}$$

$$\text{где } \xi = \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|} \quad \text{и} \quad \eta = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|}.$$

Пусть $A_k = B_k^{-1}$, $A_{k+1} = B_{k+1}^{-1}$. Для метода (5) справедливы неравенства

$$\|A_{k+1}(x_{k+1}-x^*)\|^2 \leq \|A_k(x_k-x^*)\|^2 - \left(\frac{m(f(x_k)-f^*)}{\|B_k^T \partial f(x_k)\|} \right)^2, \quad k=0,1,\dots \quad (6)$$

Неравенства (6) означают, что в методе (5) преобразование пространства таково, что в каждом очередном преобразованном пространстве переменных гарантируется уменьшение расстояния до точки минимума.

Антиовражная техника в методе (5) направлена на уменьшение степени овражности поверхностей уровня выпуклых функций подобно тому, как это сделано в r -алгоритмах [3]. Для овражных функций это обеспечивает ускоренную сходимость метода (5) по отношению к методу (2) при произвольной начальной стартовой точке x_0 . Так, например, для овражной функции двух переменных $f(x_1, x_2) = |x_1| + k|x_2|$ метод (5) будет находить точку x^* не более чем за две итерации независимо от значения k . Ускоренная сходимость метода (5) для существенно овражной функции $f_2(x_1, x_2)$ проиллюстрирована на рис. 2. Как видим, здесь уже нет зигзагообразной траектории, и метод за 30 итераций находит такую точку x_r^* , в которой функция $f_2(x_1, x_2)$ принимает значение равное $f_r^* < 1.00001$.

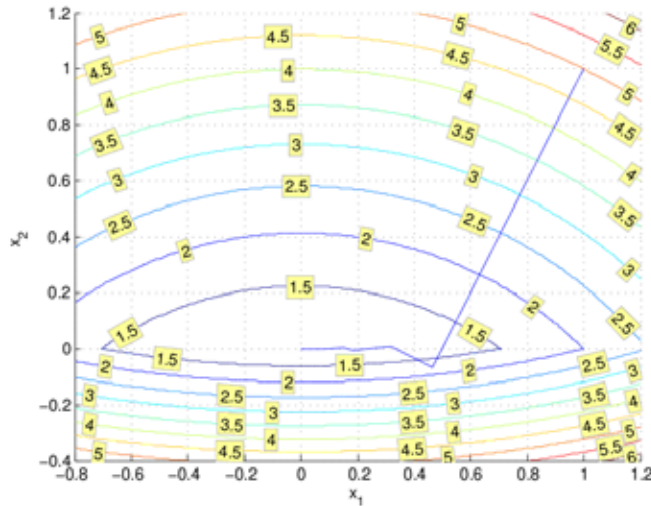


Рис.2

Метод (5) является ускоренным методом Поляка за счет преобразования пространства переменных, которое свойственно r -алгоритмам Шора [3]. Действительно, метод (5) обладает двумя характерными чертами r -алгоритмов. Во-первых, движение из точки осуществляется в направлении антисубградиента и, во-вторых, растяжение пространства производится в направлении разности нормированных последовательных субградиентов, если угол между ними тупой. Если нормы субградиентов одинаковы, то это направление будет совпадать с разностью двух последовательных субградиентов, но в отличие от r -алгоритмов второй субградиент получен не согласно шагу наискорейшего спуска, а согласно шагу Поляка в преобразованном пространстве переменных.

Метод amsg2p. В его основу положен второй из субградиентных методов с преобразованием пространства и регулировкой шага Агмона-Моцкина-Шенберга [4]. В названии "ams" указывает на способ регулировки шага в направлении нормированного антисубградиента, а "g2p" указывает, что аms-шаг используется в пространстве переменных, преобразованном с помощью двух последних субградиентов (g2) и агрегатного вектора (p).

В [5] метод amsg2p адаптирован для произвольного значения функции f_{min} и позволяет либо находить такую точку, где значение выпуклой функции $f(x)$ меньше или равно $f_{min} + \varepsilon$, либо гарантировать достаточное условие, что в шаре заданного радиуса не существует такой точки, где значение $f(x)$ равно f_{min} . В первом случае метод amsg2p находит точку $x_\varepsilon^* \in \{x : f(x) - f_{min} \leq \varepsilon\}$ и соответствующий ей номер итерации k_ε^* , а во втором — останавливается с сообщением "точки не существует". Метод реализован следующей алгоритм функцией.

$$\text{Алгоритм-функция amsg2p: } (x_\varepsilon^*, k_\varepsilon^*) = \text{amsg2p}(x_0, \varepsilon, r_0, f_{min}, \gamma)$$

На итерации $k = 0$ заданы: начальное приближение $x_0 \in R^n$; начальный радиус r_0 такой, что $\|x_0 - x^*\| \leq r_0$; достаточно малое $\varepsilon > 0$. Вычислим $f(x_0)$ и $\partial f(x_0)$. Если $f(x_0) - f_{min} \leq \varepsilon$, то $x_\varepsilon^* = x_0$, $k_\varepsilon^* = 0$ и окончание алгоритма. Иначе положим $h_0 = \frac{m(f(x_0) - f_{min})}{\|\partial f(x_0)\|}$, $\xi_0 = \frac{\partial f(x_0)}{\|\partial f(x_0)\|} \in R^n$, $p_0 = 0 \in R^n$, $B_0 = I$ — единичная $n \times n$ -матрица. Перейдем к следующей итерации.

Пусть на k -й итерации получены $x_k \in R^n$, h_k , r_k , $\xi_k \in R^n$, $p_k \in R^n$, B_k — матрица $n \times n$. Для $(k + 1)$ -й итерации выполним пп. 1-5.

- 1) Вычислим $t_k = h_k / r_k$. Если $t_k > 1$, то "точки не существует" и

окончание алгоритма. Иначе положим $r_{k+1} = r_k \sqrt{1 - t_k^2}$ и вычислим очередное приближение

$$x_{k+1} = x_k - h_k B_k \xi_k.$$

- 2) Вычислим $f(x_{k+1})$ и $\partial f(x_{k+1})$. Если $f(x_{k+1}) - f_{min} \leq \varepsilon$, то $x_\varepsilon^* = x_{k+1}$, $k_\varepsilon^* = k + 1$ и окончание алгоритма. Иначе положим

$$\xi_{k+1} = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|}, \quad h_{k+1} = \frac{m(f(x_{k+1}) - f_{min})}{\|B_k^T \partial f(x_{k+1})\|}.$$

- 3) Вычислим $\lambda_1 = -p_k^T \xi_{k+1}$ и $\lambda_2 = -\xi_k^T \xi_{k+1}$. Положим

$$p_{k+1} = \begin{cases} \frac{\lambda_1}{\sqrt{\lambda_1^2 + \lambda_2^2}} p_k + \frac{\lambda_2}{\sqrt{\lambda_1^2 + \lambda_2^2}} \xi_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 > 0, \\ p_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 \leq 0, \\ \xi_k, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 > 0, \\ 0, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 \leq 0. \end{cases}$$

- 4) Вычислим $\mu_k = p_{k+1}^T \xi_{k+1}$. Если $-1 < \mu_k < 0$, то вычислим

$$B_{k+1} = B_k + (B_k \eta) \xi_{k+1}^T, \text{ где } \eta = \left(\frac{1}{\sqrt{1 - \mu_k^2}} - 1 \right) \xi_{k+1} - \frac{\mu_k}{\sqrt{1 - \mu_k^2}} p_{k+1},$$

и пересчитаем

$$h_{k+1} = \frac{h_{k+1}}{\sqrt{1 - \mu_k^2}}, \quad p_{k+1} = \frac{1}{\sqrt{1 - \mu_k^2}} (p_{k+1} - \mu_k \xi_{k+1}).$$

Иначе положим $B_{k+1} = B_k$ и $p_{k+1} = 0$.

- 5) Перейдем к следующей итерации с x_{k+1} , h_{k+1} , r_{k+1} , ξ_{k+1} , p_{k+1} , B_{k+1} .

Теорема 1 Если $f_{min} \geq f^*$, то для метода `amsg2p` справедливы неравенства

$$\|A_{k+1}(x_{k+1} - x^*)\|^2 \leq \|A_k(x_k - x^*)\|^2 - \left(\frac{m(f(x_k) - f_{min})}{\|B_k^T \partial f(x_k)\|} \right)^2, \quad k = 0, 1, \dots,$$

Здесь $A_k = B_k^{-1}$, $A_{k+1} = B_{k+1}^{-1}$.

Теорема 1 означает, что в каждом очередном преобразованном пространстве переменных расстояние до точки минимума уменьшается. Благодаря этому для каждой итерации $k > 1$ имеем неравенство

$$\|A_k(x_k - x^*)\|^2 \leq \|x_0 - x^*\|^2 - \sum_{i=0}^{k-1} \left(\frac{m(f(x_i) - f_{min})}{\|B_i^T \partial f(x_i)\|} \right)^2 = r_0^2 - \sum_{i=0}^{k-1} h_i^2 = r_k^2,$$

с помощью которого обеспечивается достаточное условие отсутствия точки x_ε^* при $f_{min} < f^* - \varepsilon$ (реализовано в п. 1 алгоритм-функции `amsg2p`).

Антиовражная техника в методе `amsg2p` направлена на уменьшение степени овражности поверхностей уровня выпуклых функций подобно тому, как это сделано в r -алгоритмах [3]. Преобразование пространства реализуется с помощью однорангового эллипсоидального оператора и только на тех итерациях метода, когда тупым является хотя бы один из углов — угол между двумя последовательными субградиентами, либо угол между последним субградиентом и агрегатным вектором, который есть выпуклой комбинацией вычисленных на предыдущих итерациях субградиентов. Детерминант матрицы B_k стремится к нулю, так как, если на k -м шаге реализуется преобразование пространства, то

$$\det(B_{k+1}) = \det(B_k) \det(I + \eta \xi_{k+1}^T) = \det(B_k) \sqrt{1 - \mu_k^2}.$$

Для овражных функций это обеспечивает ускоренную сходимость метода `amsg2p` при произвольной начальной стартовой точке x_0 и достаточно малых значениях параметра ε ($\varepsilon \sim 10^{-10} - 10^{-14}$).

В таблице приведены результаты вычислительных экспериментов для квадратичных функций от $n = 200$ переменных с различной степенью овражности (рассматривалась функция $f(x) = \sum_{i=1}^n q^{i-1} x_i^2$, степень ее овражности характеризовала величина $Q = q^{n-1}$). При расчетах использовался параметр $m = 2$, $f_{min} = f^* = 0$, $x_0 = (1, \dots, 1)^T$. Для ряда стремящихся к нулю значений ε в таблице указано число итераций, необходимых при степенях овражности $Q1 = 10$, $Q2 = 100$, $Q3 = 1000$, $Q4 = 10^6$, $Q5 = 10^9$. Из таблицы видим, что в наихудшем случае, который соответствует степени овражности $Q5 = 10^9$, методу требуется порядка $1000 = 5 \times 200$ итераций, т.е. около $5n$ итераций. Если поверхности уровня квадратичной функции слабо вытянуты, то методу требуется меньше итераций, чем количество переменных n .

eps	itn(Q1)	itn(Q2)	itn(Q3)	itn(Q4)	itn(Q5)
1.00E-003	11	36	84	361	773
1.00E-005	15	46	99	405	826
1.00E-007	18	56	113	430	868
1.00E-009	22	65	128	461	916
1.00E-011	25	73	142	493	947
1.00E-013	29	81	154	517	979
1.00E-015	32	89	167	541	1006
1.00E-017	35	96	180	560	1027
1.00E-019	39	102	189	574	1042
1.00E-020	41	105	196	585	1048

Метод `amsq2p` можно использовать для нахождения достаточно точного приближения к единственной точке минимума существенно овражных функций. Проиллюстрируем это на примере известной тестовой задачи `maxquad` [6], которая связана с минимизацией существенно овражной выпуклой кусочно-квадратичной функции $f(x) = \max_{1 \leq k \leq 5} \varphi_k(x)$, $x \in R^{10}$.

Здесь $\varphi_k(x) = x^T H_k x - b_k^T x$, H_k – симметричные 10×10 -матрицы, такие что $H_{kij} = e^{i/j} \cos(ij) \sin k$, если $i < j$, и $H_{kii} = i |\sin k| / 10 + \sum_{j \neq i} |H_{kij}|$, а компоненты векторов b_k равны $b_{ki} = e^{i/k} \sin(ik)$.

Затраты метода `amsq2p` для нахождения в задаче `maxquad` единственного решения с достаточно высокой точностью (до 14-ти значащих цифр) позволяет оценить приведенный ниже фрагмент численных расчетов. Здесь $m = 1$, $x_0 = (1, \dots, 1)^T$ и $\varepsilon = \text{epsf}$.

```

Maxquad: f(x0)      5.3370664293114e+003
          fmin = -8.4140833459641e-001
..epsf..   ....f(itn).....  .itn.
1.0e-001   -7.7355266120112e-001   17
1.0e-003   -8.4084776169123e-001   29
1.0e-004   -8.4132394277880e-001   35
1.0e-005   -8.4140078034524e-001   41
1.0e-006   -8.4140807664455e-001   49
1.0e-011   -8.4140833458913e-001   94
1.0e-012   -8.4140833459555e-001  101
1.0e-013   -8.4140833459633e-001  110
1.0e-014   -8.4140833459640e-001  116
1.0e-015   -8.4140833459641e-001  122

```

Литература

1. Поляк Б.Т., *Введение в оптимизацию*, М.: Наука (1983).
2. Стецюк П.И., “Ортогонализующие линейные операторы в выпуклом программировании”, *Кибернетика и системный анализ*, No. 3, 97–119 (1997).
3. Шор Н.З., *Методы минимизации недифференцируемых функций и их приложения*, Киев: Наукова думка, (1979).
4. Стецюк П. И. *Субградиентные методы переменной метрики, использующие шаг Агмона-Моцкина и одноранговый эллипсоидальный оператор* // Труды АТИК – 2007–2008. Кишинэу: Эврика. – 2009. – Т. I (XII). – С. 16–25.
5. Стецюк П. И. *Метод atmsg2p для овражных выпуклых функций* // Информационный бюллетень Ассоциации математического программирования. № 12. Научное издание. – Екатеринбург: УрО РАН, 2011. – С. 57–58.
6. Lemarechal C. *Numerical experiments in nonsmooth optimization* // In: Progress in nondifferentiable optimization /Ed. E. A. Nurminski. CP-82-58. – Laxenburg: International Institute for Applied System Analysis, 1982. – P. 61–68.

Ускорение субградиентного метода Поляка

П. И. Стецюк

Теорія оптимальних рішень. – 2012. – №11. – С. 151–160.

Исследуются свойства субградиентного метода Поляка для нахождения точки минимума выпуклой функции. Показано, что для овражных функций сходимость метода можно ускорить за счет линейного преобразования пространства переменных. Изложен субградиентный метод Поляка с преобразованием пространства в случае тупого угла между двумя последовательными субградиентами, что существенно сокращает количество итераций для гладких и негладких овражных функций.

Введение. В этом году украинскому ученому, академику НАН Украины Н.З. Шору, исполнилось бы 75 лет. На протяжении многих лет статьи Н.З.Шора и его учеников публиковались в сборнике научных трудов "Теория оптимальных решений". Первая статья "Алгебраический подход к исследованию задачи о четырех красках" опубликована в соавторстве с Г.А.Донцом в 1967 г. Последняя статья "Двойственные оценки для специальной оптимизационной задачи квадратичного типа на многообразии Штиффеля" (соавторы – П.И.Стецюк и О.А.Березовский) опубликована в 2004 г. Названия этих статей лишь подчеркивают центральный результат Н.З.Шора за последние 20 лет – двойственный метод получения и уточнения оценок целевой функции в невыпуклых квадратичных моделях. Этот метод нашел красивое воплощение в NP -трудной задаче нахождения взвешенного максимального независимого множества вершин графа, где двойственные оценки Шора равны известным числам Ловаса. Эта задача используется в теории информации и кодировании, проектировании различных устройств при определенных условиях несовместности; она тесно связана с известными задачами выбора, разбиения множеств, раскраски графов и другими комбинаторными задачами, имеющими, в свою очередь, массу приложений.

Большое количество статей в "Теории оптимальных решений" посвящено исследованию модификаций r -алгоритмов и их применений в сложных экстремальных задачах. Данная статья связана с использованием линейных неортогональных преобразований пространства для улучшения обусловленности овражных функций – идеи Н.З. Шора, на которой построены эффективные модификации r -алгоритмов. В ней

рассмотрены субградиентные методы с преобразованием пространства, основанные на монотонном уменьшении расстояния к точке минимума. Сделано это на примере известного субградиентного метода, предложенного Б.Т.Поляком в 1969 г. для нахождения точки минимума выпуклой функции при известном оптимальном значении функции.

1. Субградиентный метод Поляка. Пусть $f(x)$ – выпуклая функция векторного аргумента $x \in R^n$ и ее субградиент $\partial f(x)$ удовлетворяет условию:

$$(x - x^*, \partial f(x)) \geq m(f(x) - f^*), \quad \forall x \in R^n, \quad \forall x^* \in X^*, \quad m \geq 1. \quad (1)$$

Здесь R^n – евклидово пространство размерности n со скалярным произведением (x, y) ; X^* – множество точек минимума функции $f(x)$; f^* – минимальное значение функции $f(x)$: $f^* = f(x^*)$, $x^* \in X^*$. Параметр m введен для учета специальных классов выпуклых функций: например, для кусочно-линейной негладкой функции $m = 1$, для квадратичной гладкой функции $m = 2$.

Пусть известно f^* . Для нахождения точки $x^* \in X^*$ применяется субградиентный метод Поляка [1]:

$$x_{k+1} = x_k - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|}, \quad h_k = \frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|}, \quad k = 0, 1, 2, \dots \quad (2)$$

Величину h_k называют шагом Поляка или шагом Агмона-Моцкина-Шенберга. Этот шаг тесно связан с результатами И.И.Еремина о сходимости итерационных методов аппроксимации неподвижных точек с помощью операторов, обладающих свойством квазисжимаемости (фейеровости) [2].

Шаг Поляка обладает интересными свойствами, которые ему обеспечивает априорное знание минимального значения функции и связанное с ним неравенство (1).

Теорема 1. Для всех точек, генерируемых методом (2), справедливы неравенства

$$\|x_{k+1} - x^*\|^2 \leq \|x_k - x^*\|^2 - \left(\frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|} \right)^2, \quad \forall x^* \in X^*, \quad k = 0, 1, \dots, \quad (3)$$

и неравенства

$$(x^* - x_{k+1}, -\partial f(x_k)) \geq 0, \quad \forall x^* \in X^*, \quad k = 0, 1, \dots, \quad (4)$$

Доказательство. Для любого $x^* \in X^*$ и произвольного k ($k \geq 0$) имеем

$$\begin{aligned}\|x_{k+1} - x^*\|^2 &= \left\| x_k - x^* - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|} \right\|^2 = \\ &= \|x_k - x^*\|^2 - 2h_k \frac{(x_k - x^*, \partial f(x_k))}{\|\partial f(x_k)\|} + h_k^2.\end{aligned}$$

Учитывая, что из (1) следует неравенство

$$\frac{(x_k - x^*, \partial f(x_k))}{\|\partial f(x_k)\|} \geq \frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|} = h_k,$$

имеем

$$\|x_{k+1} - x^*\|^2 \leq \|x_k - x^*\|^2 - h_k^2 = \|x_k - x^*\|^2 - \left(\frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|} \right)^2,$$

что дает неравенства (3).

Неравенства (4) следуют из того, что на основании (1) имеем

$$\begin{aligned}& (x^* - x_{k+1}, -\partial f(x_k)) = (x_{k+1} - x^*, \partial f(x_k)) = \\ &= \left(x_k - x^* - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|}, \partial f(x_k) \right) = (x_k - x^*, \partial f(x_k)) - h_k \|\partial f(x_k)\| = \\ &= (x_k - x^*, \partial f(x_k)) - m(f(x_k) - f^*) \geq 0.\end{aligned}$$

Теорема 1 доказана.

В теореме отражены два центральных свойства шага Поляка. Первое свойство следует из неравенств (4). Оно означает, что шаг h_k определяет величину максимального сдвига в направлении нормированного антисубградиента, при котором угол между антисубградиентом и направлением из точки x_{k+1} на произвольную точку из множества минимумов будет нетупым. Второе свойство следует из неравенств (3) и означает, что если множество X^* состоит из единственной точки x^* , то шаг h_k выбирается таким, чтобы расстояние от точки x_{k+1} к точке минимума x^* было минимальным.

2. Проблема овражности. Медленную скорость сходимости метода Поляка проиллюстрируем на примере трех выпуклых функций от двух переменных, первые две функции – негладкие (овражная и существенно овражная), а третья – квадратичная овражная функция.

В табл. 1 приведено количество итераций метода (2) для нахождения последовательно уточняемых приближений к единственной точке минимума $x^* = (0, 0)$. Эти приближения заданы десятью последовательно уменьшающимися (на порядок) значениями $epsf = \varepsilon_f$ (первая колонка в табл. 1). Метод Поляка прекращал работу или на итерации $k = itn$, для которой $f(x_{itn}) - f^* \leq \varepsilon_f$, или если превышено максимальное количество итераций, равное 20000.

Колонки $itn1$, $itn2$ и $itn3$ отвечают овражной кусочно-линейной функции $f_1(x_1, x_2) = |x_1| + t|x_2|$ для значений t равных 3, 9 и 27 соответственно. При расчетах использовалась стартовая точка $x_0 = (1, 1)$ и параметр $m = 1$. Из табл. 1 видим, что количество итераций существенно увеличивается с ростом t . Для того чтобы найти точку, где значение функции $f_1(x_1, x_2) = |x_1| + 27|x_2|$ отличается от минимального $f_1^* = 0$ не более, чем на $\varepsilon_f = 10^{-10}$, методу Поляка потребовалось 8634 итерации. Такую же точность для функции $f_1(x_1, x_2) = |x_1| + t|x_2|$ при $t = 100$ получить нереально даже при ста тысячах итераций метода Поляка.

Для $f_2(x_1, x_2) = \max\{x_1^2 + (2x_2 - 2)^2 - 3, x_1^2 + (x_2 + 1)^2\}$ — существенно овражной кусочно-квадратичной функции зигзагообразная траектория последовательных приближений метода Поляка дана на рис. 1. Здесь метод Поляка находит приближение, где значение функции $f_2(x_1, x_2)$ отличается от минимального $f_2^* = 1$ не более, чем на 0.0001, только за 16004 итерации (см. колонку $itn4$ в табл. 1). Медленная сходимость проявляется со второй итерации, за одну итерацию метод спускается на дно оврага, а далее метод зигзагообразно движется вдоль русла оврага.

Для функций $f_1(x_1, x_2)$ и $f_2(x_1, x_2)$ использовался параметр $m = 1$, который можно применять для произвольной выпуклой функции. Для квадратичной функции $f_3(x_1, x_2) = x_1^2 + tx_2^2$, $t > 0$ использовался параметр $m = 2$. Он в два раза увеличивает длину шага Поляка, соответствующего параметру $m = 1$, и обеспечивает более быструю скорость сходимости с различных начальных приближений. Количество итераций метода Поляка со стартовой точки $x_0 = (0, 0)$ приведено в колонках $itn5$, $itn6$ и $itn7$ и соответствует значениям t равным 100, 10000 и 1000000. Количество итераций для всех 10 значений $epsf$ одинаково и не зависит от степени вытянутости поверхности квадратичной функции. Отличия наблюдаются лишь при достаточно малых $\varepsilon_f = 10^{-12}, 10^{-14}, 10^{-16}, 10^{-18}, 10^{-20}$, для которых количество итераций приведено в скобках после количеств итераций для шести последних значений $epsf$. Чтобы найти точку минимума квадратичной функции с такой же точностью, как и точку минимума функции $f_1(x_1, x_2)$, нужно взамен ε_f использовать ε_f^2 .

Таблица 1. Сходимость метода Поляка для двумерных овражных функций

epsf	Itn1	itn2	itn3	itn4	itn5	Itn6	itn7
1.0e-01	14	119	1080	16	6	6	6
1.0e-02	24	212	1919	162	10	10	10
1.0e-03	34	305	2759	1604	12	12	12
1.0e-04	45	398	3598	16004	16	16	16
1.0e-05	55	492	4437	20000	20	20	20
1.0e-06	65	585	5277	20000	22(42)	22(42)	22(42)
1.0e-07	76	678	6116	20000	26(48)	26(52)	26(52)
1.0e-08	86	771	6955	20000	28(54)	30(56)	30(56)
1.0e-09	96	865	7795	20000	32(62)	32(62)	32(62)
1.0e-10	107	958	8634	20000	36(68)	36(70)	36(70)

3. Метод Поляка с преобразованием пространства. Пусть произведена замена переменных $x = By$, где B – неособенная $n \times n$ -матрица (т.е. существует обратная матрица $A = B^{-1}$). Субградиент выпуклой функции $f(x)$ в точке x_k удовлетворяет неравенству

$$f(x) \geq f(x_k) + \left(\partial f(x_k), x - x_k \right) \quad \forall x \in R^n,$$

откуда, осуществляя замену переменных $x = By$, получаем

$$\varphi(y) \geq \varphi(y_k) + (B^T \partial f(x_k), y - y_k) \quad \forall y \in R^n.$$

Вектор $\partial \varphi(y_k) = B^T \partial f(x_k)$ удовлетворяет неравенству

$$\varphi(y) \geq \varphi(y_k) + (\partial \varphi(y_k), y - y_k) \quad \forall y \in R^n$$

и является субградиентом выпуклой функции $\varphi(y) = f(By)$ в точке $y_k = Ax_k$ преобразованного пространства переменных $y = Ax$ [3].

Для нахождения точки $x^* \in X^*$ субградиентный метод Поляка с преобразованием пространства (определяется невырожденной матрицей B) имеет следующий вид:

$$x_{k+1} = x_k - h_k \frac{B^T \partial f(x_k)}{\|B^T \partial f(x_k)\|}, \quad h_k = \frac{m(f(x_k) - f^*)}{\|B^T \partial f(x_k)\|}, \quad k = 0, 1, 2, \dots \quad (5)$$

Здесь величина h_k – шаг Поляка (шаг Агмона-Мощкина-Шенберга), но в преобразованном пространстве переменных $y = Ax$. Это следует из

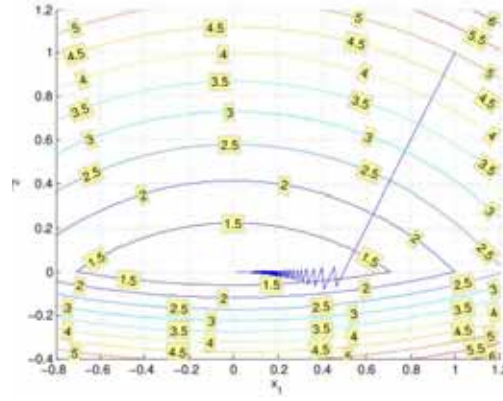


Рис. 1. Траектория метода Поляка для $f_2(x_1, x_2)$ ($x_0 = (1, 1)$)

того, что в преобразованном пространстве переменных метод (5) записывается как субградиентный процесс

$$y_{k+1} = y_k - h_k \frac{\partial \varphi(y_k)}{\|\partial \varphi(y_k)\|}, \quad h_k = \frac{m(\varphi(y_k) - \varphi^*)}{\|\partial \varphi(y_k)\|}, \quad k = 0, 1, 2, \dots \quad (6)$$

Шаг Поляка в преобразованном пространстве переменных обладает такими же свойствами как и шаг Поляка в исходном пространстве. Их обеспечивает ему априорное знание минимального значения функции и связанное с ним неравенство (1).

Теорема 2. Для всех точек, генерируемых методом (5), справедливы неравенства

$$\begin{aligned} & \|A(x_{k+1} - x^*)\|^2 \leq \\ & \leq \|A(x_k - x^*)\|^2 - \left(\frac{m(f(x_k) - f^*)}{\|B^T \partial f(x_k)\|} \right)^2, \quad \forall x^* \in X^*, \quad k = 0, 1, \dots, \end{aligned} \quad (7)$$

и неравенства

$$(A(x^* - x_{k+1}), -B^T \partial f(x_k)) \geq 0, \quad \forall x^* \in X^*, \quad k = 0, 1, \dots \quad (8)$$

Доказательство теоремы 2 аналогично доказательству теоремы 1.

Очевидно, что если матрицу B выбрать такой, чтобы овражная функция в преобразованном пространстве переменных становилась менее овражной, то субградиентный метод Поляка с преобразованием пространства (5) окажется эффективнее, чем этот метод без преобразования пространства (2). Это подтверждает количество итераций метода (5) для нахождения десяти последовательно уточняемых приближений функции $f_1(x_1, x_2) = |x_1| + 10|x_2|$ для различных матриц B , которые получены в результате растяжения пространства переменных в направлении x_2 с коэффициентами растяжения $\alpha = 1; 1.5; 2; 3; 4; 5$. Из табл. 2 легко видеть, что количество итераций метода Поляка с преобразованием пространства увеличивается по мере того, как уменьшается степень овражности функции $\varphi_1(y_1, y_2) = |y_1| + \frac{10}{\alpha}|y_2|$ в преобразованном пространстве переменных.

Таблица 2. Сходимость метода Поляка с преобразованием пространства

epsf	itn1	itn2	itn3	itn4	itn5	itn6
1.0e-01	147	63	33	6	10	9
1.0e-02	262	114	62	19	17	13
1.0e-03	377	165	91	31	24	18
1.0e-04	492	216	119	44	31	22
1.0e-05	607	268	148	57	38	27
1.0e-06	722	319	177	70	45	31
1.0e-07	837	370	206	82	53	36
1.0e-08	952	421	234	95	60	40
1.0e-09	1000	472	263	108	67	45
1.0e-10	1000	523	292	121	74	49

4. Ускоренный субградиентный метод Поляка. Медленную сходимость метода Поляка для овражных функций определяет угол между двумя последовательными субградиентами: $\partial f(x_k)$ и $\partial f(x_{k+1})$. Чем ближе этот угол к 180 градусам, тем более медленной будет скорость сходимости. Тупой угол между векторами $\partial f(x_k)$ и $\partial f(x_{k+1})$ можно преобразовать в прямой с помощью "однорангового эллипсоидального оператора" [4]. Если на каждой итерации избавляться от тупого угла между последовательными субградиентами в преобразованном пространстве переменных, то для овражных функций повысится ско-

рость сходимости метода. Этот принцип реализован в изложенной ниже модификации метода Поляка.

Ускоренный субградиентный метод Поляка имеет следующий вид:

$$x_{k+1} = x_k - h_k B_k \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}, \quad h_k = \frac{m(f(x_k) - f^*)}{\|B_k^T \partial f(x_k)\|},$$

$$k = 0, 1, 2, \dots, \quad (9)$$

где матрица $B_0 = I$, а матрица B_{k+1} размерности $n \times n$ вычисляется по правилу:

$$B_{k+1} = \begin{cases} B_k, & \text{если } \mu_k \geq 0 \\ B_{k+1} = B_k + (B_k \eta) \xi_{k+1}^T, & \text{иначе} \end{cases} \quad (10)$$

где

$$\mu_k = (\xi_k, \xi_{k+1}), \quad \xi_k = \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}, \quad \xi_{k+1} = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|},$$

$$\eta = \left(\frac{1}{\sqrt{1 - \mu_k^2}} - 1 \right) \xi_{k+1} - \frac{\mu_k}{\sqrt{1 - \mu_k^2}} \xi_k.$$

Метод (9), (10) можно назвать ускоренным методом Поляка за счет антиогражного приема, подобного тому, который использован в r -алгоритмах Шора [3]. Действительно, на k -й итерации растяжение пространства производится в направлении разности нормированных последовательных субградиентов в преобразованном пространстве переменных: $y = A_k x = B_k^{-1} x$, где B_k – невырожденная матрица размерности $n \times n$. Если нормы субградиентов одинаковы, то это направление совпадает с разностью двух последовательных субградиентов, по которой реализуется растяжение пространства в r -алгоритмах. Отличие состоит в том, что в преобразованном пространстве переменных для r -алгоритмов второй субградиент определяется согласно шагу наискорейшего спуска в направлении антисубградиента, а в методе (9), (10) – согласно шагу Поляка.

Теорема 3. Пусть $A_k = B_k^{-1}$, $A_{k+1} = B_{k+1}^{-1}$. Для всех точек, генерируемых методом (9), (10), справедливы неравенства

$$\|A_{k+1}(x_{k+1} - x^*)\|^2 \leq \|A_k(x_k - x^*)\|^2 - \left(\frac{m(f(x_k) - f^*)}{\|B_k^T \partial f(x_k)\|} \right)^2,$$

$$\forall x^* \in X^*, \quad k = 0, 1, \dots \quad (11)$$

Доказательство теоремы 3 аналогично доказательству теоремы 2 [4].

Преобразование пространства в методе (9), (10) направлено на уменьшение степени вытянутости поверхностей уровня овражных выпуклых функций. Неравенства (11) означают, что в каждом очередном преобразованном пространстве переменных гарантируется уменьшение расстояния до множества точек минимума. В связи с этим для каждой итерации $k > 1$ имеет место неравенство

$$\|A_k(x_k - x^*)\|^2 \leq \|x_0 - x^*\|^2 - \sum_{i=0}^{k-1} \left(\frac{m(f(x_i) - f^*)}{\|B_i^T \partial f(x_i)\|} \right)^2.$$

Для овражных функций детерминант матрицы B_k стремится к нулю, так как, если на k -м шаге реализуется преобразование пространства, то

$$\begin{aligned} \det(B_{k+1}) &= \det(B_k) \det(I + \eta \xi_{k+1}^T) = \det(B_k) \sqrt{1 - \mu_k^2} = \\ &= \det(B_k) \sqrt{1 - \cos^2 \varphi_k}, \end{aligned}$$

где φ_k – тупой угол между двумя последовательными субградиентами.

Для овражных негладких функций метод (9), (10) обладает ускоренной сходимостью по отношению к методу (2). Так, например, для кусочно-линейной функции двух переменных $f_1(x_1, x_2) = |x_1| + t|x_2|$ при любом значении параметра $t > 1$ и произвольной стартовой точке x_0 методом (9), (10) можно найти точку минимума $x^* = (0, 0)$ не более, чем за две итерации. Метод (2) сходится к x^* со скоростью геометрической прогрессии со знаменателем $\sqrt{1 - 1/t^2}$ и требует существенного количества итераций даже при сравнительно небольших значениях t . Траектория ускоренного метода Поляка для кусочно-квадратичной функции $f_2(x_1, x_2)$ дана на рис. 2, здесь метод находит точку, где $f_2(x) < 1 + 10^{-5}$ всего за 16 итераций, а где $f_2(x) < 1 + 10^{-10}$ – за 31 итерацию.

Заключение. Несмотря на то, что r -алгоритмы используются уже 40 лет, проблема обоснования их сходимости для всего класса выпуклых функций остается открытой и в настоящее время. Еще в 1982 г. Н.З.Шор и В.И.Гершович в работе [5] отметили: "Теория всего класса алгоритмов с растяжением пространства далека от совершенства. Нам кажется достаточно реалистичной целью – построение такого алгоритма, который по своей практической эффективности не уступал бы r -алгоритму и был столь же хорошо обоснован, как метод эллипсоидов". Шагом в этом направлении можно считать алгоритм (9), (10), где для преобразования специального эллипсоида в шар используется антиовражный

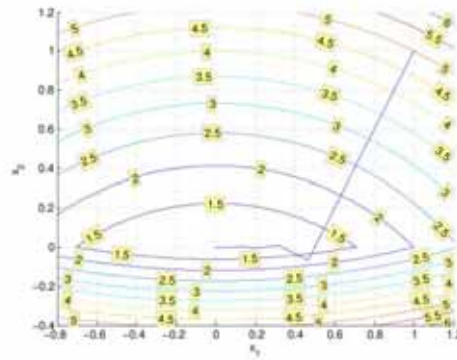


Рис. 2. Траектория ускоренного метода Поляка для $f_2(x_1, x_2)$ ($x_0 = (1, 1)$)

прием, близкий к тому, который имеет место в r -алгоритмах. Однако, здесь растяжение пространства реализуется в направлении разности двух нормированных субградиентов, и близким к направлению разности двух субградиентов оно будет только тогда, когда нормы субградиентов близки.

Литература

1. Поляк Б.Т. Введение в оптимизацию. – М.: Наука, 1983. – 384 с.
2. Васин В.В., Еремин И.И. Операторы и итерационные процессы фейеровского типа (теория и приложения). – Москва; Ижевск: Институт компьютерных исследований, НИЦ "Регулярная и хаотическая динамика", 2005. – 200 с.
3. Шор Н.З. Методы минимизации недифференцируемых функций и их приложения. – К.: Наук. думка, 1979. – 199 с.
4. Стецюк П.И. Ортогонализирующие линейные операторы в выпуклом программировании // Кибернетика и системный анализ. – 1997. – № 3. – С. 97–119.
5. Шор Н.З., Гершович В.И. Метод эллипсоидов, его обобщения и приложения // Кибернетика. – 1982. – № 5. – С. 61–69.

Ускоренный по Шору субградиентный метод Поляка¹

П. И. Стецюк

Конструктивный негладкий анализ и смежные вопросы. Тезисы докладов международной конференции. – СПб.: Изд-во Санкт-Петербургского университета, 2012. – Р. 157–160.

Пусть $f(x)$ – выпуклая функция векторного аргумента $x \in R^n$ и ее субградиент $\partial f(x)$ удовлетворяет условию:

$$(x - x^*, \partial f(x)) \geq m(f(x) - f^*), \quad \text{где } m \geq 1, \quad (1)$$

для $\forall x \in R^n$ и $\forall x^* \in X^*$. Здесь R^n – евклидово пространство размерности n со скалярным произведением (x, y) ; X^* – множество точек минимума функции $f(x)$; f^* – минимальное значение функции $f(x)$: $f^* = f(x^*)$, $x^* \in X^*$. Параметр m задает величину максимального сдвига по выпуклости функции $f(x)$ и введен для учета специальных классов выпуклых функций. Для кусочно-линейной негладкой функции $m = 1$, для квадратичной гладкой функции $m = 2$.

Пусть известно f^* . Для нахождения точки $x^* \in X^*$ можно использовать субградиентный метод Поляка [1]:

$$x_{k+1} = x_k - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|}, \quad h_k = \frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|}, \quad k = 0, 1, 2, \dots \quad (2)$$

Здесь шаг h_k задает величину максимального сдвига в направлении нормированного антисубградиента, при котором для выпуклой функции $f(x)$ условие (1) гарантирует, что угол между антисубградиентом и направлением из точки x_{k+1} на точку минимума будет нетупым.

Обоснование сходимости метода (2) очень простое, так как для всех точек итерационного процесса справедливы неравенства

$$\|x_{k+1} - x^*\|^2 \leq \|x_k - x^*\|^2 - \left(\frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|} \right)^2, \quad k = 0, 1, \dots$$

Отсюда для каждой итерации $K > 1$ имеем неравенство

$$\|x_K - x^*\|^2 \leq \|x_0 - x^*\|^2 - \sum_{k=0}^{K-1} \left(\frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|} \right)^2,$$

¹Shor's acceleration for Polyak's subgradient method

из которого, используя расходимость ряда, легко прийти к противоречию, что $\|x_{\overline{K}} - x^*\|^2 < 0$ для некоторого \overline{K} .

Медленную сходимость метода Поляка для овражных функций определяет тупой угол между двумя последовательными субградиентами. Чем ближе угол к 180 градусам, тем более медленной будет сходимость. Так, например, для негладкой функции двух переменных $f(x_1, x_2) = |x_1| + k|x_2|$ скорость сходимости метода (2) определяется геометрической прогрессией со знаменателем $\sqrt{1 - 1/k^2}$, который близок к 1 даже при сравнительно небольших значениях k .

Ускорить метод (2) можно, если пространство переменных преобразовать так, чтобы тупой угол между двумя последовательными субградиентами уменьшался. Тупой угол между двумя векторами можно преобразовать в прямой с помощью „однорангового эллипсоидального оператора“ [2]. Для двух нормированных векторов ξ и η из R^n это реализует линейный оператор из R^n в R^n , который в матричной форме имеет вид:

$$T_1(\xi, \eta) = I - \frac{1}{1 - (\xi, \eta)^2} \left(\left(1 - \sqrt{1 - (\xi, \eta)^2}\right) \eta - (\xi, \eta)\xi \right) \eta^T. \quad (3)$$

Здесь $\xi, \eta \in R^n$ – векторы, такие, что $\|\xi\| = 1$, $\|\eta\| = 1$ и их скалярное произведение удовлетворяет условию $(\xi, \eta)^2 \neq 1$, I – единичная матрица размера $n \times n$. Для оператора $T_1(\xi, \eta)$ существует обратный $T_1^{-1}(\xi, \eta)$:

$$T_1^{-1}(\xi, \eta) = I + \frac{1}{\sqrt{1 - (\xi, \eta)^2}} \left(\left(1 - \sqrt{1 - (\xi, \eta)^2}\right) \eta - (\xi, \eta)\xi \right) \eta^T. \quad (4)$$

Оператор $T_1(\xi, \eta)$ связан с преобразованием в шар специального эллипсоида, описанного вокруг тела, которое получено в результате пересечения шара и двух полупространств, проходящих через центр шара.

Субградиентный метод Поляка с преобразованием пространства имеет следующий вид:

$$x_{k+1} = x_k - h_k B_k \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}, \quad h_k = \frac{m(f(x_k) - f^*)}{\|B_k^T \partial f(x_k)\|}, \quad k = 0, 1, 2, \dots, \quad (5)$$

где матрица $B_0 = I$, а матрица B_{k+1} пересчитывается по следующему правилу:

$$B_{k+1} = \begin{cases} B_k, & \text{если } (\xi, \eta) \geq 0 \\ B_k T^{-1}(\xi, \eta) & \text{иначе} \end{cases},$$

где $\xi = \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}$ и $\eta = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|}$.

Пусть $A_k = B_k^{-1}$, $A_{k+1} = B_{k+1}^{-1}$. Для $\forall x^* \in X^*$ и всех точек итерационного процесса (5) справедливы неравенства

$$\|A_{k+1}(x_{k+1} - x^*)\|^2 \leq \|A_k(x_k - x^*)\|^2 - \left(\frac{m(f(x_k) - f^*)}{\|B_k^T \partial f(x_k)\|} \right)^2, \quad k = 0, 1, \dots \quad (6)$$

Неравенства (6) означают, что в методе (5) преобразование пространства таково, что в каждом очередном преобразованном пространстве переменных гарантируется уменьшение расстояния до множества точек минимума. Благодаря этому для каждой итерации $k > 1$ имеем неравенство

$$\|A_k(x_k - x^*)\|^2 \leq \|x_0 - x^*\|^2 - \sum_{i=0}^{k-1} \left(\frac{m(f(x_i) - f^*)}{\|B_i^T \partial f(x_i)\|} \right)^2.$$

Антиовражная техника в методе (5) направлена на уменьшение степени овражности поверхностей уровня выпуклых функций подобно тому, как это сделано в r -алгоритмах [3]. Детерминант матрицы B_k стремится к нулю, так как, если на k -м шаге реализуется преобразование пространства, то

$$\det(B_{k+1}) = \det(B_k) \det(T^{-1}(\xi, \eta)) = \det(B_k) \sqrt{1 - \cos^2 \varphi},$$

где φ — угол между двумя последовательными субградиентами. Для овражных функций это обеспечивает ускоренную сходимость метода (5) по отношению к методу (2) при произвольной начальной стартовой точке x_0 . Так, например, для овражной функции двух переменных $f(x_1, x_2) = |x_1| + k|x_2|$ метод (5) будет находить точку x^* не более чем за две итерации независимо от значения k .

Отметим, что метод (5) можно назвать ускоренным методом Поляка за счет преобразования пространства переменных которое свойственно r -алгоритмам Шора [3]. Действительно, он обладает двумя характерными чертами r -алгоритмов. Во-первых, движение из точки осуществляется в направлении антисубградиента и, во-вторых, растяжение пространства производится в направлении разности нормированных последовательных субградиентов, если угол между ними тупой. Если нормы субградиентов одинаковы, то это направление будет совпадать с разностью двух последовательных субградиентов, но в отличие от r -алгоритмов второй субградиент получен не согласно шагу наискорейшего спуска, а согласно шагу Поляка в преобразованном пространстве переменных.

Литература

1. Поляк Б.Т., *Введение в оптимизацию*, М.: Наука (1983).
2. Стецюк П.И., “Ортогонализирующие линейные операторы в выпуклом программировании”, *Кибернетика и системный анализ*, No. 3, 97–119 (1997).
3. Шор Н.З., *Методы минимизации недифференцируемых функций и их приложения*, Киев: Наукова думка, (1979).

Ускоренные модификации субградиентного метода Поляка для овражных выпуклых функций

П. И. Стецюк

*Стохастическое программирование и его приложения / П.С. Кнопов,
В.И. Зоркальцев, Я.М. Иванько и др. – Иркутск: Институт систем
энергетики им. Л.А. Мелентьева СО РАН, 2012. – С. 160–184.*

Обсуждаются три субградиентных метода с преобразованием пространства для нахождения точки минимума выпуклой функции при известном минимальном значении функции. В методах используется шаг Поляка в направлении нормированного антисубградиента и гарантируется монотонное уменьшение расстояния до точки минимума в последовательно преобразованных пространствах переменных. Преобразование пространства реализуется с помощью однорангового эллипсоидального оператора и направлено на уменьшение степени овражности поверхностей уровня выпуклых функций.

Ключевые слова. Субградиентный метод, шаг Поляка, шаг Агмона-Моцкина-Шенберга, овражная функция, преобразование пространства.

Данная статья связана с использованием линейных неортогональных преобразований пространства для улучшения обусловленности овражных функций – идеи Н.З. Шора, на которой построены эффективные модификации r -алгоритмов. В ней рассмотрены субградиентные методы с шагом Поляка и преобразованием пространства, обеспечивающие монотонное уменьшение расстояния до точки минимума. Эти методы можно рассматривать как расширение предложенного Б.Т. Поляком в 1969 году субградиентного метода для нахождения точки минимума выпуклой функции при известном оптимальном значении функции. В них длина шага в направлении антисубградиента вычисляется (а не подбирается как в r -алгоритмах из решения одномерной задачи минимизации функции по направлению), причем для ее вычисления требуется лишь знание минимального значения функции.

1. Субградиентный метод Поляка

Пусть $f(x)$ – выпуклая функция векторного аргумента $x \in R^n$ и ее субградиент $\partial f(x)$ удовлетворяет условию:

$$(x - x^*, \partial f(x)) \geq m(f(x) - f^*), \quad \forall x \in R^n, \forall x^* \in X^*, \quad m \geq 1. \quad (1)$$

Здесь R^n – евклидово пространство размерности n со скалярным произведением (x, y) ; X^* – множество точек минимума функции $f(x)$; f^* – минимальное значение функции $f(x)$: $f^* = f(x^*)$, $x^* \in X^*$. Параметр m считается известным, он введен для учета специальных классов выпуклых функций, например, для квадратичной гладкой функции $m = 2$. Значение $m = 1$ обеспечивает выполнение условия (1) для произвольной выпуклой функции.

Пусть известно f^* . Для нахождения точки $x^* \in X^*$ применяется субградиентный метод Поляка [1,2]:

$$x_{k+1} = x_k - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|}, \quad h_k = \frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|}, \quad k = 0, 1, 2, \dots \quad (2)$$

При $m = 1$ метод (2) имеет простой геометрический смысл. Функция $f(x)$ аппроксимируется линейной $\tilde{f}(x) = f(x_k) + (f'(x_k), x - x_k)$ и шаг выбирается так, чтобы эта аппроксимирующая функция стала равной f^* (т. е. $\tilde{f}(x_{k+1}) = f^*$). Для одномерного случая метод Поляка совпадает с методом Ньютона для решения уравнения $f(x) = f^*$.

Величину h_k называют шагом Поляка или шагом Агмона-Моцкина-Шенберга. Этот шаг тесно связан с результатами И.И. Еремина о сходимости итерационных методов аппроксимации неподвижных точек с помощью операторов, обладающих свойством квазисжимаемости (фейеровости) [3]. Впервые такой шаг был использован в [4, 5] в релаксационном (субградиентном) методе для нахождения хотя бы одного решения совместной системы линейных неравенств.

Теорема 1. *Для всех точек, генерируемых методом (2), справедливы неравенства*

$$\|x_{k+1} - x^*\|^2 \leq \|x_k - x^*\|^2 - \left(\frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|} \right)^2, \quad \forall x^* \in X^*, \quad k = 0, 1, \dots, \quad (3)$$

и неравенства

$$(x^* - x_{k+1}, -\partial f(x_k)) \geq 0, \quad \forall x^* \in X^*, \quad k = 0, 1, \dots, \quad (4)$$

Доказательство. Для любого $x^* \in X^*$ и произвольного k ($k \geq 0$) имеем

$$\begin{aligned} \|x_{k+1} - x^*\|^2 &= \left\| x_k - x^* - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|} \right\|^2 = \\ &= \|x_k - x^*\|^2 - 2h_k \frac{(x_k - x^*, \partial f(x_k))}{\|\partial f(x_k)\|} + h_k^2. \end{aligned}$$

Учитывая, что из (1) следует неравенство

$$\frac{(x_k - x^*, \partial f(x_k))}{\|\partial f(x_k)\|} \geq \frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|} = h_k,$$

имеем

$$\|x_{k+1} - x^*\|^2 \leq \|x_k - x^*\|^2 - h_k^2 = \|x_k - x^*\|^2 - \left(\frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|} \right)^2,$$

что дает неравенства (3).

Неравенства (4) следуют из того, что на основании (1) имеем

$$\begin{aligned} (x^* - x_{k+1}, -\partial f(x_k)) &= (x_{k+1} - x^*, \partial f(x_k)) = \\ &= (x_k - x^* - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|}, \partial f(x_k)) = \\ &= (x_k - x^*, \partial f(x_k)) - h_k \|\partial f(x_k)\| = \\ &= (x_k - x^*, \partial f(x_k)) - m(f(x_k) - f^*) \geq 0. \end{aligned}$$

Теорема 1 доказана.

В теореме 1 отражены два центральных свойства шага Поляка. Первое свойство следует из неравенств (4). Оно означает, что шаг h_k определяет величину максимального сдвига в направлении нормированного антисубградиента, при котором угол между антисубградиентом и направлением из точки x_{k+1} на произвольную точку из множества минимумов будет нетупым. Второе свойство следует из неравенств (3) и означает, что если множество X^* состоит из единственной точки x^* , то шаг h_k выбирается таким, чтобы расстояние от точки x_{k+1} к точке минимума x^* было минимальным.

2. Проблема овражности

В [1, 2] показано, что метод (2) сходится со скоростью геометрической прогрессии, и отмечается что во многих практических задачах знаменатель прогрессии может оказаться близким к единице. Для гладкого случая этот факт также хорошо известен (градиентные методы медленно сходятся для функций овражного типа). Тот же эффект имеет место и в негладком случае, например, для функции двух переменных $f(x) = |x_1| + t|x_2|$ метод (2) сходится со скоростью геометрической прогрессии со знаменателем $(1 - \frac{1}{t^2})^{\frac{1}{2}}$, т. е. очень медленно для больших t .

Медленную скорость сходимости метода Поляка проиллюстрируем на примере трех выпуклых функций от двух переменных, первые две функции – негладкие (овражная и существенно овражная), а третья – квадратичная овражная функция. В табл. 1 приведено количество итераций метода (2) для нахождения последовательно уточняемых приближений к единственной точке минимума $x^* = (0, 0)$. Эти приближения заданы десятью последовательно уменьшающимися (на порядок) значениями $\text{epsf} = \varepsilon_f$ (первая колонка в табл. 1). Метод Поляка прекращал работу или на итерации $k = \text{itn}$, для которой $f(x_{\text{itn}}) - f^* \leq \varepsilon_f$, или если превышено максимальное количество итераций, равное 20000.

Колонки itn1 , itn2 и itn3 отвечают овражной кусочно-линейной функции $f_1(x_1, x_2) = |x_1| + t|x_2|$ для значений t равных 3, 9 и 27 соответственно. При расчетах использовалась стартовая точка $x_0 = (1, 1)$ и параметр $m = 1$. Из табл. 1 видим, что количество итераций существенно увеличивается с ростом t . Для того, чтобы найти точку, где значение функции $f_1(x_1, x_2) = |x_1| + 27|x_2|$ отличается от минимального $f_1^* = 0$ не более, чем на $\varepsilon_f = 10^{-10}$, методу Поляка потребовалось 8634 итерации. Такая же точность для функции $f_1(x_1, x_2) = |x_1| + t|x_2|$ при $t = 100$ не обеспечивается даже при ста тысячах итераций метода Поляка.

Для $f_2(x_1, x_2) = \max \{x_1^2 + (2x_2 - 2)^2 - 3, x_1^2 + (x_2 + 1)^2\}$ – существенно овражной кусочно-квадратичной функции зигзагообразная траектория последовательных приближений метода Поляка дана на рис. 1. Здесь метод Поляка находит приближение, где значение функции $f_2(x_1, x_2)$ отличается от минимального $f_2^* = 1$ не более, чем на 0.0001, только за 16004 итераций (см. колонку itn4 в табл. 1).

Для функций $f_1(x_1, x_2)$ и $f_2(x_1, x_2)$ использовался параметр $m = 1$, который можно применять для произвольной выпуклой функции. Для квадратичной функции $f_3(x_1, x_2) = x_1^2 + tx_2^2, t > 0$ использовался параметр $m = 2$. Он в два раза увеличивает длину шага Поляка, соот-

Таблица 1.

Сходимость метода Поляка для двумерных овражных функций

epsf	itn1	itn2	itn3	itn4	itn5	itn6	itn7
1.0e-01	14	119	1080	16	6	6	6
1.0e-02	24	212	1919	162	10	10	10
1.0e-03	34	305	2759	1604	12	12	12
1.0e-04	45	398	3598	16004	16	16	16
1.0e-05	55	492	4437	20000	20	20	20
1.0e-06	65	585	5277	20000	22(42)	22(42)	22(42)
1.0e-07	76	678	6116	20000	26(48)	26(52)	26(52)
1.0e-08	86	771	6955	20000	28(54)	30(56)	30(56)
1.0e-09	96	865	7795	20000	32(62)	32(62)	32(62)
1.0e-10	107	958	8634	20000	36(68)	36(70)	36(70)

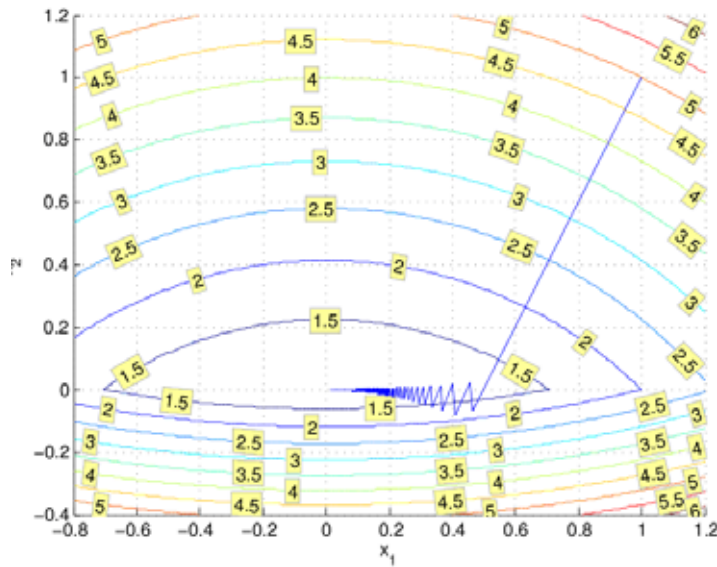


Рис. 1. Траектория метода Поляка для $f_2(x_1, x_2)$ ($x_0 = (1, 1)$)

ветствующего параметру $m = 1$, и обеспечивает более быструю скорость сходимости с различных начальных приближений. Количество итераций метода Поляка со стартовой точки $x_0 = (0, 0)$ приведено в колонках itn5, itn6 и itn7 и соответствует значениям t равным 100, 10000 и 1000000. Количество итераций для всех 10 значений epsf одинаково и не зависит от степени вытянутости поверхности квадратичной функции $f_3(x_1, x_2)$. Отличия наблюдаются лишь при достаточно малых $\varepsilon_f = 10^{-12}, 10^{-14}, 10^{-16}, 10^{-18}, 10^{-20}$, для которых количество итераций приведено в скобках после количества итераций для шести последних значений epsf. Чтобы найти точку минимума квадратичной функции $f_3(x_1, x_2)$ с такой же точностью, как и точку минимума функции $f_1(x_1, x_2)$, нужно взамен ε_f использовать ε_f^2 .

Следовательно, вопрос об ускорении сходимости метода (2) является актуальным и здесь отметим два способа его ускорения. Первый способ состоит в том, чтобы вблизи минимума функцию $f(x)$ аппроксимировать кусочно-линейной функцией, используя информацию из предыдущих итераций. Самый общий метод, реализующий этот способ, предложен в [2] и имеет следующий вид:

$$x_{k+1} = P_{Q_k}(x_k), \quad Q_k = \{x : f(x_i) + (\partial f(x_i), x - x_i) \leq f^*, \quad i \in I_k\}, \quad (2a)$$

где I_k – любое подмножество индексов из $0, 1, \dots, k$, которое обязательно содержит индекс k , P_{Q_k} – оператор проектирования на множество Q_k .

Метод (2a) также сходится со скоростью геометрической прогрессии и не медленнее, чем метод (2), а его частные случаи очень тесно связаны с другими известными методами. Так, если $I_k = \{0, 1, \dots, k\}$, то метод (2a) дает точный минимум для кусочно-линейной функции, причем он идейно близок к методу Келли [6]. Если множество $I_k = \{k-1, k\}$, то проекцию на множество Q_k можно выписать явно. В этом случае метод (2a) близок к методу из [7], который использует всего два вектора и для построения очередного направления спуска использует линейную комбинацию направления субградиента и направления движения на предыдущем шаге (по типу сопряженных градиентов) с тем, чтобы оно составляло более острый угол с направлением на минимум.

Второй способ ускорения метода (2) базируется на идее Н.З.Шора, связанной с использованием линейных неортогональных преобразований пространства для улучшения обусловленности овражных функций. На этой идее построены эффективные модификации r -алгоритмов [8,9]. В статье рассмотрим субградиентные методы с шагом Поляка и преобразованием пространства переменных, которое обеспечивает монотонное уменьшение расстояния до точки минимума и направлено на умень-

шение степени овражности поверхностей уровня выпуклых функций подобно тому, как это сделано в r -алгоритмах Шора.

3. Метод Поляка с преобразованием пространства

Пусть произведена замена переменных $x = By$, где B — неособенная $n \times n$ -матрица (т.е. существует обратная матрица $A = B^{-1}$). Субградиент выпуклой функции $f(x)$ в точке x_k удовлетворяет неравенству

$$f(x) \geq f(x_k) + (\partial f(x_k), x - x_k) \quad \forall x \in R^n,$$

откуда, осуществляя замену переменных $x = By$, получаем

$$\phi(y) \geq \phi(y_k) + (B^T \partial f(x_k), y - y_k) \quad \forall y \in R^n,$$

где $\phi(y) = f(By)$. Вектор $\partial \phi(y_k) = B^T \partial f(x_k)$ удовлетворяет неравенству

$$\phi(y) \geq \phi(y_k) + (\partial \phi(y_k), y - y_k) \quad \forall y \in R^n$$

и является субградиентом выпуклой функции $\phi(y)$ в точке $y_k = Ax_k$ преобразованного пространства переменных $y = Ax$.

Для нахождения точки $x^* \in X^*$ субградиентный метод Поляка с преобразованием пространства (определяется невырожденной матрицей B) имеет следующий вид:

$$x_{k+1} = x_k - h_k \frac{B^T \partial f(x_k)}{\|B^T \partial f(x_k)\|}, \quad h_k = \frac{m(f(x_k) - f^*)}{\|B^T \partial f(x_k)\|}, \quad k = 0, 1, 2, \dots \quad (5)$$

Здесь величина h_k — шаг Поляка (шаг Агмона-Моцкина-Шенберга), но в преобразованном пространстве переменных $y = Ax$. Это следует из того, что в преобразованном пространстве переменных метод (5) записывается как субградиентный процесс

$$y_{k+1} = y_k - h_k \frac{\partial \phi(y_k)}{\|\partial \phi(y_k)\|}, \quad h_k = \frac{m(\phi(y_k) - \phi^*)}{\|\partial \phi(y_k)\|}, \quad k = 0, 1, 2, \dots \quad (6)$$

Шаг Поляка в преобразованном пространстве переменных обладает такими же свойствами как и шаг Поляка в исходном пространстве. Это следует из априорного знания минимального значения функции и связанного с ним неравенства (1).

Теорема 2. Пусть $A = B^{-1}$. Для всех точек, генерируемых методом (5), справедливы неравенства

$$\begin{aligned} \|A(x_{k+1} - x^*)\|^2 &\leq \\ &\leq \|A(x_k - x^*)\|^2 - \left(\frac{m(f(x_k) - f^*)}{\|B^T \partial f(x_k)\|} \right)^2, \forall x^* \in X^*, k = 0, 1, \dots, \end{aligned} \quad (7)$$

и неравенства

$$(A(x^* - x_{k+1}), -B^T \partial f(x_k)) \geq 0, \quad \forall x^* \in X^*, k = 0, 1, \dots \quad (8)$$

Доказательство теоремы 2 аналогично доказательству теоремы 1.

Очевидно, что если матрицу B выбрать такой, чтобы вытянутые поверхности уровня овражной функции в преобразованном пространстве переменных становились менее вытянутыми, то субградиентный метод Поляка с преобразованием пространства (5) окажется эффективнее, чем субградиентный метод Поляка без преобразования пространства (2). Это согласуется с количеством итераций метода (5) для нахождения десяти последовательно уточняемых приближений к точке минимума функции $f_1(x_1, x_2) = |x_1| + 10|x_2|$ для шести различных матриц B , каждой из которых соответствует свой столбец `itn` в табл. 2. Матрицы B получены в результате растяжения пространства переменных в направлении x_2 с коэффициентами растяжения $\alpha = 1; 1.5; 2; 3; 4; 5$. Матрица B имеет вид

$$B = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1}{\alpha} \end{pmatrix}$$

и если $\alpha = 1$, то она совпадает с единичной матрицей. Это соответствует случаю, когда метод Поляка с преобразованием пространства тождественно равен методу (2) – методу Поляка без преобразования пространства. В табл. 2 ему отвечает столбец `itn1`, где приведено число итераций метода Поляка (2) для нахождения последовательно уточняемых приближений к единственной точке минимума $x^* = (0, 0)$.

Из табл. 2 видим, что количество итераций метода Поляка с преобразованием пространства монотонно уменьшается по мере того, как уменьшается степень овражности функции $\phi_1(y_1, y_2) = |y_1| + \frac{10}{\alpha}|y_2|$ в преобразованном пространстве переменных (соответствует увеличению коэффициента растяжения α).

Таблица 2.

Сходимость метода Поляка с преобразованием пространства для кусочно-линейной функции $f_1(x_1, x_2) = |x_1| + 10|x_2|$

epsf	itn1	itn2	itn3	itn4	itn5	itn6
1.0e-01	147	63	33	6	10	9
1.0e-02	262	114	62	19	17	13
1.0e-03	377	165	91	31	24	18
1.0e-04	492	216	119	44	31	22
1.0e-05	607	268	148	57	38	27
1.0e-06	722	319	177	70	45	31
1.0e-07	837	370	206	82	53	36
1.0e-08	952	421	234	95	60	40
1.0e-09	1000	472	263	108	67	45
1.0e-10	1000	523	292	121	74	49

Если для овражной негладкой функции можно подобрать такую матрицу B , чтобы ускорить сходимость метода Поляка, то для существенно овражной функции это сделать практически невозможно. Об этом говорят результаты о числе итераций метода Поляка с преобразованием пространства для тех же шести матриц B , что и раньше, но для существенно овражной функции

$$f_2(x_1, x_2) = \max \{x_1^2 + (2x_2 - 2)^2 - 3, x_1^2 + (x_2 + 1)^2\}.$$

Из табл. 3 видим, что монотонного уменьшения количества итераций по мере увеличения α — коэффициента растяжения пространства в направлении x_2 здесь не наблюдается. Зато наблюдается некоторый разрыв, который происходит при коэффициенте $\alpha = 2$ (соответствует столбцу itn3). При всех остальных коэффициентах α метод требует максимального количества итераций для достижения точности $\varepsilon_f = 10^{-6}$.

Ниже рассмотрим две модификации метода Поляка с преобразованием пространства, где матрица B будет изменяться, если обнаружена возможность зигзагообразного движения вдоль русла оврага. В этих модификациях будет использована одноранговая коррекция несимметричной матрицы B только на тех итерациях, когда угол между двумя последовательными субградиентами тупой.

Сходимость метода Поляка с преобразованием пространства для кусочно-квадратичной

$$f_2(x_1, x_2) = \max \{x_1^2 + (2x_2 - 2)^2 - 3, x_1^2 + (x_2 + 1)^2\}$$

epsf	itn1	itn2	itn3	itn4	itn5	itn6
1.0e-01	16	4	4	5	7	8
1.0e-02	162	37	4	6	7	9
1.0e-03	1604	679	5	6	8	9
1.0e-04	10000	7079	5	6	9	446
1.0e-05	10000	10000	6	8	8061	6206
1.0e-06	10000	10000	6	10000	10000	10000
1.0e-07	10000	10000	6	10000	10000	10000
1.0e-08	10000	10000	6	10000	10000	10000
1.0e-09	10000	10000	6	10000	10000	10000
1.0e-10	10000	10000	6	10000	10000	10000

4. Одноранговый эллипсоидальный оператор

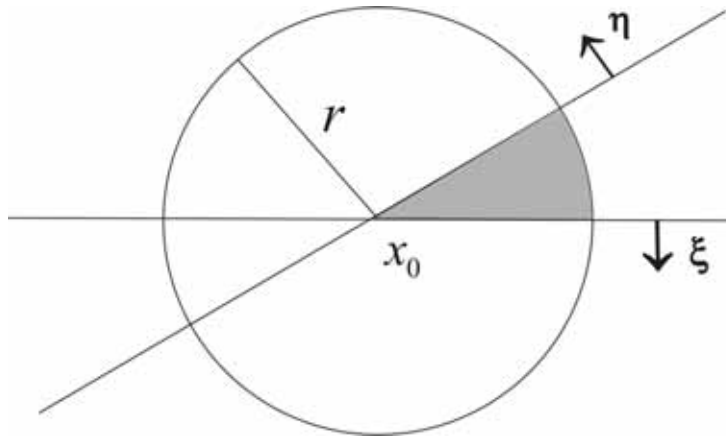
Тупой угол между двумя нормированными векторами ξ и η из R^n можно преобразовать в прямой с помощью линейного оператора из R^n в R^n , который в матричной форме представим

$$T_1(\xi, \eta) = I - \frac{1}{1 - (\xi, \eta)^2} \left(\left(1 - \sqrt{1 - (\xi, \eta)^2} \right) \eta - (\xi, \eta) \xi \right) \eta^T. \quad (9)$$

Здесь $\xi, \eta \in R^n$ – векторы, такие что $\|\xi\| = 1$, $\|\eta\| = 1$ и их скалярное произведение удовлетворяет условию $(\xi, \eta)^2 \neq 1$, I – единичная матрица размера $n \times n$. Для оператора $T_1(\xi, \eta)$ существует обратный $T_1^{-1}(\xi, \eta)$

$$T_1^{-1}(\xi, \eta) = I + \frac{1}{\sqrt{1 - (\xi, \eta)^2}} \left(\left(1 - \sqrt{1 - (\xi, \eta)^2} \right) \eta - (\xi, \eta) \xi \right) \eta^T \quad (10)$$

Оператор $T_1(\xi, \eta)$ введен в [10] под названием "одноранговый эллипсоидальный оператор". Он связан с преобразованием в шар специального эллипсоида, описанного вокруг тела W (см. рис. 2), которое получено в результате пересечения шара и двух полупространств, проходящих через центр шара. В случае тупого угла между нормальными полупространств этот эллипсоид содержит тело W и является минимальным по объему в рамках семейства эллипсоидов, центр которых совпадает с центром шара (см. рис. 3).

Рис. 2. Тело W (проекция на плоскость)

Минимальный объем эллипсоида меньше, чем объем шара, и это уменьшение определяется величиной $\sqrt{1 - (\xi, \eta)^2}$. Преобразование минимального по объему эллипсоида в шар требует растяжения пространства в направлении $\frac{\xi - \eta}{\|\xi - \eta\|}$ с коэффициентом $\alpha_1 = \frac{1}{\sqrt{1 + (\xi, \eta)}}$ и "сжатия" пространства в ортогональном направлении $\frac{\xi + \eta}{\|\xi + \eta\|}$ с коэффициентом $\alpha_2 = \frac{1}{\sqrt{1 - (\xi, \eta)}}$. В преобразованном пространстве эллипсоид станет шаром, а образы векторов ξ и η будут ортогональными (см. рис. 4).

5. Ускоренный субградиентный метод Поляка

Медленную сходимость метода Поляка для овражных функций определяет угол между двумя последовательными субградиентами: $\partial f(x_k)$ и $\partial f(x_{k+1})$. Чем ближе этот угол к 180 градусам, тем более медленной будет скорость сходимости. Тупой угол между векторами $\partial f(x_k)$ и $\partial f(x_{k+1})$ можно преобразовать в прямой с помощью "однорангового эллипсоидального оператора" [10]. Если на каждой итерации избавляться от тупого угла между последовательными субградиентами в преобразованном пространстве переменных, то для овражных функций следует ожидать повышения скорости сходимости метода. Этот принцип реализован в изложенной ниже модификации метода Поляка.

Ускоренный субградиентный метод Поляка имеет следующий вид:

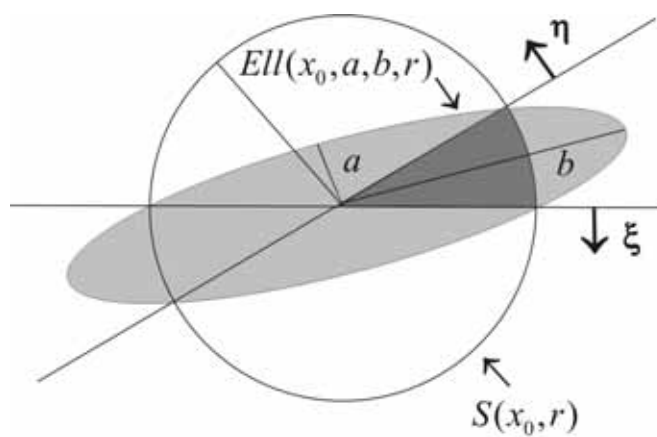


Рис. 3. Специальный эллипсоид, содержащий тело W

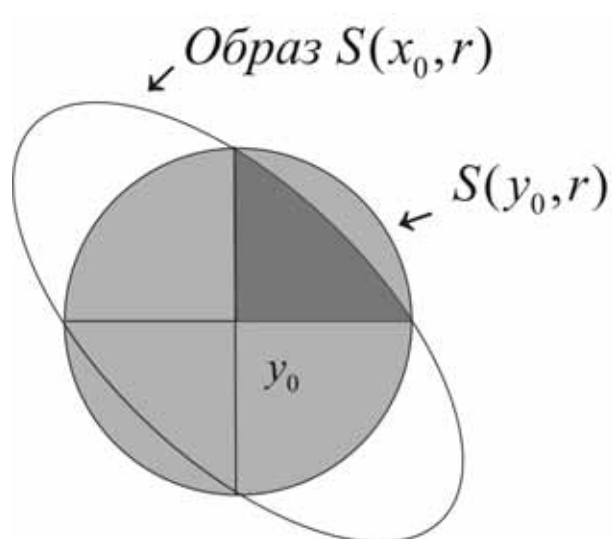


Рис. 4. Оптимальный эллипсоид после преобразования

$$x_{k+1} = x_k - h_k B_k \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}, \quad h_k = \frac{m(f(x_k) - f^*)}{\|B_k^T \partial f(x_k)\|}, \quad k = 0, 1, 2, \dots, \quad (11)$$

где матрица $B_0 = I$, а матрица B_{k+1} размера $n \times n$ вычисляется по правилу:

$$B_{k+1} = \begin{cases} B_k, & \mu_k \geq 0 \\ B_k + (B_k \eta) \xi_{k+1}^T, & \mu_k < 0 \end{cases} \quad (12)$$

где

$$\mu_k = (\xi_k, \xi_{k+1}), \quad \xi_k = \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}, \quad \xi_{k+1} = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|},$$

$$\eta = \left(\frac{1}{\sqrt{1 - \mu_k^2}} - 1 \right) \xi_{k+1} - \frac{\mu_k}{\sqrt{1 - \mu_k^2}} \xi_k.$$

Метод (11), (12) естественно назвать ускоренным методом Поляка за счет антиогражного приема, подобного тому, который использован в r -алгоритмах Шора [8]. Действительно, на k -й итерации растяжение пространства производится в направлении разности нормированных последовательных субградиентов в преобразованном пространстве переменных: $y = A_k x = B_k^{-1} x$, где B_k – невырожденная матрица размерности $n \times n$. Если нормы субградиентов одинаковы, то это направление совпадает с разностью двух последовательных субградиентов, по которой реализуется растяжение пространства в r -алгоритмах. Отличие состоит в том, что в преобразованном пространстве переменных для r -алгоритмов второй субградиент определяется согласно шагу наискорейшего спуска в направлении антисубградиента, а в методе (9), (10) – согласно шагу Поляка.

Теорема 3. Пусть $A_k = B_k^{-1}$, $A_{k+1} = B_{k+1}^{-1}$. Для всех точек, генерируемых методом (11), (12), справедливы неравенства

$$\|A_{k+1}(x_{k+1} - x^*)\|^2 \leq \|A_k(x_k - x^*)\|^2 - \left(\frac{m(f(x_k) - f^*)}{\|B_k^T \partial f(x_k)\|} \right)^2, \quad \forall x^* \in X^*, k = 0, 1, \dots \quad (13)$$

и неравенства

$$(A_k(x^* - x_{k+1}), -B_k^T \partial f(x_k)) \geq 0, \quad \forall x^* \in X^*, k = 0, 1, \dots \quad (14)$$

Доказательство теоремы 3 аналогично доказательству теоремы 2 [10].

Преобразование пространства в методе (11), (12) направлено на уменьшение степени вытянутости поверхностей уровня овражных выпуклых функций. Неравенства (13) означают, что в каждом очередном преобразованном пространстве переменных гарантируется уменьшение расстояния до множества точек минимума. Поэтому для каждой итерации $k > 1$ имеет место неравенство

$$\|A_k(x_k - x^*)\|^2 \leq \|x_0 - x^*\|^2 - \sum_{i=0}^{k-1} \left(\frac{m(f(x_i) - f^*)}{\|B_i^T \partial f(x_i)\|} \right)^2.$$

Для овражных функций детерминант матрицы B_k уменьшается, а следовательно, уменьшается объем эллипсоида, локализирующего точку x^* . Действительно, если на k -м шаге реализуется преобразование пространства, то

$$\begin{aligned} \det(B_{k+1}) &= \det(B_k) \det(I + \eta \xi_{k+1}^T) = \\ &= \det(B_k) \sqrt{1 - \mu_k^2} = \det(B_k) \sqrt{1 - \cos^2 \phi_k}, \end{aligned}$$

где ϕ_k – тупой угол между двумя последовательными субградиентами.

Для овражных негладких функций метод (11), (12) это обеспечивает ускоренную сходимость по отношению к методу (2). Так, например, для кусочно-линейной функции двух переменных $f_1(x_1, x_2) = |x_1| + t|x_2|$ при любом значении параметра $t > 1$ и произвольной стартовой точке x_0 метод (11), (12) находит точку минимума $x^* = (0, 0)$ не более, чем за три итерации. Метод (2) сходится к x^* со скоростью геометрической прогрессии со знаменателем $\sqrt{1 - 1/t^2}$ и требует существенного количества итераций даже при сравнительно небольших значениях t . Траектория ускоренного метода Поляка для кусочно-квадратичной функции $f_2(x_1, x_2)$ дана на рис. 5, здесь метод находит точку, где $f_2(x) < 1 + 10^{-5}$ всего за 16 итераций, а где $f_2(x) < 1 + 10^{-10}$ – за 31 итерацию.

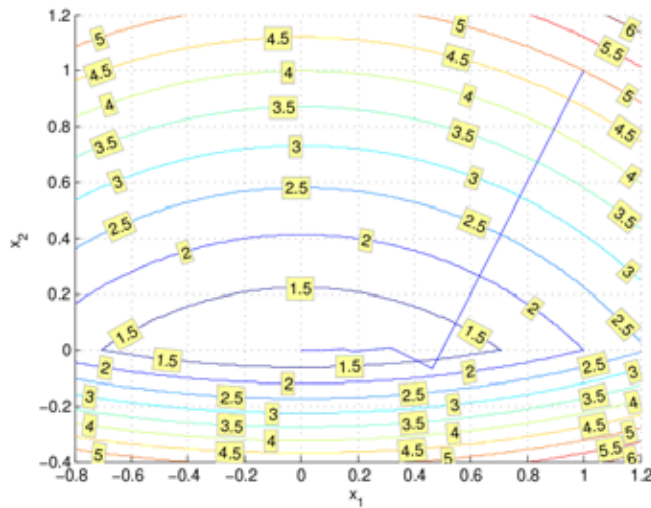


Рис. 5. Траектория ускоренного метода Поляка для $f_2(x_1, x_2)(x_0 = (1, 1))$

6. Метод msg2p

Метод (11), (12) можно усилить более радикальным уменьшением объема эллипсоида, локализующего точку x^* . Если на k -итерации была реализована операция преобразования пространства, то в преобразованном пространстве $Y = A_{k+1}X = B_{k+1}^{-1}X$ для функции $\phi_{k+1}(y) = f(B_{k+1}y)$ субградиенты будут ортогональны и в силу шага Поляка (шага Агмона-Мощкина-Шенберга) полупространства, определяемые ими в точке $y_{k+2} = A_{k+1}x_{k+2}$, не отсекают точку $y^* = A_{k+1}x^*$. Очевидно, что если вычисленный в точке y_{k+2} субградиент образует тупой угол с двумя предыдущими, то можно уменьшить объем области локализации x^* . Для этого достаточно выбрать в качестве векторов, определяющих оператор $T_1(\xi, \eta)$, субградиент в точке y_{k+2} и вектор, являющийся выпуклой комбинацией первых двух, так, чтобы угол между ними был максимально тупым. Для очередной итерации (если реализуется преобразование пространства) эта ситуация повторяется, только в качестве первого вектора уже будет использоваться не субградиент, а вектор, являющийся выпуклой комбинацией двух предыдущих субградиентов в очередном преобразованном пространстве. Это позволяет замкнуть

цикл вычислений и построить конструктивное правило использования в качестве одного из векторов, определяющих оператор $T_1(\xi, \eta)$, агрегированного вектора.

Метод, построенный на этом принципе, условимся называть `amsg2p` [11, 12]. В его названии "ams" указывает на способ регулировки шага в направлении нормированного антисубградиента, а "g2p" указывает, что `ams`-шаг используется в пространстве переменных, преобразованном с помощью двух последних субградиентов (`g2`) и агрегатного вектора (`p`). Преобразование пространства реализуется с помощью однорангового эллипсоидального оператора и только на тех итерациях метода, когда тупым является хотя бы один из углов – угол между двумя последовательными субградиентами, либо угол между последним субградиентом и агрегатным вектором, который является выпуклой комбинацией вычисленных на предыдущих итерациях субградиентов.

В [13] метод `amsg2p` расширен на случай произвольного значения f_{min} и позволяет либо найти такую точку, где значение выпуклой функции $f(x)$ меньше или равно $f_{min} + \varepsilon$, либо гарантирует достаточное условие того, что точки, где значение $f(x)$ равно f_{min} , в шаре заданного радиуса не существует. В первом случае метод `amsg2p` находит точку $x_\varepsilon^* \in \{x : f(x) - f_{min} \leq \varepsilon\}$ и соответствующий ей номер итерации k_ε^* , а во втором – останавливается с сообщением "точки не существует". Метод `amsg2p` состоит в следующем.

На итерации $k = 0$ заданы: начальное приближение $x_0 \in R^n$; начальный радиус r_0 такой, что $\|x_0 - x^*\| \leq r_0$; достаточно малое $\varepsilon > 0$. Вычислим $f(x_0)$ и $\partial f(x_0)$. Если $f(x_0) - f_{min} \leq \varepsilon$, то $x_\varepsilon^* = x_0$, $k_\varepsilon^* = 0$ и окончание алгоритма. Иначе положим $h_0 = \frac{m(f(x_0) - f_{min})}{\|\partial f(x_0)\|}$, $\xi_0 = \frac{\partial f(x_0)}{\|\partial f(x_0)\|} \in R^n$, $p_0 = 0 \in R^n$, $B_0 = I$ – единичная $n \times n$ -матрица. Перейдем к следующей итерации.

Пусть на k -й итерации получены $x_k \in R^n$, $h_k, r_k, \xi_k \in R^n$, $p_k \in R^n$, B_k – матрица $n \times n$. Для $(k + 1)$ -й итерации выполним пп. 1-5.

- 1) Вычислим $t_k = h_k / r_k$. Если $t_k > 1$, то "точки не существует" и окончание алгоритма. Иначе положим $r_{k+1} = r_k \sqrt{1 - t_k^2}$ и вычислим очередное приближение

$$x_{k+1} = x_k - h_k B_k \xi_k.$$

- 2) Вычислим $f(x_{k+1})$ и $\partial f(x_{k+1})$. Если $f(x_{k+1}) - f_{min} \leq \varepsilon$, то $x_\varepsilon^* = x_{k+1}$, $k_\varepsilon^* = k + 1$ и окончание алгоритма. Иначе положим

$$\xi_{k+1} = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|}, \quad h_{k+1} = \frac{m(f(x_{k+1}) - f_{min})}{\|B_k^T \partial f(x_{k+1})\|}.$$

3) Вычислим $\lambda_1 = -p_k^T \xi_{k+1}$ и $\lambda_2 = -\xi_k^T \xi_{k+1}$. Положим

$$p_{k+1} = \begin{cases} \frac{\lambda_1}{\sqrt{\lambda_1^2 + \lambda_2^2}} p_k + \frac{\lambda_2}{\sqrt{\lambda_1^2 + \lambda_2^2}} \xi_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 > 0, \\ p_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 \leq 0, \\ \xi_k, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 > 0, \\ 0, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 \leq 0. \end{cases}$$

4) Вычислим $\mu_k = p_{k+1}^T \xi_{k+1}$. Если $-1 < \mu_k < 0$, то вычислим

$$B_{k+1} = B_k + (B_k \eta) \xi_{k+1}^T, \text{ где } \eta = \left(\frac{1}{\sqrt{1 - \mu_k^2}} - 1 \right) \xi_{k+1} - \frac{\mu_k}{\sqrt{1 - \mu_k^2}} p_{k+1},$$

и пересчитаем

$$h_{k+1} = \frac{h_{k+1}}{\sqrt{1 - \mu_k^2}}, \quad p_{k+1} = \frac{1}{\sqrt{1 - \mu_k^2}} (p_{k+1} - \mu_k \xi_{k+1}).$$

Иначе положим $B_{k+1} = B_k$ и $p_{k+1} = 0$.

1) Перейдем к следующей итерации с x_{k+1} , h_{k+1} , r_{k+1} , ξ_{k+1} , p_{k+1} , B_{k+1} .

Теорема 4. Пусть $A_k = B_k^{-1}$, $A_{k+1} = B_{k+1}^{-1}$. Если $f_{min} \geq f^*$ и $X^* = x^*$, то для всех точек, генерируемых методом атсг2р, справедливы неравенства

$$\|A_{k+1}(x_{k+1} - x^*)\|^2 \leq \|A_k(x_k - x^*)\|^2 - \left(\frac{m(f(x_k) - f_{min})}{\|B_k^T \partial f(x_k)\|} \right)^2, \quad k = 0, 1, \dots, \quad (15)$$

и неравенства

$$(A_k(x^* - x_{k+1}), -B_k^T \partial f(x_k)) \geq 0, \quad \forall x^* \in X^*, \quad k = 0, 1, \dots \quad (16)$$

Таблица 4.

Сходимость amsg2p для квадратичных функций, $n = 200$					
eps	itn(Q1)	itn(Q2)	itn(Q3)	itn(Q4)	itn(Q5)
1.00E-003	11	36	84	361	773
1.00E-005	15	46	99	405	826
1.00E-007	18	56	113	430	868
1.00E-009	22	65	128	461	916
1.00E-011	25	73	142	493	947
1.00E-013	29	81	154	517	979
1.00E-015	32	89	167	541	1006
1.00E-017	35	96	180	560	1027
1.00E-019	39	102	189	574	1042
1.00E-020	41	105	196	585	1048

Теорема 4 означает, что в каждом очередном преобразованном пространстве переменных расстояние до точки минимума уменьшается. Благодаря этому для каждой итерации $k > 1$ имеем неравенство

$$\|A_k(x_k - x^*)\|^2 \leq \|x_0 - x^*\|^2 - \sum_{i=0}^{k-1} \left(\frac{m(f(x_i) - f_{min})}{\|B_i^T \partial f(x_i)\|} \right)^2 = r_0^2 - \sum_{i=0}^{k-1} h_i^2 = r_k^2,$$

с помощью которого обеспечивается достаточное условие отсутствия точки x_ε^* при $f_{min} < f^* - \varepsilon$ (реализовано в п. 1 метода **amsg2p**).

Антиовражная техника в методе **amsg2p** направлена на уменьшение степени овражности поверхностей уровня выпуклых функций подобно тому, как это сделано в r -алгоритмах [8]. Детерминант матрицы B_k уменьшается, так как, если на k -м шаге реализуется преобразование пространства, то $\det(B_{k+1}) = \det(B_k) \det(I + \eta \xi_{k+1}^T) = \det(B_k) \sqrt{1 - \mu_k^2}$. Для овражных функций это обеспечивает ускоренную сходимость метода **amsg2p** при произвольной начальной стартовой точке x_0 и достаточно малых значениях параметра ε ($\varepsilon \sim 10^{-10} - 10^{-14}$).

В табл. 4 приведены результаты вычислительных экспериментов для квадратичных функций от $n = 200$ переменных с различной степенью овражности (рассматривалась функция $f(x) = \sum_{i=1}^n q^{i-1} x_i^2$, а степенью овражности считалась величина $Q = q^{n-1}$).

При расчетах использовался параметр $m = 2$, $f_{min} = f^* = 0$, $x_0 = (1, \dots, 1)^T$. Для ряда стремящихся к нулю ε в таблице даны затраты в числе итераций, которые требуются при степенях овражности $Q1 = 10$, $Q2 = 100$, $Q3 = 1000$, $Q4 = 10^6$, $Q5 = 10^9$.

Метод `amsg2p` можно использовать для того, чтобы достаточно точно найти приближение к единственной точке минимума существенно овражных функций. Проиллюстрируем это на примере известной тестовой задачи `maxquad` [14], которая связана с минимизацией существенно овражной выпуклой кусочно-квадратичной функции $f(x) = \max_{1 \leq k \leq 5} \phi_k(x)$, $x \in R^{10}$. Здесь $\phi_k(x) = x^T H_k x - b_k^T x$, H_k – симметрические 10×10 -матрицы, такие что $H_{kij} = e^{i/j} \cos(ij) \sin k$, если $i < j$, и $H_{kii} = i |\sin k| / 10 + \sum_{j \neq i} |H_{kij}|$, а компоненты векторов b_k определяются $b_{ki} = e^{i/k} \sin(ik)$.

Затраты метода `amsg2p` для нахождения в задаче `maxquad` единственного решения с достаточно высокой точностью (до 14-ти значащих цифр) позволяет оценить приведенный ниже фрагмент численных расчетов с одноименной программой `amsg2p` на языке `octave` [15].

```
Maxquad: f(x0)    5.3370664293114e+003
           fmin = -8.4140833459641e-001
...epsf...      ....f(itn)..... .itn.
1.0e-001  -7.7355266120112e-001    17
1.0e-003  -8.4084776169123e-001    29
1.0e-004  -8.4132394277880e-001    35
1.0e-005  -8.4140078034524e-001    41
1.0e-006  -8.4140807664455e-001    49
1.0e-011  -8.4140833458913e-001    94
1.0e-012  -8.4140833459555e-001   101
1.0e-013  -8.4140833459633e-001   110
1.0e-014  -8.4140833459640e-001   116
1.0e-015  -8.4140833459641e-001   122
```

Здесь $m = 1$, $x_0 = (1, \dots, 1)^T$ и $\varepsilon = \text{eps} f$.

7. Вычислительные эксперименты

Теорема 4 обеспечивает обоснование сходимости метода `amsg2p` аналогично тому как теорема 3 – сходимость метода (11), (12). Но более сильное уменьшение объема эллипсоида, локализирующего точку x^* , обеспечивает для овражных функций его ускоренную сходимость по сравнению с методом (11), (12). По количеству итераций метод `amsg2p` сравним

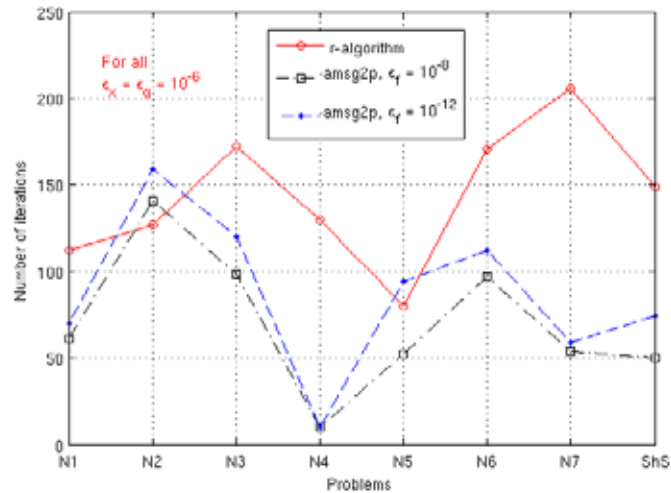


Рис. 6. Сравнение msg2p и r -алгоритма для восьми функций [11].

с r -алгоритмом, а в ряде случаев и превосходит его. Это подтверждают результаты тестовых экспериментов из [11], которые приведены на рис. 6. Рассматривались 7 известных тестовых задач безусловной минимизации гладких и негладких выпуклых функций [16] (стр. 279–282) и кусочно-квадратичная функция [8] (стр. 176). Количество переменных в тестовых примерах было от 5 до 50. Все примеры решались r -алгоритмом с критериями останова $\varepsilon_x = 10^{-6}$ и $\varepsilon_g = 10^{-6}$ и методом msg2p при достаточно малых $\varepsilon_f = 10^{-8}$ и $\varepsilon_f = 10^{-12}$. Число затраченных методами итераций дано на рис. 4. Из него видим, что только в одном случае метод msg2p уступил r -алгоритму (пример № 2).

На рис. 7 приведены количества итераций метода msg2p и программной реализации r -алгоритма, которую выполнил Д.Л. Крошко, для пяти тестовых примеров из [16] (их номера находятся под столбцами диаграммы). Из диаграммы видим, что для четырех тестовых примеров из пяти, выигрыш msg2p по количеству итераций составляет более чем в два раза.

Вычислительную эффективность метода (11), (12) и метода msg2p по отношению к методу Поляка приведем на примере кусочно-линейной и квадратичной функций от 20 переменных из работы [17]. Основным

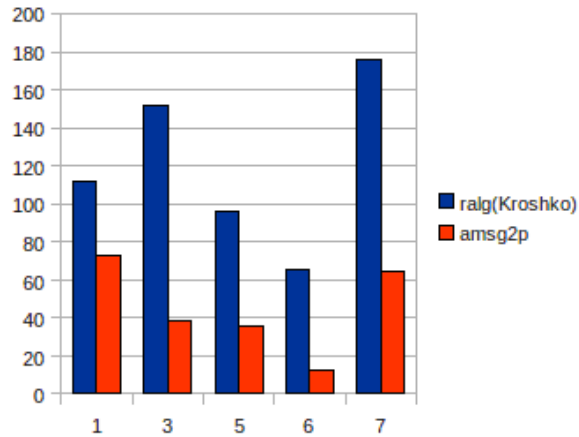


Рис. 7. Сравнение `amsq2p` и r -алгоритма в реализации Д.Л. Крошко для пяти тестовых примеров из монографии [16], с. 279–282

показателем эффективности методов будем считать не время вычислений, а число итераций, т.е. число вычислений $f(x)$ и $\partial f(x)$. Рассматривались кусочно-линейная функция $f(x) = \sum_{i=1}^{20} (1 + 1/4)^{i-1} |x_i|$ (обозначена *Sabs*(1.25)) и квадратичная функция $f(x) = \sum_{i=1}^{20} (1 + 1/2)^{i-1} x_i^2$ (обозначена *Squad*(1.5)). Для функции *Sabs*(1.25) использован параметр $m = 1$, а для функции *Squad*(1.5) – параметр $m = 2$. Количество итераций для всех трех методов (*polyak*, *amsq2* и *amsq2p*) при разных значениях ε приведены в табл. 5, где "прочерк" означает, что метод не решил задачу за 10000 итераций.

Из табл. 5 видим, что методы (11), (12) и *amsq2p* намного эффективнее, чем метод Поляка без преобразования пространства. Число итераций для них увеличивается слабо при значительном уменьшении ε . Это подтверждает, что преобразования пространства переменных, направленные на уменьшение степени вытянутости поверхностей уровня выпуклых функций, способны для овражных функций значительно ускорить сходимость субградиентных методов с шагом Поляка.

Таблица 5. Вычислительные эксперименты из [17]

ε	sabs(1.25), $n=20$, $m=1$			squad(1.5), $n=20$, $m=2$		
	polyak	amsg2	amsg2p	polyak	amsg2	amsg2p
101	289	48	20	601	32	15
100	1163	84	31	1205	36	19
10^{-1}	2737	102	42	2047	46	20
10^{-2}	4306	108	48	3079	51	23
10^{-3}	5869	113	55	4237	56	25
10^{-4}	7425	119	68	5463	58	25
10^{-5}	8943	161	78	6719	61	25
10^{-6}	—	197	95	7985	65	26
10^{-8}	—	214	107	—	68	27
10^{-10}	—	228	119	—	71	32
$F(x_0)$	5567.1151			2194649.4419		

Заключение

Несмотря на то, что r -алгоритмы используются уже 40 лет, проблема обоснования их сходимости для всего класса выпуклых функций остается открытой и в настоящее время. Еще в 1982 г. Н.З. Шор и В.И. Гершович в работе [18] отметили: "Теория всего класса алгоритмов с растяжением пространства далека от совершенства. Нам кажется достаточно реалистичной целью — построение такого алгоритма, который по своей практической эффективности не уступал бы r -алгоритму и был столь же хорошо обоснован, как метод эллипсоидов". Шагом в этом направлении можно считать алгоритм (11), (12), где для преобразования специального эллипсоида в шар используется антивражный прием, близкий к тому, который имеет место в r -алгоритмах. Однако, здесь растяжение пространства реализуется в направлении разности двух нормированных субградиентов, и близким к направлению разности двух субградиентов оно будет только тогда, когда нормы субградиентов близки.

В заключение отметим, что ускоренные варианты субградиентных методов на основе однорангового эллипсоидального оператора можно построить и для других способов регулировки шага. Замечательной чертой таких методов есть полная определенность в параметрах преобразования пространства.

Литература

1. Поляк Б.Т. Введение в оптимизацию. – М.: Наука, 1983. – 384 с.
2. Поляк Б.Т. Минимизация негладких функционалов // Журн. вычислит. математики и матем. физики. – 1969. – Т.9, № 3. – С. 507–521.
3. Васин В.В., Еремин И.И. Операторы и итерационные процессы фейеровского типа (теория и приложения). – Москва; Ижевск: Институт компьютерных исследований, НИЦ «Регулярная и хаотическая динамика», 2005. – 200 с.
4. Agmon S. The relaxation method for linear inequalities // Canadian Journal of Mathematics. – 1954. – 6. – P. 382–392.
5. Motzkin T., Schoenberg I.J. The relaxation method for linear inequalities // Canadian Journal of Mathematics. – 1954. – 6. – P. 393–404.
6. Kelley J. E.. The cutting plane method for solving convex programs. J. Soc. for Industr. and Appl. Math., 1960, 8, № 4, 703–712.
7. Camerini P., Fratta L., Maffioli F. On improving relaxation methods by modified gradient techniques // Math. Program. – 1975. – Study 3. – P. 26–34.
8. Шор Н.З. Методы минимизации недифференцируемых функций и их приложения. – Киев: Наук. думка, 1979. – 199 с.
9. Шор Н.З., Журбенко Н.Г. Метод минимизации, использующий операцию растяжения пространства в направлении разности двух последовательных градиентов // Кибернетика. – 1971. – № 3. – С.51–59.
10. Стецюк П.И. Ортогонализирующие линейные операторы в выпуклом программировании (Часть I) // Кибернетика и системный анализ. – 1997. – № 3. – С. 97–119.
11. Стецюк П. И. Субградиентные методы переменной метрики, использующие шаг Агмона-Моткина и одноранговый эллипсоидальный оператор // Труды АТИК – 2007–2008. Кишинэу: Эврика. – 2009. – Т. I (XII). – С. 16–25.
12. Стецюк П.И. Ускоренные по Шору модификации субградиентного метода Поляка // Математическое моделирование, оптимизация

и информационные технологии: материалы 3-й Междунар. науч. конф. (Кишинэу, 19-23 марта 2012 г.) : Кишинэу: Эврика, 2012. – С. 509–519.

13. Стецюк П.И. Релаксационный субградиентный метод минимизации овражных выпуклых функций // Проблемы теоретической кибернетики. Материалы XVI Международной конференции (Нижний Новгород, 20-25 июня 2011 г.) / Под ред. Ю.И.Журавлева. Нижний Новгород: Изд-во Нижегородского госуниверситета, 2011. – С. 449–453.
14. Lemarechal C. Numerical experiments in nonsmooth optimization // In: Progress in nondifferentiable optimization /Ed. E. A. Nurminski. CP-82-58. – Laxenburg: International Institute for Applied System Analysis, 1982. – P. 61–68.
15. Octave [Электронный ресурс]: <http://www.octave.org/> – Режим доступа: свободный.
16. Ржевский С.В. Монотонные методы выпуклого программирования. – Киев: Наукова думка, 1993. – 324 с.
17. Журбенко М.Г., Стецюк П.І. Субградієнтні методи змінної метрики для розв'язування яружних задач оптимізації. – Київ, 2009. – 27 с. – (Препр. / НАН України. Ін-т кібернетики імені В.М.Глушкова; 2009-3)
18. Гершович В.И., Шор Н.З. Метод эллипсоидов, его обобщения и приложения // Кибернетика. – 1982. – № 5. – С. 61–69.

Ускоренный субградиентный метод Поляка

П. И. Стецюк, Л. Б. Кошлай

*Моделирование, управление и устойчивость (MCS-2012): между-
конф.; Севастополь, 10–14 сентября 2012 г. / отв. ред.*

*О.В. Анашкин; Таврический нац. ун-т имени В.И. Вернадского. –
Симферополь: ДИАЙПИ, 2012. – С. 186–187.*

В докладе рассмотрена модификация метода Поляка для нахождения точки минимума выпуклой функции. В ней используется линейное преобразование пространства переменных, существенно сокращающее количество итераций в случае гладких и негладких овражных функций.

Пусть $f(x)$ – выпуклая функция векторного аргумента $x \in R^n$ и ее субградиент $\partial f(x)$ удовлетворяет условию:

$$(x - x^*, \partial f(x)) \geq m(f(x) - f^*), \quad \text{где } m \geq 1,$$

для $\forall x \in R^n$ и $\forall x^* \in X^*$. Здесь R^n – евклидово пространство размерности n со скалярным произведением (x, y) ; X^* – множество точек минимума функции $f(x)$; f^* – минимальное значение функции $f(x)$: $f^* = f(x^*)$, $x^* \in X^*$. Параметр m введен для учета специальных классов выпуклых функций: например, для кусочно-линейной негладкой функции $m = 1$, для квадратичной гладкой функции $m = 2$.

Пусть известно f^* . Для нахождения точки $x^* \in X^*$ применяется субградиентный метод Поляка [1]:

$$x_{k+1} = x_k - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|}, \quad h_k = \frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|}, \quad k = 0, 1, 2, \dots \quad (1)$$

Шаг h_k называют шагом Поляка, он определяет величину максимального сдвига в направлении нормированного антисубградиента, при котором угол между антисубградиентом и направлением из точки x_{k+1} на точку минимума будет нетупым.

Медленную сходимость метода Поляка для овражных функций определяет угол между двумя последовательными субградиентами: $\partial f(x_k)$ и $\partial f(x_{k+1})$. Чем ближе этот угол к 180 градусам, тем более медленной будет скорость сходимости. Тупой угол между векторами $\partial f(x_k)$ и $\partial f(x_{k+1})$ можно преобразовать в прямой с помощью „однорангового

эллипсоидального оператора“ [2]. Если на каждой итерации избавляться от тупого угла между последовательными субградиентами в преобразованном пространстве переменных, то для овражных функций это повысит скорость сходимости метода. Этот принцип реализован в изложенной ниже модификации метода Поляка.

Ускоренный субградиентный метод Поляка имеет следующий вид:

$$x_{k+1} = x_k - h_k B_k \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}, \quad h_k = \frac{m(f(x_k) - f^*)}{\|B_k^T \partial f(x_k)\|}, \quad k = 0, 1, 2, \dots, \quad (2)$$

где матрица $B_0 = I$, а матрица B_{k+1} размерности $n \times n$ вычисляется по правилу:

$$B_{k+1} = \begin{cases} B_k, & \text{если } \mu_k \geq 0 \\ B_{k+1} = B_k + (B_k \eta) \xi_{k+1}^T, & \text{иначе} \end{cases}, \quad \text{где } \mu_k = (\xi_k, \xi_{k+1}),$$

$$\xi_k = \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}, \quad \xi_{k+1} = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|},$$

$$\eta = \left(\frac{1}{\sqrt{1 - \mu_k^2}} - 1 \right) \xi_{k+1} - \frac{\mu_k}{\sqrt{1 - \mu_k^2}} \xi_k.$$

Метод (2) можно назвать ускоренным методом Поляка за счет антиовражного приема, подобного тому, который использован в r -алгоритмах Шора [3]. Действительно, на k -й итерации растяжение пространства производится в направлении разности нормированных последовательных субградиентов в преобразованном пространстве переменных: $y = A_k x = B_k^{-1} x$, где B_k – невырожденная матрица размерности $n \times n$. Если нормы субградиентов одинаковы, то это направление совпадает с разностью двух последовательных субградиентов, по которой реализуется растяжение пространства в r -алгоритмах. Отличие состоит в том, что в преобразованном пространстве переменных для r -алгоритмов второй субградиент определяется согласно шагу наискорейшего спуска в направлении антисубградиента, а в методе (2) – согласно шагу Поляка.

Теорема. Пусть $A_k = B_k^{-1}$, $A_{k+1} = B_{k+1}^{-1}$. Для $\forall x^* \in X^*$ и всех точек, генерируемых методом (2), справедливы неравенства

$$\|A_{k+1}(x_{k+1} - x^*)\|^2 \leq$$

$$\leq \|A_k(x_k - x^*)\|^2 - \left(\frac{m(f(x_k) - f^*)}{\|B_k^T \partial f(x_k)\|} \right)^2, \quad k = 0, 1, \dots \quad (3)$$

Доказательство теоремы аналогично доказательству теоремы 2 [2].

Преобразование пространства в методе (2) направлено на уменьшение степени вытянутости поверхностей уровня овражных выпуклых функций. Неравенства (3) означают, что в каждом очередном преобразованном пространстве переменных гарантируется уменьшение расстояния до множества точек минимума. В связи с этим для каждой итерации $k > 1$ имеет место неравенство

$$\|A_k(x_k - x^*)\|^2 \leq \|x_0 - x^*\|^2 - \sum_{i=0}^{k-1} \left(\frac{m(f(x_i) - f^*)}{\|B_i^T \partial f(x_i)\|} \right)^2.$$

Для овражных функций детерминант матрицы B_k стремится к нулю, так как, если на k -м шаге реализуется преобразование пространства, то

$$\begin{aligned} \det(B_{k+1}) &= \det(B_k) \det(I + \eta \xi_{k+1}^T) = \det(B_k) \sqrt{1 - \mu_k^2} = \\ &= \det(B_k) \sqrt{1 - \cos^2 \varphi_k}, \end{aligned}$$

где φ_k — тупой угол между двумя последовательными субградиентами.

Для овражных негладких функций метод (2) обладает ускоренной сходимостью по отношению к методу (1). Так, например, для функции двух переменных $f(x_1, x_2) = |x_1| + t|x_2|$ при любом значении параметра $t > 1$ и произвольной стартовой точке x_0 методом (2) можно найти точку минимума $x^* = (0, 0)$ не более, чем за две итерации. Метод (1) сходится к x^* со скоростью геометрической прогрессии со знаменателем $\sqrt{1 - 1/t^2}$ и требует существенного количества итераций даже при сравнительно небольших значениях t .

Ускоренный метод Поляка можно использовать для одновременного нахождения оптимальных решений прямой и двойственной задач выпуклого программирования. В экономических приложениях это позволяет исследовать точечно-множественные отображения между ресурсами и технологиями (переменные в прямой задаче) и ценами на них (переменные в двойственной задаче).

Литература. [1] Поляк Б. Т. Введение в оптимизацию, М., 1983. [2] Стецюк П. И. Ортогонализирующие линейные операторы в выпуклом программировании, Кибернетика и системный анализ, № 3, 1997. [3] Шор Н. З. Методы минимизации недифференцируемых функций и их приложения, Киев, 1979.

Ускоренные по Шору модификации метода Поляка для овражных функций

П. И. Стецюк

*Всероссийская молодежная школа-семинар „Дискретные модели и
методы принятия решений“: Материалы школы-семинара
(г. Новосибирск, 21-23 июня 2013). – Новосибирск: Изд-во Ин-та
математики, 2013. – С. 142–153.*

Обсуждаются два субградиентных метода с преобразованием пространства для нахождения точки минимума выпуклой функции при известном минимальном значении. В методах используется шаг Поляка в направлении нормированного антисубградиента и гарантируется монотонное уменьшение расстояния до точки минимума в последовательно преобразованных пространствах переменных. Преобразование пространства реализуется с помощью однорангового оператора и направлено на уменьшение степени овражности поверхностей уровня выпуклых функций, подобно тому, как это сделано в r -алгоритмах Шора.

Пусть $f(x)$ – выпуклая функция векторного аргумента $x \in E^n$ и ее субградиент $\partial f(x)$ удовлетворяет условию:

$$(x - x^*, \partial f(x)) \geq m(f(x) - f^*), \quad \forall x \in E^n, \quad m \geq 1. \quad (1)$$

Здесь E^n – евклидово пространство размерности n со скалярным произведением (x, y) ; x^* – единственная точка минимума функции $f(x)$; f^* – минимальное значение функции $f(x)$: $f^* = f(x^*)$. Параметр m считается известным, он введен для учета специальных классов выпуклых функций, например, для квадратичной гладкой функции $m = 2$. Значение $m = 1$ обеспечивает выполнение условия (1) для произвольной выпуклой функции.

Субградиентный метод Поляка и проблема овражности. Пусть известно значение f^* . Для нахождения точки x^* можно применить субградиентный метод:

$$x_{k+1} = x_k - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|}, \quad h_k = \frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|}, \quad k = 0, 1, 2, \dots \quad (2)$$

Метод (2) называют методом Поляка. Для случая $m = 1$ он предложен Б.Т. Поляком в 1969 году [1] и имеет следующий геометрический смысл.

Функция $f(x)$ аппроксимируется линейной $\tilde{f}(x) = f(x_k) + (f'(x_k), x - x_k)$ и шаг выбирается так, чтобы эта аппроксимирующая функция стала равной f^* (т. е. $\tilde{f}(x_{k+1}) = f^*$).

Величину h_k называют шагом Поляка или шагом Агмона-Моцкина-Шенберга. Впервые такой шаг был использован в 1954 году в релаксационном (субградиентном) методе для нахождения хотя бы одного решения совместной системы линейных неравенств [2, 3]. В 1965 году И.И. Еремин обобщил этот шаг для решения систем неравенств с выпуклыми функциями [4].

Шаг h_k задает величину максимального сдвига в направлении нормированного антисубградиента, при котором угол между антисубградиентом и направлением из точки x_{k+1} в точку минимума будет нетупым. Это гарантирует уменьшение расстояния до точки минимума x^* (в общем случае до множества минимумов X^*) на каждой итерации метода (2).

Теорема 1. *Для всех точек, генерируемых методом (2), справедливы неравенства*

$$\|x_{k+1} - x^*\|^2 \leq \|x_k - x^*\|^2 - \left(\frac{m(f(x_k) - f^*)}{\|\partial f(x_k)\|} \right)^2, \quad k = 0, 1, \dots \quad (3)$$

Метод Поляка сходится со скоростью геометрической прогрессии, однако во многих практических задачах знаменатель прогрессии может оказаться близким к единице. Для гладкого случая этот факт также хорошо известен (градиентные методы медленно сходятся для функций овражного типа). Тот же эффект имеет место и в негладком случае, например, для функции двух переменных $f(x) = |x_1| + t|x_2|$ метод (2) сходится со скоростью геометрической прогрессии со знаменателем $\sqrt{1 - \frac{1}{t^2}}$, т. е. очень медленно для больших t . Так, например, чтобы найти точку, где значение функции $f(x_1, x_2) = |x_1| + 27|x_2|$ отличается от минимального $f^* = 0$ не более, чем на $\varepsilon_f = 10^{-10}$, методу Поляка, стартующему из точки $x_0 = (1, 1)$, требуется 8634 итерации. Для функции $f_1(x_1, x_2) = |x_1| + 100|x_2|$ указанная точность не обеспечивается даже при ста тысячах итераций.

Об ускорении метода Поляка. В 1969 году Б.Т. Поляк предложил один из способов ускорения метода (2) [1]. Он состоит в том, чтобы вблизи минимума функцию $f(x)$ аппроксимировать кусочно-линейной функцией, используя информацию из предыдущих итераций. Самый общий метод, реализующий способ Поляка, имеет следующий вид:

$$x_{k+1} = P_{Q_k}(x_k), \quad Q_k = \{x : f(x_i) + (\partial f(x_i), x - x_i) \leq f^*, \quad i \in I_k\}, \quad (2a)$$

где I_k – любое подмножество индексов из $0, 1, \dots, k$, которое обязательно содержит индекс k , P_{Q_k} – оператор проектирования на множество Q_k .

Метод (2а) также сходится со скоростью геометрической прогрессии и не медленнее, чем метод (2). Если $I_k = \{0, 1, \dots, k\}$, то метод (2а) дает точный минимум для кусочно-линейной функции. Если множество $I_k = \{k-1, k\}$, то проекцию на множество Q_k можно выписать явно. В этом случае метод (2а) использует всего два вектора и для построения очередного направления спуска использует линейную комбинацию направления субградиента и направления движения на предыдущем шаге с тем, чтобы оно составляло более острый угол с направлением на минимум.

В том же 1969 году Н.З. Шор предложил оригинальный способ ускорения сходимости субградиентных методов [5], который можно использовать в том числе и для ускорения метода (2). Способ Шора связан с использованием линейных неортогональных преобразований пространства для улучшения обусловленности овражных функций. Ниже будут рассмотрены субградиентные методы с шагом Поляка и преобразованием пространства переменных, которое обеспечивает монотонное уменьшение расстояния до точки минимума и направлено на уменьшение степени овражности поверхностей уровня выпуклых функций подобно тому, как это сделано в r -алгоритмах Шора-Журбенко [6]. Преобразование пространства осуществляется только тогда, когда обнаружено зигзагообразное движение вдоль русла оврага, или другими словами, когда угол между двумя последовательными субградиентами тупой.

Тупой угол между двумя нормированными векторами ξ и η из E^n можно преобразовать в прямой с помощью линейного оператора из E^n в E^n , который в матричной форме представим

$$T_1(\xi, \eta) = I - \frac{1}{1 - (\xi, \eta)^2} \left((1 - \sqrt{1 - (\xi, \eta)^2}) \eta - (\xi, \eta) \xi \right) \eta^T. \quad (4)$$

Здесь $\xi, \eta \in E^n$ – векторы, такие что $\|\xi\| = 1$, $\|\eta\| = 1$ и их скалярное произведение удовлетворяет условию $(\xi, \eta)^2 \neq 1$, I – единичная матрица размера $n \times n$. Для оператора $T_1(\xi, \eta)$ существует обратный $T_1^{-1}(\xi, \eta)$

$$T_1^{-1}(\xi, \eta) = I + \frac{1}{\sqrt{1 - (\xi, \eta)^2}} \left((1 - \sqrt{1 - (\xi, \eta)^2}) \eta - (\xi, \eta) \xi \right) \eta^T \quad (5)$$

Оператор $T_1(\xi, \eta)$ введен в [7] под названием "одноранговый эллипсоидальный оператор". Он связан с преобразованием в шар специального эллипсоида, описанного вокруг тела W , которое получено в результате пересечения шара и двух полупространств, проходящих через центр

шара [6]. В случае тупого угла между нормальными к границам полупространств этот эллипсоид содержит тело W и является минимальным по объему среди семейства эллипсоидов, центр которых совпадает с центром шара.

Если угол φ между векторами ξ и η тупой, то специальный эллипсоид содержит тело W . Его объем меньше, чем объем шара, и это уменьшение равно $\sqrt{1 - (\xi, \eta)^2} = \sqrt{1 - (\cos \varphi)^2}$. Если преобразовать пространство с помощью оператора $T_1(\xi, \eta)$, то в преобразованном пространстве эллипсоид станет шаром, а образы векторов ξ и η будут ортогональными. Это позволяет „расширить“ конус подходящих направлений убывания функции для субградиентного процесса в преобразованном пространстве переменных, подобно тому как это делается в r -алгоритмах.

Ускоренный субградиентный метод Поляка (метод amsg2). Сходимость метода Поляка для овражных функций зависит от угла между двумя последовательными субградиентами: $\partial f(x_k)$ и $\partial f(x_{k+1})$. Чем ближе этот угол к 180 градусам, тем более медленной будет скорость сходимости. Если на каждой итерации избавляться от тупого угла между последовательными субградиентами в преобразованном пространстве переменных, то для овражных функций следует ожидать повышения скорости сходимости такого метода. Этот принцип реализован в изложенной ниже модификации метода Поляка и сделано это с помощью однорангового эллипсоидального оператора.

Ускоренный субградиентный метод Поляка имеет вид:

$$x_{k+1} = x_k - h_k B_k \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}, \quad h_k = \frac{m(f(x_k) - f^*)}{\|B_k^T \partial f(x_k)\|}, \quad k = 0, 1, 2, \dots, \quad (6)$$

где матрица $B_0 = I$, а матрица B_{k+1} размера $n \times n$ вычисляется по правилу:

$$B_{k+1} = \begin{cases} B_k, & \text{если } \mu_k = (\xi_k, \xi_{k+1}) \geq 0, \\ B_k + (B_k \eta) \xi_{k+1}^T, & \text{иначе,} \end{cases} \quad (7)$$

где

$$\xi_k = \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}, \quad \xi_{k+1} = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|},$$

$$\eta = \left(\frac{1}{\sqrt{1 - \mu_k^2}} - 1 \right) \xi_{k+1} - \frac{\mu_k}{\sqrt{1 - \mu_k^2}} \xi_k.$$

Метод (6), (7) условимся называть методом `amsg2`. Здесь „ams“ указывает на способ регулировки шага в направлении нормированного антисубградиента, а „g2“ указывает, что `ams`-шаг (Agmon-Motzkin-Schoenberg step, шаг Поляка) используется в пространстве переменных, преобразованном с помощью двух последовательных субградиентов (g2).

Метод `amsg2` является методом Поляка, ускоренным за счет антиовражного приема, подобного тому, который использован в r -алгоритмах Шора. Действительно, на k -й итерации растяжение пространства производится в направлении разности нормированных последовательных субградиентов в преобразованном пространстве переменных: $y = A_k x = B_k^{-1} x$, где B_k – невырожденная матрица размерности $n \times n$. Если нормы субградиентов одинаковы, то это направление совпадает с разностью двух последовательных субградиентов, по которой осуществляется растяжение пространства в r -алгоритмах. Отличие состоит в том, что в преобразованном пространстве переменных для r -алгоритмов второй субградиент определяется согласно шагу наискорейшего спуска в направлении антисубградиента, а для метода `amsg2` – согласно шагу Поляка (шагу Агмона-Моткина-Шенберга).

Теорема 2. Пусть $A_k = B_k^{-1}$, $A_{k+1} = B_{k+1}^{-1}$. Для всех точек, генерируемых методом (6), (7), справедливы неравенства

$$\|A_{k+1}(x_{k+1} - x^*)\|^2 \leq \|A_k(x_k - x^*)\|^2 - \left(\frac{m(f(x_k) - f^*)}{\|B_k^T \partial f(x_k)\|} \right)^2, k = 0, 1, \dots \quad (8)$$

Неравенства (8) означают, что в каждом очередном преобразованном пространстве переменных гарантируется уменьшение расстояния до точки минимума x^* . Поэтому для каждой итерации $k > 1$ имеет место неравенство

$$\|A_k(x_k - x^*)\|^2 \leq \|x_0 - x^*\|^2 - \sum_{i=0}^{k-1} \left(\frac{m(f(x_i) - f^*)}{\|B_i^T \partial f(x_i)\|} \right)^2.$$

Для овражных функций детерминант матрицы B_k уменьшается, а следовательно уменьшается объем эллипсоида $\|A_k(x_k - x^*)\|$, локализующего точку x^* . Действительно, если на k -м шаге выполняется преобразование пространства, то

$$\begin{aligned} \det(B_{k+1}) &= \\ &= \det(B_k) \det(I + \eta \xi_{k+1}^T) = \det(B_k) \sqrt{1 - \mu_k^2} = \det(B_k) \sqrt{1 - \cos^2 \varphi_k}, \end{aligned}$$

Таблица 1. Итерации метода Поляка (itn1) и метода amsg2 (itn2) по нахождению с точностью epsf минимума кусочно-линейной функции $f(x) = \sum_{i=1}^{10} q^{(i-1)/9} |x_i - 1|$ при трех степенях вытянутости: $q = 3$, $q = 9$ и $q = 27$.

epsf	itn1	itn2	itn1	itn2	itn1	itn2
1.e-01	31	15	220	37	1645	64
1.e-02	72	24	458	44	3257	73
1.e-03	113	29	695	49	4871	78
1.e-04	155	38	933	54	6481	80
1.e-05	196	43	1170	59	8083	84
1.e-06	237	50	1407	62	9633	91
1.e-07	279	54	1642	66	10000	93
1.e-08	320	59	1874	74	10000	100
1.e-09	362	62	2101	78	10000	108
1.e-10	403	65	2322	85	10000	113

где φ_k – тупой угол между двумя последовательными субградиентами.

Для овражных негладких функций метод amsg2 имеет ускоренную сходимость по сравнению с методом Поляка – методом (2). Так, например, для кусочно-линейной функции $f(x_1, x_2) = |x_1| + t|x_2|$ при любом значении параметра $t > 1$ и произвольной стартовой точке x_0 метод amsg2 находит точку минимума $x^* = (0, 0)$ не более, чем за три итерации. Заметим, что метод Поляка сходится к x^* со скоростью геометрической прогрессии со знаменателем $\sqrt{1 - 1/t^2}$ и требует существенного количества итераций даже при сравнительно небольших значениях t . В таблице 1 дано сравнение обоих методов для кусочно-линейной функции от 10 переменных при трех различных степенях вытянутости. Расчеты выполнены при $m = 1$ и $x_0 = (0, \dots, 0)^T$. Из таблицы легко видеть, что по мере увеличения степени вытянутости разрыв между методом Поляка и методом amsg2 по количеству итераций увеличивается.

Метод amsg2p. В его названии "ams" указывает на способ регулировки шага в направлении нормированного антисубградиента, а "g2p" указывает, что ams-шаг используется в пространстве переменных, преобразованном с помощью двух последних субградиентов (g2) и агрегатного вектора (p). Преобразование пространства реализуется с помощью однорангового эллипсоидального оператора и только на тех итерациях метода, когда тупым является хотя бы один из углов – угол между двумя последовательными субградиентами, либо угол между последним

субградиентом и агрегатным вектором. Агрегатный вектор является выпуклой комбинацией вычисленных на предыдущих итерациях субградиентов и позволяет обеспечить более сильное уменьшение объема эллипсоида, локализирующего точку x^* , чем в методе amsg2. Агрегатный вектор автоматически обновляется в зависимости от того, можно ли на его основе построить тупой угол, или нет.

Метод amsg2p учитывает случай произвольного значения $f_{min} \geq f^*$. Он позволяет либо найти такую точку, где значение выпуклой функции $f(x)$ меньше или равно $f_{min} + \varepsilon$, либо гарантирует достаточное условие того, что точки, где значение $f(x)$ равно f_{min} , в шаре заданного радиуса не существует. В первом случае метод amsg2p находит точку $x_\varepsilon^* \in \{x : f(x) - f_{min} \leq \varepsilon\}$ и соответствующий ей номер итерации k_ε^* , а во втором – останавливается с сообщением "точки не существует".

Метод amsg2p состоит в следующем. На итерации $k = 0$ заданы: начальное приближение $x_0 \in E^n$; начальный радиус r_0 такой, что $\|x_0 - x^*\| \leq r_0$; достаточно малое $\varepsilon > 0$. Вычислим $f(x_0)$ и $\partial f(x_0)$. Если $f(x_0) - f_{min} \leq \varepsilon$, то $x_\varepsilon^* = x_0$, $k_\varepsilon^* = 0$ и окончание работы алгоритма. Иначе положим $h_0 = \frac{m(f(x_0) - f_{min})}{\|\partial f(x_0)\|}$, $\xi_0 = \frac{\partial f(x_0)}{\|\partial f(x_0)\|} \in E^n$, $p_0 = 0 \in E^n$, $B_0 = I$ – единичная $n \times n$ -матрица. Перейдем к следующей итерации.

Пусть на k -й итерации получены $x_k \in E^n$, h_k , r_k , $\xi_k \in E^n$, $p_k \in E^n$, B_k – матрица $n \times n$. Для $(k + 1)$ -й итерации выполним пп. 1-5.

- 1) Вычислим $t_k = h_k / r_k$. Если $t_k > 1$, то "точки не существует" и окончание алгоритма. Иначе положим $r_{k+1} = r_k \sqrt{1 - t_k^2}$ и вычислим очередное приближение

$$x_{k+1} = x_k - h_k B_k \xi_k.$$

- 2) Вычислим $f(x_{k+1})$ и $\partial f(x_{k+1})$. Если $f(x_{k+1}) - f_{min} \leq \varepsilon$, то $x_\varepsilon^* = x_{k+1}$, $k_\varepsilon^* = k + 1$ и окончание работы алгоритма. Иначе положим

$$\xi_{k+1} = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|}, \quad h_{k+1} = \frac{m(f(x_{k+1}) - f_{min})}{\|B_k^T \partial f(x_{k+1})\|}.$$

- 3) Вычислим $\lambda_1 = -p_k^T \xi_{k+1}$ и $\lambda_2 = -\xi_k^T \xi_{k+1}$. Положим

$$p_{k+1} = \begin{cases} \frac{\lambda_1}{\sqrt{\lambda_1^2 + \lambda_2^2}} p_k + \frac{\lambda_2}{\sqrt{\lambda_1^2 + \lambda_2^2}} \xi_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 > 0, \\ p_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 \leq 0, \\ \xi_k, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 > 0, \\ 0, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 \leq 0. \end{cases}$$

4) Вычислим $\mu_k = p_{k+1}^T \xi_{k+1}$. Если $-1 < \mu_k < 0$, то вычислим

$$B_{k+1} = B_k + (B_k \eta) \xi_{k+1}^T,$$

$$\text{где } \eta = \left(\frac{1}{\sqrt{1 - \mu_k^2}} - 1 \right) \xi_{k+1} - \frac{\mu_k}{\sqrt{1 - \mu_k^2}} p_{k+1},$$

и пересчитаем

$$h_{k+1} = \frac{h_{k+1}}{\sqrt{1 - \mu_k^2}}, \quad p_{k+1} = \frac{1}{\sqrt{1 - \mu_k^2}} (p_{k+1} - \mu_k \xi_{k+1}).$$

Иначе положим $B_{k+1} = B_k$ и $p_{k+1} = 0$.

5) Перейдем к следующей итерации с x_{k+1} , h_{k+1} , r_{k+1} , ξ_{k+1} , p_{k+1} , B_{k+1} .

Теорема 3. Пусть $A_k = B_k^{-1}$, $A_{k+1} = B_{k+1}^{-1}$. Если $f_{min} \geq f^*$, то для генерируемых методом *amsg2p* точек справедливы неравенства

$$\|A_{k+1}(x_{k+1} - x^*)\|^2 \leq \|A_k(x_k - x^*)\|^2 - \left(\frac{m(f(x_k) - f_{min})}{\|B_k^T \partial f(x_k)\|} \right)^2, k = 0, 1, \dots \quad (9)$$

Теорема 3 означает, что в каждом очередном преобразованном пространстве переменных расстояние до точки минимума уменьшается. Благодаря этому для каждой итерации $k > 1$ имеем неравенство

$$\|A_k(x_k - x^*)\|^2 \leq \|x_0 - x^*\|^2 - \sum_{i=0}^{k-1} \left(\frac{m(f(x_i) - f_{min})}{\|B_i^T \partial f(x_i)\|} \right)^2 = r_0^2 - \sum_{i=0}^{k-1} h_i^2 = r_k^2,$$

с помощью которого обеспечивается достаточное условие отсутствия точки x_ε^* при $f_{min} < f^* - \varepsilon$ (реализовано в п. 1 метода *amsg2p*).

Для овражных функций метод *amsg2p* обладает ускоренной сходимостью при произвольной начальной стартовой точке x_0 и достаточно малых значениях параметра ε ($\varepsilon \sim 10^{-10} - 10^{-14}$). Это подтверждают приведенные в табл. 2 результаты вычислительных экспериментов для квадратичных функций от $n = 200$ переменных с различной степенью овражности (рассматривалась функция $f(x) = \sum_{i=1}^n q^{i-1} x_i^2$, а степенью овражности считалась величина $Q = q^{n-1}$). При расчетах использовался параметр $m = 2$, $f_{min} = f^* = 0$, $x_0 = (1, \dots, 1)^T$. Для ряда

Таблица 2. Сходимость `amsg2p` для квадратичных функций, $n = 200$

epsf	itn(Q_1)	itn(Q_2)	itn(Q_3)	itn(Q_4)	itn(Q_5)
1.00e-003	11	36	84	361	773
1.00e-005	15	46	99	405	826
1.00e-007	18	56	113	430	868
1.00e-009	22	65	128	461	916
1.00e-011	25	73	142	493	947
1.00e-013	29	81	154	517	979
1.00e-015	32	89	167	541	1006
1.00e-017	35	96	180	560	1027
1.00e-019	39	102	189	574	1042
1.00e-020	41	105	196	585	1048

стремящихся к нулю $\varepsilon = \text{epsf}$ в таблице приведены количества итераций, которые требуются при степенях овражности $Q_1 = 10$, $Q_2 = 100$, $Q_3 = 1000$, $Q_4 = 10^6$, $Q_5 = 10^9$.

Для овражных функций метод `amsg2p` оказался значительно эффективнее, чем метод Поляка без преобразования пространства. Число итераций для него увеличивается слабо при значительном уменьшении ε , он сравним с r -алгоритмом, а в ряде случаев и превосходит его по количеству итераций [8, 9]. Это означает, что преобразования пространства переменных, направленные на уменьшение степени вытянутости поверхностей уровня выпуклых функций, способны для овражных функций значительно ускорить сходимость субградиентных методов с шагом Поляка.

Метод `amsg2p` можно использовать для нахождения достаточно точного приближения к единственной точке минимума существенно овражных функций. Проиллюстрируем это на известной тестовой задаче `maxquad`, которая связана с минимизацией существенно овражной выпуклой кусочно-квадратичной функции $f(x) = \max_{1 \leq k \leq 5} \varphi_k(x)$, $x \in R^{10}$.

Здесь $\varphi_k(x) = x^T H_k x - b_k^T x$, H_k – симметричные 10×10 -матрицы, такие что $H_{kij} = e^{i/j} \cos(ij) \sin k$, если $i < j$, и $H_{kii} = i |\sin k| / 10 + \sum_{j \neq i} |H_{kij}|$, а компоненты векторов b_k определяются $b_{ki} = e^{i/k} \sin(ik)$. Трудоемкость метода `amsg2p` для нахождения решения задачи `maxquad` с достаточно высокой точностью (до 14-ти значащих цифр) позволяет оценить следующий фрагмент

```

Maxquad: f(x0)    5.3370664293114e+003
          fmin = -8.4140833459641e-001
..epsf..   ....f(itn).....  .itn.
1.0e-001  -7.7355266120112e-001    17
1.0e-003  -8.4084776169123e-001    29
1.0e-004  -8.4132394277880e-001    35
1.0e-005  -8.4140078034524e-001    41
1.0e-006  -8.4140807664455e-001    49
1.0e-011  -8.4140833458913e-001    94
1.0e-012  -8.4140833459555e-001   101
1.0e-013  -8.4140833459633e-001   110
1.0e-014  -8.4140833459640e-001   116
1.0e-015  -8.4140833459641e-001   122

```

численных расчетов с одноименной octave-программой `amsg2p`. В программе использовались $m = 1$, $x_0 = (1, \dots, 1)^T$ и $\varepsilon = \textit{epsf}$.

В заключение отметим, что ускоренные варианты субградиентных методов с помощью однорангового эллипсоидального оператора можно построить и для других способов регулировки шага. Преимуществом таких методов есть полная определенность в параметрах преобразования пространства. В этом смысле метод `amsg2p` может служить некоторым эталоном, в котором отсутствуют подлежащие настройке параметры. Кроме того, в нем оригинально используется агрегатный вектор, который автоматически обновляется в зависимости от того, можно ли на его основе уменьшить объем эллипсоида, локализирующего точку минимума, или нет.

Работа поддержана совместным грантом НТЦУ и НАНУ (проект № 5710).

ЛИТЕРАТУРА

1. Б.Т. Поляк. Минимизация негладких функционалов // Журн. вычисл. математики и мат. физики. 1969. № 3. С. 507-521.
2. S. Agmon. The relaxation method for linear inequalities // Canadien J. of Mathematics. 1954. № 6. P. 382-392.
3. T. Motzkin, I.J. Schoenberg. The relaxation method for linear inequalities // Canadien J. of Mathematics. 1954. № 6. P. 393-404.
4. И.И. Еремин. Обобщение релаксационного метода Агмона-Моцкина // Успехи мат. наук. 1965. Т. XX. Вып. 2(122). С. 183-187.

5. Н.З. Шор, В.И. Билецкий. Метод растяжения пространства для ускорения сходимости в задачах овражного типа // Тр. семинара Науч. совета АН УССР по кибернетике „Теория оптимальных решений“. – Киев: Ин-т кибернетики им. В.М.Глушкова НАН Украины. – 1969. № 2. С. 3–18.
6. Н.З. Шор, Н.Г. Журбенко. Метод минимизации, использующий операцию растяжения пространства в направлении разности двух последовательных градиентов // Кибернетика. 1971. № 3. С. 51-59.
7. П.И. Стецюк. Ортогонализирующие линейные операторы в выпуклом программировании // Кибернетика и системный анализ. 1997. № 3. С. 97-119.
8. П.И. Стецюк. r -алгоритмы и эллипсоиды // Кибернетика и системный анализ. 1996. № 1. С. 113-134.
9. П.И. Стецюк. Субградиентные методы переменной метрики, использующие шаг Агмона-Мощкина и одноранговый эллипсоидальный оператор // Труды АТИК. 2007–2008. Кишинэу: Эврика, 2009. I (XII). С. 16-25.
10. П.И. Стецюк. Ускоренные модификации субградиентного метода Поляка для овражных выпуклых функций // Стохастическое программирование и его приложения / Под ред. П.С. Кнопова, В.И. Зоркальцева, Я.М. Иваньо. – Иркутск: Институт систем энергетики им. Л.А. Мелентьева СО РАН, 2012. С. 160-184.

Р А З Д Е Л VI

***r*-Алгоритмы и квазиньютоновские методы**

В раздел включены статьи:

- 6.1. Квазиньютоновские методы и *r*-алгоритмы (*Стецюк, 1996*) 352
- 6.2. Линейные операторы в квазиньютоновских методах
(*Стецюк, 1998*) 375
- 6.3. Субградиентные методы с преобразованием пространства для
минимизации овражных выпуклых функций (*Стецюк, 2011*) .. 382
- 6.4. *r*-Алгоритмы: теория и практика (*Стецюк, 2013*) 391

Квазиньютоновские методы и r -алгоритмы.

П. И. Стецюк

*Препр. / НАН Украины. Ин-т кибернетики им. В. М. Глушкова.
– Киев, 1996. – 96–10. – 21 с.*

Обсуждается ряд вопросов, связанных с квазиньютоновскими методами и r -алгоритмами. Построен метод с преобразованием пространства для минимизации гладких выпуклых функций. Показана его эффективность для минимизации сильно овражных квадратичных функций. Обсуждается перенос аналогичных идей на негладкий случай.

Обсуждается специальная форма (B -форма) методов квазиньютоновского типа, которая позволяет просто интерпретировать эти методы, как градиентные в преобразованном соответствующим образом пространстве аргументов. Приведена B -форма метода Давидона – Флетчера – Пауэлла и на ее основе проведено сравнение этого метода с r -алгоритмами. Для минимизации гладких выпуклых функций построен градиентный метод с преобразованием пространства, сочетающий свойства как квазиньютоновских методов, так и r -алгоритмов. Обсуждаются возможные схемы такого типа методов для минимизации негладких выпуклых функций.

О квазиньютоновских методах. Теория квазиньютоновских методов [1]–[4] опирается на возможность аппроксимации кривизны нелинейной функции без явного формирования ее матрицы Гессе, т.е. данные о матрице Гессе накапливаются в них на основе наблюдения за изменением градиента функции во время итераций спуска. Они требуют на итерации существенно меньшего объема вычислений, чем ньютоновские методы, и при определенных условиях способны обеспечить скорость сходимости, характерную для методов второго порядка. Это выгодно отличает квазиньютоновские методы от методов ньютоновского типа и на практике им часто отдают предпочтение при решении задач безусловной оптимизации нелинейных гладких функций.

Первым методом квазиньютоновского типа был метод, предложенный Давидоном в [5] и развитый Флетчером и Пауэллом в [6] (ДФП-метод). Он является одним из самых эффективных среди методов квазиньютоновского типа. Так, в [2] он отнесен к разряду наи-

лучших методов нелинейного программирования при отсутствии ограничений, исходя из времени решения задач. В [4] ДФП-метод также рекомендуется в качестве эффективного, наряду с еще одним методом квазиньютоновского типа – методом Бroyдена – Флетчера – Шенно. Хотя различных методов квазиньютоновского типа не так уж и мало [2], наилучшие результаты на практике дают два вышеупомянутых метода.

Семейство методов квазиньютоновского типа можно расширить, используя методики из [3] и [4], суть которых заключается в выборе, согласно некоторому правилу, поправок с помощью матриц первого или второго ранга для коррекции симметричной матрицы, используемой на очередном шаге метода. Однако анализ некоторых квазиньютоновских методов можно упростить, если от шага к шагу использовать одноранговую коррекцию несимметричной матрицы. Другими словами, имеются две возможности представления одного и того же квазиньютоновского метода. Будем придерживаться для них тех названий, которые закрепились за подобными схемами методов субградиентного типа с растяжением пространства [7]. Методы, где корректируется матрица обратного преобразования B (она может быть и несимметричной), будем называть методами в B -форме, а методы, использующие коррекцию симметричной матрицы BB^T , – методами в H -форме. Каждая из этих форм имеет как преимущества, так и недостатки. Сочетание обеих форм позволяет проводить более основательный анализ методов квазиньютоновского типа. Это же справедливо для любого метода, который от шага к шагу использует коррекцию положительно определенной симметричной матрицы, в частности для ряда методов сопряженных направлений.

О H - и B -формах квазиньютоновских методов. Пусть имеется задача

$$\min f(x), \quad (1)$$

где $f(x)$ – выпуклая дважды дифференцируемая функция векторного аргумента $x \in X$, $X = R^n$. R^n – евклидово пространство размерностью n со скалярным произведением (x, y) ; x^* – экстремальная точка задачи (1).

Не ограничивая общности, рассмотрим только такие варианты методов квазиньютоновского типа, которые используют точный поиск минимума функции по направлению и условие окончания процесса за n -шагов для квадратичной целевой функции. Эти методы являются определенными вариантами методов сопряженных направлений, если $f(x)$ – квадратичная функция.

Пусть начальная точка x_0 выбрана в достаточно малой окрестности

минимума x^* , т. е. $f(x)$ будет хорошо аппроксимироваться квадратичной функцией

$$\varphi(x) = \frac{1}{2} \left(\nabla^2 f(x^*) (x - x^*), x - x^* \right) + f(x^*).$$

Здесь $\nabla^2 f(x^*)$ – матрица Гессе в точке минимума. Пусть H_0 – симметричная положительно определенная матрица размером $n \times n$. Обычно полагают $H_0 = I$, где I – единичная матрица.

Методы квазиньютоновского типа в H -форме генерируют последовательность точек $\{x_k, k = 0, 1, \dots, n\}$ по следующему правилу:

$$x_{k+1} = x_k - h_k H_k \nabla f(x_k), \quad (2)$$

где $\nabla f(x_k)$ – градиент $f(x)$ в точке x_k ; H_k – симметричная матрица размером $n \times n$; h_k – шаг, соответствующий минимуму $f(x)$ в направлении $-H_k \nabla f(x_k)$, т. е.

$$h_k = \operatorname{argmin}_{h \geq 0} \left\{ f \left(x_k - h H_k \nabla f(x_k) \right) \right\}. \quad (3)$$

Пересчет матрицы H от шага к шагу осуществляется следующим образом:

$$H_{k+1} = H_k + \Delta H_k, \quad (4)$$

где ΔH_k – матрица небольшого ранга, построенная на основе поведения градиентов $\nabla f(x_k)$ и $\nabla f(x_{k+1})$ так, чтобы $H_n \approx \left[\nabla^2 f(x^*) \right]^{-1}$. Возможность такого пересчета обеспечивает так называемое квазиньютоновское условие

$$H_{k+1} \left(\nabla f(x_{k+1}) - \nabla f(x_k) \right) = -h_k H_k \nabla f(x_k). \quad (5)$$

Для квазиньютоновских методов, в которых отброшено условие окончания процесса за n шагов при минимизации квадратичных функций, условие (5) обеспечивает выполнение свойства, состоящего в том, что собственные числа матрицы H_k стремятся к собственным числам матрицы $\left[\nabla^2 f(x^*) \right]^{-1}$. Фактически разница между методами квазиньютоновского типа в H -форме состоит в различных формулах пересчета матрицы H_{k+1} , удовлетворяющих квазиньютоновскому условию (5) и одному из указанных выше способов приближения H_k к $\left[\nabla^2 f(x^*) \right]^{-1}$.

Любую положительно определенную симметричную матрицу H_k всегда можно представить в виде $H_k = B_k B_k^T$, где B_k – невырожденная матрица размером $n \times n$. Соотношение (2), реализующее переход в очередную точку для H -формы квазиньютоновских методов, можно записать так:

$$x_{k+1} = x_k - h_k B_k B_k^T \nabla f(x_k) \quad (6)$$

или же

$$y_{k+1} = y_k - h_k B_k^T \nabla f(x_k) = y_k - h_k \nabla \varphi_k(y_k), \quad (7)$$

где $y_{k+1} = B_k^{-1} x_{k+1}$ и $y_k = B_k^{-1} x_k$ – образы точек x_{k+1} и x_k из X в преобразованном посредством линейного оператора $A_k = B_k^{-1}$ пространстве аргументов. $\nabla \varphi_k(y_k) = B_k^T \nabla f(x_k)$ – градиент функции $\varphi_k(y) = f(B_k y)$, определенной в пространстве аргументов $Y_k = A_k X$, в точке y_k .

Процесс (7) можно интерпретировать как градиентный метод наискорейшего спуска в преобразованном пространстве аргументов $Y_k = A_k X$ для минимизации выпуклой гладкой функции $\varphi_k(y) = f(B_k y)$. На самом деле, соотношение (3), определяющее выбор шагового множителя в квазиньютоновских методах, равносильно следующему:

$$\begin{aligned} h_k &= \operatorname{argmin}_{h \geq 0} \left\{ \varphi_k \left(y_k - h B_k^T \nabla f(x_k) \right) \right\} = \\ &= \operatorname{argmin}_{h \geq 0} \left\{ \varphi_k \left(y_k - h \nabla \varphi_k(y_k) \right) \right\}, \end{aligned} \quad (8)$$

где $\varphi_k(y) = f(B_k y)$ – функция, определенная в Y_k .

Соотношения (6)–(8) в сочетании с процедурой малоранговой коррекции матрицы B_k позволяют описывать квазиньютоновские методы (и не только их) в B -форме. При этом процессы имеют достаточно простую градиентную природу в преобразованном пространстве аргументов. Заметим, что в силу неоднозначности разложения $H_k = B_k B_k^T$ для конкретного метода в H -форме следует существование разных методов в B -форме. Но это не так уж плохо, так как при построении B -метода легко учитывать и то обстоятельство, которое обеспечивает численную устойчивость метода. Например, при одноранговой коррекции матрицы $B_{k+1} = B_k T_k$ матрицу T_k выбираем так, чтобы отношение $\lambda_{\max}(T_k) / \lambda_{\min}(T_k)$ было по возможности меньшим. Здесь $\lambda_{\max}(T_k)$ ($\lambda_{\min}(T_k)$) – максимальное (минимальное) собственное число матрицы T_k .

B -форма для квазиньютоновских методов обладает и рядом других преимуществ.

Во-первых, она позволяет обосновать процедуру наискорейшего спуска в смысле минимизации числа вычислений $f(x)$ посредством использования в преобразованном пространстве аргументов не только значения функции в точке, но и градиента в ней. По-видимому, здесь можно найти объяснение и ряду замечаний, отмеченных в [2], которые связаны с тем, что квазиньютоновские методы часто показывают не худшую сходимость при «грубом» поиске минимума $f(x)$ по направлению, чем при достаточно точном.

Во-вторых, положительная определенность матрицы $H_k = B_k B_k^T$ здесь следует автоматически в силу невырожденности B_k . В [1] и [3] этот факт устанавливается посредством теорем с довольно емким доказательством. Сюда же примыкает и становится очевидным тот факт, что метод квазиньютоновского типа должен сходиться и при старте с любой строго положительно определенной матрицы $H_0 = \tilde{H}_0$. На самом деле, это равносильно квазиньютоновскому методу с $H_0 = I$, но в преобразованном пространстве аргументов $Y_0 = \tilde{B}_0^{-1} X$, где $\tilde{B}_0 \tilde{B}_0^T = \tilde{H}_0$.

В-третьих, для B -формы квазиньютоновских методов достаточно наглядный вид приобретают и процедуры рестарта для задачи (1), когда начальное приближение x_0 выбрано неудачно.

Однако все доводы в пользу B -формы методов сводятся на нет двумя преимуществами H -формы. Это – более экономное хранение матрицы и возможность обойтись меньшим числом арифметических операций на итерации. Поэтому здесь нужен некий компромисс между исследованием и анализом методов и их реализацией на ЭВМ. В качестве такового может служить, например, разработка численно устойчивого метода квазиньютоновского типа в B -форме, а уже как следствие рассматривать H -форму этого метода с целью экономии памяти и вычислений. Тем более что разработке такого метода ничего не мешает. На самом деле, квазиньютоновское условие можно записать так:

$$B_{k+1} B_{k+1}^T \left(\nabla f(x_{k+1}) - \nabla f(x_k) \right) = -h_k B_k B_k^T \nabla f(x_k).$$

Обозначим $\xi = B_k^T \nabla f(x_{k+1}) - B_k^T \nabla f(x_k)$ и $\eta = B_k^T \nabla f(x_k)$ – соответственно векторы разности последовательных градиентов и текущего градиента в преобразованном пространстве аргументов. Пусть для пересчета B_{k+1} используется одноранговая коррекция достаточно общего вида:

$$B_{k+1} = B_k \left(I + t_1 (\xi + t_2 \eta) (\xi + t_3 \eta)^T \right),$$

где t_1, t_2, t_3 – неизвестные скалярные параметры. Тогда, для того чтобы выполнить квазиньютоновское условие, параметры t_1, t_2, t_3

должны удовлетворять следующему соотношению:

$$\left(I + t_1(\xi + t_2 \eta) (\xi + t_3 \eta)^T \right) \left(I + t_1(\xi + t_3 \eta) (\xi + t_2 \eta)^T \right) \xi = -h_k \eta, \quad (9)$$

связывающему векторы ξ и η в преобразованном пространстве аргументов.

Из соотношения (9) следуют два уравнения при трех неизвестных параметрах t_1 , t_2 и t_3 . Выбор некоторых из этих параметров дает уже известные методы квазиньютоновского типа. Так, например, положив $t_3 = 0$ и тогда, однозначно определив t_1 и t_2 , получим ДФП-метод. Но выбор этих параметров порождает и ряд новых методов квазиньютоновского типа, которые используют одноранговую коррекцию матрицы B_k , и вполне возможно, что среди этих методов можно отыскивать и такие, которые по эффективности не будут уступать ДФП-методу и методу Бройдена – Флетчера – Шенно. Если для пересчета матрицы B_k использовать двухранговую коррекцию, то число рассматриваемых методов можно увеличить. Однако этот вопрос требует более детального исследования. Остановимся только на сравнении обеих форм для ДФП-метода. Это обусловлено еще и тем, что этот метод близок к эффективным на практике r -алгоритмам [7], хотя последние применимы к решению задач безусловной минимизации для более широкого класса функций.

H - и B -формы ДФП-метода. Обе формы для задачи (1) опишем в предположении, что x_0 – хорошее начальное приближение. H -форму ДФП-метода приведем согласно [1] с точностью до незначительных переобозначений.

Шаг 0. Выбрать $x_0 \in R^n$. Если $\nabla f(x_0) = 0$, то остановиться, положив $x^* = x_0$. Иначе положить $H_0 = I$, где I – единичная матрица размером $n \times n$, $g_0 = \nabla f(x_0)$, $k = 0$ и перейти к Шагу 1.

Шаг 1. Положить

$$\xi_k = H_k g_k. \quad (10)$$

Шаг 2. Вычислить

$$h_k = \operatorname{argmin}_{h \geq 0} \left\{ f(x_k - h \xi_k) \right\}. \quad (11)$$

Шаг 3. Положить

$$x_{k+1} = x_k - h_k \xi_k. \quad (12)$$

Шаг 4. Если $\nabla f(x_{k+1}) = 0$, то остановиться, положив $x^* = x_{k+1}$.
Иначе положить

$$g_{k+1} = \nabla f(x_{k+1}), \quad \Delta g_k = g_{k+1} - g_k, \quad \Delta x_k = x_{k+1} - x_k, \quad (13)$$

$$H_{k+1} = H_k - \frac{H_k \Delta g_k (\Delta g_k)^T H_k}{(\Delta g_k, H_k \Delta g_k)} = \frac{\Delta x_k (\Delta x_k)^T}{(\Delta g_k, \Delta x_k)}. \quad (14)$$

Шаг 5. Положить $k = k + 1$ и перейти к Шагу 1.

Для описания B -формы ДФП-метода нам понадобятся некоторые вспомогательные результаты. Так, используя (12) и (13), пересчет матрицы H_{k+1} , согласно (14), можно записать следующим образом:

$$\begin{aligned} H_{k+1} &= H_k - \frac{H_k \Delta g_k (\Delta g_k)^T H_k}{(\Delta g_k, H_k \Delta g_k)} + h_k \frac{H_k g_k g_k^T H_k}{(\Delta g_k, H_k g_k)} = \\ &= B_k B_k^T - \frac{B_k B_k^T \Delta g_k (\Delta g_k)^T B_k B_k^T}{(\Delta g_k, B_k B_k^T \Delta g_k)} + h_k \frac{B_k B_k^T g_k g_k^T B_k B_k^T}{(\Delta g_k, B_k B_k^T g_k)} = \\ &= B_k \left(I - \xi_k \xi_k^T + t_k^2 \eta_k \eta_k^T \right) B_k^T = B_{k+1} B_{k+1}^T, \end{aligned} \quad (15)$$

$$\text{где} \quad \xi_k = \frac{B_k^T g_{k+1} - B_k^T g_k}{\|B_k^T g_{k+1} - B_k^T g_k\|}, \quad \eta_k = \frac{B_k^T g_k}{\|B_k^T g_k\|} \quad \text{и}$$

$$t_k^2 = h_k \frac{\|B_k^T g_k\|^2}{\left(B_k^T g_k, B_k^T g_k - B_k^T g_{k+1} \right)}.$$

Положительность параметра t_k^2 обеспечивается выбором h_k из условия наискорейшего спуска для гладкой функции в преобразованном пространстве аргументов. При точной реализации наискорейшего спуска $t_k^2 = h_k$, так как $(B_k^T g_k, B_k^T g_{k+1}) = 0$. Для того чтобы обеспечить пересчет матрицы H_{k+1} как в (15), достаточно корректировать $B_{k+1} = B_k T_k$, где $T_k = \left(I - (\xi_k + t_k \eta_k) \xi_k^T \right)$. На самом деле, при такой коррекции B_{k+1} для $T_k T_k^T$ имеем:

$$\begin{aligned} T_k T_k^T &= \left(I - (\xi_k + t_k \eta_k) \xi_k^T \right) \left(I - \xi_k (\xi_k + t_k \eta_k)^T \right) = \\ &= I - (\xi_k + t_k \eta_k) \xi_k^T - \xi_k (\xi_k + t_k \eta_k)^T + (\xi_k + t_k \eta_k) (\xi_k + t_k \eta_k)^T = \\ &= I - \xi_k \xi_k^T + t_k^2 \eta_k \eta_k^T, \end{aligned}$$

что обеспечивает средний множитель в правой части (15).

Следовательно, для одноранговой коррекции матрицы B_{k+1} подходит следующая формула:

$$B_{k+1} = B_k \left(I - \left(\frac{\tilde{g}_{k+1} - \tilde{g}_k}{\|\tilde{g}_{k+1} - \tilde{g}_k\|} + \sqrt{h_k \frac{\|\tilde{g}_k\|^2}{(\tilde{g}_k, \tilde{g}_k - \tilde{g}_{k+1})} \frac{\tilde{g}_k}{\|\tilde{g}_k\|}} \right) \times \right. \\ \left. \times \left(\frac{\tilde{g}_{k+1} - \tilde{g}_k}{\|\tilde{g}_{k+1} - \tilde{g}_k\|} \right)^T \right), \quad (16)$$

где \tilde{g}_{k+1} и \tilde{g}_k — субградиенты функции $\varphi_k(y) = f(B_k y)$ в точках $y_{k+1} = B_k^{-1} x_{k+1}$ и $y_k = B_k^{-1} x_k$ соответственно. С учетом (16), B -форма ДФП-метода приобретает следующий вид.

Шаг 0. Выбрать $x_0 \in R^n$. Если $\nabla f(x_0) = 0$, то остановиться, положив $x^* = x_0$. Иначе положить $B_0 = I$, где I — единичная матрица размером $n \times n$, $g_0 = \nabla f(x_0)$, $k = 0$ и перейти к Шагу 1.

Шаг 1. Положить

$$\tilde{g}_k = B_k^T g_k. \quad (17)$$

Шаг 2. Положить

$$\xi_k = B_k \tilde{g}_k. \quad (18)$$

Шаг 3. Вычислить

$$h_k = \operatorname{argmin}_{h \geq 0} \left\{ f(x_k - h \xi_k) \right\}. \quad (19)$$

Шаг 4. Положить

$$x_{k+1} = x_k - h_k \eta_k. \quad (20)$$

Шаг 5. Если $\nabla f(x_{k+1}) = 0$, то остановиться, положив $x^* = x_{k+1}$. Иначе положить $g_{k+1} = \nabla f(x_{k+1})$, $\tilde{g}_{k+1} = B_k^T g_{k+1}$,

$$\xi_k = \frac{\tilde{g}_{k+1} - \tilde{g}_k}{\|\tilde{g}_{k+1} - \tilde{g}_k\|} + \sqrt{h_k \frac{\|\tilde{g}_k\|^2}{(\tilde{g}_k, \tilde{g}_k - \tilde{g}_{k+1})} \frac{\tilde{g}_k}{\|\tilde{g}_k\|}}, \quad \eta_k = \frac{\tilde{g}_{k+1} - \tilde{g}_k}{\|\tilde{g}_{k+1} - \tilde{g}_k\|}, \quad (21)$$

$$B_{k+1} = B_k (I - \xi_k \eta_k^T). \quad (22)$$

Шаг 6. Положить $k = k + 1$ и перейти к Шагу 1.

Приведенный ДФП-метод в B -форме (17)–(22) не является оптимальным в смысле использования арифметических операций. Его можно улучшить, убрав операцию умножения матрицы на вектор в (17) за счет того, что $B_{k+1}^T g_{k+1} = (I - \eta_k \xi_k^T) B_k^T g_{k+1} = \tilde{g}_{k+1} - \eta_k (\xi_k, \tilde{g}_{k+1})$. Но даже в этом случае он будет уступать ДФП-методу в H -форме (10)–(14) по числу арифметических операций, хотя этот разрыв будет не таким уж и большим ($4n^2$ умножений по отношению к $3n^2$ умножений).

Тем не менее B -форма позволяет прояснить ряд моментов для ДФП-метода. Во-первых, положительная определенность матрицы $H_{k+1} = B_{k+1} B_{k+1}^T$ является следствием невырожденности матрицы B_{k+1} , если матрица B_k – невырождена. На самом деле, определитель матрицы B_{k+1} будет таким:

$$\begin{aligned} \det(B_{k+1}) &= \det\left(B_k \begin{pmatrix} I - \xi_k \eta_k^T \end{pmatrix}\right) = \det(B_k) \left(1 - (\xi_k, \eta_k)\right) = \\ &= \det(B_k) \left(1 - \left(\frac{\tilde{g}_{k+1} - \tilde{g}_k}{\|\tilde{g}_{k+1} - \tilde{g}_k\|} + \right.\right. \\ &\quad \left.\left. + \sqrt{h_k \frac{\|\tilde{g}_k\|^2}{(\tilde{g}_k, \tilde{g}_k - \tilde{g}_{k+1})}} \frac{\tilde{g}_k}{\|\tilde{g}_k\|}, \frac{\tilde{g}_{k+1} - \tilde{g}_k}{\|\tilde{g}_{k+1} - \tilde{g}_k\|}\right)\right) = \\ &= \det(B_k) \sqrt{h_k \frac{\|\tilde{g}_k\|^2}{(\tilde{g}_k, \tilde{g}_k - \tilde{g}_{k+1})}} \frac{(\tilde{g}_k, \tilde{g}_k - \tilde{g}_{k+1})}{\|\tilde{g}_{k+1} - \tilde{g}_k\| \cdot \|\tilde{g}_k\|} = \\ &= \det(B_k) \sqrt{h_k \frac{(\tilde{g}_k, \tilde{g}_k - \tilde{g}_{k+1})}{\|\tilde{g}_{k+1} - \tilde{g}_k\|^2}} \neq 0. \end{aligned}$$

При точном наискорейшем спуске

$$\det(B_{k+1}) = \det(B_k) \sqrt{h_k \frac{\|\tilde{g}_k\|^2}{\|\tilde{g}_k\|^2 + \|\tilde{g}_{k+1}\|^2}}$$

и он равен нулю тогда, когда либо $h_k = 0$, либо $\|\tilde{g}_k\| = 0$, что для гладких выпуклых функций обозначает выполнение достаточного условия оптимальности в преобразованном пространстве аргументов. Во-вторых, параметр $\Delta_k = \frac{(\tilde{g}_k, \tilde{g}_{k+1})}{\|\tilde{g}_k\|^2} = \frac{(g_k, H_k g_{k+1})}{(g_k, H_k g_k)}$, который следует из (21) при $\frac{\tilde{g}_k}{\|\tilde{g}_k\|}$, задает точность выполнения условия наискорейшего

спуска в преобразованном пространстве аргументов. Его или близкое к нему условие естественно использовать в квазиньютоновских методах с приближенным вычислением минимума функции по направлению в качестве критерия останова при одномерном спуске по направлению. В-третьих, Δ_k в сочетании с шагом наискорейшего спуска h_k разумно использовать для процедуры рестарта методов квазиньютоновского типа, учитывая, что в процессе пересчета матрицы B_{k+1} возможно накопление ошибок. Именно эти параметры и в большей мере шаг наискорейшего спуска h_k сказываются на точности пересчета матрицы B_{k+1} .

ДФП-метод и r -алгоритмы. Для r -алгоритмов [7, 8] отмечена их «близость» к ДФП-методу в смысле формул пересчета матрицы H_{k+1} . Используя B -форму ДФП-метода, можно проинтерпретировать эту «близость» более содержательно посредством анализа, происходящего в преобразованном пространстве аргументов. Сделаем это для ДФП-метода (17)–(22) и тех вариантов r -алгоритмов, которые используют точный поиск минимума функции по направлению ($r_\mu(\alpha)$ -алгоритм ($\alpha = 2 - 4$) и предельный вариант r -алгоритма ($\alpha = \infty$)). В качестве характеристики для сравнения выберем изменение угла между двумя последовательными субградиентами, которое характерно для одной итерации этих методов при переходе из Y_k в Y_{k+1} . Естественно, эти сравнения справедливы только для гладких функций.

Пусть g_k и g_{k+1} – субградиенты $\varphi_k(y) = f(B_k y)$, определенной в преобразованном пространстве аргументов $Y_k = B_k^{-1}X$, которые получены согласно точному шагу наискорейшего спуска h_k^* в пространстве аргументов Y_k . Тогда их образы \tilde{g}_k и \tilde{g}_{k+1} в преобразованном пространстве $Y_{k+1} = B_{k+1}^{-1}X$, учитывая, что $(g_k, g_{k+1}) = 0$, будут

$$\begin{aligned}
 \tilde{g}_k &= \left(I - \frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} \left(\frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} + \sqrt{h_k^*} \frac{g_k}{\|g_k\|} \right)^T \right) g_k = \\
 &= g_k + \left(\frac{\|g_k\|^2}{\|g_{k+1}\|^2 + \|g_k\|^2} - \sqrt{h_k^*} \frac{\|g_k\|^2}{\|g_{k+1}\|^2 + \|g_k\|^2} \right) (g_{k+1} - g_k) = \\
 &= \left(\frac{\|g_{k+1}\|^2}{\|g_{k+1}\|^2 + \|g_k\|^2} + \sqrt{h_k^*} \frac{\|g_k\|^2}{\|g_{k+1}\|^2 + \|g_k\|^2} \right) g_k + \\
 &+ \left(\frac{\|g_k\|^2}{\|g_{k+1}\|^2 + \|g_k\|^2} - \sqrt{h_k^*} \frac{\|g_k\|^2}{\|g_{k+1}\|^2 + \|g_k\|^2} \right) g_{k+1}, \quad (23)
 \end{aligned}$$

$$\begin{aligned}
\tilde{g}_{k+1} &= \left(I - \frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} \left(\frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} + \sqrt{h_k^*} \frac{g_k}{\|g_k\|} \right)^T \right) g_{k+1} = \\
&= g_k - \frac{\|g_{k+1}\|^2}{\|g_{k+1}\|^2 + \|g_k\|^2} (g_{k+1} - g_k) = \\
&= \frac{\|g_{k+1}\|^2}{\|g_{k+1}\|^2 + \|g_k\|^2} g_k + \frac{\|g_k\|^2}{\|g_{k+1}\|^2 + \|g_k\|^2} g_{k+1}. \tag{24}
\end{aligned}$$

Из (23), (24) следует, что косинус угла ψ_k между векторами \tilde{g}_k и \tilde{g}_{k+1} будет

$$\cos \psi_k = \left(\frac{\tilde{g}_k}{\|\tilde{g}_k\|}, \frac{\tilde{g}_{k+1}}{\|\tilde{g}_{k+1}\|} \right) = 1 / \sqrt{1 + h_k \frac{\|g_k\|^2 + \|g_{k+1}\|^2}{\|g_k\|^2}}. \tag{25}$$

Следовательно, один шаг ДФП-метода из Y_k в Y_{k+1} приводит к уменьшению угла между последовательными субградиентами от $\pi/2$ к острому ψ_k , определяемому из условия (25). Это свойство ДФП-метода характерно и для r -алгоритмов, которые построены на идее «расширить» конус возможных направлений убывания функции посредством операции растяжения пространства аргументов в направлении разности двух последовательных субградиентов.

Из (25) следует, что когда $h_k \ll 1$, тогда итерация ДФП-метода близка к итерации предельного варианта r -алгоритмов, который преобразует субградиенты g_k и g_{k+1} так, чтобы в преобразованном пространстве $\cos \psi_k = 1$. Этот же факт следует и из (16), так как когда $h_k \ll 1$, то операция преобразования пространства в ДФП-методе будет близка к растяжению пространства с очень большим коэффициентом в направлении разности двух последовательных градиентов. Такая ситуация будет иметь место для сильно овражных функций, когда находимся в точке, близкой ко «дну» оврага.

Для $r_\mu(\alpha)$ -алгоритма, который использует постоянный коэффициент растяжения пространства в направлении $g_{k+1} - g_k$, характерно преобразование прямого угла в острый ψ_k , косинус которого равен

$$\begin{aligned}
\cos \psi_k &= (\alpha^2 - 1) / \sqrt{\alpha^4 + \alpha^2 \left(\frac{\|g_k\|^2}{\|g_{k+1}\|^2} + \frac{\|g_{k+1}\|^2}{\|g_k\|^2} \right) + 1} \leq \\
&\leq 1 - \frac{2}{\alpha^2 + 1}.
\end{aligned}$$

Здесь преобразование угла между последовательными градиентами прямо не связано с h_k^* , хотя косвенную связь обеспечивает соотношение между нормами векторов g_k и g_{k+1} .

Итак, для всех трех методов преобразования пространства для гладких функций направлены на одну и ту же цель — «расширить» конус подходящих направлений убывания $f(x)$. Если же в r -алгоритмах эта цель достигается за счет операции растяжения пространства в направлении разности последовательных градиентов, то в ДФП-методе — за счет преобразования, которое при $h_k \ll 1$ близко к этой операции. Для гладких функций преобразование, используемое в ДФП-методе, выглядит более логичным в том смысле, что когда достигается локальный успех за счет шага наискорейшего спуска, то слабее «растягивается» конус подходящих направлений убывания $f(x)$, а когда шаг наискорейшего спуска мал, то он влечет более сильное «растяжение» конуса. Однако для негладких функций, где h_k может равняться нулю, такое преобразование в принципе не применимо.

Тем не менее это обстоятельство можно использовать для модификации $r_\mu(\alpha)$ -алгоритма, чтобы расширить область его применения для класса почти дифференцируемых функций. На самом деле, построенные в [9] «ловушки» для минимизирующей последовательности $r_\mu(\alpha)$ -алгоритма основаны на использовании им постоянного коэффициента растяжения пространства. Заменяя на шаге $r_\mu(\alpha)$ -алгоритма постоянный коэффициент растяжения на переменный, зависящий от параметров h_k^* , $\|g_k\|$ и $\|g_{k+1}\|$, для такого варианта r -алгоритмов вполне возможно обеспечить выход из «плохих» угловых точек для почти дифференцируемых функций, сохранив при этом для него максимальную «близость» к $r_\mu(\alpha)$ -алгоритму.

Некоторые общие замечания. Излагая вышеприведенное, мы преследовали две конкретных цели.

Во-первых, показать, что, даже для такого хорошо исследованного раздела методов как квазиньютоновские, есть ряд вопросов, которые требуют более тщательного анализа, и B -форма является некоторым разумным инструментом исследования этих методов. Тем более, что она больше, чем H -форма, соответствует первоначальному названию этих методов, как методов «переменной метрики».

Во-вторых, проинтерпретировать геометрическую связь ДФП-метода и эффективных на практике r -алгоритмов.

Однако параллельно с этими вопросами приходим к некоторой более общей постановке вопроса в методах математического программирования, а именно к семейству методов «переменной метрики»

(т. е. использующих линейные неортогональные преобразования пространства). Идея таких методов принадлежит Н.З.Шору [10] и состоит в том, что применение линейных неортогональных преобразований позволяет изменять углы между направлением субградиента и направлением на оптимум и тем самым обеспечивает условия, близкие к наилучшим, для субградиентных (ε -субградиентных) процессов в преобразованном пространстве аргументов. По отношению к пространству аргументов X эта идея фактически приводит к изменению понятия «проектирование» или более общо понятия «субградиента».

Одним из достаточно хорошо разработанных направлений в рамках методов «переменной метрики» являются субградиентные методы, использующие операцию растяжения пространства аргументов [7]. Так, растяжение пространства в направлении субградиента дало метод эллипсоидов, имеющий значительные теоретические приложения, а растяжение пространства в направлении разности двух последовательных субградиентов – эффективный на практике $r(\alpha)$ -алгоритм.

Достаточно общая схема метода центров тяжести простых тел (МЦТПТ) [11], которая наряду с растяжением пространства предполагает использование и других линейных неортогональных преобразований пространства, позволяет получить экономные по обработке информации методы «переменной метрики» для задачи выпуклого программирования. В основу МЦТПТ положен принцип монотонного уменьшения объема области локализации множества экстремумов. На этом же принципе в [12] построены методы «переменной метрики» для минимизации выпуклой функции $f(x)$ при известном значении $f(x^*) = f^*$. Градиентную природу этих методов обеспечивает классический фейеровский шаг в направлении субградиента в преобразованном пространстве аргументов, благодаря априорному знанию f^* , а преобразования пространства направлены на то, чтобы расширить «узкий» конус возможных направлений убывания $f(x)$ посредством преобразования его в «прямоугольный».

Однако преобразования пространства, используемые в [12], не очень подходящие для обоснования методов «переменной метрики» монотонных по функционалу. Для того чтобы это сделать, приходится отказываться от чисто субградиентной природы методов и использовать в качестве направления движения векторы агрегатного типа. В то же время r -алгоритмы являются чисто субградиентными методами, используют на каждом шаге всего два субградиента и, кроме того, активно используют идею наискорейшего спуска в направлении субградиента. В $r_\mu(\alpha)$ -алгоритме и предельном варианте r -алгоритма предполагается

достаточно точная реализация наискорейшего спуска. В $r(\alpha)$ -алгоритме она заменена на адаптивную регулировку шагового множителя, которая ближе к наискорейшему спуску, чем к классическому фейеровскому шагу. В силу этого $r(\alpha)$ -алгоритм принято считать почти монотонным по функционалу.

Однако сходимость r -алгоритмов для минимизации выпуклых функций обоснована лишь в частных случаях [7]. Поэтому хотелось бы построить метод «переменной метрики» субградиентного типа, использующий всего два субградиента и монотонный по функционалу (или почти монотонный, как $r(\alpha)$ -алгоритм), но с обоснованием его сходимости для всего класса выпуклых функций. При этом желательно, чтобы на практике этот метод не уступал $r(\alpha)$ -алгоритму.

Одну из возможностей предоставляет замена операции растяжения пространства в направлении разности двух последовательных субградиентов на преобразование пространства, которое лежит в основе ДФП-метода. Такой путь позволяет построить семейство методов «переменной метрики» чисто субградиентного типа, использующих процедуру наискорейшего спуска в направлении субградиента (или некоторый ее аналог), что обеспечивает возможность создания субградиентных методов монотонных (почти монотонных) по функционалу. Далее ограничимся рассмотрением метода применительно к минимизации гладких выпуклых функций $f(x)$ и анализом этого метода, когда $f(x)$ – квадратичные функции, в том числе и сильно овражные.

Об одном методе «переменной метрики». Для задачи (1), при условии, что мы умеем точно реализовывать процедуру наискорейшего спуска, построим метод «переменной метрики», промежуточный между ДФП-методом и $r_\mu(\alpha)$ -алгоритмом. При этом сохраним как достоинства первого (разумный выбор очередного направления движения), так и второго, в том смысле, что этот метод можно перенести и на негладкий случай.

Пусть g_k и g_{k+1} – градиенты $\varphi_k(y) = f(B_k y)$ в точках y_k и y_{k+1} . Здесь y_{k+1} получена согласно шагу наискорейшего спуска в направлении $-g_k$ из точки y_k в пространстве аргументов $Y_k = B_k^{-1}X$. Тогда $(g_k, g_{k+1}) = 0$. Преобразование пространства из Y_k в Y_{k+1} , которое лежит в основе ДФП-метода, имеет одно «замечательное» свойство – независимо от параметра t_k , движение в пространстве Y_k , которое соответствует направлению образа градиента g_{k+1} в Y_{k+1} , осуществляется по кратчайшему вектору выпуклой комбинации g_k и g_{k+1} . На самом деле, направление движения в Y_k , которое соответствует движению по

\tilde{g}_{k+1} в Y_{k+1} определяются:

$$\begin{aligned}
 p_k &= T_k(g_k, g_{k+1}) T_k^T(g_k, g_{k+1}) g_{k+1} = \\
 &= \left(I - \left(\frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} + t_k \frac{g_k}{\|g_k\|} \right) \left(\frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} \right)^T \right) \times \\
 &\quad \times \left(I - \left(\frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} \right) \left(\frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} + t_k \frac{g_k}{\|g_k\|} \right)^T \right) g_{k+1} = \\
 &= \left(I - \left(\frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} \right) \left(\frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} \right)^T + t_k^2 \frac{g_k}{\|g_k\|} \frac{g_k}{\|g_k\|}^T \right) g_{k+1} = \\
 &= g_{k+1} - (g_{k+1} - g_k) \frac{\|g_{k+1}\|^2}{\|g_{k+1}\|^2 + \|g_k\|^2} = \\
 &= \frac{\|g_{k+1}\|^2}{\|g_{k+1}\|^2 + \|g_k\|^2} g_k + \frac{\|g_k\|^2}{\|g_{k+1}\|^2 + \|g_k\|^2} g_{k+1}, \tag{26}
 \end{aligned}$$

в силу того, что $(g_k, g_{k+1}) = 0$. Вектор p_k , определяемый согласно (26), есть не что иное, как решение следующей задачи:

$$\min \frac{1}{2} \|\lambda_1 g_k + \lambda_2 g_{k+1}\|^2, \tag{27}$$

$$(g_k, g_{k+1}) = 0; \quad \lambda_1 + \lambda_2 = 1; \quad \lambda_1 \geq 0; \quad \lambda_2 \geq 0. \tag{28}$$

Поэтому в основу метода целесообразно поставить преобразование, которое лежит в основе ДФП-метода, так как оно обеспечивает достаточно удачное направление движения на шаге в Y_k . Так, для квадратичных функций $f(x)$ размерностью пространства $n = 2$ такое направление движения из точки наискорейшего спуска будет задавать точное направление на минимум независимо от начальной точки. Поэтому для таких функций при произвольном выборе параметра t_k метод будет требовать не более чем два шага наискорейшего спуска.

Однако сходимость метода за n шагов для квадратичных функций при $n \geq 3$ (аналогично, как в ДФП-методе) будет обеспечивать далеко не каждый выбор параметра t_k . Поэтому отбросим условие окончания процесса минимизации за n шагов для квадратичных функций и выбор t_k подчиним условию, чтобы метод приближался к $r_\mu(\alpha)$ -алгоритму в

том смысле, чтобы преобразование пространства было «сжимающим» пространство субградиентов по типу, как и в $r_\mu(\alpha)$ -алгоритме, т. е.

$$\det(B_{k+1}) = \frac{1}{\alpha} \det(B_k), \quad \alpha > 1, \quad \alpha = 2 - 4.$$

Для этого достаточно для коррекции матрицы $B_{k+1} = B_k T_k(g_k, g_{k+1})$ выбрать

$$T_k(g_k, g_{k+1}) = \left(I - \left(\frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} + \frac{1}{\alpha} \frac{\|g_{k+1} - g_k\|}{\|g_k\|} \frac{g_k}{\|g_k\|} \right) \times \right. \\ \left. \times \left(\frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} \right)^T \right), \quad (29)$$

так как

$$\begin{aligned} \det(T_k(g_k, g_{k+1})) &= \\ &= 1 - \left(\frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} + \frac{1}{\alpha} \frac{\|g_{k+1} - g_k\|}{\|g_k\|} \frac{g_k}{\|g_k\|}, \frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} \right) = \\ &= 1 - 1 - \frac{1}{\alpha} \frac{\|g_{k+1} - g_k\|}{\|g_k\|} \left(\frac{g_k}{\|g_k\|}, \frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} \right) = \\ &= \frac{1}{\alpha} \frac{\|g_{k+1} - g_k\|}{\|g_k\|} \frac{\|g_k\|}{\|g_{k+1} - g_k\|} = \frac{1}{\alpha}. \end{aligned}$$

Из вышеприведенных соображений приходим к следующему методу «переменной метрики» (условимся называть его $\text{dfpr}(\alpha)$ -алгоритмом) применительно к решению задачи (1).

dfpr(α)-алгоритм

Перед началом вычислений имеем $\alpha > 1$, $\varepsilon_g > 0$, $x_0 \in R^n$, $B_0 = I_n$ — единичная матрица размером $n \times n$, $g_0 = \nabla f(x_0)$. Здесь ε_g — достаточно малое число, задающее критерий останова по норме градиента.

Если $\|g_0\| \leq \varepsilon_g$, то x_0 — искомая точка и останов. В противном случае переходим к очередному шагу.

Пусть на k -м шаге получены $x_k \in R^n$, $g_k \in R^n$, B_k — матрица $n \times n$. Здесь $g_k = \nabla f(x_k)$ — градиент $f(x)$ в точке x_k . Тогда $(k+1)$ -й шаг характеризует следующая последовательность операций.

1. Положить $\tilde{g}_k = B_k^T g_k$.
2. Положить $\xi_k = B_k \tilde{g}_k$.
3. Вычислить $h_k = \operatorname{argmin}_{h \geq 0} \{f(x_k - h\xi_k)\}$.
4. Вычислить очередное приближение $x_{k+1} = x_k - h_k \eta_k$.
5. Положить $g_{k+1} = \nabla f(x_{k+1})$. Если $\|g_{k+1}\| \leq \varepsilon_g$, то останов и x_{k+1} – искомая точка. В противном случае вычислим

$$\tilde{g}_{k+1} = B_k^T g_{k+1}, \quad t_k = \frac{1}{\alpha} \sqrt{1 + \frac{\|\tilde{g}_{k+1}\|^2}{\|\tilde{g}_k\|^2}},$$

$$\eta_1 = \frac{\tilde{g}_{k+1} - \tilde{g}_k}{\|\tilde{g}_{k+1} - \tilde{g}_k\|} + t_k \frac{\tilde{g}_k}{\|\tilde{g}_k\|}, \quad \eta_2 = \frac{\tilde{g}_{k+1} - \tilde{g}_k}{\|\tilde{g}_{k+1} - \tilde{g}_k\|},$$

$$B_{k+1} = B_k \left(I - \eta_1 \eta_2^T \right).$$

6. Переходим к очередному шагу с x_{k+1} , B_{k+1} , g_{k+1} .

Очевидно, что сходимость $\operatorname{dfpr}(\alpha)$ -алгоритма к решению x^* для задачи (1) будет обеспечивать процедура наискорейшего спуска в направлении антиградиента. Кроме того, он как и ДФП-метод и r -алгоритмы использует преобразование пространства для того, чтобы расширить конус возможных направлений убывания функции на следующем шаге метода. На самом деле, угол между последовательными градиентами в Y_{k+1} определяет следующее соотношение:

$$\cos \psi_k = 1 / \sqrt{1 + \left(\frac{1}{\alpha} \left(\frac{\|g_k\|}{\|g_{k+1}\|} + \frac{\|g_{k+1}\|}{\|g_k\|} \right) \right)^2}, \quad (30)$$

и чем больше α , тем сильнее он будет уменьшаться. При $\alpha = \infty$ $\operatorname{dfpr}(\alpha)$ -алгоритм будет равносильен предельному варианту r -алгоритмов и для квадратичных функций будет обеспечивать сходимость к x^* за n шагов. Из (30) следует, что если в $\operatorname{dfpr}(\alpha)$ -алгоритме заменить постоянный коэффициент α на переменный $\alpha_k = \frac{\|g_k\|}{\|g_{k+1}\|} + \frac{\|g_{k+1}\|}{\|g_k\|} \geq 2$, то угол между последовательными градиентами будет уменьшаться ровно в два раза, т. е. от $\pi/2$ до $\pi/4$.

Еще одну интересную модификацию $\text{dfpr}(\alpha)$ -алгоритма обеспечивает следующий выбор α_k :

$$\alpha_k = \frac{\|g_{k+1}\|}{\|g_k\|} \left(\frac{\|g_k\|}{\|g_{k+1}\|} + \frac{\|g_{k+1}\|}{\|g_k\|} \right) = 1 + \frac{\|g_{k+1}\|^2}{\|g_k\|^2} > 1,$$

при котором угол между последовательными градиентами будет уменьшаться согласно соотношению

$$\cos \psi_k = 1 / \sqrt{1 + \frac{\|g_{k+1}\|^2}{\|g_k\|^2}}.$$

При таком выборе α_k очень простой вид принимают как одноранговая матрица обратного преобразования $T_k(g_k, g_{k+1})$

$$T_k(g_k, g_{k+1}) = I - \frac{1}{\|g_{k+1} - g_k\|^2} g_{k+1} (g_{k+1} - g_k)^T,$$

так и матрица преобразования пространства аргументов

$$T_k^{-1}(g_k, g_{k+1}) = I - \frac{1}{\|g_k\|^2} g_{k+1} (g_{k+1} - g_k)^T.$$

Учитывая, что преобразование пространства направлено на уменьшение угла между двумя последовательными градиентами, $\text{dfpr}(\alpha)$ -алгоритм должен быть эффективен для минимизации гладких овражных функций в широком диапазоне значений параметра α , в том числе и при $\alpha = 2 - 4$. Подтверждением этому могут служить приведенные в табл. 1, 2 результаты численных экспериментов для минимизации квадратичных функций

$$f(x) = \frac{1}{2} x^T A x = \frac{1}{2} \sum_i^n q^{i-1} x_i^2 = \text{Quad}(q, n),$$

в том числе и сильно овражных. Для сравнения здесь приведены и результаты работы $r_\mu(\alpha)$ -алгоритма при тех же значениях α и параметре $\mu = 0$ ($r_0(\alpha)$ -алгоритм). Оба метода работали в одинаковых условиях: начальная стартовая точка $x_0 = (1, 1, \dots, 1)$; критерий останова $-\varepsilon_g = 10^{-10}$; шаг наискорейшего спуска вычислялся аналитически $-h_k = \frac{(g_k, \xi_k)}{(A\xi_k, \xi_k)}$ и при этом точность выполнения наискорейшего спуска в Y_k была достаточно высокой, т. е.

$$\left| \left(\frac{\tilde{g}_k}{\|\tilde{g}_k\|}, \frac{\tilde{g}_{k+1}}{\|\tilde{g}_{k+1}\|} \right) \right| < 10^{-14}.$$

Таблица 1. Численные эксперименты при малых значениях α

$Quad(q, n)$	dfpr(α)-алгоритм			$r_0(\alpha)$ -алгоритм		
	$\alpha = 2$	$\alpha = 3$	$\alpha = 4$	$\alpha = 2$	$\alpha = 3$	$\alpha = 4$
$Quad(1.1, 200)$	732	581	508	1168	885	775
$Quad(1.1, 130)$	288	241	218	496	419	398
$Quad(1.1, 70)$	88	79	74	178	176	183
$Quad(1.2, 100)$	337	273	239	627	494	457
$Quad(1.2, 50)$	80	69	66	191	176	173
$Quad(2.0, 30)$	103	83	76	273	218	206

Следовательно, можно считать, что результаты, приведенные в табл. 1 и табл. 2, показывают, насколько преобразование насколько преобразование по типу ДФП-метода лучше, чем растяжение пространства в направлении разности двух последовательных градиентов, второй из которых получен согласно шагу наискорейшего спуска в преобразованном пространстве аргументов. Для несильно овражных квадратичных функций ($Quad(1.1, 70)$, $Quad(1.2, 50)$) работа dfpr(α)-алгоритма при небольших α близка к работе ДФП-метода в том смысле, что обеспечивает сходимость за число шагов, очень близкое к n . Конечно, разрыв между dfpr(α) -алгоритмом и $r_0(\alpha)$ -алгоритмом должен сокращаться при увеличении α , однако это имеет место при очень больших значениях α . Так, при $\alpha = 1000$ (табл. 2), который нельзя считать малым, этот разрыв все еще достаточно большой. Если dfpr(α)-алгоритм гарантирует сходимость к x^* практически за n шагов, то для $r_0(\alpha)$ -алгоритма это не так. Одна из причин такого поведения методов – слишком большая точность решения задач $\varepsilon_g = 10^{-10}$, что равносильно $f(x_k) - f^* < 10^{-20}$. В случае менее жесткого критерия останова разрыв будет меньше, но и число итераций для dfpr(α)-алгоритма будет меньше. Однако такая точность по функционалу говорит в пользу dfpr(α)-алгоритма в том смысле, что он достаточно устойчив к точности решения даже для сильно овражных задач.

Таблица 2. Численные эксперименты при больших значениях α

$Quad(q, n)$	dfpr(α)-алгоритм			$r_0(\alpha)$-алгоритм		
	$\alpha = 10$	$\alpha = 100$	$\alpha = 1000$	$\alpha = 10$	$\alpha = 100$	$\alpha = 1000$
$Quad(1.1, 200)$	379	271	221	702	692	550
$Quad(1.1, 130)$	177	131	130	391	384	290
$Quad(1.1, 70)$	70	70	70	212	177	140
$Quad(1.2, 100)$	181	133	107	422	365	276
$Quad(1.2, 50)$	54	50	50	185	143	106
$Quad(2.0, 30)$	58	42	36	178	120	87

Следовательно, dfpr(α)-алгоритм можно считать более устойчивым, чем $r_0(\alpha)$ -алгоритм, а точнее – преобразование пространства по типу ДФП-метода следует признать более рациональным, чем растяжение пространства в направлении разности двух последовательных градиентов.

К субградиентным методам «переменной метрики», использующим преобразование пространства по типу ДФП-метода.

dfpr(α)-алгоритм имеет «идеализированный» характер в том смысле, что он предполагает точную процедуру наискорейшего спуска, которая в принципе нереализуема даже для гладких функций. Кроме того, для негладких функций точная процедура наискорейшего спуска гарантирует более сильное условие, чем существование антисубградиента, ортогонального направлению спуска. В частности, угол может быть тупым, т. е. $(g_k, g_{k+1}) < 0$. Поэтому перенос основных принципов dfpr(α)-алгоритма на общий случай минимизации выпуклых функций требует решения двух главных вопросов. Первый – замена преобразования (29) преобразованием, которое бы обеспечивало выбор направления движения, как в dfpr(α)-алгоритме, не только при ортогональных последовательных субградиентах. Второй – замена точного наискорейшего спуска в направлении субградиента некоторой другой регулировкой шагового множителя, которая бы требовала небольшого числа вычислений $f(x)$ и $\partial f(x)$.

Первый вопрос не представляет особых проблем, поскольку независимо от угла между g_k и g_{k+1} в $Y_k = B_k^{-1}X$, выбор направления движения как в $\text{dfpr}(\alpha)$ -алгоритме можно сохранить, если для коррекции матрицы $B_{k+1} = B_k T_k(g_k, g_{k+1})$ выбрать

$$T_k(g_k, g_{k+1}) = \left(I - \left(\frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} + t_k \left(g_k - \frac{(g_{k+1}, g_k)}{\|g_{k+1}\|^2} g_{k+1} \right) \right) \times \right. \\ \left. \times \left(\frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} \right)^T \right). \quad (31)$$

Здесь t_k – некоторый скалярный параметр, от выбора которого зависит сходимость методов.

Второй вопрос, связанный с регулировкой шагового множителя в направлении субградиента, более сложный и приводит к разного рода модификациям. Так, классический фейеровский шаг в направлении субградиента или некоторый его аналог, когда f^* неизвестно, приводит к немонотонным по функционалу субградиентным методам «переменной метрики» на основе преобразования (31). При этом очевидно, что преобразование пространства требуется на тех шагах метода, когда для кратчайшей выпуклой комбинации векторов g_k и g_{k+1} выполнено $\lambda_1 > 0$ и $\lambda_2 > 0$. Преобразование пространства (31) сильнее расширяет конус возможных направлений убывания функции, чем «ортогонализация» в подобных методах [12]. Обоснование сходимости таких методов можно построить аналогично тому, как это сделано в [12], убрав даже требование на ограниченность евклидовой нормы матрицы B_k . Такой способ регулировки шагового множителя требует не более одного вычисления $f(x)$ и $\partial f(x)$ на каждом шаге метода и позволяет за конечное число шагов K либо найти $f(x_K)$ с требуемым значением функции, либо получить достаточные условия расходимости фейеровского процесса, когда вместо f^* используется заниженное значение.

Адаптивный способ регулировки шагового множителя [8], который используется в $r(\alpha)$ -алгоритме, позволяет получить почти монотонные по функционалу методы. Формально он приводит к замене точного поиска минимума по направлению на приближенный, т. е. $h_k > h_k^*$, но так, чтобы h_k было близко к h_k^* . Здесь h_k^* – точный шаг наискорейшего спуска в направлении антисубградиента. Учитывая, что гарантированное уменьшение нормы субградиента в преобразованном пространстве позволяет просто уточнять шаг наискорейшего спуска в преобразованном пространстве, при такой регулировке шагового множителя достаточно на каждом шаге метода использовать в среднем 2-3 вычисления

$f(x)$ и $\partial f(x)$. При адаптивной регуливке шагового множителя можно переходить и в точки с меньшим шагом, чем h_k^* , так одновременное выполнение условий $\|g_k\|^2 > (g_k, g_{k+1})$ и $\|g_{k+1}\|^2 > (g_k, g_{k+1})$ позволяет применять преобразование (31) и когда $(g_k, g_{k+1}) > 0$.

Итак, для минимизации выпуклых функций как первый, так и второй способы регуливки шагового множителя позволяют получить практически реализуемые субградиентные методы «переменной метрики» на основе преобразования по типу ДФП-метода в условиях минимального использования информации (только два последовательных субградиента).

Отметим, что вышеобсуждаемые субградиентные методы «переменной метрики» более соответствуют названию работы [13], чем методы, предлагаемые в этой работе, которые послужили основой для создания ϵ -субградиентных методов в негладкой оптимизации. На самом деле, в ϵ -субградиентных методах из ДФП-метода взят только принцип движения по направлению, противоположному кратчайшему вектору к выпуклой оболочке двух последовательных субградиентов, и обобщен на случай большего числа векторов. Однако содержательный смысл ДФП-метода состоит не столько в выборе такого направления движения, сколько в применяемом линейном преобразовании пространства, которое позволяет улучшать структуру поверхностей уровня минимизируемой функции.

Литература

1. ПОЛАК Э. Численные методы оптимизации. Единый подход. — М.: Мир, 1974. — 376 с.
2. ХИММЕЛЬБЛАУ Д. Прикладное нелинейное программирование. — М.: Мир, 1975. — 535 с.
3. ПШЕНИЧНЫЙ Б. Н., ДАНИЛИН Ю. М. Численные методы в экстремальных задачах. — М.: Наука, 1975. — 319 с.
4. ПОЛЯК Б. Т. Введение в оптимизацию. — М.: Наука, 1983. — 384с.
5. DAVIDON W.C. Variable metric methods for minimization, AEC Research and Development Rept. ANL 5990 (Rev.), 1959.
6. FLETCHER R., POWELL M. J. D. A rapidly convergent descent method for minimization // Comput. J. — 1963. — 6. — №2. — P. 163–168.

7. ШОР Н. З. Методы минимизации недифференцируемых функций и их приложения. – К.: Наук. думка, 1979. – 199 с.
8. ШОР Н. З., СТЕЦЕНКО С. И. Квадратичные экстремальные задачи и недифференцируемая оптимизация. – К.: Наук. думка, 1989. – 208 с.
9. СТЕЦЮК П. И. К вопросу сходимости r -алгоритмов // Кибернетика и систем. анализ. – 1995. – №6. – С. 173–177.
10. ШОР Н. З., БИЛЕЦКИЙ В. И. Метод растяжения пространства для ускорения сходимости в задачах овражного типа // Тр. семинара науч. совета АН УССР по кибернетике "Теория оптимальных решений". – 1969. – №2. – С. 3–18.
11. СТЕЦЮК П. И. Об одной схеме методов отсечений. – Киев, 1995. – 34 с. – (Препр. / НАН Украины. Ин-т кибернетики им. В. М. Глушкова; **95–25**).
12. СТЕЦЮК П. И. Классические фейеровские методы с преобразованием пространства // Методы решения экстремальных задач. – К.: Ин-т кибернетики им. В. М. Глушкова НАН Украины, 1996. – С. 3–9.
13. LEMARÉCHAL C. An extension of Davidon methods to nondifferentiable problems // Math. Progr. Study 3. – 1975. – P. 95–109.

Линейные операторы в квазиньютоновских методах¹

П. И. Стецюк

Теория и приложения методов оптимизации. – Киев, 1998. – С. 3–8.

Анализируются линейные операторы преобразования пространства для двух квазиньютоновских методов: Давидона – Флетчера – Пауэлла и Бройдена – Флетчера – Шенно. Для этих методов дана геометрическая интерпретация градиентного шага в преобразованном пространстве переменных.

По своей структуре r -алгоритмы [1] «близки» к методам квазиньютоновского типа в том смысле, что формулы пересчета симметричной матрицы в модифицированных вычислительных схемах r -алгоритмов [2] очень напоминают формулы пересчета аналогичных матриц в некоторых методах квазиньютоновского типа. Это означает, что как первые, так и вторые используют близкие «антиовражные» приемы, чем объясняется их эффективность при минимизации овражных функций. Однако, если геометрический смысл «антиовражного» приема в r -алгоритмах достаточно «прозрачен» (реализуется с помощью операции растяжения пространства в направлении разности двух последовательных субградиентов [3, с. 112], то для квазиньютоновских методов, которые обычно интерпретируют с позиций методов ньютоновского типа или сопряженных направлений, это не совсем так. Рассмотрим геометрическую интерпретацию некоторых квазиньютоновских методов, используя соответствующие им линейные операторы преобразования пространства.

О квазиньютоновских методах. Для минимизации выпуклой дважды дифференцируемой функции $f(x)$ ($x \in X$, $X = R^n$, R^n – евклидово пространство размерностью n) методы квазиньютоновского типа генерируют последовательность точек $\{x_k, k = 0, 1, 2, \dots\}$ по следующему правилу:

$$x_{k+1} = x_k - h_k H_k \nabla f(x_k), \quad (1)$$

где $\nabla f(x_k)$ – градиент $f(x)$ в точке x_k ; H_k – симметричная матрица размером $n \times n$; h_k – шаг, соответствующий минимуму $f(x)$ в направлении $-H_k \nabla f(x_k)$. Пересчет матрицы H от шага к шагу осуществляется с

¹Работа выполнена при поддержке Федерального Управления Иностранных Дел и Национального Научного Фонда Швейцарии в рамках программы научного сотрудничества между государствами Западной Европы и новыми независимыми государствами.

обязательным выполнением так называемого квазиньютоновского условия

$$H_{k+1} \left(\nabla f(x_{k+1}) - \nabla f(x_k) \right) = -h_k H_k \nabla f(x_k), \quad (2)$$

которое гарантирует, что $H_n \approx [\nabla^2 f(x^*)]^{-1}$, если $f(x)$ хорошо аппроксимируется квадратичной функцией

$$\varphi(x) = \frac{1}{2} \left(\nabla^2 f(x^*) (x - x^*), x - x^* \right) + f(x^*).$$

Любой из способов пересчета матрицы H , удовлетворяющий условию (2), однозначно определяет соответствующий этому способу квазиньютоновский метод. Для метода Давидона – Флетчера – Пауэлла (ДФП-метод) пересчет матрицы H_{k+1} определен формулой

$$H_{k+1} = H_k - \frac{H_k \Delta g_k (\Delta g_k)^T H_k}{(\Delta g_k, H_k \Delta g_k)} + h_k \frac{H_k g_k g_k^T H_k}{(\Delta g_k, H_k g_k)}, \quad (3)$$

а для метода Бройдена – Флетчера – Шенно (БФШ-метод) – формулой

$$\begin{aligned} H_{k+1} = H_k - \frac{H_k g_k (\Delta g_k)^T H_k + H_k \Delta g_k g_k^T H_k}{(\Delta g_k, H_k g_k)} - \\ - \left(h_k - \frac{(\Delta g_k, H_k \Delta g_k)}{(\Delta g_k, H_k g_k)} \right) \frac{H_k g_k g_k^T H_k}{(\Delta g_k, H_k g_k)}. \end{aligned} \quad (4)$$

Здесь $g_k = \nabla f(x_k)$ – градиент в точке, из которой осуществляется движение; $\Delta g_k = \nabla f(x_{k+1}) - \nabla f(x_k)$ – разность градиентов в двух последовательных точках минимизирующей последовательности. Если на каждом шаге ДФП и БФШ-методов шаговый множитель h_k выбирается из условия минимума функции по направлению $-H_k \nabla f(x_k)$, то матрицы H_{k+1} , генерируемые каждым из методов, являются положительно определенными.

Два указанных метода наиболее используемые на практике, причем некоторое предпочтение отдается БФШ-методу. Почему это так, попытаемся разобраться с помощью анализа линейных операторов, которые лежат в основе этих методов.

Линейные операторы в ДФП- и БФШ-методах. Для положительно определенной матрицы H_{k+1} квазиньютоновское условие можно записать так:

$$B_{k+1} B_{k+1}^T \left(\nabla f(x_{k+1}) - \nabla f(x_k) \right) = -h_k B_k B_k^T \nabla f(x_k), \quad (5)$$

где B_k и B_{k+1} – вещественные матрицы размером $n \times n$.

Пусть матрица $B_{k+1} = B_k T_k$, где T_k – неособенная матрица размером $n \times n$. Обозначим $g_k = B_k^T \nabla f(x_k)$, $g_{k+1} = B_{k+1}^T \nabla f(x_{k+1})$. Они являются градиентами функции $\varphi_k(y) = f(B_k y)$, определенной в преобразованном линейным оператором $A_k = B_k^{-1}$ пространстве переменных $Y_k = A_k X$.

Применительно к преобразованному пространству переменных Y_k квазиньютоновское условие (5) можно записать таким образом:

$$T_k T_k^T (g_{k+1} - g_k) = -h_k g_k. \quad (6)$$

Условие (6) выполняется как для однорангового оператора

$$\begin{aligned} T_1(g_k, g_{k+1}) = \\ = I - \left(\frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} - \sqrt{\frac{h_k(g_k, g_k)}{(g_k, g_k - g_{k+1})} \frac{g_k}{\|g_k\|}} \right) \left(\frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} \right)^T, \end{aligned} \quad (7)$$

так и для однорангового оператора

$$\begin{aligned} T_2(g_k, g_{k+1}) = \\ = I + \frac{g_k}{(g_k - g_{k+1}, g_k)} \left(\sqrt{\frac{h_k(g_k, g_k - g_{k+1})}{(g_k, g_k)}} g_k + g_{k+1} - g_k \right)^T. \end{aligned} \quad (8)$$

Оператор $T_1(g_k, g_{k+1})$ будем называть ДФП-оператором, так как он обеспечивает пересчет матрицы H_{k+1} согласно (3), а оператор $T_2(g_k, g_{k+1})$ будем называть БФШ-оператором, так как он соответствует пересчету матрицы H_{k+1} по формуле (4). Отметим, что здесь речь идет об обратных операторах к операторам преобразования пространства переменных (именно они используются в вычислительных схемах методов переменной метрики). Преобразование пространства переменных реализуется для ДФП-метода согласно оператору $T_1^{-1}(g_k, g_{k+1})$, а для БФШ-метода – согласно оператору $T_2^{-1}(g_k, g_{k+1})$.

Некоторые свойства ДФП- и БФШ-операторов. Рассмотрим один шаг, связанный с переходом из преобразованного пространства переменных Y_k в очередное преобразованное пространство переменных $Y_{k+1} = T_k^{-1} Y_k$. Градиенты функции $\varphi_k(y)$ в двух последовательных точках y_k и y_{k+1} преобразованного пространства переменных Y_k будем обозначать как и раньше: g_k и g_{k+1} . Здесь точка y_{k+1} получена согласно шагу наискорейшего спуска h_k (как правило, приближенного) в направлении $-g_k$ из точки y_k . Точный шаг наискорейшего спуска будем обозначать h_k^* , в этом случае $(g_k, g_{k+1}) = 0$.

1. При переходе в очередное преобразованное пространство переменных $Y_{k+1} = T_k^{-1}Y_k$ пространство градиентов функции $\varphi_{k+1}(y)$ пересчитывается по формуле $\nabla\varphi_{k+1}(y) = T_k^T\nabla\varphi_k(y)$. Поэтому градиент, по которому будет осуществляться процесс спуска в преобразованном пространстве переменных Y_{k+1} , для ДФП-метода будет иметь вид

$$\begin{aligned}\tilde{g}_{k+1} &= \frac{(g_{k+1}, g_{k+1} - g_k)}{\|g_{k+1} - g_k\|^2} g_k + \frac{(g_k, g_k - g_{k+1})}{\|g_{k+1} - g_k\|^2} g_{k+1} + \\ &\quad + \sqrt{\frac{h_k(g_k, g_{k+1})^2}{(g_k, g_k - g_{k+1})}} \cdot \frac{g_{k+1} - g_k}{\|g_{k+1} - g_k\|} \sim \\ &\sim \frac{\|g_{k+1}\|^2}{\|g_{k+1}\|^2 + \|g_k\|^2} g_k + \frac{\|g_k\|^2}{\|g_{k+1}\|^2 + \|g_k\|^2} g_{k+1},\end{aligned}\quad (9)$$

а для БФШ-метода

$$\begin{aligned}\tilde{g}_{k+1} &= \frac{(g_k, g_k)}{(g_k - g_{k+1}, g_k)} g_{k+1} + \\ &+ \frac{(g_k, g_{k+1})}{(g_k - g_{k+1}, g_k)} \left(\sqrt{\frac{h_k(g_k, g_k - g_{k+1})}{(g_k, g_k)}} - 1 \right) g_k \sim g_{k+1}.\end{aligned}\quad (10)$$

При точной реализации наискорейшего спуска для ДФП-метода \tilde{g}_{k+1} в точности совпадает с минимальной по норме выпуклой комбинацией градиентов g_k и g_{k+1} , а для БФШ-метода \tilde{g}_{k+1} совпадает с g_{k+1} . Этим обстоятельством в какой-то мере объясняется практическое превосходство БФШ-метода над ДФП-методом. Дело в том, что преобразование пространства, которое переводит градиент в кратчайший вектор выпуклой комбинации двух последовательных градиентов, более чувствительно к точности процедуры наискорейшего спуска (особенно когда нормы векторов g_k и g_{k+1} сильно различаются), чем преобразование пространства, которое сохраняет неизменяющимся последний вычисленный градиент.

2. Направление движения в Y_k из точки $y_{k+1} = B_k^{-1}x_{k+1}$, которое будет соответствовать направлению движения в Y_{k+1} по градиенту \tilde{g}_{k+1} , определяется по формуле $p_{k+1} = T_k T_k^T \tilde{g}_{k+1} = T_k \tilde{g}_{k+1}$. Для ДФП-метода

$$p_{k+1} = \frac{(g_{k+1}, g_{k+1} - g_k)}{\|g_{k+1} - g_k\|^2} g_k + \frac{(g_k, g_k - g_{k+1})}{\|g_{k+1} - g_k\|^2} g_{k+1} +$$

$$\begin{aligned}
& + \frac{h_k(g_k, g_{k+1})}{(g_k, g_k - g_{k+1})} g_k \sim \\
& \sim \frac{\|g_{k+1}\|^2}{\|g_{k+1}\|^2 + \|g_k\|^2} g_k + \frac{\|g_k\|^2}{\|g_{k+1}\|^2 + \|g_k\|^2} g_{k+1}, \quad (11)
\end{aligned}$$

а для БФШ-метода

$$\begin{aligned}
p_{k+1} &= \frac{(g_k, g_k)}{(g_k - g_{k+1}, g_k)} g_{k+1} + \\
&+ \left(\frac{h_k(g_k, g_{k+1})}{(g_k, g_k)(g_k - g_{k+1}, g_k)} + \frac{(g_k, g_k) \cdot (g_{k+1} - g_k, g_{k+1})}{(g_k - g_{k+1}, g_k)^2} \right) g_k \sim \\
&\sim g_{k+1} + \frac{\|g_{k+1}\|^2}{\|g_k\|^2} g_k. \quad (12)
\end{aligned}$$

Из формул (11), (12) видим, что при точном наискорейшем спуске для обоих методов направление движения совпадает с направлением кратчайшей выпуклой комбинации двух последовательных градиентов (свойство сопряженных направлений). Однако, если для ДФП-метода p_k в точности равен кратчайшему вектору, то для БФШ-метода — с точностью до некоторого множителя.

3. Практическая эффективность квазиньютоновских методов (в общем случае — методов переменной метрики в H -форме) во многом зависит от того, насколько хорошо сохраняется положительная определенность матрицы H_k на каждом шаге. Для указанных методов при достаточно точной реализации наискорейшего спуска теоретически это не представляет проблем. Однако при минимизации существенно овражных функций ошибки вычислений будут по-разному влиять на положительную определенность матриц H_k .

Рассмотрим определители матриц обратных преобразований: для ДФП-оператора

$$\det(T_1(g_k, g_{k+1})) = \sqrt{h_k \frac{(g_k, g_k - g_{k+1})}{\|g_{k+1} - g_k\|^2}} \sim \sqrt{h_k^* \frac{\|g_k\|^2}{\|g_{k+1}\|^2 + \|g_k\|^2}}, \quad (13)$$

для БФШ-оператора

$$\det(T_2(g_k, g_{k+1})) = \sqrt{h_k \frac{(g_k, g_k)}{(g_k, g_k - g_{k+1})}} \sim \sqrt{h_k^*}. \quad (14)$$

Из формул (13), (14) следует, что БФШ-метод должен обеспечивать численно более устойчивый пересчет матрицы H_{k+1} , чем ДФП-метод, так как он лучше сохраняет положительную определенность матрицы H_{k+1} при небольшом шаге h_k^* .

4. И, наконец, рассмотрим такую характеристику, как изменение угла между последовательными градиентами при переходе в преобразованное пространство переменных Y_{k+1} . Для указанных методов при точном наискорейшем спуске угол уменьшается от $\pi/2$ к острому ψ_k , косинус которого для ДФП-метода определяется из условия

$$\cos \psi_k = \left(\frac{\tilde{g}_k}{\|\tilde{g}_k\|}, \frac{\tilde{g}_{k+1}}{\|\tilde{g}_{k+1}\|} \right) = 1 / \sqrt{1 + h_k^* \frac{\|g_k\|^2 + \|g_{k+1}\|^2}{\|g_k\|^2}},$$

а для БФШ-метода – из условия

$$\cos \psi_k = \left(\frac{\tilde{g}_k}{\|\tilde{g}_k\|}, \frac{\tilde{g}_{k+1}}{\|\tilde{g}_{k+1}\|} \right) = 1 / \sqrt{1 + h_k^* \frac{\|g_k\|^2}{\|g_{k+1}\|^2}}.$$

Уменьшение угла между последовательными градиентами свойственно и для итерации r -алгоритмов при минимизации гладких функций. Так, оператор растяжения пространства в направлении разности двух последовательных градиентов с коэффициентом α ($\alpha > 1$) преобразует прямой угол в острый ψ_k , косинус которого равен

$$\cos \psi_k = (\alpha^2 - 1) / \sqrt{\alpha^4 + \alpha^2 \left(\frac{\|g_k\|^2}{\|g_{k+1}\|^2} + \frac{\|g_{k+1}\|^2}{\|g_k\|^2} \right)} + 1 \leq 1 - \frac{2}{\alpha^2 + 1}.$$

Поэтому можно считать, что преобразование пространства в ДФП- и БФШ-методах имеют такую же направленность, как и в r -алгоритмах, т. е. «расширяется» конус возможных направлений убывания функции в преобразованном пространстве переменных.

Для минимизации выпуклых функций ДФП- и БФШ-операторы позволяют строить методы переменной метрики по типу r -алгоритмов с тем, чтобы уменьшить степень вытянутости поверхностей функции в преобразованном пространстве переменных. При этом методы в B -форме (корректируется матрица обратного преобразования) всегда гарантируют положительную определенность $B_k B_k^T$ и, как следствие, будут численно более устойчивыми, чем методы в H -форме.

Литература

1. Шор Н. З. Методы минимизации недифференцируемых функций и их приложения. – К.: Наук. думка, 1979. – 199 с.
2. Скоков В.А. Замечания к методам минимизации, использующим операцию растяжения пространства // Кибернетика. – 1974. – №4. – С. 115–117.
3. Михалевич В. С., Трубин В. А., Шор Н. З. Оптимизационные задачи производственно-транспортного планирования. – М.: Наука, 1986. – 264 с.

Субградиентные методы с преобразованием пространства для минимизации овражных выпуклых функций

П. И. Стецюк

*Современные проблемы прикладной математики и механики:
теория, эксперимент и практика / Международная конференция,
посвященная 90-летию со дня рождения академика Н.Н. Яненко,
Новосибирск, Россия, 30 мая – 4 июня 2011 г.*

Приведены Octave-функция `ralgb5` (реализует r -алгоритм с адаптивной регулировкой шага и постоянным коэффициентом растяжения пространства) и алгоритм-функция `amsg2p` (реализует релаксационный метод с преобразованием пространства, которое использует два последовательных субградиента и агрегатный вектор, являющийся выпуклой комбинацией вычисленных ранее субградиентов). Даны результаты экспериментов для существенно овражной кусочно-квадратичной и кусочно-линейной функции, которая связана с разрешимостью интервальной линейной задачи о допусках.

Функции с разрывным градиентом часто встречаются в различных задачах интервального анализа как следствие кусочной гладкости интервальных арифметических операций. Поэтому владение численными методами минимизации негладких выпуклых функций дает разработчику действенный инструмент при построении эффективных алгоритмов для решения задач математического программирования в интервальном анализе. Подробное изложение основных источников, порождающих задачи негладкой оптимизации, и приложение к ним методов минимизации недифференцируемых функций можно найти в [1]. Очень информативны на этот счет сборники научных трудов [2, 3], в которые включены основные работы Н.З. Шора по методам негладкой оптимизации и их применению в задачах большой размерности, квадратичного, булевого, матричного и стохастического программирования. Сборники доступны в электронном варианте по адресу <http://elis.dvo.ru/?q=node/114> (режим доступа — свободный).

Негладкие выпуклые функции, как правило, характеризуются овражной или существенно овражной структурой поверхностей уровня. Качество методов минимизации негладких выпуклых функций главным

образом определяется способностью избежать зигзагообразной траектории итерационного процесса в точках близких ко дну «русла оврага». Если метод с этим легко справляется то, как правило, этот метод обладает ускоренной сходимостью при минимизации негладких функций. Ниже обсудим алгоритмы из двух семейств субградиентных методов оптимизации с преобразованием пространства, имеющих ускоренную сходимость при минимизации выпуклых функций с овражной структурой поверхностей уровня.

Первый метод — $r(\alpha)$ -алгоритм (реализован функцией `ralgb5` на языке Octave [4]) относится к семейству субградиентных методов минимизации негладких функций, которое известно как r -алгоритмы Н.З. Шора. r -Алгоритмы базируются на процедуре наискорейшего спуска в преобразованном пространстве переменных и обеспечивают монотонность (или почти монотонность) по значениям минимизируемой функции. r -Алгоритмы используют операцию растяжения пространства в направлении разности двух последовательных субградиентов, которая улучшает свойства овражной функции в преобразованном пространстве переменных.

Второй метод (реализован алгоритм-функцией `amsg2p`) относится к семейству методов минимизации негладких функций, которые используют релаксационный шаг (известен как шаг Поляка или шаг Агмона-Моцкина), и использует априорное знание минимального значения функции. Здесь применяется антиовражная техника, подобно тому, как это сделано в r -алгоритмах Шора. Но преобразование пространства реализуется с помощью линейного оператора [5], который позволяет обеспечить уменьшение расстояния до точки минимума в очередном преобразованном пространстве переменных.

1. Octave-функция `ralgb5`

Программа находит точку минимума x_r^* выпуклой функции $f(x)$ от n переменных и делает это с помощью $r(\alpha)$ -алгоритма — вариант r -алгоритмов с постоянным на каждой итерации коэффициентом растяжения пространства α ($\alpha > 1$) и адаптивной регулировкой шага в направлении нормированного антисубградиента. Программа использует Octave-функцию `function [f,g] = calcfg(x)`, которая вычисляет значение функции $f = f(x)$ и её субградиента $g = \partial f(x)$ в точке x . Программа использует следующие параметры.

% Входные параметры:

```
%   calcfg -- имя функции вида calcfg(x) для вычисления f и g
%   x -- начальная точка x(n) (на выходе портится)
%   alpha -- коэффициент растяжения пространства
%   h0, nh, q1, q2 -- параметры адаптивной регулировки шага
%   epsx, epsg, maxitn -- параметры останова
%   Выходные параметры:
%   xr -- найденная точка минимума функции xr(n)
%   fr -- значение функции в точке минимума
%   itn -- число затраченных итераций
%   ncalls -- число вызовов функции calcfg
%   istop -- код останова (2=epsg, 3=epsx, 4=maxitn, 5=error)
```

Адаптивная регулировка шага в $r(\alpha)$ -алгоритме выполняет одномерный спуск в направлении нормированного антисубградиента в преобразованном пространстве переменных и реализуется с помощью параметров h_0 , q_1 , n_h , q_2 . Здесь h_0 — величина начального шага (используется на 1-й итерации, на каждой последующей итерации эта величина уточняется); q_1 — коэффициент уменьшения шага ($q_1 \leq 1$), если условие завершения спуска по направлению выполняется за один шаг; q_2 — коэффициент увеличения шага ($q_2 \geq 1$); натуральное число n_h задает число шагов одномерного спуска ($n_h > 1$), через каждые из которых шаг будет увеличиваться в q_2 раз. Подробные рекомендации по выбору коэффициента растяжения пространства и параметров адаптивной регулировки шага даны в [6], с. 45–47. Их суть состоит в том, чтобы адаптивный способ регулировки шага позволял увеличивать точность поиска минимума функции по направлению в процессе счета и при этом число шагов по направлению не должно быть большим.

Параметры ε_x и ε_g определяют условия завершения $r(\alpha)$ -алгоритма: метод останавливается в точке x_{k+1} , если выполнено $\|x_{k+1} - x_k\| \leq \varepsilon_x$ (останов по аргументу); метод останавливается в точке x_{k+1} , если выполнено условие $\|g_f(x_{k+1})\| \leq \varepsilon_g$ (останов по норме субградиента, используется для гладких функций). Аварийное завершение программы связано либо с тем, что функция $f(x)$ неограничена снизу, либо h_0 слишком мал и его требуется увеличить.

```
# ralgb5 -- Octave-function for Shor's r-algorithm
function [xr,fr,itn,ncalls,istop]=ralgb5(cfg,x,alpha,h0,q1,
                                     q2,nh,epsg,epsx,maxitn);

itn=0; hs=h0; B=eye(length(x)); xr=x; # row001
ncalls = 1; [fr,g0] = cfg(xr); # row002
printf("itn %4d f %14.6e fr %14.6e ls %2d ncalls %4d\n", # row003
       itn, fr, fr, 0, ncalls);
```



```

if(norm(g0) < epsg) istop = 2; return; endif # row004
for (itn = 1:maxitn) # row005
    dx = B * (g1 = B' * g0)/norm(g1); # row006
    d = 1; ls = 0; ddx = 0; # row007
    while (d > 0) # row008
        x -= hs * dx; ddx += hs * norm(dx); # row009
        ncalls ++; [f, g1] = calcfg(x); # row010
        if (f < fr) fr = f; xr = x; endif # row011
        if(norm(g1) < epsg) istop = 2; return; endif # row012
        ls ++; (mod(ls,nh)==0) && (hs *= q2); # row013
        if(ls > 500) istop = 5; return; endif # row014
        d = dx' * g1; # row015
    endwhile # row016
    (ls == 1) && (hs *= q1); # row017
    printf("itn %4d f %14.6e fr %14.6e ls %2d ncalls %4d\n", # row018
        itn, f, fr, ls, ncalls);
    if(ddx < epsx) istop = 3; return; endif # row019
    xi = (dg = B' * (g1 - g0))/norm(dg); # row020
    B += (1 / alpha - 1) * B * xi * xi'; # row021
    g0 = g1; # row022
endfor # row023
istop = 4; # row024
endfunction

```

При минимизации негладких функций рекомендуется следующий выбор параметров: $\alpha = 2 \div 3$, $h_0 = 1.0$, $q_1 = 1.0$, $q_2 = 1.1 \div 1.2$, $n_h = 2 \div 3$. Если известна априорная оценка расстояния от начальной точки x_0 до точки минимума x^* , то начальный шаг h_0 целесообразно выбирать порядка $\|x_0 - x^*\|$. При минимизации гладких функций рекомендуемые параметры такие же, за исключением q_1 ($q_1 = 0.8 \div 0.95$). Это обусловлено тем, что дополнительное измельчение шага способствует увеличению точности поиска минимума функции по направлению, что при минимизации гладких функций обеспечивает более быструю скорость сходимости. При таком выборе параметров, как правило, число спусков по направлению редко превосходит два, а за n шагов точность по функции улучшается в три-пять раз. Параметры останова $\varepsilon_x, \varepsilon_g \sim 10^{-6} \div 10^{-5}$ при минимизации выпуклой функции даже существенно овражной структуры обеспечивает нахождение x_r^* со значением функции, достаточно близким к оптимальному. При этом обычно

$$\frac{f(x_r^*) - f(x^*)}{|f(x^*)| + 1} \sim 10^{-6} \div 10^{-5} \text{ — для негладких}$$

и

$$\frac{f(x_r^*) - f(x^*)}{|f(x^*)| + 1} \sim 10^{-12} \div 10^{-10} \text{ — для гладких функций,}$$

что подтверждается результатами многочисленных тестовых и реальных расчетов.

2. Алгоритм-функция `amsg2p`

Алгоритм `amsg2p` находит точку минимума выпуклой функции $f(x)$ при известном её минимальном значении f^* . В его основу положен второй из субградиентных методов с преобразованием пространства и регуляризацией шага Агмона-Мощкина-Шонберга (AMS-шаг) в преобразованном пространстве переменных [7]. Алгоритм использует величину максимального сдвига по выпуклости функции $f(x)$ (задается параметром γ), для которой субградиент $\partial f(x)$ удовлетворяет следующему условию:

$$\langle x - x^*, \partial f(x) \rangle \geq \gamma(f(x) - f^*), \quad \text{где } \gamma \geq 1, \quad (1)$$

для любого $x \in R^n$ и произвольного x^* из множества X^* точек минимума функции $f(x)$. Неравенство (1) позволяет реализовать более сильные AMS-шаги для специальных классов функций. Так, например, для квадратичной гладкой функции рекомендуется использовать $\gamma = 2$.

$$\text{Алгоритм-функция } \text{amsg2p}: (x_\varepsilon^*, k_\varepsilon^*) = \text{amsg2p}(x_0, \varepsilon, f^*, \gamma)$$

На итерации $k = 0$ имеем начальное приближение $x_0 \in R^n$ и достаточно малое $\varepsilon > 0$. Вычислим $f(x_0)$ и $\partial f(x_0)$. Если $f(x_0) - f^* \leq \varepsilon$, то $x_\varepsilon^* = x_0$, $k_\varepsilon^* = 0$ и окончание работы алгоритма. Иначе положим

$$h_0 = \frac{\gamma(f(x_0) - f^*)}{\|\partial f(x_0)\|}, \quad \xi_0 = \frac{\partial f(x_0)}{\|\partial f(x_0)\|} \in R^n, \quad p_0 = 0 \in R^n,$$

$B_0 = I_n$ — единичная матрица размера $n \times n$. Перейдем к следующей итерации.

Пусть на k -й итерации получены $x_k \in R^n$, $h_k, \xi_k \in R^n$, $p_k \in R^n$, B_k — матрица $n \times n$. Для $(k + 1)$ -й итерации выполним пп. 1–5.

1) Вычислим очередное приближение

$$x_{k+1} = x_k - h_k B_k \xi_k.$$

- 2) Вычислим $f(x_{k+1})$ и $\partial f(x_{k+1})$. Если $f(x_{k+1}) - f^* \leq \varepsilon$, то $x_\varepsilon^* = x_{k+1}$, $k_\varepsilon^* = k + 1$ и окончание алгоритма. Иначе положим

$$\xi_{k+1} = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|}, \quad h_{k+1} = \frac{\gamma(f(x_{k+1}) - f^*)}{\|B_k^T \partial f(x_{k+1})\|}.$$

- 3) Вычислим $\lambda_1 = -p_k^T \xi_{k+1}$ и $\lambda_2 = -\xi_k^T \xi_{k+1}$. Положим

$$p_{k+1} = \begin{cases} \frac{\lambda_1}{\sqrt{\lambda_1^2 + \lambda_2^2}} p_k + \frac{\lambda_2}{\sqrt{\lambda_1^2 + \lambda_2^2}} \xi_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 > 0, \\ p_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 \leq 0, \\ \xi_k, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 > 0, \\ 0, & \text{если } \lambda_1 \leq 0 \text{ и } \lambda_2 \leq 0. \end{cases}$$

- 4) Вычислим $\mu_k = p_{k+1}^T \xi_{k+1}$. Если $-0.98 \leq \mu_k \leq 0$, то вычислим

$$B_{k+1} = B_k + (B_k \eta) \xi_{k+1}^T, \quad \text{где } \eta = \left(\frac{1}{\sqrt{1 - \mu_k^2}} - 1 \right) \xi_{k+1} - \frac{\mu_k}{\sqrt{1 - \mu_k^2}} p_{k+1}$$

и пересчитаем

$$h_{k+1} = \frac{h_{k+1}}{\sqrt{1 - \mu_k^2}}, \quad p_{k+1} = \frac{1}{\sqrt{1 - \mu_k^2}} (p_{k+1} - \mu_k \xi_{k+1}).$$

Иначе положим $B_{k+1} = B_k$ и $p_{k+1} = 0$.

- 5) Перейдем к следующей итерации с x_{k+1} , h_{k+1} , ξ_{k+1} , p_{k+1} , B_{k+1} .

3. Вычислительный эксперимент

Использование Octave-функции `ralgb5` проиллюстрируем для задачи `maxquad` [8], которая связана с минимизацией существенно вращающей выпуклой кусочно-квадратичной функции $\varphi(x)$ от 10 переменных. Здесь $\varphi(x) = \max_{1 \leq k \leq 5} f_k(x)$, где $f_k(x) = x^T A_k x - b_k^T x$, A_k — симметричные 10×10 -матрицы, такие что $A_{kij} = e^{i/j} \cos(ij) \sin k$, если $i < j$, и $A_{kii} = i |\sin k| / 10 + \sum_{j \neq i} |A_{kij}|$, а компоненты векторов b_k определяются $b_{ki} = e^{i/k} \sin(ik)$. В качестве начального приближения `maxquad` использует точку $x^0 = (1, \dots, 1)^T \in R^{10}$. Подготовку данных для задачи `maxquad` и вызов `ralgb5` реализует такой Octave-код

```

global B b m;
n = 10; m = 5; B = zeros(n, n, m); b = zeros(m, n); en = [ 1:n ];
a1 = en'*ones(1,n); a2 = exp(min(a1, a1')) ./ max(a1, a1'); a3 = cos(en'*en);
for k = 1:m
    A = a2 .* a3 * sin(k); A = A - diag(diag(A));
    B(:, :, k) = A + diag(sum(abs(A)) + abs(sin(k))*en/n);
    b(k, :) = exp(en/k) .* sin(en*k);
endfor
alpha = 2, h0 = 1.0, nh = 3, q1 = 1.0, q2 = 1.1
epsx = 1.e-6, epsg = 1.e-6, maxitn = 1000, x0 = ones(n,1)
[xr,fr,itn,ncalls,istop]=ralgb5(@maxquad,x0,alpha,h0,q1,q2,nh,epsg,epsx,maxitn);
xr,fr,itn,ncalls,istop

```

в котором вычисление $\varphi(x)$ и $\partial\varphi(x)$ выполняется с помощью Octave-функции `maxquad`

```

function [f,g] = maxquad(x)
global B b m;
ff = zeros(1,m);
for k = 1:m
    ff(k) = x'*B(:, :, k)*x - b(k, :)*x;
endfor
[ f indx ] = max(ff);
g = 2*x'*B(:, :, indx) - b(indx, :); g = g';
endfunction

```

Затраты $r(\alpha)$ -алгоритма и метода `amsq2p` для нахождения в задаче `maxquad` единственного решения с достаточно высокой точностью позволяет оценить приведенный ниже фрагмент численных расчетов с обеими программами.

Maxquad: f(x0)		5.3370664293114e+003	fmin = -8.4140833459641e-001	gamma=1		
..epsx..fr(itn).....	itn(ncalls)	..epsf..f(itn).....	itn.	
1.0e-001	-7.3721660556183e-001	35(40)	1.0e-001	-7.7355266120112e-001	17	
1.0e-002	-8.3982079259954e-001	68(74)	1.0e-003	-8.4084776169123e-001	29	
1.0e-003	-8.4138254819439e-001	107(117)	1.0e-004	-8.4132394277880e-001	35	
1.0e-004	-8.4139971517765e-001	120(131)	1.0e-005	-8.4140078034524e-001	41	
1.0e-005	-8.4140785230390e-001	148(164)	1.0e-006	-8.4140807664455e-001	49	
1.0e-006	-8.4140830366048e-001	175(195)	1.0e-011	-8.4140833458913e-001	94	
1.0e-007	-8.4140833400334e-001	211(236)	1.0e-012	-8.4140833459555e-001	101	
1.0e-008	-8.4140833455704e-001	240(267)	1.0e-013	-8.4140833459633e-001	110	
1.0e-009	-8.4140833459582e-001	278(309)	1.0e-014	-8.4140833459640e-001	116	
1.0e-010	-8.4140833459641e-001	330(369)	1.0e-015	-8.4140833459641e-001	122	

Как видим затраты по числу итераций на нахождение точки минимума при известном $f^* = fmin$ в несколько раз меньше, чем затраты программы `ralgb5`. Это может помочь при анализе разрешимости интервальных систем линейных уравнений [9]. Так, например, для интер-

вальной линейной 7×7 -системы с матрицей Ноймайера

$$\begin{pmatrix} 10.5 & [0, 2] & \cdots & [0, 2] \\ [0, 2] & 10.5 & \cdots & [0, 2] \\ \vdots & \vdots & \ddots & \vdots \\ [0, 2] & [0, 2] & \cdots & 10.5 \end{pmatrix} x = \begin{pmatrix} [-1, 1] \\ [-1, 1] \\ \vdots \\ [-1, 1] \end{pmatrix}$$

на нахождение максимума распознающего функционала допускового множества решений [10] программа `ralgb5` затратила 246 итераций, а алгоритм `amsg2p` всего за 30 итераций находит точку, где оптимальное значение (равно 1) реализуется с точностью $\varepsilon = 10^{-10}$. Для вычисления суперградиента кусочно-линейной вогнутой функции использовалась реализованная С.П. Шарым функция `calcsg`.

Приведенные алгоритмы можно использовать при решении негладких задач из различных областей приложений. Так как гладкая функция с очень быстро изменяющимся градиентом близка по своим свойствам к негладкой функции, то наши алгоритмы обладают ускоренной сходимостью при оптимизации овражных гладких функций. Матрично-векторные вычисления для обоих семейств алгоритмов легко поддаются параллельной обработке, что может быть полезным при их реализации на параллельных ЭВМ.

Литература

1. Шор Н.З., Журбенко Н.Г., Лиховид А.П., Стецюк П.И. Развитие алгоритмов недифференцируемой оптимизации и их приложения // Кибернетика и Системный Фнализ. 2003. № 4. С. 80–94.
2. Шор Н.З. Методы недифференцируемой оптимизации и сложные экстремальные задачи: Сб. избр. тр. Кишинэу: Эврика, 2008. 270 с.
3. Шор Н.З. Методы минимизации негладких функций и матричные задачи оптимизации: Сб. избр. тр. Кишинэу: Эврика, 2009. 240 с.
4. Octave [Электронный ресурс] <http://www.octave.org>. – Режим доступа: свободный.
5. Стецюк П.И. Ортогонализирующие линейные операторы в выпуклом программировании // Кибернетика и Системный Анализ. 1997. № 3. С. 97–119.

6. ШОР Н.З., СТЕЦЕНКО С.И. Квадратичные экстремальные задачи и недифференцируемая оптимизация. Киев: Наук. думка, 1989. 208 с.
7. СТЕЦЮК П.И. Субградиентные методы переменной метрики, использующие шаг Агмона-Мощкина и одноранговый эллипсоидальный оператор // Труды АТИК - 2007–2008. Кишинэу: Эврика, 2009. Том. I (XII). С. 16–25.
8. LEMARECHAL C. Numerical experiments in nonsmooth optimization // Progress in nondifferentiable optimization / Ed. E.A.Nurminski. CP-82-58. International Institute for Applied System Analysis: Laxenburg, Austria, 1982. P. 61–84.
9. ШАРЫЙ С.П. Интервальный анализ или методы Монте-Карло? // Вычислительные Технологии. 2007. Том 12, № 1. С. 103–115.
10. SHARY S.P. Solving the linear interval tolerance problem // Mathematics and Computers in Simulation. 1995. Vol. 39. P. 53–85.

r-алгоритмы: теория и практика

П. И. Стецюк

*Матеріали XI Міжнародної науково-практичної конференції
„Математичне та програмне забезпечення інтелектуальних систем“
(20-22 листопада 2013 р., м. Дніпропетровськ). – С. 227–232.*

Субградиентные методы с растяжением пространства в направлении разности двух последовательных субградиентов стали одним из центральных результатов докторской диссертации Н.З. Шора (1970) [1, 2]. Они получили название r -алгоритмов (от русского слова "разность"). Их ускоренная сходимость для овражных функций связана с выбором шага из условия точного (приближенного) минимума функции по направлению сдвига, благодаря чему определяются те два последовательных субградиента, растяжение по разности которых улучшает свойства овражной функции в преобразованном пространстве переменных. При определенной регулировке шага и коэффициентов растяжения пространства r -алгоритмы являются монотонными (или почти монотонными) по минимизируемой функции. Отметим имеющиеся теоретические и практические результаты для тех модификаций r -алгоритмов, которые используют постоянный коэффициент растяжения пространства на каждой итерации.

$r(\alpha)$ -алгоритм. Пусть $f(x)$ – негладкая выпуклая функция от n переменных, $\alpha > 1$ – коэффициент растяжения пространства. $r(\alpha)$ -алгоритмом минимизации $f(x)$ называется итеративная процедура построения последовательностей векторов $\{x_k\}_{k=0}^{\infty}$ и матриц $\{B_k\}_{k=0}^{\infty}$ по следующему правилу:

$$x_{k+1} = x_k - h_k B_k \xi_k, \quad B_{k+1} = B_k R_\beta(\eta_k), \quad k = 0, 1, 2, \dots, \quad (1)$$

где

$$\xi_k = \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|}, \quad h_k \geq h_k^* = \arg \min_{h \geq 0} f(x_k - h B_k \xi_k), \quad (2)$$

$$\eta_k = \frac{B_k^T r_k}{\|B_k^T r_k\|}, \quad r_k = g_f(x_{k+1}) - g_f(x_k), \quad \beta = \frac{1}{\alpha} < 1. \quad (3)$$

Здесь x_0 – начальное приближение, $B_0 = I_n$ – единичная $n \times n$ -матрица (в качестве матрицы B_0 часто выбирают диагональную матрицу D_n с положительными элементами на диагонали, с помощью которой осуществляется масштабирование переменных), h_k – шаговый

множитель (не меньше, чем шаг h_k^* из условия минимума функции в направлении нормированного антисубградиента в преобразованном пространстве переменных), $R_\beta(\eta) = I_n + (\beta - 1)\eta\eta^T$ – оператор "сжатия" пространства субградиентов в нормированном направлении η с коэффициентом $\beta = \frac{1}{\alpha} < 1$, $g_f(x_k)$ и $g_f(x_{k+1})$ – произвольные субградиенты функции $f(x)$ в точках x_k и x_{k+1} . Если $g_f(x_k) = 0$, то x_k является точкой минимума функции $f(x)$ и процесс (1)–(3) останавливается.

Теоретические результаты. Для минимизации гладких функций r -алгоритмы по своей формальной структуре близки к алгоритмам квазиньютоновского типа с переменной метрикой. Этим определяются два следующих теоретических результата. Так, предельный вариант r -алгоритма с бесконечным коэффициентом растяжения (здесь $\beta = 0$, $h_k = h_k^*$) является проективным вариантом метода сопряженных градиентов [2]. Для задачи минимизации выпуклой непрерывно дифференцируемой функции $f(x)$ предельный вариант r -алгоритма с восстановлением матрицы B_k после каждых n итераций обладает квадратичной скоростью сходимости при обычных условиях гладкости и регулярности $f(x)$ [1].

Наиболее общим результатом о сходимости r -алгоритмов является теорема, доказанная Н.З. Шором в [3]. Для класса почти дифференцируемых кусочно-гладких функций она определяет достаточные условия, при которых "идеализированный" вариант r -алгоритмов ($r_\mu(\alpha)$ -алгоритм) сходится к локальному минимуму. Из-за громоздкости не будем приводить эти условия, отметим лишь, что они слишком сильные и не выполняются даже для кусочно-линейных функций. Наиболее типичная ситуация, когда это происходит, связана с нарушением линейной независимости множества $G_f(x)$ – множества почти-градиентов $f(x)$ в точке x .

В статье [4] построен пример выпуклой кусочно-линейной функции, для которой возможно "зацикливание" $r_\mu(\alpha)$ -алгоритма. Для $\alpha = 3$ таким примером служит выпуклая функция

$$f(x) = \max \left\{ \max_{i=1}^4 (-g_i, x - a_1), \max_{i=1}^4 (g_i, x - a_2) \right\}, \quad (4)$$

где $a_1 = (0, -1)^T$; $a_2 = (0, 1)^T$; $g_1 = (10, 1)^T$; $g_2 = (-6, 9)^T$; $g_3 = (-10, 1)^T$; $g_4 = (6, 9)^T$. Минимальное значение функция (4) принимает в точке $x^* = (0, 0)$ и $f^* = -1$. Кроме оптимальной, для нее существуют две точки $z_1 = (0, -1)$ и $z_2 = (0, 1)$: $f(z_1) = f(z_2) = 0$, в которых нарушено условие линейной независимости семейств почти-градиентов $G_f(z_1)$ и $G_f(z_2)$. Любой из антипочти-градиентов в этих точках не явля-

ется направлением убывания функции, так как существует антипочти-градиент, образующий с ним тупой угол.

Пусть $\alpha = 3$, $x_0 = z_2$ — начальная стартовая точка для $r_\mu(\alpha)$ -алгоритма и $g_0 = g_1$ — почти-градиент в ней. Тогда в качестве последовательных почти-градиентов по итерациям выберем последовательность $\{g_1, g_2\}, \{g'_2, g'_3\}, \{g''_3, g''_4\}, \{g'''_4, g'''_1\}$. Здесь "штрихами" обозначены почти-градиенты в преобразованном пространстве: одним — после первого растяжения, двумя — после двух и т. д. Такая последовательность удовлетворяет требованиям $r_\mu(\alpha)$ -алгоритма. При этом шаг на каждой итерации будет нулевым, т. е. сдвига с точки x_0 происходить не будет. Картина же почти-градиентов после четырех итераций останется такой же, как и вначале, только длины всех векторов уменьшатся в $\alpha^2 = 9$ раз. Это связано с тем, что растяжение пространства осуществляется в двух взаимно-ортогональных направлениях.

Если этот цикл повторять, то $r_\mu(\alpha)$ -алгоритм с коэффициентом растяжения пространства $\alpha = 3$ не сможет выбраться из точки $x_0 = (0, 1)$, хотя она не является оптимальной для функции (4). Поэтому, такие точки могут служить "ловушками" для минимизирующей последовательности $r_\mu(\alpha)$ -алгоритма. В практических вариантах r -алгоритмов, в отличие от $r_\mu(\alpha)$ -алгоритма, шаговый множитель h_k выбирается из условия приближенного поиска минимума по направлению и так, чтобы выполнялось условие $h_k > h_k^*$, где h_k^* соответствует минимуму функции по направлению. Такой выбор шага позволяет преодолевать точки, которые являются "ловушками" для минимизирующей последовательности $r_\mu(\alpha)$ -алгоритма.

Практические результаты. Замечательное свойство r -алгоритмов заключается в том, что его конкретные реализации показывают очень хорошие результаты при минимизации овражных функций. Одним из эффективных зарекомендовал себя вариант $r(\alpha)$ -алгоритма с адаптивным способом регулировки шага. В нем величина h_k настраивается в процессе выполнения одномерного спуска в направлении нормированного антисубградиента в преобразованном пространстве переменных с помощью параметров h_0, q_1, n_h, q_2 . Здесь h_0 — величина начального шага (используется на первой итерации, на каждой последующей итерации уточняется); q_1 — коэффициент уменьшения шага ($q_1 \leq 1$), если условие завершения спуска по направлению выполняется всего за один шаг одномерного спуска; q_2 — коэффициент увеличения шага ($q_2 \geq 1$); натуральное число n_h задает число шагов одномерного спуска ($n_h > 1$), через каждые из которых шаг будет увеличиваться в q_2 раз. Подробные рекомендации по выбору коэффициента растяжения пространства

и параметров адаптивной регулировки шага даны в [5, с. 45–47]. Суть их выбора состоит в том, чтобы адаптивный способ регулировки шага позволял увеличивать точность поиска минимума функции по направлению в процессе счета и при этом число шагов по направлению не должно превышать в среднем двух-трех на одну итерацию.

На основе адаптивного способа и его модификаций разработано ряд программных реализаций r -алгоритма [6]. Одной из простых (включает всего 24 строки) есть octave-программа `galgb5` [7], которая находит x_r^* – приближение к точке минимума функции $f(x)$. Название программы связано с тем, что пересчет матрицы B в алгоритме (1)–(3) требует $5n^2$ арифметических операций. Условия завершения $r(\alpha)$ -алгоритма определяются параметрами ε_x и ε_g : метод останавливается в точке x_{k+1} , если выполнено $\|x_{k+1} - x_k\| \leq \varepsilon_x$ (останов по аргументу); метод останавливается в точке x_{k+1} , если выполнено условие $\|g_f(x_{k+1})\| \leq \varepsilon_g$ (останов по норме градиента, используется для гладких функций). Аварийное завершение программы связано либо с тем, что функция $f(x)$ неограничена снизу, либо начальный шаг h_0 слишком мал и его требуется увеличить.

При минимизации негладких функций рекомендуется следующий выбор параметров: $\alpha = 2 \div 3$, $h_0 = 1, 0$, $q_1 = 1, 0$, $q_2 = 1, 1 \div 1, 2$, $n_h = 2 \div 3$. Если известна априорная оценка расстояния от начальной точки x_0 до точки минимума x^* , то начальный шаг h_0 целесообразно выбирать порядка $\|x_0 - x^*\|$. При минимизации гладких функций рекомендуемые параметры такие же, за исключением q_1 , их следует выбирать $q_1 = 0, 8 \div 0, 95$. Это обусловлено тем, что дополнительное измельчение шага способствует увеличению точности поиска минимума функции по направлению, что при минимизации гладких функций обеспечивает более быструю скорость сходимости. При таком выборе параметров, как правило, число спусков по направлению редко превосходит два, а за n шагов точность по функции улучшается в три-пять раз. Параметры останова $\varepsilon_x, \varepsilon_g \sim 10^{-6} \div 10^{-5}$ при минимизации выпуклой функции (даже существенно овражной структуры) обеспечивают нахождение x_r^* со значением функции, достаточно близким к оптимальному. При этом обычно для негладких функций выполняется условие $\frac{f(x_r^*) - f(x^*)}{|f(x^*)| + 1} \sim 10^{-6} \div 10^{-5}$ ($\sim 10^{-12} \div 10^{-10}$ – для гладких функций). Это подтверждается результатами многочисленных тестовых и реальных расчетов в задачах линейного и нелинейного программирования, блочных задачах с различными схемами декомпозиции, при решении минимаксных и матричных задач оптимизации, для вычисления двойственных лагранжевых оценок в многоэкстремальных и комбинаторных задачах оптимизации и

др.

Список литературы

1. **Шор Н.З.** Методы минимизации недифференцируемых функций и их приложения. – Киев: Наук. думка, 1979. – 199 с.
2. **Шор Н.З.** Метод минимизации, использующий операцию растяжения пространства в направлении разности двух последовательных градиентов / Н.З. Шор, Н.Г. Журбенко // Кибернетика. – 1971. – № 3. – С. 51–59.
3. **Шор Н.З.** Исследование сходимости метода градиентного типа с растяжением пространства в направлении разности двух последовательных субградиентов // Кибернетика. – 1975. – № 4. – С. 48–53.
4. **Стецюк П.И.** К вопросу сходимости r -алгоритмов // Кибернетика и систем. анализ. – 1995. – № 6. – С. 173–177.
5. **Шор Н.З.** Квадратичные экстремальные задачи и недифференцируемая оптимизация / Н. З. Шор, С.И. Стеценко. – Киев: Наук. думка, 1989. – 208 с.
6. **Шор Н.З.** Развитие алгоритмов недифференцируемой оптимизации и их приложения / Н.З. Шор, Н.Г. Журбенко, А.П. Лиховид, П.И. Стецюк // Кибернетика и системный анализ. – 2003. – № 4, С. 80–94.
7. **Стецюк П.И.** Субградиентные методы с преобразованием пространства для минимизации овражных выпуклых функций // Современные проблемы прикладной математики и механики: теория, эксперимент и практика [Электронный ресурс] / Международная конференция, посвященная 90-летию со дня рождения академика Н.Н. Яненко, Новосибирск, Россия, 30 мая – 4 июня 2011 г., Новосибирск, ИВТ СО РАН, 2011, № гос. регистрации – 0321101160, Режим доступа: <http://conf.nsc.ru/files/conferences/niknik-90/fulltext/37828/53979/Stetsyuk.pdf>, свободный. – Загл. с экрана (дата обращения: 04.10.2013).

Р А З Д Е Л VII

Юбилейная статья

В раздел включены статьи:

- 7.1. Об одной схеме отсечений (*Стецюк, 1995*) 398
- 7.2. О трех научных идеях Н.З.Шора (*Сергиенко, Стецюк, 2012*) .. 433

Об одной схеме методов отсечений.

П. И. Стецюк

*Препр. / НАН Украины. Ин-т кибернетики им. В. М. Глушкова.
– Киев, 1995. – 95–25. – 33 с.*

Предложена достаточно общая конструкция метода центров тяжести простых тел (МЦТПТ), которая для задач выпуклого программирования позволяет обосновывать методы, сходящиеся к оптимуму по последовательности рекордов минимизируемой функции. Она основана на внешней аппроксимации множества экстремумов простыми выпуклыми телами (эллипсоид, параллелепипед и симплекс) с достаточным уменьшением их объема. Последующая их регуляризация (преобразование этих тел в более простые тела) использует линейные преобразования пространства и позволяет оценить сверху сложность обработки информации для методов, создаваемых в рамках этой конструкции.

Для специалистов в области создания численных методов в математическом программировании.

Цель работы – обсуждение возможности построения и обоснования для задачи выпуклого программирования практически работоспособных алгоритмов первого порядка (т. е. использующих значения функции и субградиента), для которых можно было бы гарантировать сходимость по функции со скоростью геометрической прогрессии. Исходя из анализа существующих методов типа отсечений в выпуклом программировании, попытаемся выделить характерные черты, которыми должны обладать такого рода алгоритмы, и сформулировать основные требования к ним, чтобы обеспечить их субоптимальность в смысле информационной трудоемкости. Тогда для методов типа отсечений рассмотрим достаточно общую конструкцию, которая для задач выпуклого программирования позволяет обосновывать алгоритмы, сходящиеся к оптимуму по последовательности рекордов минимизируемой функции. Она основана на внешней аппроксимации множества экстремумов простыми геометрическими телами с гарантированным уменьшением их объема на каждом шаге алгоритма и позволяет в ее рамках создавать экономные по сложности обработки информации алгоритмы.

Выбранный путь для построения методов в рамках этой конструкции геометрически «прозрачен» и основан на интерпретации их как методов типа отсечений в некотором преобразованном посредством линейного оператора пространстве аргументов, которое получено в результа-

те последовательных регуляризаций тел, локализующих множество экстремумов. При этом область локализации множества экстремумов задается пересечением конечного числа полупространств, определяемых значениями функции и субградиента в ряде точек. Этот путь позволяет разумно связать шаговый множитель с направлением движения для субградиентных методов и обеспечивает достаточно конструктивное решение ряда проблем, характерных для методов типа отсечений.

Рассмотрим следующую задачу:

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

где $f(x)$ – выпуклая функция векторного аргумента, $x \in X$, $X = R^n$; $Q \subset R^n$ – выпуклое ограниченное многогранное множество; R^n – евклидово пространство размерностью n со скалярным произведением (x, y) . Задачу (1) условимся называть стандартной задачей выпуклого программирования (ЗВП), т. е. от функции $f(x)$, кроме выпуклости, ничего больше требовать не будем. Оптимальное значение $f(x)$ для нее обозначим f^* , множество экстремумов – X^* . X будем называть исходным или прямым пространством аргументов.

Значительная часть методов для решения стандартной ЗВП обладает «предельной» сходимостью, т. е. теоремы об их сходимости построены на том факте, что в пределе удовлетворяются некоторые условия экстремума. К ним можно отнести ε -субградиентные методы, методы возможных направлений, методы штрафных функций и барьеров и др. Для этих методов наличие оценок скорости сходимости обусловлено, как правило, значительным сужением стандартной ЗВП.

Но в то же время существует ряд методов, которые для стандартной ЗВП обладают «гарантированной» скоростью сходимости, т. е. существует некоторая скалярная величина, для которой можно гарантировать ее уменьшение на каждом шаге алгоритма в определенное число раз и, кроме того, она является мажорантой сверху для рекордов $f(x)$. Эти методы относятся к классу методов отсечений и чаще всего такой скалярной величиной для них является объем области локализации множества экстремумов.

Методы этого типа для задачи (1) гарантируют сходимость по функции в следующем смысле:

$$f_k^* - f^* \leq q^k \max_{x \in Q} (f(x) - f^*), \quad (2)$$

где f_k^* – рекордное значение $f(x)$, полученное за k -шагов. Здесь $q < 1$ – знаменатель геометрической прогрессии, характеризующей скорость

сходимости метода. Сходимость, согласно (2), – это сходимость по последовательности рекордов минимизируемой функции и она является следствием уменьшения объема некоторых тел в R^n , локализующих X^* для задачи (1).

Так как мы обсуждаем возможность построения практически эффективных алгоритмов, сходящихся по функции со скоростью геометрической прогрессии для стандартной ЗВП, то естественным представляется рассмотреть этот вопрос в рамках методов типа отсечений и ориентироваться на сходимость в смысле рекордов по функции. Прежде чем переходить к его обсуждению, кратко проанализируем наиболее известные из существующих методов типа отсечений применительно к задаче (1). При анализе алгоритмов значительное внимание уделим основным идеям, положенным в их основу, наличию у них «гарантированных» оценок скорости сходимости для стандартной ЗВП, а также вычислительной трудоемкости одной итерации.

Краткий анализ методов типа отсечений. Первым в этом ряду стоит метод центров тяжести (МЦТ), предложенный А.Ю. Левиным [1]. Он основывается на том факте, что гиперплоскость, проходящая через центр тяжести n -мерного выпуклого тела G объема $\text{vol}(G)$, «отрезает» от него часть, которая составляет по объему не менее чем $\frac{1}{e} \text{vol}(G)$. МЦТ для стандартной ЗВП гарантирует сходимость рекордов по функции со скоростью геометрической прогрессии со знаменателем $q_{\text{МЦТ}} = \left(1 - \frac{1}{ne}\right)$. Основной его недостаток состоит в том, что при $n \geq 3$ отыскание центра тяжести выпуклого многогранника общего вида в R^n является чрезвычайно громоздкой задачей, и метод вряд ли может быть реализован. Поэтому МЦТ представляет в основном теоретический интерес.

Попытка модифицировать МЦТ для устранения его основного недостатка привела Д.Б. Юдина и А.С. Немировского к созданию метода эллипсоидов (МЭ) [2]. Этот же метод был получен независимо Н. З. Шором [3], исходя из алгоритмов с растяжением пространства в направлении субградиента. Основу МЭ составляет тот факт, что вокруг полушара в R^n можно описать содержащий этот полушар эллипсоид, объем которого меньше, чем объем исходного шара. МЭ также сходится в смысле (2) со скоростью геометрической прогрессии. Однако знаменатель прогрессии здесь существенно ближе к 1, чем в МЦТ, а именно $q_{\text{МЭ}} = \left(1 - \frac{1}{2n^2}\right)$, вместо $q_{\text{МЦТ}} = \left(1 - \frac{1}{ne}\right)$. Но в отличие от МЦТ, он

легко реализуем и сложность обработки информации для итерации МЭ составляет $O(n^2)$ арифметических операций.

Третьим в ряду методов с гарантированной оценкой скорости сходимости для стандартной ЗВП стоит метод вписанных эллипсоидов (МВЭ), предложенный в [4]. В МВЭ в исходную выпуклую область Q вписывается эллипсоид максимального объема. Через его центр проводится гиперплоскость, отделяющая точку минимума. В ту часть Q , в которой оказалась точка, отвечающая минимуму целевой функции, вписывается эллипсоид максимального объема, через его центр проводится отсекающая гиперплоскость, и процедура повторяется для усеченного тела. Оказывается, что скорость убывания объемов вписанных эллипсоидов близка к той, которая имеет место в МЦТ, а именно $q = 0.843$ по отношению к $q = \left(1 - \frac{1}{e}\right) = 0.632$ в МЦТ. Итерация МВЭ в принципе реализуема за полиномиальное время, в отличие от итерации МЦТ, но по трудоемкости гораздо сложнее, чем в МЭ.

На каждой итерации МВЭ требуется решать задачу нахождения γ -максимального эллипсоида для многогранника, которая является специальной задачей выпуклого программирования. Для нее, в отличие от задачи (1), задана, определяемая параметром γ , точность, с которой требуется ее решать. Это позволяет оценить трудоемкость решения этой задачи, и к настоящему времени наилучшая из оценок для нее $O(m^{3.5})$ арифметических операций [5], где m – число линейных ограничений, задающих многогранник. Но, даже при такой оценке, задача нахождения γ -максимального эллипсоида является достаточно трудоемкой и для ее решения требуются более эффективные алгоритмы. Интерес в этом смысле представляет алгоритм [6], применяющий операцию растяжения пространства. Хотя не получены теоретические оценки для его трудоемкости, тем не менее он позволяет обеспечить приближенное решение задачи нахождения γ -максимального эллипсоида и очень прост в реализации.

Гарантированной оценкой скорости сходимости для стандартной ЗВП обладают и методы, которые для локализации множества экстремумов используют такое выпуклое тело в R^n , как симплекс [7], [8]. Их принято называть методами симплексов (МС). Методы в [7] и [8] отличаются способом построения описанных симплексов. МС сходятся в смысле рекордов по функции со скоростью геометрической прогрессии со знаменателем $q_{\text{МС}} = \left(1 - \frac{1}{n^3}\right)$. Трудоемкость итерации здесь такого же порядка, как и в МЭ. Однако, несмотря на худшую теоретическую оценку, чем

в МЭ, на практике они часто оказываются эффективнее, чем МЭ. Это обусловлено тем, что оценка для МС получена в расчете на наихудший случай уменьшения объема. В МЭ, напротив, переход к следующему эллипсоиду происходит с постоянной скоростью уменьшения объема. В [9] приведена усредненная оценка уменьшения объема для МС, которая оказывается чуть лучше, чем в теоретическом варианте МЭ. Отметим два довода в пользу МС по отношению к МЭ, которые приведены в [7] и понадобятся нам в дальнейшем. Во-первых, погружение допустимой области в симплекс для задачи с линейными ограничениями представляется более естественным, чем в эллипсоид. Во-вторых, начальный симплекс может быть сформирован неравенствами-ограничениями задачи.

МЦТ, МЭ, МВЭ и МС исчерпывают список методов, которые для стандартной ЗВП обладают гарантированной оценкой скорости сходимости, которая зависит только от размерности пространства аргументов, но не зависит от свойств функции $f(x)$. Далее будем их называть первой группой методов отсечений. Они применимы для более широкого класса задач, чем общая ЗВП, например, таких, как: задача отыскания седловых точек выпукло-вогнутых функций, частные случаи задач решения вариационных неравенств, а также специальные классы задач линейной и нелинейной дополнителности.

Но все эти методы имеют один и тот же существенный недостаток, а именно: они представляют интерес в теоретическом плане и неэффективны в практических приложениях. Это объясняется тем, что более предпочтительные по оценкам скорости сходимости МЦТ и МВЭ обладают трудоемкими в вычислительном отношении итерациями. МЭ и МС, для которых итерации легко реализуемы, напротив, имеют слабые оценки скорости сходимости. Так, например, чтобы на порядок улучшить значение функции, в МЭ требуется $4.6n^2$ итераций, что не так мало даже при небольших значениях n .

Поэтому на практике зарекомендовала себя вторая группа методов отсечений, построенных на идее кусочно-линейной аппроксимации выпуклой функции $f(x)$. Наиболее известные среди них – метод отсекающей гиперплоскости (МОГ) [10], метод чебышевских центров (МЧЦ) [11], группа агрегатных методов [12] и метод уровней (МУ) [13].

В МОГ в качестве очередной точки x_{k+1} выбирается точка минимума кусочно-линейной аппроксимации $f(x)$, определяемая значениями $f(x_i)$ и $\partial f(x_i)$, $i = 0, 1, \dots, k$, на множестве Q . В МЧЦ в качестве очередного приближения выбирается чебышевский центр многогранника, т. е. точка, максимум расстояния от которой до граней многогранника

минимален. На каждой итерации, как в МОГ, так и в МЧЦ, требуется решать задачу линейного программирования (ЗЛП). Для этих методов вопросы сходимости исследованы слабо и, по-видимому, для многомерных задач нельзя обеспечить линейную скорость сходимости даже для гладких сильно выпуклых функций.

Для МОГ и МЧЦ очередное приближение может быть очень неинформативным, например, точка x_{k+1} может сильно отличаться от x_k . Этому недостатка лишены методы агрегатного типа [12]. Основная идея этих методов состоит в кусочно-линейной аппроксимации снизу целевой функции с использованием субградиентов, вычисленных в предыдущих точках. При выборе направления спуска используется квадратичная регуляризирующая добавка, которая характерна для метода линейаризации [14] и позволяет ограничить приращение вектора x в точке x_k с тем, чтобы решение линейаризованной задачи в точке x_k не уходило слишком далеко от x_k , оставаясь в такой окрестности x_k , в которой линейаризация еще справедлива. Поэтому они требуют решения на каждом шаге вспомогательной подзадачи, которая является задачей квадратичного программирования (ЗКП). Для агрегатных методов «гарантированные» оценки скорости сходимости для стандартной ЗВП отсутствуют.

Основываясь на анализе множителей Лагранжа для вспомогательной подзадачи, агрегатные методы используют некоторые способы, для того чтобы ограничить число запоминаемых субградиентов. Один из них состоит в отбрасывании ограничений, входящих в решение вспомогательной подзадачи с нулевыми значениями множителей Лагранжа, и приводит к запоминанию не более чем $(n + 1)$ субградиентов, вычисленных ранее. Второй заключается в построении и использовании некоторого комбинированного ограничения, которое получено на основе множителей Лагранжа и представляет собой линейную комбинацию ограничений вспомогательной подзадачи, которые определяются вычисленными ранее субградиентами. Хотя такой механизм и не дает четких теоретических гарантий, что мы не отбросим тот из субградиентов, который может в дальнейшем оказаться активным, тем не менее практически он позволяет эффективно решать вспомогательные подзадачи.

В смысле оценки скорости сходимости наибольший интерес среди методов типа отсечений второй группы представляет МУ, который объединяет идеи как МОГ, так и агрегатных методов. Он предназначен для решения ЗВП (1) с определенной точностью ε по функционалу. МУ представляет собой итеративную процедуру, на каждом шаге которой очередная точка x_k выбирается из минимума кусочно-линейной аппрок-

симации выпуклой функции по информации, полученной в предыдущих точках, если отклонение значения функции в этой точке от оптимального не превышает заданной точности ε . В противном случае переход осуществляется в точку, которая является проекцией x_k на множество уровня полученной аппроксимации $f(x)$.

Для МУ в [13] получена следующая оценка скорости сходимости:

$$f(x_k) - f^* \leq 2.3(DL)k^{-\frac{1}{2}}, \quad k = 1, 2, \dots, \quad (3)$$

где D — диаметр многогранника Q , а L — постоянная Липшица функции $f(x)$. Итерация МУ более сложная, чем итерация МОГ или МЧЦ, так как, кроме решения ЗЛП, она требует, правда, не на каждом шаге, нахождения проекции точки x_k на множество линий уровня аппроксимации $f(x)$, что сводится к решению ЗКП.

Ограничимся приведенными выше методами, которые уже стали классическими, так как они дают достаточно информации для дальнейших рассуждений. Однако следует отметить, что в последнее время более популярными принято считать так называемые методы внутренних точек, часть из которых также можно интерпретировать как методы типа отсечений. Наиболее полно это направление представлено в [15].

Об информационной трудоемкости методов первого порядка. Информация, которая используется во всех этих методах, — это значения функции и субградиента в точках. Поэтому естественным является вопрос: Чего можно ждать от методов, использующих такую информацию? Или, другими словами, какова потенциальная граница методов первого порядка? Ответ на него получен в монографии А.С. Немировского и Д.Б. Юдина [16]. Они показали, что сложность «всех вообще» выпуклых экстремальных задач на выпуклом компакте Q размерности n допускает оценку

$$c_1 \leq \frac{N^*(\nu)}{1 + n \ln\left(\frac{1}{\nu}\right)} \leq \left(\ln\left(\frac{e}{e-1}\right)\right)^{-1}, \quad (4)$$

где $N^*(\nu)$ — сложность данного класса задач как функция погрешности ν , равная минимально возможной трудоемкости метода, решающего всякую задачу класса с относительной погрешностью по функции, не превышающей ν ; c_1 — положительная абсолютная константа. В качестве трудоемкости метода рассматривается число обращений к оракулу, который предоставляет значение функции и субградиент в точке. Правое

неравенство верно при всех $\nu \leq 1$, а левое в асимптотике по $\nu \rightarrow 0$. Константа c_1 и «момент установления» асимптотики зависят от аффинных свойств Q . Например, если Q – параллелепипед, то $N^*(\nu) \cong n \ln \left(\frac{1}{\nu} \right)$ при $\nu \leq \frac{1}{4}$ и всех n , так что сложность при этом растет линейно с ростом n .

Методы, которые обладают трудоемкостью, согласно (4), т. е. сходятся по функции в смысле (2) со скоростью геометрической прогрессии со знаменателем $\left(1 - \frac{c}{n}\right)$, где c – некоторая константа, или равномерно по размерности как $O\left(k^{-\frac{1}{2}}\right)$, принято называть субоптимальными в смысле информационной трудоемкости, и попытки создать более быстро сходящиеся методы для общего класса ЗВП заведомо обречены на неудачу. Так, МЦТ и МВЭ являются в этом смысле субоптимальными. Оценка скорости сходимости (3) гарантирует субоптимальность МУ при невысоких требованиях к точности решения задачи (1).

Однако к полученному в [16] выводу надо относиться с некоторой долей осторожности. Во-первых, он характеризует самый наихудший случай для широкого класса «всех выпуклых функций». Во-вторых, при оценке скорости сходимости в методе учитывается только число вычислений $f(x)$ и $\partial f(x)$ и не принимается в расчет объем вычислительной работы, необходимый для того, чтобы задать очередной вопрос оракулу. Кроме того, оракул возвращает методу значение функции в точке и один из ее субградиентов. Как изменится ситуация, если оракул будет возвращать более обширную информацию о точке, например, информацию об субградиентах в ε -окрестности точки? Хотя такого рода вопросами здесь заниматься не будем, тем не менее в дальнейшем увидим, что для задачи (1) они не совсем беспочвенны. На самом деле, какой смысл решать такую трудоемкую задачу, как отыскание центра тяжести выпуклого многогранника, ради одного вычисления $f(x)$ и $\partial f(x)$. А сейчас вернемся к выяснению, что же мы собираемся делать?

Итак, с точки зрения информационной сложности, можно считать разрешенным вопрос о теоретически субоптимальном алгоритме для стандартной ЗВП. Одним из них можно считать МЦТ, МВЭ или МУ. В практическом отношении среди них можно выделить только МУ, хотя и он уступает ряду методов, для которых проведены численные эксперименты в [16]. Учитывая трудоемкость итерации МУ (требуется решать ЗЛП и ЗКП), вряд ли он способен на практике составить действенную конкуренцию алгоритмам, которые обладают простыми в вычислитель-

ном отношении итерациями, например r -алгоритму [17]. Тем не менее кажется, что от методов отсечений можно ожидать значительно большего в практическом отношении.

Поэтому будем исходить из практических приложений и основной вопрос, на который попытаемся ответить, сформулируем следующим образом:

Можно ли в рамках методов первого порядка построить алгоритм для решения стандартной ЗВП, который бы обладал нетрудоемкой итерацией и на каждой из них практически реализовывал теоретически оптимальную оценку скорости сходимости или даже лучше, и что для этого нужно?

Но как это сделать? Естественным представляется искать такой алгоритм в рамках методов типа отсечений (пока будем называть его искомым алгоритмом) и при этом он должен удовлетворять ряду формальных требований. Во-первых, очевидно, что следует отказаться от идей, на которых реализованы МЦТ и МВЭ, так как поиск центра тяжести для первого – нереализуемая задача, а для второго – итерация хотя и реализуема, но по сложности близка к решению стандартной ЗВП. Тогда понятно, что для локализации области экстремумов целесообразно рассматривать геометрически более разумные области, чем выпуклый многогранник, например, как в МЭ или МС. Во-вторых, в отличие от классических вариантов МЭ и МС, искомый алгоритм должен использовать информацию о функции в существенно большем числе точек, чем одна, что характерно для теоретически оптимальных алгоритмов МЦТ, МВЭ и МУ. В-третьих, следует отказаться и от решения на каждом шаге таких трудоемких задач, как ЗЛП и ЗКП, что имеет место в МУ, и заменить их менее трудоемкими в вычислительном отношении задачами.

Для того чтобы искомый алгоритм, применительно к задаче (1), обладал геометрической скоростью сходимости в смысле (2), естественно для него в качестве критерия выбрать уменьшение объема области локализации множества экстремумов. Для того чтобы обеспечить для него простоту итерации, целесообразно остановиться на идее внешней аппроксимации множества экстремумов более простыми выпуклыми телами, чем общий выпуклый многогранник в МЦТ. В качестве выпуклых тел разумно использовать такие геометрически простые тела в R^n , как эллипсоид, симплекс или параллелепипед. Далее будем называть их локализаторами, подразумевая под этим, что они содержат множество экстремумов задачи (1) – X^* . При этом информация, которая может быть использована для построения локализаторов, это значения

функции и субградиента $f(x)$ в ряде точек, где они задают отсекающие множество X^* гиперплоскости вида

$$(x - x_i, \partial f(x_i)) \leq f^* - f(x_i) \leq f_k^* - f(x_i), \quad x \in X^*, \quad i = 1, 2, \dots, m. \quad (5)$$

Здесь f_k^* – рекордное значение $f(x)$, полученное к k -му шагу искомого алгоритма.

Отметим, что наличие локализующего X^* тела сколь угодно малого объема еще не дает оснований говорить о сходимости метода в смысле рекордов по функции. Но ситуация упрощается, когда это тело служит хорошим локализатором ограниченного многогранного множества, полученного в результате пересечения полупространств вида (5). Техника такого рода оценок не представляет особых проблем и достаточно хорошо разработана как для выпуклых, так и для квазивыпуклых функций (см. например, [18]). Поэтому в дальнейшем не будем уделять внимание этим вопросам, а ограничимся принципиальной схемой построения искомого алгоритма, подразумевая, что мы строим локализаторы, которые достаточно хорошо аппроксимируют множество экстремумов.

Пусть уже выбран локализатор одного из указанных типов, обозначим его L_1 . Тогда возникает вопрос о построении очередного локализатора, например, этого же типа (обозначим его L_2), который был бы по объему меньше, чем L_1 . Для этого естественным представляется в качестве очередной точки, относительно которой можно строить L_2 , выбрать точку, совпадающую с «центром» локализатора L_1 , и обеспечить в каком-то смысле равноправность этой точки относительно его границ. Это легко сделать, преобразовав локализатор L_1 в геометрически простое тело в R^n , так, эллипсоид – в шар, параллелепипед – в куб, произвольный симплекс – в правильный симплекс. Шар, куб и правильный симплекс будем называть геометрически простыми, а эллипсоид, параллелепипед и неправильный симплекс – геометрически сложными телами в R^n . Операцию преобразования геометрически сложного тела в геометрически простое тело условимся называть регуляризацией локализатора. Для того чтобы ее осуществить достаточно использовать линейное преобразование пространства. В силу того, что для указанных тел регуляризация определена с точностью до константы, условимся среди определяющих их регуляризацию матриц выбирать ту, для которой детерминант равен отношению объемов текущего и очередного локализаторов. Это не имеет принципиального значения, так как субградиентные методы инвариантны относительно однородного масштабирования всего пространства R^n .

Заметим, что регуляризация некоторой области играет немаловажную роль в математическом программировании. Так, этот факт имеет место в методах отсечений первой группы, которые обладают теоретически «красивыми» оценками скорости сходимости. Для МЭ и МС регуляризация области носит явно выраженный характер, а для МЦТ и МВЭ она «скрыта» в терминах «центр тяжести многогранника» и «центр эллипсоида максимального объема, вписанного в многогранник». Кроме того, она характерна и для ряда других методов, которые хорошо зарекомендовали себя на практике. Например, в рамках методов нулевого порядка – это методы деформируемых многогранников [19], в рамках линейного программирования – проективный алгоритм Дикина-Кармаркара [20], [21].

Пусть L_1 – геометрически простой локализатор одного из указанных типов. Пусть построен очередной локализатор L_2 одного из этих типов, не обязательно геометрически простой, но такой, чтобы выполнялось условие $\text{vol}(L_1) > \text{vol}(L_2)$. При этом совершенно не обязательно, чтобы L_2 был вложенным в L_1 . Тогда переходим в центр тяжести L_2 и регуляризируем L_2 посредством линейного преобразования пространства. Повторяя этот же цикл для L_2 и L_3 , L_3 и L_4 и т.д., получим замкнутую в вычислительном отношении процедуру в том смысле, что на каждом ее шаге обеспечивается возможность однотипного построения очередного локализатора.

Эта процедура приводит к тому, что на k -м ее шаге необходимо работать в том из преобразованных пространств $Y_k = A_k X$, где текущий локализатор L_k является телом геометрически простой формы. Здесь A_k – невырожденная матрица размера $n \times n$, полученная в результате последовательных регуляризаций локализаторов за k шагов. Поэтому отсекающие гиперплоскости вида (5) нам нужны в пространстве Y_k . Это не представляет особых проблем, так как соотношение (5) может быть записано

$$\left(y - y_i, (A_k^{-1})^T \partial f(x_i) \right) \leq f_k^* - f(x_i), \quad y \in Y^*, \quad i = 1, 2, \dots, m, \quad (6)$$

где $y = A_k x$ и $y_i = A_k x_i$ – образы точек из X в преобразованном пространстве Y_k , $Y^* = A_k X^*$ – образ множества экстремумов в Y_k .

Согласно (6), $(A_k^{-1})^T \partial f(x_i)$ можно рассматривать как субградиент в точке $y_i = A_k x_i$ для функции $\varphi_k(y) = f(A_k^{-1} y)$, определенной в преобразованном пространстве Y_k . Переход из Y_k в Y_{k+1} осуществляется посредством пересчета матрицы $A_{k+1} = T_{k+1} A_k$, где T_{k+1} – невырожденная матрица, задающая для локализатора L_{k+1} преобразование из геометрически сложной формы в геометрически простую. Следова-

но, субградиенты при переходе от одного пространства к другому будут пересчитываться $\partial\varphi_{k+1}(y) = (T_{k+1}^{-1})^T \partial\varphi_k(y)$.

Отсекающие гиперплоскости вида (5) в X и их образы (6) в Y_k позволяют легко ориентироваться в преобразованном пространстве Y_k относительно любой точки, в том числе и относительно точки y_k — центра тяжести локализатора L_k в Y_k . Пусть переход в центр тяжести L_{k+1} в Y_k задает соотношение

$$y_{k+1} = y_k + \sum_{i=1}^m \alpha_i \partial\varphi_k(y_i), \quad (7)$$

где $\alpha_i, i = 1, 2, \dots, m$, — некоторые скаляры. Тогда процесс (7) в Y_k равносильен в X соотношению

$$x_{k+1} = x_k + A_k^{-1} \sum_{i=1}^m \alpha_i (A_k^{-1})^T \partial f(x_i), \quad (8)$$

где x_k и x_{k+1} — прообразы в X центров тяжести локализаторов L_k и L_{k+1} в Y_k .

Для реализации процесса типа (8) на k -м его шаге достаточно знать матрицу $B_k = A_k^{-1}$, x_k — прообраз в X центра тяжести локализатора L_k в Y_k , субградиенты $\partial\varphi_k(y_i) = B_k^T \partial f(x_i)$, $i = 1, 2, \dots, m$, а также правило, согласно которому вычисляется очередной центр тяжести L_{k+1} , т. е. $\alpha_i, i = 1, 2, \dots, m$. Для перехода в очередное преобразованное пространство аргументов Y_{k+1} достаточно знать матрицу T_{k+1}^{-1} , где T_{k+1} — невырожденная матрица размера $n \times n$, которая задает преобразование L_{k+1} из геометрически сложной в геометрически простую форму. В частности, если такое преобразование не требуется, $T_{k+1} = I_n$, где I_n — единичная матрица размера $n \times n$.

Основанный на внешней аппроксимации множества экстремумов простыми телами процесс (7) имеет достаточно простую и геометрически наглядную интерпретацию на k -м шаге в соответствующем ему преобразованном пространстве аргументов Y_k . Но эта простота отсутствует в исходном пространстве аргументов X и его можно, согласно (8), рассматривать как зеркальное отображение в X процесса в Y_k . В этом смысле такая интерпретация процесса близка к некоторой достаточно общей конструкции так называемых методов зеркального спуска [16].

Итак, искомый алгоритм будем интерпретировать как метод типа отсечений в некотором преобразованном пространстве аргументов, основываясь на информации о функции и ее субградиенте в ряде точек.

В качестве локализатора множества экстремумов выберем одно из указанных выпуклых тел в R^n , обеспечив гарантированное уменьшение его объема по отношению к предыдущему локализатору, а также переход в центр тяжести очередного локализатора и его регуляризацию. Поэтому откажемся от названия «искомый алгоритм» и заменим его более подходящим для такого процесса названием – метод центров тяжести простых тел (МЦТПТ). При этом под МЦТПТ будем понимать не столько конкретный метод, сколько некоторую общую конструкцию, в рамках которой для задачи (1) можно описывать сходящиеся в смысле рекордов по функции алгоритмы.

При дальнейшем обсуждении МЦТПТ удобно разделить основные его функции. Выбор локализующих множество экстремумов тел с гарантированным уменьшением объема и их регуляризацию условимся называть внешним алгоритмом. Сюда же отнесем и пересчет субградиентов при переходе в очередное преобразованное пространство аргументов. При этом под итерацией МЦТПТ в смысле перехода в очередную точку будем понимать именно итерацию внешнего алгоритма, т. е. переход в центр тяжести построенного локализатора. Построение же эффективных для внешнего алгоритма отсекающих гиперплоскостей, т. е. фактически определение точек, в которых требуется вычислять $f(x)$ и $\partial f(x)$, будем называть внутренним алгоритмом. Естественно также и внутренний алгоритм наделить своим критерием останова и в качестве такового рассмотрим некий аналог необходимого и достаточного условия экстремума выпуклой функции, а именно $0 \in \partial f(x)$, где $\partial f(x)$ – субдифференциал. Если для внешнего алгоритма все рассуждения должны быть математически строгими, то для внутреннего это не обязательно и он может допускать различного рода нестрогие эвристические процедуры.

На самом деле функции внешнего и внутреннего алгоритмов не так легко разделить и в МЦТПТ их естественно в значительной мере совместить, особенно в той их части, которая касается построения очередного локализатора. Поэтому от внутреннего алгоритма потребуется, чтобы он взял на себя часть функций по подготовке информации (обозначим ее Ω), используемой внешним алгоритмом при выборе очередного локализатора. К Ω отнесем и информацию о текущем геометрически простом локализаторе, и информацию об отсекающих гиперплоскостях, определяющих образ выпуклого многогранного множества Q в преобразованном пространстве аргументов, а также информацию, обеспечивающую конструктивный критерий останова для МЦТПТ, о которой речь пойдет ниже.

Итак, МЦТПТ для решения задачи (1) фактически должен обладать двумя критериями останова. Первый из них – достаточное уменьшение объема области локализации множества экстремумов, второй связан с приближенным выполнением условий оптимальности для задачи выпуклого программирования. При этом первый из критериев должен быть нацелен на то, чтобы для стандартной ЗВП гарантировать теоретически оптимальную оценку скорости сходимости по последовательности рекордов минимизируемой функции.

С учетом вышесказанного, опуская подробную детализацию ряда вопросов для внешнего и внутреннего алгоритмов, опишем общую вычислительную схему МЦТПТ для решения задачи (1), в основном с целью оценить для него сложность обработки информации (ОИ-трудоемкость). Для преобразованного пространства аргументов примем следующие обозначения. Локализаторы геометрически простой формы (шар, куб, правильный симплекс) будем обозначать L . Для обозначения локализаторов не обязательно геометрически простой формы (эллипсоид, параллелепипед или неправильный симплекс) будем использовать обозначения L' .

Перед началом вычислений имеем x_0 – центр тяжести L_0 , с тем чтобы $X^* \subset L_0$; $q = 1 - \frac{1}{e}$ – константа, характеризующая гарантированное уменьшение объема на каждой итерации МЦТПТ; m_0 – максимальное число хранимых субградиентов ($m_0 \leq 2n$); $B_0 = I_n$ – единичная матрица размером $n \times n$.

МЦТПТ строит итерационную последовательность $\{x_k\}_{k=0}^{\infty}$ следующим образом:

Шаг 1. Вычислим согласно некоторым правилам (внутренний алгоритм) $(x_i, f(x_i), \partial f(x_i), \partial \varphi_k(y_i) = B_k^T \partial f(x_i), i = 1, 2, \dots, m)$, $m \leq m_0$ и множество Ω . Если получены необходимые и достаточные условия экстремума, то останов.

Шаг 2. Используя Ω , в пространстве аргументов $Y_k = B_k^{-1}X$ относительно центра тяжести L_k (т. е. точки $y_k = B_k^{-1}x_k$) выбираем локализатор L'_{k+1} , такой, чтобы $Y_k^* = B_k^{-1}X^* \subset L'_{k+1}$ и $q^* = \text{vol}(L'_{k+1})/\text{vol}(L_k)$ было минимально. Если $q^* \geq q$, то переходим к Шагу 1.

Шаг 3. Вычислим очередное приближение

$$x_{k+1} = x_k + B_k \sum_{i=1}^m \alpha_i \partial \varphi_k(y_i),$$

которое задает прообраз в X центра тяжести L'_{k+1} в Y_k . Здесь α_i ,

$i = 1, \dots, m$ – скаляры, задающие в Y_k центр тяжести L'_{k+1} относительно центра тяжести L_k .

Шаг 4. $B_{k+1} = B_k T_k^{-1}$ (регуляризация L'_{k+1}). T_k – неособенная матрица размером $n \times n$, задающая преобразование L'_{k+1} в L_{k+1} .

Шаг 5. $\partial\varphi_{k+1}(y_i) = (T_k^{-1})^T \partial\varphi_k(y_i)$, $i = 1, 2, \dots, m$ (пересчет субградиентов при переходе в очередное преобразованное пространство аргументов).

Шаг 6. Положим $k = k + 1$ и переходим к Шагу 2, модифицируя при этом Ω или некоторую его часть.

Заметим, что в приведенной схеме МЦТПТ фактически отсутствует останов по уменьшению объема. Это не случайно. Дело в том, что принятая схема аппроксимации множества экстремумов простыми телами позволяет достаточно хорошо рассчитывать нижнюю оценку f^* посредством последовательных уточнений ее в каждом из преобразованных пространств. Следовательно, используя рекордную точку и нижнюю оценку f^* , внутренний алгоритм для задачи (1) обеспечивает конструктивный критерий останова в терминах точности ее решения по функционалу.

Кроме того, к Ω отнесены и определяющие многогранное множество Q в X гиперплоскости, образы которых в Y_k задают отсекающие гиперплоскости для его образа $Q_k = \bar{\pi}_k^{-1} Q$ в Y_k и также требуют соответствующего пересчета и хранения. Но для оценки ОИ-трудоемкости итерации МЦТПТ это не существенно, поскольку для линейных неравенств общего вида это увеличивает число отсекающих гиперплоскостей на некоторую константу порядка n . Когда многогранник Q в X задан параллелепипедом $Q = \{x : l_i \leq x_i \leq u_i, i = 1, \dots, n\}$, то такого хранения не требуется, так как в этом случае нормали к отсекающим Q_k в Y_k гиперплоскостям будут задавать строки матрицы B_k .

ОИ-трудоемкость итерации МЦТПТ определяется ОИ-трудоемкостью шагов 1 – 6, где шаг 1 определяет внутренний алгоритм, шаги 2 – 6 – внешний. Шаг 3 требует $O(n^2)$ арифметических операций, шаги 4 и 5 – $O(n^3)$ в самом наихудшем случае, т. е. при использовании в качестве локализаторов эллипсоидов, параллелепипедов или симплексов самого общего вида. Поэтому, обеспечив легкий выбор внешним алгоритмом очередного локализатора (шаг 2) и для внутреннего алгоритма ОИ-трудоемкость порядка $O(n^3)$, получим общую ОИ-трудоемкость итерации МЦТПТ не более $O(n^3)$. Если при этом для МЦТПТ на каждом шаге мы сможем гарантировать уменьшение объема в $(1 - \frac{1}{e})$ раз, то он будет реализовывать теоретически оптимальную оценку в смыс-

ле рекордов по функции и его общая ОИ-трудоемкость будет – $O(n^4)$ арифметических операций. Используя специальные локализаторы одного из указанных типов и учитывая тот факт, что работа внутреннего алгоритма нужна не на каждой итерации, для МЦТПТ практически можно обеспечить в среднем ОИ-трудоемкость итерации – $O(n^2)$.

Следовательно, замена многогранника общего вида, как в МЦТ, на более простые выпуклые тела, центры тяжести которых легко находить, позволяет разумно оценить трудоемкость итерации методов типа отсечений. При этом такая замена выглядит более осмысленной, чем попытка заменить вычисление центра тяжести общего выпуклого многогранника такой трудоемкой процедурой, как в МВЭ.

Приведенная достаточно общая схема МЦТПТ позволяет, конкретизируя внутренний и внешний алгоритмы, получать для решения стандартной ЗВП вполне конструктивные сходящиеся в смысле рекордов по функции методы. Так, например, если ориентироваться на выбор локализатора одного и того же типа, то получим метод описанных эллипсоидов (МОЭ), метод описанных параллелепипедов (МОП) или же метод описанных симплексов (МОС). Кроме выбора локализаторов, она требует и определенного механизма построения отсекающих гиперплоскостей (внутренний алгоритм), который обеспечивал бы легкое построение эффективных требуемых локализаторов. Ниже рассмотрим эти вопросы более подробно и при их обсуждении главное внимание уделим легко реализуемым алгоритмам в рамках этой схемы, ориентируясь на использование локализаторов специального типа, для того чтобы обеспечить для них практическую ОИ-трудоемкость итерации – $O(n^2)$ арифметических операций.

Принятый подход к созданию искомого алгоритма не является совершенно новым. Такого рода алгоритмы, в качестве локализаторов для которых используются эллипсоиды, рассматривались в [22] – [24]. Их принято называть методами эллипсоидов с более глубокими отсечениями и в отличие от МЦТПТ они используют только один критерий для установления их сходимости – достаточное уменьшение объема эллипсоидов, локализующих множество экстремумов. Так, в [22] рассмотрены алгоритмы, основанные на эллипсоидах минимального объема описанных вокруг следующих выпуклых тел: сегмент, слой и s -пирамида. Но они не используют информацию, полученную на предыдущих шагах. Алгоритм в [23] также основан на эллипсоиде минимального объема, содержащего шаровой слой, и уже использует полученную ранее информацию. Но параметры шарового слоя для него вычисляются на основании информации только о двух гиперплоскостях и вряд ли при

этом следует ожидать хороших оценок скорости сходимости. Поэтому и не удивительно, что вычислительные эксперименты показали незначительные преимущества этих методов над классическим вариантом МЭ.

Что касается методов эллипсоидов с более глубокими отсечениями, то можно отметить, что их основной недостаток заключается в том, что, уделяя главным образом внимание технике описания эллипсоидов оптимального объема, содержащих различного рода выпуклые тела, они практически игнорируют вопросы «хорошего» построения этих тел. Так, например, константа, характеризующая уменьшение объема для эллипсоида оптимального объема, содержащего шаровой слой, существенно зависит от двух параметров – расстояния от центра шара до нижней границы шарового слоя и от самой его толщины. Поэтому естественно уделить внимание увеличению первого и уменьшению второго параметров, для чего явно недостаточно использования только двух отсекающих гиперплоскостей. В дальнейшем при обсуждении конкретных вариантов МОЭ ограничимся достаточно простыми локализаторами и главное внимание уделим вопросам их эффективного построения.

О внешнем алгоритме. Что выбрать в качестве локализатора множества экстремумов? Из рассмотренных тел предпочтительнее остальных является эллипсоид, так как при этом обеспечивается возможность на каждой итерации внешнего алгоритма уменьшить объем области локализации множества экстремумов, используя информацию о субградиенте только в одной точке – центре шара в преобразованном пространстве, содержащего образ множества экстремумов (классический МЭ). Из этих соображений в качестве локализатора можно было бы избрать и симплекс. Но, во-первых, он теоретически хуже уменьшает объем, чем эллипсоид и доводы в его пользу, приведенные в [7], вряд ли состоятельны. Тот факт, что для задач с линейными ограничениями симплекс является более естественным, пожалуй, бесспорный. Но доводы о том, что его лучше использовать, чем эллипсоид для внешней аппроксимации множества экстремумов, основаны только на средних оценках уменьшения объема в МС и не совсем убедительны. Так, например, для классического варианта МЭ при определенных условиях можно добиться уменьшения объема в $\sqrt{\frac{n+1}{e}}$ раз на одной итерации. Но эти условия выполняются настолько редко, насколько редко могут быть выполнены и условия для наилучших отсечений по объему и в МС.

Во-вторых, в случае, когда решение находится в подпространстве пространства R^n , основной недостаток МС связан со сложностью пере-

хода от симплекса в n -мерном пространстве к симплексу в пространстве меньшей размерности. В то же время этот недостаток легко устраним для метода типа эллипсоидов и фактически приводит к проектированию на подпространство меньшей размерности. При этом сохраняется возможность рассуждений на уровне шара в подпространстве пространства R^n .

В-третьих, локализуемый множество экстремумов эллипсоид преобразовывается в шар, для которого обеспечивается минимум дополнительной информации о локализаторе, в том смысле, что достаточно знать центр шара в преобразованном пространстве (на самом деле достаточно знать его прообраз в X) и его радиус, что немаловажно с точки зрения информационной трудоемкости МЦТПТ.

Но приведенные в пользу эллипсоида доводы не значат, что остальные из указанных тел нецелесообразно использовать в качестве локализаторов. Так, например, естественно использовать в качестве локализаторов параллелепипед, грани которого параллельны осям координат, или же симплекс, n граней которого параллельны осям координат, пусть даже они будут служить более грубыми локализаторами множества экстремумов. В этом случае регуляризация локализаторов использует линейное преобразование пространства, которое фактически сводится к масштабированию переменных задачи. Это позволяет обойтись запоминанием n элементов, в отличие от того, что при использовании эллипсоида их требуется $n \times n$, и, кроме того, при решении внутренних подзадач использовать мощный арсенал средств для работы с разреженными матрицами.

Что касается рекомендаций относительно выбора локализаторов для внешней аппроксимации множества экстремумов, то здесь сложно давать какие-то готовые рецепты. Ответ на вопрос о том, на какой из методов МОЭ, МОП или МОС в рамках МЦТПТ ориентироваться, далеко не однозначен. У каждого из них есть ряд преимуществ, которые для других могут служить недостатками. Так, например, с общих геометрических позиций более рационально использовать эллипсоид или параллелепипед, так как они являются центрально-симметричными телами в R^n . Симплекс таковым не является. При этом, если для эллипсоида простота его построения связана с переходом к шару, то такой простоты не наблюдается в случае перехода от параллелепипеда к кубу. Но, если ориентироваться на построение параллелепипеда даже самого общего вида, оси которого всегда будут образами нормалей к отсекающим гиперплоскостям в X , можно сохранить возможность работы с разреженными матрицами, т. е. максимально учесть структурные осо-

бенности задачи. В то же время симплекс позволяет легко работать с неправильными симплексами в X , что характерно для различных модификаций метода симплексов Нелдера – Мида [25], сохраняя при этом возможность рассуждений в преобразованном пространстве на уровне правильного симплекса.

Интересным также представляется использование в качестве локализаторов всех рассмотренных тел в различных комбинациях. На одном из таких вариантов, который использует в качестве локализаторов все три тела, немного подробнее остановимся в дальнейшем при обсуждении проблемы «рестарта» МЦТПТ. При этом эллипсоид будет использоваться для внешней аппроксимации множества экстремумов, симплекс и параллелепипед для построения эффективных аппроксимирующих множество экстремумов эллипсоидов. Кроме того, параллелепипед будет использоваться и для «рестарта» МЦТПТ. Поэтому остановимся на анализе тех эллипсоидов, использование которых целесообразно для локализации множества экстремумов.

Итак, пусть мы выбрали в качестве локализатора эллипсоид. Тогда для того чтобы обеспечить простоту итерации внешнего алгоритма, достаточно отказаться от рассмотрения сложных оптимальных эллипсоидов, порожденных, например, таким телом, как s -пирамида [22] и ограничиться эллипсоидами специального вида $\text{Ell}(\xi, a, b)$, т. е. такими, для которых в R^n длина полуоси в направлении ξ равна a , а длины полуосей в $(n - 1)$ – ортогональных к ξ направлениях одинаковы и равны b . Это обусловлено тем, что для его преобразования в шар достаточно одномерного оператора растяжения пространства [26].

Простое построение таких эллипсоидов в R^n для шара $S(y_0, r)$ допускают такие тела, как шаровой слой – тело, полученное в результате пересечения шара $S(y_0, r)$ и множества $PU(y_0, \xi, a_1, a_2) = \{y : a_1 \leq (y - y_0, \xi) \leq a_2\}$, и n -мерный ограниченный круговой конус – пересечение шара $S(y_0, r)$, множества $PU(y_0, \xi, a_1, a_2)$ и множества $KS(y_0, \xi, a) = \{y : (y - y_0, \xi) \geq a\|y - y_0\|\}$. Здесь ξ – ненулевой вектор единичной длины, a , a_1 и a_2 – скаляры, такие, что $0 \leq a_1 \leq a_2 \leq r$ и $0 < a < 1$. $PU(y_0, \xi, a_1, a_2)$ условимся называть плоским слоем в R^n в направлении ξ , $KS(y_0, \xi, a)$ – n -мерным круговым конусом в направлении ξ . Именно рациональному построению этих тел и следует уделить главное внимание. Для этого целесообразно использовать такие выпуклые тела в R^n , как $S(m)$ -пирамида [22] (тело, являющееся пересечением шара и полупространств, определяемых m ($m \leq n$) линейно-независимыми субградиентами) и симплекс, которые позволяют эффективно решать эти задачи. Анализ $S(m)$ -пирамид и симплексов позволяет избежать реше-

ния на итерации таких трудоемких задач, как ЗЛП и ЗКП, и обойтись при этом достаточно простыми вычислительными операциями.

Пусть $S(2)$ -пирамида в R^n получена в результате пересечения шара $S(y_0, r)$ и двух полупространств P_1 и P_2 , определяемых векторами нормалей p_1 и p_2 . Пусть центр шара y_0 принадлежит границам полупространств P_1 и P_2 и угол между векторами p_1 и p_2 — тупой, т. е. $(p_1, p_2) < 0$. Тогда существует эллипсоид типа $\text{Ell}(\xi_1, \xi_2, a, b, r)$ с центром в y_0 , содержащий эту $S(2)$ -пирамиду, и его объем меньше, чем объем шара $S(y_0, r)$. Константа, характеризующая уменьшение его объема по отношению к объему шара, зависит только от косинуса угла между векторами p_1 и p_2 . Здесь a и b — длины полуосей эллипсоида в взаимно-ортогональных направлениях ξ_1 и ξ_2 , r — длины полуосей в $(n-2)$ — ортогональных к ξ_1 и ξ_2 направлениях. Преобразование $\text{Ell}(\xi_1, \xi_2, a, b, r)$ в шар требует двух последовательных растяжений пространства и по смыслу оно близко к преобразованию, применяемому в r -алгоритмах [26].

В отличие от эллипсоида типа $\text{Ell}(\xi, a, b)$, он позволяет уменьшать объем области локализации множества экстремумов, не сдвигаясь из центра шара, и это обстоятельство обеспечивает достаточно эффективный механизм для уменьшения объема $S(m)$ -пирамиды посредством погружения ее в $S(2)$ -пирамиды, которые допускают внешнюю локализацию эллипсоидом типа $\text{Ell}(\xi_1, \xi_2, a, b, r)$. Этот факт предопределяет преимущество $\text{Ell}(\xi_1, \xi_2, a, b, r)$ по отношению к эллипсоиду оптимального объема, содержащему $S(2)$ -пирамиду, центр которого будет находиться внутри $S(2)$ -пирамиды, что приводит к необходимости каждый раз искать новую $S(m)$ -пирамиду. Кроме того, в случае, когда известен достаточно хороший рекорд по $f(x)$ для задачи (1), $\text{Ell}(\xi_1, \xi_2, a, b, r)$ позволяет значительно уменьшать объем области локализации X^* , ограничиваясь при этом хранением всего двух — трех субградиентов.

Если ограничиться для внешней аппроксимации множества экстремумов локализаторами типа $\text{Ell}(\xi, a, b)$ и $\text{Ell}(\xi_1, \xi_2, a, b, r)$, в рамках МЦТПТ получим МОЭ, обеспечив при этом достаточную простоту итерации внешнего алгоритма. Так, шаги 3 — 5 потребуют $3n(n + \frac{2}{3}m)$ арифметических операций умножения и приблизительно столько же сложений для $\text{Ell}(\xi, a, b)$ и в два раза больше для $\text{Ell}(\xi_1, \xi_2, a, b, r)$. Здесь m — число хранимых субградиентов.

Насколько реально при этом создание МОЭ, обеспечивающего теоретически оптимальную оценку скорости сходимости или даже лучше? Чтобы обеспечить для него оценку скорости сходимости $\left(1 - \frac{1}{ne}\right)$ в

смысле рекордов по функции, достаточно уменьшать объем области локализации множества экстремумов на каждой из итераций внешнего алгоритма не менее чем в $\left(1 - \frac{1}{e}\right)$ раз. Для этого вполне достаточно гарантировать для шара радиуса r выполнение одного из следующих условий. Для шарового слоя либо расстояние от нижней его границы до центра шара будет порядка r/n , либо его ширина будет порядка r/\sqrt{n} . Для $S(2)$ -пирамид, которые допускают построение эллипсоида типа $\text{Ell}(\xi_1, \xi_2, a, b, r)$, достаточно, чтобы косинус угла между определяющими их векторами был меньше, чем $-\sqrt{\frac{2}{e} - \frac{1}{e^2}} = -0.775$.

Выполнение одного из этих условий представляется не таким уж невозможным, если использовать информацию о функции в ряде точек. Построение же эллипсоида, содержащего n -мерный ограниченный круговой конус, или эллипсоида, содержащего шаровой слой в расчете на «хорошее» отсечение от центра шара для следующего шага внешнего алгоритма, позволяют в значительной мере улучшить эту оценку.

Такой вариант МОЭ может служить одной из возможных интерпретаций, которая геометрически достаточно наглядна для методов с растяжением пространства [26], если, согласно [28], операцию растяжения пространства понимать как преобразование такого рода эллипсоидов в шар. На этом пути легко объясняются как растяжение пространства аргументов, так и его сжатие. Так, для $\text{Ell}(\xi_1, \xi_2, a, b, r)$ присутствуют оба этих элемента. Для $\text{Ell}(\xi, a, b)$, когда он содержит шаровой слой, всегда требуется растяжение пространства аргументов. Когда он содержит n -мерный ограниченный круговой конус, при одних условиях требуется растяжение пространства аргументов, а при других – его сжатие.

Следует отметить, что, исходя из уменьшения объема области локализации множества экстремумов, главенствующая роль должна принадлежать эллипсоидам типа $\text{Ell}(\xi, a, b)$. Привлечение эллипсоидов типа $\text{Ell}(\xi_1, \xi_2, a, b, r)$ целесообразно только тогда, когда они очень хороши для уменьшения объема, как в вышеупомянутых случаях. Положить $\text{Ell}(\xi_1, \xi_2, a, b, r)$ в основу методов разумнее при интерпретации их как методов типа отсечений в рамках методов фееровского типа, используя при этом $\text{Ell}(\xi, a, b)$ в том случае, когда не удастся легко улучшить рекордное значение $f(x)$.

О внутреннем алгоритме. Но для создания алгоритмов в рамках МЦТПТ требуется и механизм построения эффективных отсекающих гиперплоскостей. Поэтому попытаемся прояснить некоторые моменты, связанные с внутренним алгоритмом. При этом будем учитывать, что

от его работы требуется подготовка информации для внешнего алгоритма с тем, чтобы обеспечить легкое построение локализаторов. Так, например, для эллипсоида типа $\text{Ell}(\xi, a, b)$ требуется построение «хороших» шаровых слоев и n -мерных ограниченных круговых конусов, для эллипсоида типа $\text{Ell}(\xi_1, \xi_2, a, b, r)$ – эффективных $S(m)$ -пирамид специального типа.

При обсуждении внутреннего алгоритма будем исходить из следующих соображений. Во-первых, он должен строить эффективные отсекающие гиперплоскости для внешнего алгоритма. Во-вторых, известна некоторая точка x_k , образ которой $y_k = A_k x_k$ является центром тяжести геометрически простого локализатора множества экстремумов в преобразованном пространстве аргументов $Y_k = A_k X$. Здесь A_k – невырожденная матрица размером $n \times n$, полученная в результате последовательных регуляризаций аппроксимирующих множество экстремумов локализаторов. Поэтому вполне естественно вопрос о построении эффективных отсекающих гиперплоскостей рассматривать в преобразованном пространстве относительно точки y_k .

Не ограничивая общности, основные принципы внутреннего алгоритма для задачи (1) рассмотрим на примере исходного пространства аргументов X – области определения $f(x)$. Тем более, что эта процедура требуется на начальном шаге МЦТПТ, когда никакой информации о поведении $f(x)$ еще не имеется. Перенос же ее в преобразованное линейным оператором пространство осуществляется сравнительно просто и приводит к этим же задачам на наборе векторов, которые получены в результате применения линейного оператора к векторам исходного пространства.

С учетом вышесказанного, основываясь на идеях методов отсечений второй группы, построим для нашего случая некоторую «идеализированную» процедуру, позволяющую строить эффективные отсекающие гиперплоскости для задачи (1), используя для этого значения функции и ее субградиента в точке. Назовем ее «ИЗМЕРИТЕЛЬ» и под измерением будем понимать вычисление в точке x_i значений $f(x_i)$ и $\partial f(x_i)$, т. е. фактически обращение к оракулу первого порядка.

Пусть перед началом вычислений для МЦТПТ мы имеем некоторую точку x_0 , которая принадлежит многограннику Q и является центром тяжести геометрически простого локализатора L_0 , который содержит множество X^* задачи (1). $f(x_0), \partial f(x_0)$ – вычисленные в x_0 значения функции и субградиента. $f_0^* = f(x_0)$ – рекордное значение $f(x)$.

Опишем шаг процедуры «ИЗМЕРИТЕЛЬ» применительно к локализатору L_0 . Пусть уже имеется m гиперплоскостей, заданных вычис-

ленными ранее субградиентами, т. е. $x_i, \partial f(x_i), f(x_i), i = 0, 1, \dots, m-1$. f_0^* – достигнутое рекордное значение $f(x)$ для задачи (1). Тогда в качестве первой из точек возьмем решение задачи, как в МОГ, с тем отличием, чтобы точка минимума кусочно-линейной аппроксимации $f(x)$, построенная по имеющейся информации, принадлежала локализатору L_0 :

$$\min t \quad (9)$$

$$f(x_i) + (\partial f(x_i), x - x_i) \leq t, \quad i = 0, 1, \dots, m-1 \quad (10)$$

$$x \in L_0, \quad x \in Q. \quad (11)$$

Обозначим эту точку x_A и оптимальное значение $t^* - f_A$. Так как $f^* \geq f_A$, то f_A может служить нижней оценкой f^* для задачи (1). Кроме того, точка x_A – строго внутренняя точка многогранного множества, заданного ограничениями вида (10), если параметр $t = f_0^*$.

Используя $f_0^* \geq f^* \geq f_A$, определим некий «зазор» $\Delta = f_0^* - f_A$, который локализует оптимальное значение f^* задачи (1). Его целесообразно использовать для построения эффективных отсекающих гиперплоскостей относительно точки x_0 . Здесь напрашивается использование идеи, близкой к используемой в МУ, а именно проектировать точку x_0 на множество линий уровня полученной аппроксимации $f(x)$, задавшись некоторым $f_\Delta = f_0^* - \frac{\Delta}{2}$. Поэтому в качестве второй точки (обозначим ее x_B) возьмем решение следующей задачи:

$$\min \|x_0 - x\|^2 \quad (12)$$

$$f(x_i) + (\partial f(x_i), x - x_i) \leq f_\Delta, \quad i = 0, 1, \dots, m-1 \quad (13)$$

$$x \in L_0, \quad x \in Q. \quad (14)$$

Очевидно, что отрезок в R^n , который соединяет точки x_B и x_A целиком принадлежит многогранному множеству, описанному ограничениями вида (10), если параметр $t = f_0^*$. Поэтому целесообразно проводить измерения в точках этого отрезка. Для определенности выберем три измерения и будем проводить их последовательно в точках x_B , $(x_A + x_B)/2$ и x_A . Очевидно, что, если гиперплоскость с нормалью $\xi = -\frac{\partial f(x_B)}{\|\partial f(x_B)\|}$, проведенная в точке x_B со сдвигом в направлении ξ на величину $\frac{f(x_B) - f_A}{\|\partial f(x_B)\|}$, отсекает точку x_A , то измерения можно прекратить, так как в этом случае можно улучшить f_A и, следовательно,

уменьшить «зазор» Δ . Это же касается и точки $(x_B + x_A)/2$, если измерения не прекратились ранее. В наихудшем случае это приводит к трем вычислениям $f(x)$ и $\partial f(x)$. К очередному шагу процедуры «ИЗМЕРИТЕЛЬ» переходим добавив к имеющимся субградиентам вычисленные описанным выше способом и уточнив при этом f_0^* , если это возможно.

Измерения, проводимые процедурой «ИЗМЕРИТЕЛЬ», представляются эффективными в том смысле, что $f(x)$ и $\partial f(x)$ вычисляются в той области многогранного множества Q , в которой имеется недостаточно информации о поведении $f(x)$, и не вычисляются там, где заранее известно, что значение функции больше, чем полученный рекорд. Более того, если из задач (9)–(11) и (12)–(14) убрать ограничение $x \in L_0$ и дополнить модифицированную таким образом процедуру «ИЗМЕРИТЕЛЬ» некоторым правилом перехода в точку x_{k+1} (например, в точку улучшения рекорда f_k^*), получим сходящийся «в пределе» алгоритм для решения задачи (1) без преобразования пространства. При этом он должен работать значительно лучше, чем МОГ. Решение трудоемких ЗКП и ЗЛП на каждой итерации такого алгоритма более содержательно, чем в МУ, в том смысле, что их решение позволяет построить несколько содержательных отсекающих гиперплоскостей. Но вряд ли от него следует ожидать очень хорошей работы и тем более получить для него гарантированную оценку скорости сходимости для стандартной ЗВП.

Для точки x_0 , если в ней достигается $f(x_0) = f_0^*$, процедура «ИЗМЕРИТЕЛЬ» фактически реализует поиск направления убывания $f(x)$ из этой точки (если оно существует) и в пределе напоминает процедуры, которые используются в ε -субградиентных методах [29]. В процессе вычислений она существенно использует значения функции в точках вычисления субградиентов и нижнюю оценку f^* – параметры, неиспользование которых для процедур в [29] сильно затрудняет регулировку ε . Если остановиться на использовании процедуры типа «ИЗМЕРИТЕЛЬ» для внутреннего алгоритма, то естественно при этом для него в качестве критерия останова выбрать условие, аналогичное как и в ε -субградиентных методах – $0 \in \partial_\varepsilon f(x)$, где ε легко подсчитывается на основании полученной в процессе вычислений информации.

В практическом плане процедура «ИЗМЕРИТЕЛЬ» приводит к решению трудоемких подзадач (9)–(11) и (12)–(14) и обеспечить при этом простоту итерации МЦТПТ невозможно. Поэтому целесообразно заменить ее процедурой, которая пусть делала бы измерения более грубо, но была просто реализуема. При этом желательно, чтобы она реализовывала их идейно, как можно ближе к процедуре «ИЗМЕРИТЕЛЬ». Так как сходимость МЦТПТ обеспечивается за счет итерации внеш-

него алгоритма, то нет особой надобности в точном решении подзадач (9)–(11) и (12)–(14). Поэтому логично заменить их некоторой процедурой, которая позволяла бы одновременно получить приближенные решения как первой, так и второй подзадач. При этом она должна поставлять содержательную информацию для выбора локализаторов внешним алгоритмом.

Такой механизм легко обеспечить на уровне анализа некоторых выпуклых ограниченных тел в R^n , полученных в результате пересечения локализатора и $K(m)$ -конуса в R^n , который задан пересечением не более чем m ($m \leq n$) полупространств вида (10), определяемых линейно независимыми субградиентами $\partial f(x_i)$, $i = 1, 2, \dots, m$. В качестве полупространств, задающих $K(m)$ -конус, могут использоваться и полупространства, которые задают выпуклый многогранник Q . Для $K(m)$ -конуса точку, которая принадлежит пересечению границ всех m полупространств и при этом находится на кратчайшем расстоянии от центра локализатора x_0 , условимся называть его вершиной относительно x_0 .

Кратко объясним смысл проводимых при этом измерений на примере $K(m)$ -конуса, построенного по субградиентам $f(x)$, т. е. фактически в предположении, что $X^* \subset \text{int} Q$. В случае, если решение задачи (1) достигается на границе Q , механизм измерений остается таким же, с тем отличием, что формирование $K(m)$ -конуса требует привлечения полупространств, задающих некоторые из границ Q .

Пусть выбраны m линейно независимых субградиентов, вычисленных ранее, т. е. $(x_i, \partial f(x_i), f(x_i))$, $i = 1, 2, \dots, m$. Рассмотрим систему линейных уравнений

$$f(x_i) + (\partial f(x_i), x - x_i) = t, \quad i = 1, 2, \dots, m, \quad (15)$$

где t – параметр. Решение системы (15) относительно точки x_0 имеет следующий вид:

$$x(t) = x_0 + G(G^T G)^{-1}(te + a) \quad (16)$$

и для определяемого этими субградиентами $K(m)$ -конуса задает его вершину относительно точки x_0 при разных значениях t . Здесь G – матрица размером $n \times m$, столбцами которой являются $\partial f(x_i)$, $i = 1, 2, \dots, m$; a – m -мерный вектор, i -й компонент которого равен $(\partial f(x_i), x_i - x_0) - f(x_i)$, для всех $i = 1, 2, \dots, m$; e – m -мерный вектор, все компоненты которого равны единице.

Управление параметром t позволяет легко анализировать $K(m)$ -конус при разных значениях t . Так при $t = f_0^*$ он будет содержать множество X^* . Формально это равносильно анализу упрощенной задачи квадратичного программирования типа (12)–(14) с m ограничениями вида (13)

без учета ограничений (14) при различных значениях параметра f_Δ . При этом вектор $u(t) = (G^T G)^{-1}(te + a)$ для нее играет роль множителей Лагранжа и позволяет легко проверять, является ли вершина $x(t)$, вычисленная, согласно (16), ее решением. Если $u(f_0^*) \geq 0$, то гиперплоскость с нормалью $\frac{x(f_0^*) - x_0}{\|x(f_0^*) - x_0\|}$, проведенная на расстоянии $\|x(f_0^*) - x_0\|$ от точки x_0 , отсекает множество X^* задачи (1). Минимальное значение параметра $t = t^*$, при котором $u(t^*) \geq 0$ и полупространство, заданное гиперплоскостью с нормалью $\frac{x(t^*) - x_0}{\|x(t^*) - x_0\|}$ на расстоянии $\|x(t^*) - x_0\|$ от точки x_0 , не имеет общих точек с локализатором L_0 , может служить уточнением нижней оценки f_A для f^* . Для проведения измерений подходящим является отрезок $[x(f_0^* - \delta), x(f_A)]$, где $0 < \delta \leq \frac{\Delta}{2}$, если вершина $x(f_0^* - \delta)$ не отсекается от многогранного множества типа (10) при $t = f_0^* - \delta$ какой-либо из гиперплоскостей, не включенных в рассматриваемый $K(m)$ -конус. В противном случае можно перейти к очередному $K(m_1)$ -конусу, увеличивая при этом расстояние $\|x_0 - x(f_0^* - \delta)\|$.

Такой механизм позволяет строить эффективные отсекающие гиперплоскости также, как и процедура «ИЗМЕРИТЕЛЬ», обходясь незначительным числом арифметических операций. Так, самая трудоемкая из них, обеспечивающая переход от $K(m)$ -конуса к $K(m+1)$ -конусу, требует $\sim m \cdot n$ арифметических операций. При этом правило, согласно которому следует включить очередной субградиент, чтобы измерения были эффективными, геометрически достаточно «прозрачно» и использует как значения функции в точках, так и субградиенты в них, а точнее, их нормы. Он легко переносится в преобразованное посредством линейного оператора A_k пространство аргументов $Y_k = A_k X$, где A_k – невырожденная матрица размером $n \times n$. Для (m) -конуса в Y_k его вершина $y(t)$ относительно точки y_k , которая является центром геометрически простого локализатора $L_k \in Y_k$, вычисляется как в (16), с тем отличием, что векторы $\partial f(x_i)$ заменяются на векторы $\partial \varphi_k(y_i) = (A_k^{-1})^T \partial f(x_i)$ для всех $i = 1, 2, \dots, m$.

Основанная на таком принципе процедура может быть интерпретирована как ε -субградиентная в соответствующем пространстве аргументов, где ε выбирается исходя из рекорда по функции и нижней оценки f^* , с тем отличием, что решение внутренних подзадач (переход от $K(m)$ -конуса к $K(m_1)$ -конусу) обеспечивают итерации симплексного типа. При этом в качестве критерия останова для нее могут быть использованы как одно из условий:

- а) $f_k^* - f_A \leq \gamma_1 r \|\partial \varphi_k(y_k)\|$;
- б) $\|y_k - y(f_k^* - \varepsilon)\| \geq \gamma_2 r$;
- в) максимальное число вычисленных субградиентов;
- г) ограничение на трудоемкость внутреннего алгоритма $-O(n^3)$, так и различного рода их комбинации.

Здесь r — минимальный радиус содержащего локализатор шара в преобразованном пространстве аргументов, γ_1, γ_2 — скаляры, такие, что $0 < \gamma_1 \leq 0.1$, $\frac{1}{n} \leq \gamma_2 < 1$. Но, так как эта процедура реализует измерения более грубо, чем «ИЗМЕРИТЕЛЬ», то ее можно рассматривать относительно нескольких точек в R^n , например, центра тяжести локализатора и точки, в которой достигается рекорд f_k^* .

Так как процедура построения эффективных отсечений основана на работе с ограниченными выпуклыми телами, то отсюда автоматически следует ответ на вопрос об информации для внешнего алгоритма. При использовании внешнего локализатора типа параллелепипеда или симплекса таковой может служить информация о вершинах некоторых локализующих множество экстремумов $K(n)$ -конусов и списке образующих их субградиентов. Но в качестве основной целесообразно ориентироваться на информацию о ширине множества экстремумов в R^n в направлениях, заданных вычисленными субградиентами, т. е. на плоские слои вида $PU(y_0, \xi, a_1, a_2)$, локализующие множество X^* . Она легко переносится в преобразованное пространство аргументов Y_k для образа множества X^* , т. е. $Y_k^* = A_k X^*$, и ее вычисление и уточнение не представляет особых проблем для некоторых выпуклых тел в R^n , локализующих множество экстремумов. Так, например, это очень легко для шара. В значительном ряде случаев для этих целей подходит $S(m)$ -пирамида, которая является пересечением шара и $K(m)$ -конуса. $S(m)$ -пирамиды легко строить на уровне правильных локализаторов в R^n . Но наиболее подходящим телом для вычисления ширины множества в различных направлениях в R^n является локализующий это множество симплекс. $K(n)$ -конус с вершиной $y(f_k^*)$ относительно точки y_k позволяет посредством добавления $(n+1)$ -й отсекающей гиперплоскости легко анализировать различные симплексы, локализующие $Y^* = A_k X^*$ для задачи (1).

Вычисление ширины множества экстремумов в некотором направлении требует для симплекса $O(n^2)$, для $S(m)$ -пирамиды $-O(nm)$ арифметических операций. Следовательно, если проводить ее вычисление

для всех хранимых субградиентов всего несколько раз для построенных внутренних локализаторов, то в целом ее трудоемкость будет не более $O(n^3)$ арифметических операций. На самом деле в практическом плане вычисление ширины множества упрощается тем, что ее легко совместить с операцией выбора очередного конуса.

Для задачи (1) информация о ширине множества X^* и его образа Y^* в направлениях вычисленных ранее субградиентов содержательна при работе внутреннего алгоритма, как для более богатого выбора отсекающих гиперплоскостей для $K(m)$ -конусов, так и для получения более точной нижней оценки f^* . Кроме того, она может быть использована непосредственно для построения локализаторов типа параллелепипеда, оси которого совпадают с n -линейно независимыми субградиентами, а также для построения шарового слоя в направлении субградиента.

Общие замечания. Итак, для общей схемы МЦТПТ мы определились с главными ее компонентами — внешним и внутренним алгоритмами. Наиболее привлекательным в рамках этой схемы выглядит МОЭ с привлечением для внешней аппроксимации множества экстремумов эллипсоидов типа $\text{Ell}(\xi, a, b)$ и $\text{Ell}(\xi_1, \xi_2, a, b, r)$, ориентируясь на работу внутреннего алгоритма, состоящего в построении эффективных локализующих множество экстремумов плоских слоев в направлениях субградиентов как на базе $S(m)$ -пирамид, так и симплексов. Для начала этого процесса достаточно знать только стартовую точку x_0 и радиус r шара с центром в x_0 , в котором содержится множество экстремумов. При этом априорное задание радиуса даже с большим запасом не столь существенно, так как он легко уточняется на уровне локализующего множество экстремумов симплекса, который содержится внутри шара. Это легко сделать как с помощью уточнения радиуса шара с центром в точке x_0 , так и с помощью эллипсоида типа $\text{Ell}(\xi, a, b)$, содержащего n -мерный ограниченный круговой конус. Шаг 2 МЦТПТ очень прост, если ориентироваться на выбор направлений движения, совпадающими с субградиентами в преобразованном пространстве. Если в качестве направления движения выбирать некоторые направления, которые являются линейной (не обязательно выпуклой) комбинацией субградиентов, то он незначительно усложнится.

При этом вполне реально создание практически эффективных алгоритмов для решения задачи (1) со средней оценкой ОИ-трудоемкости итерации $O(n^2)$. Их можно интерпретировать как методы ε -субградиентного типа с растяжением пространства, где растяжение пространства аргументов, используемое для преобразования эллипсоидов в шары, направлено как на обеспечение гарантированной оценки скорости

сходимости в смысле рекордов по функции, так и на обеспечение эффективной ε -субградиентной процедуры в преобразованном пространстве аргументов.

Однако, несмотря на близость к ε -субградиентным процедурам, все-таки направленность у внутреннего алгоритма несколько иная, в том смысле, что основная цель – не найти направление убывания функции из данной точки, а построить гиперплоскости, позволяющие хорошо аппроксимировать множество экстремумов, и только предельный его вариант можно рассматривать как ε -субградиентную процедуру. Хорошо это или плохо? В качестве аргументации попытаемся привести некоторые из проблем в выпуклом программировании, решение которых вытекает из такой схемы МЦТПТ. В основном они связаны с известными проблемами в методах отсечений, которые или не разрешены, или очень расплывчато описаны в имеющейся литературе, а также с некоторыми достаточно общими проблемами для методов в выпуклом программировании.

Для МЦТ, МВЭ и методов отсечений второй группы не решен вопрос об «отсеве» явно лишних отсекающих гиперплоскостей. В МОГ, хотя и применяются некоторые правила «отсева» [30], однако они носят эвристический характер и не обоснованы теоретически. Наиболее обоснован «отсев» гиперплоскостей в агрегатных методах, хотя и он не дает гарантий, что мы не выбросим активные. С позиций же МЦТПТ этот вопрос решается достаточно просто. Так, для отсева явно лишней отсекающей гиперплоскости внешним алгоритмом достаточно убедиться, что локализатор полностью содержится в полупространстве, которое она определяет. Эта задача тривиально решается для симплекса и шара в R^n и чуть сложнее она для куба. Более тонкий «отсев» лишних гиперплоскостей можно выполнить на уровне внутреннего алгоритма при анализе $S(m)$ -пирамид и симплексов и он будет тем лучше, чем точнее нижняя оценка f^* .

Обычно предполагают, что система ограничений, задающая выпуклый многогранник Q , совместна. Но это далеко не всегда так. Поэтому желательно иметь возможность на уровне алгоритма получать ответ на вопрос: «Совместна ли система ограничений?»

Эта проблема легко разрешима для многогранника Q , заданного системой линейных неравенств, и сигналом о ее несовместности служит тот факт, что построенный локализатор и полупространство, определяемое каким-то из неравенств, не имеют общих точек.

В случае, если выпуклое множество Q более сложной структуры, например, $Q = \{f_i(x) \leq 0, i = 1, 2, \dots, m\}$, где $f_i(x)$ – выпуклые функ-

ции, этот вопрос решается на уровне отсекающих гиперплоскостей вида $(\partial f_i(x_k), x - x_k) \leq f_i(x_k)$ для $x \in X^*$.

Автоматически решается проблема выбора шагового множителя в направлении движения для субградиентных методов. Шаг и направление движения связаны между собой в преобразованном пространстве аргументов и определяются переходом в центр тяжести выбранного локализатора множества экстремумов с меньшим объемом. Так, например, при использовании эллипсоидов типа $\text{Ell}(\xi, a, b)$ и $\text{Ell}(\xi_1, \xi_2, a, b, r)$ автоматически связываются шаг в направлении движения и коэффициент растяжения пространства аргументов в этом или некоторых других направлениях с такими характеристиками выпуклых задач, как геометрия линий уровня $f(x)$ и геометрия субдифференциального множества в точке экстремума.

С точки зрения параллельных вычислений, МЦТПТ допускает легкое распараллеливание процесса на уровне содержательных подзадач. Так, при анализе $S(m)$ -пирамиды или симплекса, каждому из подпроцессов можно поручить работу по уточнению ширины множества экстремумов в определенных для него направлениях, при анализе $K(m)$ -конуса – определение того, какой из $K(m+1)$ -конусов выбрать для проведения очередных измерений. Можно легко распараллелить и пересчет субградиентов при переходе в очередное преобразованное пространство аргументов.

Этот список вопросов можно было бы и продолжить, как, например, восстановление решения прямой задачи по решению двойственной при использовании декомпозиционного подхода, или построение достаточно хорошей нижней оценки для различного рода схем метода ветвей и границ, которые работают в прямом пространстве переменных, и др. Но не будем этим заниматься, а сосредоточим внимание только на одной достаточно общей проблеме – проблеме «рестарта» МЦТПТ, так как она приводит к нескольким неожиданным выводам.

Проблема «рестарта» связана не столько с методами типа отсечений, сколько с методами, использующими линейные преобразования пространства, и для МЦТПТ она обусловлена накоплением ошибок вычислений в матрице преобразования пространства при последовательной регуляризации внешних локализаторов. Для ее решения на уровне МЦТПТ можно привлекать как шар, так и симплекс. Но мы рассмотрим эту проблему на примере использования описанного выше варианта МОЭ для решения задачи (1), когда многогранник Q является парал-

лелепипедом, т. е.

$$Q = \{x : l_i \leq x_i \leq u_i, \quad i = 1, \dots, n\}.$$

Для того чтобы обеспечить стартовые условия для МОЭ, преобразуем пространство аргументов X , чтобы образ Q в нем был кубом Q_0 . Для этого применим последовательные растяжения пространства в направлениях e_i с коэффициентами $\max_{i=1}^n \frac{u_i - l_i}{u_i - l_i}$ для всех $i = 1, \dots, n$. Здесь e_i — n -мерный вектор, i -я компонента которого равна 1, а все остальные — нулю. В преобразованном пространстве аргументов содержащий Q_0 шар радиуса $r = \max_{i=1}^n (u_i - l_i) \frac{\sqrt{n}}{2}$ с центром, совпадающим с центром куба Q_0 , будет обеспечивать стартовую позицию для МОЭ. Все эти приготовления сводятся всего-навсего к тому, что для старта МОЭ достаточно положить $x_0 = \left\{ \frac{l_1 + u_1}{2}, \dots, \frac{l_n + u_n}{2} \right\}$;

$$B_0 = \text{diag} \left\{ \frac{u_i - l_i}{\max_{i=1}^n (u_i - l_i)} \right\}; \quad r_0 = \max_{i=1}^n (u_i - l_i) \frac{\sqrt{n}}{2}.$$

Тогда стартуем в преобразованном пространстве аргументов внутренний алгоритм, используя в качестве отсекающих гиперплоскостей и границы параллелепипеда Q , задаваемые векторами нормалей $\frac{B_0 e_i}{\|B_0 e_i\|}$.

При его работе будем определять ширину $Y^* = B_0^{-1} X^*$ как в направлениях субградиентов, так и в направлениях $\frac{B_0 e_i}{\|B_0 e_i\|}$ для всех $i = 1, \dots, n$. Это приводит к незначительному числу дополнительных арифметических операций как для $S(m)$ -пирамиды, так и для симплекса. В результате возможны два исхода: первый — мы существенно улучшили ширину Y^* в направлениях $\frac{B_0 e_i}{\|B_0 e_i\|}$, что равносильно уточнению параллелепипеда Q в X , и второй — этого не произошло. Будем ориентироваться на первый из исходов, так как при втором, применяя МОЭ с уточнением ширины $Y^* = B_k^{-1} X^*$ в направлениях $\frac{B_k e_i}{\|B_k e_i\|}$, мы рано или поздно придем к улучшению границ параллелепипеда Q в X .

Применение МОЭ в сочетании с такой схемой уточнения границ параллелепипеда, локализирующего X^* в X , позволяет решить проблему «рестарта» для МЦТПТ в том смысле, что уточненный параллелепипед в X может служить разумным начальным приближением для повторного запуска МОЭ как с полным обновлением информации об отсекающих

гиперплоскостях, так и с ее отсеком. При этом такие комбинированные варианты МЦТПТ имеют ряд преимуществ при решении стандартных задач математического программирования.

Во-первых, они более конструктивны именно для целочисленных переменных задачи (1), в том смысле, что это позволяет более сильно «зажимать» границы параллелепипеда. Так, например, если для некоторой компоненты x , скажем x_j , получен диапазон ее изменения $0.01 \leq x_j \leq 1.99$, то очевидно, что она может принимать только единственное целое значение $x_j = 1$. В их распоряжении имеется также и нижняя оценка $f^* - f_A$, которую мы можем только улучшать при переходе в очередное преобразованное пространство аргументов. Целочисленные точки, в которых следует проводить измерения, можно определять, округлив точки отрезка в X , который является образом отрезка $[y(f_k^*) - y(f_A)]$ в Y_k , чтобы они принадлежали пересечению локализатора и многогранного множества, заданного ограничениями вида (10), если параметр $t = f_k^*$.

Во-вторых, в рамках такой схемы можно строить более интересные схемы для параллельных компьютеров, которые подразумевают для алгоритмов отказ от привычной итеративной схемы $x_{k+1} = x_k + h_k p_k$. Так, например, центральный процесс, управляя методом типа отсечений, берет на себя функции контроля в X за границами параллелепипеда, нижней оценкой f^* и за шириной множества экстремумов в направлении вычисленных субградиентов. В нужный момент он может использовать для уточнения этих параметров несколько параллельных МОЭ, которые будут работать по различным центрам тяжести. Каждому из подпроцессов поручается обработка своего локализатора типа эллипсоида. Он снабжается своей матрицей преобразования и его цель — уточнить параметры для центрального процесса. При этом он может время от времени обращаться к центральному процессу как для обмена информацией по параметрам, так и по поводу решения вопросов об очередных измерениях.

В-третьих, на таком пути просматривается и выход из тупика «больших» размерностей для задач математического программирования. Так, например, для решения задач линейного программирования применение симплекс-метода ограничивается размерами памяти, требуемой для хранения матрицы обратной к базисной. Работа с разреженными матрицами позволяет увеличить размеры решаемых задач симплекс-алгоритмом. Но это характерно только для хорошо структурированных задач и в целом не снимает проблем памяти для симплекс-метода. В этом смысле надежды на проективный алгоритм Дикина — Кармаркара не

оправдались. Применение МОЭ с приведенной выше схемой уточнения границ параллелепипеда в X , учитывая, что ограничения типа равенств допускают построение эллипсоидов типа $Ell(\xi, a, b)$ объема сколь угодно близкого к нулевому, позволяют надеяться на преодоление проблем памяти для алгоритмов при решении задач линейного программирования.

Следовательно, отказ от одной из центральных идей в математическом программировании – идеи наискорейшего спуска и замена ее как можно лучшей аппроксимацией множества экстремумов не столь уж и бесперспективна.

Кроме того, следует заметить, что методы в рамках схемы МЦГПТ применимы для более широкого класса задач, чем задачи выпуклого программирования. Они могут быть использованы для поиска точек, принадлежащих некоторому стационарному множеству, если мы умеем в произвольной точке R^n убедиться, принадлежит эта точка этому множеству или строить достаточно хорошие отсекающие это множество гиперплоскости. При этом не обязательно даже, чтобы они отсекали и саму эту точку, и тем более для построения отсекающих гиперплоскостей использовать именно субградиент в точке. В этой связи эти методы применимы для задач, таких, как задача отыскания седловых точек выпукло-вогнутых функций, частные случаи задач решения вариационных неравенств, а также специальные классы задач линейной и нелинейной дополнителности.

Литература

1. ЛЕВИН А. Ю. Об одном алгоритме минимизации выпуклых функций // Докл. АН СССР. – 1965. – 160, №6. – С.1224-1227.
2. ЮДИН Д. Б., НЕМИРОВСКИЙ А. С. Информационная сложность и эффективные методы решения выпуклых экстремальных задач // Экономика и мат. методы. – 1976. – Вып.2. – С. 357–359.
3. ШОР Н. З. Метод отсечения с растяжением пространства для решения задач выпуклого программирования // Кибернетика. – 1977. – № 1. – С. 94–95.
4. ТАРАСОВ С.П., ХАЧИЯН Л. Г., ЭРЛИХ И. И. Метод вписанных эллипсоидов // Докл. АН СССР. – 1987. – 298, № 5. – С. 1081–1085.

5. KNACHIYAN L. AND TODD M. On the complexity of approximating the maximal inscribed ellipsoid for polytope // *Math. Programming.* – 1993. – **61**, N 2. – P. 137–159.
6. SHOR N. Z. AND BEREZOVSKI O. A. New algorithms for constructing optimal circumscribed and iscribed ellipsoids // *Optim. Methods and Software.* – 1992. – **1**. – P. 283–299.
7. Методы решения задач математического программирования и оптимального управления / Ащепков Л. Т., Белов Б. И., Булатов В. П., Васильев О. В., Срочко В. И., Тарасенко Н. В.. – Новосибирск: Наука, 1984. – 323 с.
8. YAMNITSKY B., LEVIN L. A. An old linear programming algorithm runs in polynomial time // 23-rd Annual. Symp. Found. Comput. Sci. Chicago III, 3–5 Nov. – Selver Spring, 1982. – P. 327–328.
9. АНЦИФЕРОВ Е. Г., БУЛАТОВ В. П. Алгоритм симплексных погружений в выпуклом программировании // *Журн. вычисл. математики и мат. физики.* – 1987. – **27**, № 3. – С. 377–384.
10. KELLEY J. E. The cutting-plane method for solving convex programs // *SIAM.* – 1960. – **8**, N 4. – P. 703–712.
11. ЗУХОВИЦКИЙ С. И., ПРИМАК М. Е. О сходимости метода чебышевских центров и метода центрированных сечений для решения задачи выпуклого программирования // *Докл. АН СССР.* – 1975. – **222**, № 2. – С. 273–276.
12. KIWIEN K. C. Methods of descent for nondifferentiable optimization. – Berlin: Springer-Verlag, 1985. – 362 p.
13. LEMARÉCHAL C., NEMIROVSKII A., NESTEROW YU. New Variants of Bundle Methods // Research Report N 1508, Institut Nationale de Recherches en Informatique et en Automatique (INRIA). – Paris, 1991. – 70 p.
14. ПШЕНИЧНЫЙ Б. Н. Метод линеаризации. – М.: Наука, 1983. – 136 с.
15. NESTEROV Y. AND NEMIROVSKII A. Interior Point Polynomial Algorithm in Convex Programming. – Philadelphia: SIAM, 1994. – 406 p.
16. НЕМИРОВСКИЙ А. С., ЮДИН Д. Б. Сложность задач и эффективность методов оптимизации. – М.: Наука, 1979. – 383 с.

17. НЕСТЕРОВА С. И., СКОКОВ В. А. Численный анализ программ негладкой безусловной оптимизации // Экономика и мат. методы. – 1994. – **30**, № 2. – С. 136–145.
18. ШОР Н. З., ЖУРБЕНКО Н. Г. Метод минимизации, использующий операцию растяжения пространства в направлении разности двух последовательных градиентов // Кибернетика. – 1971. – № 3. – С. 1–59.
19. НЕСТЕРОВ Ю. Е. Эффективные методы в нелинейном программировании. – М.: Радио и связь, 1989. – 304 с.
20. ГИЛЛ Ф., МЮРРЕЙ У., РАЙТ М. Практическая оптимизация. – М.: Мир, 1974. – 376 с.
21. ДИКИН И. И. Итеративное решение задач линейного и квадратичного программирования // Докл. АН СССР. – 1967. – **174**. – С. 747–748.
22. KARMAKAR N. A new polynomial time algorithm for linear programming // Combinatorica. – 1984. – **4**, N 4. – P. 373–395.
23. ШОР Н. З., ГЕРШОВИЧ В. И. Об одном семействе алгоритмов для решения задач выпуклого программирования // Кибернетика. – 1979. – № 4. – С. 62–67.
24. KONIG H. AND PALLASCHKE D. On Khachian's algorithm and minimal ellipsoids // Numerische Mathematik. – 1981. – **36**. – P. 211–223.
25. TODD M. J. On minimum volume ellipsoids containing part of a given ellipsoid // Math. Oper. Res. – 1982. – N 7. – P. 253–261.
26. NELDER J. A., MEAD R. A simplex method for function minimization // Comput. J. – 1965. – N 7. – P. 308–313.
27. ШОР Н. З. Методы минимизации недифференцируемых функций и их приложения. – Киев: Наук. думка, 1979. – 199 с.
28. СТЕЦЮК П. И. К обоснованию сходимости алгоритмов с растяжением пространства // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В. М. Глушкова НАН Украины, 1995. – С. 4–8.
29. HIRIART-URRUTY J.B., LEMARECHAL C. Convex analysis and minimization Algorithms. – Berlin: Springer-Verlag, 1994. – Vol I–II.
30. ПОЛЯК Б. Т. Введение в оптимизацию. – М.: Наука, 1983. – 384 с.

О трех научных идеях Н.З. Шора

И. В. Сергиенко, П. И. Стецюк

Кибернетика и системный анализ. 2012, №1. – С.4–22.

Статья посвящена 75-летию со дня рождения Н.З. Шора. Главный акцент сделан на трех его центральных идеях: обобщенном градиентном спуске (1962), использовании линейных неортогональных преобразований пространства для улучшения обусловленности овражных функций (1969), двойственном подходе к получению и уточнению оценок целевой функции в невыпуклых квадратичных моделях (1985). Приведены применения этих идей в методах и алгоритмах, разработанных в Институте кибернетики им. В.М.Глушкова НАН Украины.

Ключевые слова: недифференцируемая оптимизация, субградиентный метод, оператор растяжения пространства, метод эллипсоидов, r -алгоритм, экстремальная квадратичная задача, двойственная оценка, функционально избыточные ограничения.

Введение

В этом году исполняется 50 лет со времени опубликования Н.З. Шором работы [1], к которой восходит идея субградиентного метода минимизации негладких выпуклых функций (1962). Тогда Науму Зуселевичу исполнилось всего 25 лет. Для решения двойственных задач к транспортным задачам в матричной и сетевой формах он предложил использовать метод спуска в пространстве потенциалов. Впоследствии этот метод получил название метода обобщенного градиентного спуска (ОГС). Метод ОГС сначала применялся для минимизации кусочно-гладких выпуклых функций, появляющихся при решении транспортных и транспортно-производственных задач, а затем для класса произвольных выпуклых функций и задач выпуклого программирования. Последующие усовершенствования метода ОГС дали ряд численных методов оптимизации негладких (недифференцируемых) функций, которые существенно повлияли на развитие линейного, нелинейного, дискретного и стохастического программирования.

Настоящая статья посвящена 75-летию со дня рождения Наума Зуселевича Шора. В ней акцент сделан на трех его важнейших идеях: обобщенном градиентном спуске (1962), использовании линейных неортогональных преобразований пространства для улучшения обусловлен-

ности овражных функций (1969), двойственном подходе к получению и уточнению оценок в невыпуклых квадратичных моделях (1985). На основе первых двух идей получен ряд фундаментальных результатов в негладкой оптимизации [2], которая как самостоятельный раздел математического программирования сформировалась в конце прошлого века. На основе третьей идеи получен ряд важных результатов в многоэкстремальной оптимизации и теории графов [3].

Предложенный Н.З. Шором в [1] алгоритм, позволяющий минимизировать выпуклые функции с разрывным градиентом, приобрел большую значимость в силу многочисленных практических приложений. Именно с него началось планомерное исследование метода обобщенного градиентного спуска, который получил в дальнейшем название субградиентного метода [4]. До того времени вопросы минимизации негладких функций не привлекали большого внимания математиков, а рассматривались лишь эпизодически (например, в теории чебышевских приближений, в теории линейных неравенств). Развитие вычислительной техники, связанное с широким распространением электронных вычислительных машин (ЭВМ), стимулировало интерес к проблемам оптимизации. Оно привело к бурному развитию математического программирования, так как, с одной стороны, давало возможность реализовывать сложные алгоритмы и решать задачи высокой размерности, встречающиеся на практике; с другой стороны, появилась возможность проводить в короткие сроки проверку эффективности новых методов и алгоритмов.

О сути проблем, связанных с минимизацией недифференцируемых функций, и о вкладе Н. З. Шора в их разрешение замечательно написал Б. Т. Поляк [5]: "Основные алгоритмы минимизации гладких функций – градиентный и Ньютона – были построены на использовании линейной и квадратичной аппроксимации функции, задаваемой первыми членами ряда Тейлора. Однако, для недифференцируемой функции эта идея неприменима – такая функция не может быть хорошо аппроксимирована ни линейной, ни квадратичной функциями... Поэтому разработка методов минимизации негладких функций требует привлечения новых идей. Одна из них, принадлежащая Н. З. Шору, выглядит несколько неожиданно. Пишется прямой аналог градиентного метода с заменой градиента на произвольный субградиент $g_f(x)$ функции $f(x)$:

$$x_{k+1} = x_k - \gamma_k g_f(x_k). \quad (1)$$

... Значения функции в методе (1) не могут убывать монотонно. Оказывается, однако, что при этом монотонно убывает другая функция – расстояние до точки минимума, и в этом-то заключается основная идея

субградиентного метода (1)."

Анализу первой идеи Шора (1962) мы посвятим следующий раздел, где приведем ряд результатов по субградиентному методу (методу обобщенного градиентного спуска) и укажем на их связь с другими известными результатами.

1. Субградиентный метод

Пусть $f(x)$ – выпуклая функция, определенная в евклидовом пространстве E^n , X^* – множество минимумов (которое может быть и пустым), $x^* \in X^*$ – точка минимума; $\inf_{x \in E^n} f(x) = f^*$; $g_f(x)$ – субградиент (произвольный) функции $f(x)$ в точке x .

Определение 1. *Субградиентом выпуклой функции $f(x)$ в точке \bar{x} называется вектор $g_f(\bar{x})$, удовлетворяющий неравенству*

$$f(x) - f(\bar{x}) \geq (g_f(\bar{x}), x - \bar{x}) \quad \text{для всех } x \in E^n. \quad (2)$$

Здесь (\cdot, \cdot) – скалярное произведение векторов из E^n . Если $f(x)$ – непрерывно дифференцируема в точке \bar{x} , то субградиент $g_f(\bar{x})$ определяется однозначно и совпадает с $\nabla f(\bar{x})$ – градиентом функции $f(x)$ в точке \bar{x} . В точках негладкости функции $f(x)$ субградиент $g_f(\bar{x})$ определяется неоднозначно.

Из неравенства (2) следует, что если $f(x) < f(\bar{x})$, то субградиент $g_f(\bar{x})$ удовлетворяет неравенству

$$(-g_f(\bar{x}), x - \bar{x}) > 0. \quad (3)$$

Геометрически формула (3) означает, что антисубградиент в точке \bar{x} образует острый угол с произвольным направлением, проведенным из точки \bar{x} в точку x с меньшим значением $f(x)$. Отсюда, если X^* непусто и $\bar{x} \notin X^*$, то при сдвиге из точки \bar{x} в направлении $-g_f(\bar{x})$ с достаточно малым шагом расстояние до X^* убывает. Этот простой факт является центральной идеей субградиентного метода минимизации негладких функций, и именно о нем идет речь в приведенной выше цитате Б.Т. Поляка.

Определение 2. *Субградиентным методом называется процедура построения последовательности $\{x_k\}_{k=0}^\infty$ по правилу*

$$x_{k+1} = x_k - h_k \frac{g_f(x_k)}{\|g_f(x_k)\|}, \quad k = 0, 1, 2, \dots, \quad (4)$$

где x_0 – начальное приближение, h_k – шаговый множитель, $g_f(x_k)$ – произвольный субградиент функции $f(x)$ в точке x_k . Если $g_f(x_k) = 0$, то x_k является точкой минимума функции $f(x)$ и процесс (4) останавливается.

Наиболее общий результат о сходимости субградиентного метода связан с классическими условиями регулировки шага и содержится в следующей теореме [2].

Теорема 1. Пусть $f(x)$ – выпуклая функция с ограниченной областью минимумов X^* , $\{h_k\}_{k=0}^\infty$ – последовательность чисел, обладающая свойствами

$$h_k > 0; \quad \lim_{k \rightarrow \infty} h_k = 0; \quad \sum_{k=0}^{\infty} h_k = +\infty.$$

Тогда последовательность $\{x_k\}_{k=0}^\infty$, полученная по формуле (4), при произвольном $x_0 \in E^n$ обладает одним из следующих свойств: либо найдется такое $k = k^*$, что $x_{k^*} \in X^*$, либо

$$\lim_{k \rightarrow \infty} \min_{x \in X^*} \|x_k - x\| = 0, \quad \lim_{k \rightarrow \infty} f(x_k) = \min_{x \in E^n} f(x) = f^*.$$

Теорема 1 является одним из ярких результатов применения идеи Шора (1962). Не менее ярким результатом применения этой же идеи является предложенный Б.Т. Поляком субградиентный метод

$$x_{k+1} = x_k - h_k^* \frac{g_f(x_k)}{\|g_f(x_k)\|}, \quad h_k^* = \frac{f(x_k) - f^*}{\|g_f(x_k)\|}, \quad k = 0, 1, 2, \dots, \quad (5)$$

где регулировка шага использует априорное знание значения функции в точке минимума [5]. Здесь шаговый множитель h_k^* задает величину максимального сдвига в направлении нормированного антисубградиента, при котором угол между антисубградиентом и направлением из точки x_{k+1} в точку минимума будет нетупым. Это гарантирует уменьшение расстояния до множества X^* на каждой итерации метода (5).

Величину h_k^* называют шагом Поляка или шагом Агмона-Моцкина-Шенберга [6]. Этот шаг тесно связан с результатами И.И. Еремина о сходимости итерационных методов аппроксимации неподвижных точек с помощью операторов, обладающих свойством квазисжимаемости (фейеровости). Свойство фейеровских операторов можно считать аналогом уменьшения расстояния до множества X^* в субградиентном методе. Подробное изложение результатов по итерационным процессам фейеровского типа дано в [7].

Имеется несколько вариантов доказательства теоремы 1 или ее аналога для субградиентного процесса в форме (1). Все они основаны на изучении поведения последовательности $\{\rho_k\}_{k=0}^{\infty}$, где $\rho_k = \min_{x \in X^*} \|x_k - x\|$. Наиболее общий результат (для случая выпуклых функций, определенных в гильбертовом пространстве, когда минимизация производится при наличии ограничений) получен Б.Т. Поляком (Доклады АН СССР, 1967, 1). Аналогичный результат для конечномерного случая получен Ю. М. Ермольевым (Кибернетика, 1966, 4). Оба указанных результата используют принцип доказательства от противного и не содержат конструктивных механизмов их распространения на специальные классы выпуклых функций.

Конструктивный механизм содержит доказательство теоремы 1, принадлежащее Н.З. Шору (Труды I Зимней школы по математическому программированию, 1969, Дрогобыч). Оно использует вспомогательный результат о свойстве субградиентного процесса с постоянным шагом [2].

Лемма 1. Пусть в субградиентном методе (4) используется регуловка шага $h_k = h$, $h > 0$ для всех $k = 0, 1, \dots$. Тогда для любого $\varepsilon > 0$ и $x^* \in X^*$ найдутся такие $k = k^*$ и $\bar{x} \in E_n$, что будет выполняться свойство

$$f(\bar{x}) = f(x_{k^*}), \quad \text{причем} \quad \|\bar{x} - x^*\| < \frac{h}{2}(1 + \varepsilon).$$

Лемма 1 гарантирует уменьшение расстояния до точек области минимумов только в случае достаточно большого расстояния от множества минимумов до точек x_k . Однако с помощью этой леммы можно обосновать сходимость субградиентного метода при дополнительных предположениях о свойствах минимизируемой функции. Так, например, если множество минимумов содержит сферу радиуса $r > h/2$, то субградиентный метод с постоянным шагом h гарантирует нахождение такого k^* , что $x_{k^*} \in X^*$.

При определенных дополнительных предположениях об свойствах функции $f(x)$ были получены субградиентные методы, сходящиеся со скоростью геометрической прогрессии [2].

Теорема 2. Пусть $f(x)$ — выпуклая функция, определенная на E^n , и для всех $x \in E^n$ при некотором $\varphi (0 \leq \varphi < \pi/2)$ выполняется неравенство

$$(g_f(x), x - x^*(x)) \geq \cos \varphi \|g_f(x)\| \cdot \|x - x^*(x)\|, \quad (6)$$

где $x^*(x)$ — точка, принадлежащая множеству минимумов функции $f(x)$ и лежащая на кратчайшем расстоянии от x . Тогда, если при

заданном x_0 выбрать величину h_0 , удовлетворяющую неравенству

$$h_0 \geq \begin{cases} \|x^*(x_0) - x_0\| \cos \varphi, & \pi/4 \leq \varphi < \pi/2, \\ \|x^*(x_0) - x_0\| / (2 \cos \varphi), & 0 \leq \varphi < \pi/4, \end{cases}$$

определить $\{h_k\}_{k=0}^\infty$ в соответствии с рекуррентной формулой

$$h_{k+1} = h_k r(\varphi), \quad k = 0, 1, 2, \dots,$$

где

$$r(\varphi) = \begin{cases} \sin \varphi, & \pi/4 \leq \varphi < \pi/2, \\ 1 / (2 \cos \varphi), & 0 \leq \varphi < \pi/4, \end{cases}$$

и вычислить $\{x_k\}_{k=1}^\infty$ по формуле (4), то либо при некотором k^* имеем $g_f(x_{k^*}) = 0$ и x_{k^*} принадлежит области минимумов, либо при всех $k = 0, 1, 2, \dots$ выполняется неравенство

$$\|x_k - x^*(x_k)\| \leq \begin{cases} h_k / \cos \varphi, & \pi/4 \leq \varphi < \pi/2, \\ 2 \cos \varphi \cdot h_k, & 0 \leq \varphi < \pi/4. \end{cases}$$

Таким образом, если угол φ заранее известен, то, регулируя шаг по формулам теоремы 2, можно получить сходимость к минимуму со скоростью геометрической прогрессии со знаменателем $q = r(\varphi)$. В формуле (6) величина $\cos \varphi$ характеризует степень вытянутости поверхностей уровня функции $f(x)$. Если в некоторой окрестности минимума функции $f(x)$ не существует такого угла $\varphi < \pi/2$, что для любого x из этой окрестности выполняется (6), то такую функцию называют существенно овражной. При минимизации существенно овражных функций приведенный в теореме 2 способ регулировки шаговых множителей неприменим. В этом случае следует использовать универсальный способ выбора шаговых множителей, указанный в теореме 1.

Особо следует выделить сходящийся со скоростью геометрической прогрессии вариант субградиентного метода, где шаговый множитель остается в течение определенного числа итераций постоянным, а затем уменьшается в два раза [2].

Теорема 3. Пусть выпуклая функция $f(x)$ определена на E^n , x^* — единственная точка минимума $f(x)$ и заданы начальное приближение x_0 и числа σ и h_0 , причем $\sigma \geq 2$, $h_0 \geq \|x_0 - x^*\| / \sigma$. Рассмотрим множество

$$Y = \{y : \|y - x^*\| \leq \sigma h_0\}$$

и пусть для любой пары точек $x, z \in Y$ такой, что $f(x) = f(z) \neq f(x^*)$, выполняется условие

$$\|x - x^*\| \leq \sigma \|z - x^*\|.$$

Рассмотрим субградиентный метод (4), где $h_k = h_0 \cdot 2^{-\lfloor k/N \rfloor}$. Здесь $\lfloor a \rfloor$ — целая часть числа a . При достаточно большом h_0 и $N \geq 3\sigma 2 + 1$ выполняется неравенство

$$\|x_k - x^*\| \leq 2\sigma h_k, \quad k = 0, 1, 2, \dots$$

Теорема 3 сформулирована в терминах, характеризующих степень "вытянутости" поверхностей уровня, и легко видеть, что чем больше σ , тем меньшей будет скорость сходимости субградиентного метода.

Ступенчатая регулировка шага (как в теореме 3) была тем первым способом регулировки шага в субградиентном методе, которую в 1962 году Н.З. Шор применил при решении транспортной задачи в сетевой форме [1]. Выполнялось это на машине М-20, для которой С. В. Брановицкая составила программу. Программа имела 150 команд. Одна из задач объемом 23×467 решалась приблизительно 80 минут, задача объемом 90×243 решалась 35 минут. Анализ расчетов показал, что отклонения по стоимости от оптимального плана не превышают 0.2–0.3%. Эти расчеты на годы вперед определили модели блочных задач линейного и нелинейного программирования большой размерности (характеризуются блочной структурой матрицы ограничений и сравнительно небольшим числом связей между блоками), для которых Н.З. Шор вместе с учениками разрабатывал алгоритмы решения на основе субградиентных методов.

Ступенчатая регулировка шага, как и в теореме 3, была использована в 1967 году Н.З. Шором и М.Б. Щепакимым при построении (на основе субградиентного метода) алгоритма решения двухэтапной задачи стохастического программирования. Вопросы, связанные со ступенчатой регулировкой шага для метода ОГС, остаются актуальными и сейчас. Так, например, в недавней работе Е. А. Нурминского [8] доказана сходимость фейеровских процессов с общей адаптивной схемой управления шаговыми множителями. Она связана со ступенчатой регулировкой шага, где шаговый множитель уменьшается в несколько раз не через фиксированное количество итераций, а определяется автоматически решением вспомогательной задачи нахождения вектора минимальной длины в выпуклой оболочке накопленных ранее векторов (субградиентов).

Наиболее интенсивно исследования по субградиентным методам в Институте кибернетики проводились в 60–70 годы прошлого столетия. Параллельно они проводились учеными из других научных центров СССР, например И. И. Ереминым (Свердловск) и Б. Т. Поляком (Москва) для решения задач выпуклого программирования с ограничениями. Следует отметить, что результаты по субградиентным методам, полученные в СССР, вызвали огромный интерес за рубежом [9], когда в них увидели ключ к решению задач большой размерности. Так, например, использование схем декомпозиции для решения блочных задач линейного и нелинейного программирования приводит к сравнительно небольшим задачам минимизации негладких функций от связывающих переменных или от множителей Лагранжа для связывающих ограничений. Центральную роль при решении блочных задач математического программирования сыграли ускоренные варианты субградиентных методов, которые будут рассмотрены ниже.

Отметим, что субградиентные методы остаются актуальными и в настоящее время. Так, несмотря на огромное быстродействие и значительную оперативную память современных компьютеров, всегда найдутся задачи с таким количеством переменных, для которых они являются единственным путем их решения. Кроме того, субградиентные методы можно рассматривать как способ ускорения методов по типу покоординатного спуска, если взамен одной координаты рассматривать группу координат, зафиксировав остальные. Методы покоординатного спуска, например, пропагандируются Ю.Е. Нестеровым, как эффективный подход к решению оптимизационных задач "сверхвысоких" размеров [10].

2. Субградиентные методы с растяжением пространства

При решении практических задач субградиентные методы оказались медленно сходящимися, что привело к необходимости построения алгоритмов, которые были бы эффективными при минимизации овражных выпуклых функций. Здесь центральную роль сыграла вторая важная идея Н.З. Шора (1969). Она связана с применением линейных неортogonalных преобразований пространства для улучшения свойств оптимизируемой функции в преобразованном пространстве переменных и восходит к работе Н.З. Шора и В.И. Билецкого [11]. Суть идеи Шора состоит в следующем.

Пусть на k -ой итерации субградиентного метода производится заме-

на переменных $x = B_k y$, где B_k – неособенная $n \times n$ -матрица (т. е. существует обратная матрица $A_k = B_k^{-1}$). Субградиент выпуклой функции $f(x)$ в точке x_k удовлетворяет неравенству

$$f(x) \geq f(x_k) + (g_f(x_k), x - x_k) \quad \forall x \in E^n,$$

откуда, осуществляя замену переменных $x = B_k y$, получаем

$$\varphi(y) \geq \varphi(y_k) + (B_k^T g_f(x_k), y - y_k) \quad \forall y \in E^n.$$

Вектор $g_\varphi(y_k) = B_k^T g_f(x_k)$ удовлетворяет неравенству

$$\varphi(y) \geq \varphi(y_k) + (g_\varphi(y_k), y - y_k) \quad \forall y \in E^n$$

и является субградиентом выпуклой функции $\varphi(y) = f(B_k y)$ в точке $y_k = A_k x_k$ преобразованного пространства переменных $y = A_k x$.

Пусть к функции $\varphi(y)$ применяется субградиентный метод, где h_k – шаговый множитель в направлении нормированного антисубградиента. В преобразованном пространстве переменных $y = A_k x$ этот метод имеет вид

$$y_{k+1} = y_k - h_k \frac{g_\varphi(y_k)}{\|g_\varphi(y_k)\|} = y_k - h_k \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|} \quad (7)$$

и, следовательно, очередное приближение $x_{k+1} = B_k y_{k+1}$ будет получено по формуле

$$x_{k+1} = B_k y_k - h_k B_k \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|} = x_k - h_k B_k \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|}. \quad (8)$$

Формулы (7) и (8) являются центральными в идее Шора (1969). Очевидно, что если их дополнить правилом рекуррентного пересчета матрицы $B_{k+1} = B_k T_k$, то получим наглядную интерпретацию субградиентного метода с последовательным (от одной итерации к другой) преобразованием пространства переменных. Если матрицы T_k выбирать так, чтобы поверхность овражной функции в очередном преобразованном пространстве переменных становилась менее овражной, то такой метод окажется эффективнее, чем субградиентный метод.

Пусть x_0 – начальное приближение, B_0 – неособенная $n \times n$ -матрица. Тогда субградиентный метод с последовательным преобразованием пространства переменных имеет вид

$$x_{k+1} = x_k - h_k B_k \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|}, \quad B_{k+1} = B_k T_k, \quad k = 0, 1, \dots, \quad (9)$$

где h_k – шаговый множитель, T_k – $n \times n$ -матрица, $g_f(x_k)$ – произвольный субградиент функции $f(x)$ в точке x_k .

Метод (9) принято называть B -формой субградиентного метода с преобразованием пространства; на каждой его итерации корректируется матрица, связанная с заменой переменных $y = Bx$. Его можно записать в H -форме (по типу методов переменной метрики) с помощью симметричной матрицы $H_k = B_k B_k^T$. Если матрицы T_0, T_1, \dots, T_{k-1} – неособенные и выпуклая функция $f(x)$ – непрерывно дифференцируема, $\nabla f(x_k)$ – ее градиент в точке x_k , то направление $-B_k B_k^T \nabla f(x_k) = -H_k \nabla f(x_k)$ всегда есть направлением спуска, т. е. направлением убывания функции $f(x)$. Метод (9) в B -форме или его аналог в H -форме является компактным изложением идеи Шора.

В 1969–1971 гг. под руководством Н.З. Шора построены два семейства субградиентных методов с растяжением пространства переменных, которые различаются выбором направления растяжения. В их основе лежит оператор растяжения пространства, который в матрично-векторной форме имеет вид

$$R_\alpha(\xi) = I_n + (\alpha - 1)\xi \xi^T, \quad \xi \in E^n, \quad \|\xi\| = 1, \quad \alpha > 1,$$

где $(\cdot)^T$ означает транспонирование, I_n – единичная $n \times n$ -матрица, α – коэффициент растяжения пространства, ξ – направление растяжения. Подробно свойства оператора $R_\alpha(\xi)$ изложены в монографии [2].

При описании алгоритмов в B -форме (на каждой итерации корректируется матрица B_k) используется оператор

$$R_\beta(\xi) = R_\alpha^{-1}(\xi) = I_n + (\beta - 1)\xi \xi^T, \quad \beta = \frac{1}{\alpha} < 1.$$

Оператор $R_\beta(\xi)$ является обратным к оператору растяжения пространства $R_\alpha(\xi)$, и в методах с растяжением пространства переменных он обеспечивает пересчет матрицы B_{k+1} за $2n^2$ арифметических операций умножения. Действительно,

$$B_{k+1} = B_k R_\beta(\xi) = B_k (I_n + (\beta - 1)\xi \xi^T) = B_k + (\beta - 1)(B_k \xi) \xi^T,$$

откуда легко видеть, что вычисление вектора $\eta = B_k \xi$ требует n^2 умножений и столько же умножений требует построение одноранговой матрицы $\eta \xi^T$.

В первом семействе субградиентных методов используется операция растяжения пространства в направлении субградиента. B -форму этого семейства методов можно описать следующим образом.

Определение 3. *Субградиентным методом с растяжением пространства в направлении субградиента называется процедура построения последовательностей $\{x_k\}_{k=0}^\infty$ и $\{B_k\}_{k=0}^\infty$ по следующему правилу:*

$$x_{k+1} = x_k - h_k B_k \xi_k, \quad B_{k+1} = B_k R_{\beta_k}(\xi_k), \quad k = 0, 1, 2, \dots, \quad (10)$$

где

$$\xi_k = \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|}, \quad \beta_k = \frac{1}{\alpha_k} < 1. \quad (11)$$

Здесь x_0 — начальное приближение, $B_0 = I_n$ — единичная $n \times n$ -матрица, h_k — шаговый множитель, α_k — коэффициент растяжения пространства, $g_f(x_k)$ — произвольный субградиент функции $f(x)$ в точке x_k . Если $g_f(x_k) = 0$, то x_k является точкой минимума функции $f(x)$ и процесс (10), (11) останавливается.

Первые эксперименты показали, что, выбирая $\alpha_k = 2$ и $h_k = \text{const}$, для многих примеров выпуклых овражных функций можно получить хорошие результаты [11]. К сожалению, такой простой способ не всегда приводит к цели. Теоретически удалось обосновать такие алгоритмы, где шаговый множитель h_k и коэффициенты растяжения пространства α_k выбирались таким образом, чтобы последовательность расстояний до точки минимума в соответствующих преобразованных пространствах не возрастала [12]. Для этого использовалась дополнительная информация о функции $f(x)$ — значение функции в точке минимума f^* и так называемые постоянные роста M и N .

Теорема 4. Пусть $f(x)$ — выпуклая функция, определенная в E^n , и в некоторой сферической окрестности S_d , $S_d = \{x : \|x - x^*\| \leq d\}$, точки минимума x^* субградиент удовлетворяет двустороннему неравенству

$$N(f(x) - f(x^*)) \leq (g_f(x), x - x^*) \leq M(f(x) - f(x^*)), \quad (12)$$

где $M \geq N$ — положительные константы. Тогда если в методе (10), (11) принять

$$x_0 \in S_d, \quad h_k = \frac{2MN}{M+N} \frac{f(x_k) - f(x^*)}{\|B_k^T g_f(x_k)\|}, \quad \alpha_k = \frac{M+N}{M-N},$$

то последовательность $\{x_k\}_{k=0}^{\infty}$ удовлетворяет неравенству

$$\|A_k(x_k - x^*)\| \leq d, \quad A_k = B_k^{-1}, \quad k = 0, 1, 2, \dots \quad (13)$$

Из неравенства (13) следует локализация x^* в эллипсоиде $\Phi_k = \{x : \|A_k(x_k - x)\| \leq d\}$ с центром в точке x_k . Отношение объемов эллипсоидов Φ_{k+1} и Φ_k задается следующим равенством:

$$\frac{\text{vol}(\Phi_{k+1})}{\text{vol}(\Phi_k)} = \beta_k = \frac{M - N}{M + N}.$$

Для выпуклых функций с постоянными роста M и N теорема 4 определяет вариант субградиентного метода с растяжением пространства в направлении субградиента, который сходится со скоростью геометрической прогрессии по отклонению наилучшего достигнутого значения $f(x)$ от оптимального $f^* = f(x^*)$. Это обеспечивает выполнение неравенства (13), в соответствии с которым объем эллипсоида, в котором локализуется точка x^* , убывает со скоростью геометрической прогрессии со знаменателем $\frac{M - N}{M + N}$.

Для квадратичной положительно-определенной функции в неравенстве (12) можно выбирать $M = N = 2$. Для кусочно-линейной функции, надграфик которой представляет собой конус с вершиной в точке (x^*, f^*) , можно выбирать $M = N = 1$. Если в методе (10), (11) выбрать $\beta_{k+1} = \beta = 0$, то этим случаям соответствуют алгоритмы, которые сходятся за число шагов, не превышающее n . Решение невырожденной системы n линейных уравнений с n неизвестными $(a_i, x) + b_i = 0, \quad i = 1, \dots, n$, можно заменить нахождением минимума $f(x) = \max_{1 \leq i \leq n} |(a_i, x) + b_i|$. Если взять $f^* = 0, \beta_k = 0$ и применить метод (10), (11), то получим алгоритм, соответствующий известной конечной процедуре решения линейных алгебраических систем — методу ортогонализации градиентов.

Известный метод эллипсоидов¹ является частным случаем методов с растяжением пространства в направлении субградиента [13].

Теорема 5. Пусть $f(x)$ — выпуклая функция, определенная в E^n , и начальное приближение x_0 такое, что существует точка $x^* \in X^*$, для которой выполняется $\|x_0 - x^*\| \leq d$. Тогда, если в методе (10),

¹Первыми метод эллипсоидов в H -форме предложили Д. Б. Юдин и А. С. Немировский [14], исходя из методов последовательных отсеков.

(11) *принять*

$$h_0 = \frac{d}{n+1}, \quad h_{k+1} = h_k \frac{n}{\sqrt{n^2-1}}, \quad \alpha_k = \alpha = \sqrt{\frac{n+1}{n-1}}, \quad k = 0, 1, 2, \dots,$$

то последовательность $\{x_k\}_{k=0}^{\infty}$ удовлетворяет неравенству

$$\|A_k(x_k - x^*)\| \leq h_k(n+1), \quad A_k = B_k^{-1}, \quad k = 0, 1, 2, \dots \quad (14)$$

Множество точек x , удовлетворяющих неравенству

$$\|A_k(x_k - x)\| \leq (n+1)h_k = d \left(\frac{n}{\sqrt{n^2-1}} \right)^k,$$

представляет собой эллипсоид Φ_k . Его объем $\text{vol}(\Phi_k)$ определяется по формуле

$$\text{vol}(\Phi_k) = \frac{\nu_0 d^n \left(\frac{n}{\sqrt{n^2-1}} \right)^{nk}}{\det A_k},$$

где ν_0 — объем единичного n -мерного шара. Отсюда получаем

$$\begin{aligned} \frac{\text{vol}(\Phi_{k+1})}{\text{vol}(\Phi_k)} &= \frac{\left(\frac{n}{\sqrt{n^2-1}} \right)^n \cdot \det A_k}{\det A_{k+1}} = \frac{\left(\frac{n}{\sqrt{n^2-1}} \right)^n \cdot \det A_k}{\det R_\alpha(\xi_k) \cdot \det A_k} = \\ &= \frac{1}{\alpha} \left(\frac{n}{\sqrt{n^2-1}} \right)^n = \sqrt{\frac{n-1}{n+1}} \left(\frac{n}{\sqrt{n^2-1}} \right)^n = q_n \approx 1 - \frac{1}{2n^2}. \end{aligned}$$

Следовательно, объем эллипсоида, в котором локализуется точка x^* , в соответствии с неравенством (14) убывает со скоростью геометрической прогрессии со знаменателем q_n . Этот знаменатель зависит лишь от n — размерности пространства переменных и не зависит от свойств минимизируемой функции $f(x)$. Благодаря этому факту, метод эллипсоидов сыграл важную роль в теории сложности задач математического программирования. На его основе в 1979 г. Л. Г. Хачиян построил и обосновал первый полиномиальный алгоритм решения задачи линейного программирования с рациональными коэффициентами. Метод эллипсоидов позволил обосновать полиномиальные алгоритмы для ряда комбинаторных задач [15].

Опыт применения алгоритмов с растяжением пространства в направлении субградиента показал существенное ускорение субградиент-

ных процессов. Однако оказалось, что такие методы в принципе не могут быть монотонными. Это связано с простым геометрическим фактом: если мы находимся на границе двух "кусков" кусочно-гладкой поверхности уровня, а градиенты к этим гладким "кускам", вычисленные в данной точке, образуют тупой угол, то никакое растяжение пространства в направлении градиентов не может преобразовать этот угол в острый, он может лишь приближаться к $\pi/2$, оставаясь тупым. Применяя растяжение пространства в направлении субградиента, невозможно получить направление убывания функции в виде антиградиента к одному из кусков в растянутом пространстве. В то же время растяжение пространства в направлении разности двух указанных градиентов с достаточным коэффициентом растяжения преобразует тупой угол между градиентами в острый, т. е. соответствующие образы этих антиградиентов в растянутом пространстве становятся направлениями убывания функции.

Это стимулировало разработку второго семейства субградиентных методов, в которых используется растяжение пространства в направлении разности двух последовательных субградиентов. Эти методы получили название r -алгоритмов (от русского слова "разность") и стали одним из центральных результатов докторской диссертации Н.З. Шора (1970). Выбор шагового множителя в r -алгоритмах связан с поиском минимума функции по направлению. При определенной регулировке шага и коэффициентов растяжения пространства они являются монотонными по минимизируемой функции.

Рассмотрим описание r -алгоритмов в B -форме для минимизации выпуклой функции $f(x)$, определенной в E^n , и будем предполагать, что $f(x)$ имеет ограниченную область минимумов x^* . Для этого требуется выполнение условия $\lim_{\|x\| \rightarrow \infty} f(x) = +\infty$, которое обеспечивает корректность регулировки шага.

Определение 4. r -алгоритмом называется процедура построения последовательностей $\{x_k\}_{k=0}^{\infty}$ и $\{B_k\}_{k=0}^{\infty}$ по следующему правилу:

$$x_{k+1} = x_k - h_k B_k \xi_k, \quad B_{k+1} = B_k R_{\beta_k}(\eta_k), \quad k = 0, 1, 2, \dots, \quad (15)$$

где

$$\xi_k = \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|}, \quad h_k \geq h_k^* = \arg \min_{h \geq 0} f(x_k - h B_k \xi_k), \quad (16)$$

$$\eta_k = \frac{B_k^T r_k}{\|B_k^T r_k\|}, \quad r_k = g_f(x_{k+1}) - g_f(x_k), \quad \beta_k = \frac{1}{\alpha_k} < 1. \quad (17)$$

Здесь x_0 – начальное приближение, $B_0 = I_n$ – единичная $n \times n$ -матрица², h_k – шаговый множитель, α_k – коэффициент растяжения пространства, $g_f(x_k)$ и $g_f(x_{k+1})$ – произвольные субградиенты функции $f(x)$ в точках x_k и x_{k+1} . Если $g_f(x_k) = 0$, то x_k является точкой минимума функции $f(x)$ и процесс (15)–(17) останавливается.

Применительно к задачам минимизации гладких функций r -алгоритмы по своей формальной структуре близки к алгоритмам квазиньютоновского типа с переменной метрикой. Так, предельный вариант r -алгоритма с бесконечным коэффициентом растяжения (здесь $\beta_k = \beta = 0$, $h_k = h_k^*$) является проективным вариантом метода сопряженных градиентов [16]. Для задачи минимизации выпуклой непрерывно дифференцируемой функции $f(x)$ предельный вариант r -алгоритма с восстановлением матрицы B_k после каждой n итераций обладает квадратичной скоростью сходимости при обычных условиях гладкости и регулярности $f(x)$ [2].

Несмотря на то, что r -алгоритмы используются уже 40 лет, проблема обоснования их сходимости для всего класса выпуклых функций остается открытой и в настоящее время. Еще в 1982 г. Н.З. Шор и В.И. Гершович отметили: "Теория всего класса алгоритмов с растяжением пространства далека от совершенства. Нам кажется достаточно реалистичной целью – построение такого алгоритма, который по своей практической эффективности не уступал бы r -алгоритму и был столь же хорошо обоснован, как метод эллипсоидов". Шагом в этом направлении можно считать работу [17], где для преобразования специального эллипсоида в шар используется антиовражный прием, близкий к тому, который имеет место в r -алгоритмах. Однако, здесь растяжение пространства реализуется в направлении разности двух нормированных субградиентов, и близким к направлению разности двух субградиентов оно будет только тогда, когда нормы субградиентов близки.

Замечательное свойство r -алгоритма заключается в том, что его конкретные реализации показывают очень хорошие результаты при минимизации овражных функций. Одним из эффективных зарекомендовал себя вариант $r(\alpha)$ -алгоритма с постоянным коэффициентом растяжения пространства α и адаптивным способом регулировки шага. В нем величина h_k настраивается в процессе выполнения одномерного спуска в направлении нормированного антисубградиента в преобразованном пространстве переменных с помощью параметров h_0^0 , q_1 , n_h , q_2 . Здесь

²В качестве матрицы B_0 часто выбирают диагональную матрицу D_n с положительными элементами на диагонали, с помощью которой осуществляется масштабирование переменных.

h_0^0 – величина начального шага (используется на первой итерации, на каждой последующей итерации уточняется); q_1 – коэффициент уменьшения шага ($q_1 \leq 1$), если условие завершения спуска по направлению ($h_k^0 > h_k^*$) выполняется всего за один шаг одномерного спуска; натуральное число n_h задает число шагов одномерного спуска ($n_h > 1$), через каждые n_h шагов h_k^0 будет увеличиваться в q_2 раз, где q_2 – коэффициент увеличения шага ($q_2 \geq 1$). Подробные рекомендации по выбору коэффициента растяжения пространства и параметров адаптивной регулировки шага даны в [[18] с. 45–47]. Суть их выбора состоит в том, чтобы адаптивный способ регулировки шага позволял увеличивать точность поиска минимума функции по направлению в процессе счета и при этом число шагов по направлению не должно превышать в среднем двух-трех на одну итерацию.

За последние 40 лет в Институте кибернетики им. В. М. Глушкова НАН Украины накоплен значительный опыт решения оптимизационных задач с помощью r -алгоритма. К настоящему времени усилиями Н.З. Шора и его учеников Н.Г. Журбенко, Л.П. Шабашовой, В.И. Гершовича, А.В. Кунцевича, П.И. Стецюка, А.П. Лиховида и др. разработано несколько модификаций r -алгоритма применительно к решению различных задач оптимизации [19]–[21]. Эти модификации использовались в задачах линейного и нелинейного программирования, блочных задачах с различными схемами декомпозиции, при решении минимаксных и матричных задач оптимизации, для вычисления двойственных лагранжевых оценок в многоэкстремальных и комбинаторных задачах оптимизации. На практике они применялись для решения задач оптимального планирования, оптимального проектирования, синтеза и анализа сетей, восстановления изображений, эллипсоидальной аппроксимации и локализации и др. Модификации r -алгоритмов стали центральными методами в системах поддержки и принятия решений для планирования структурно-технологических преобразований на основе семейства оптимизационных межотраслевых моделей с переменными коэффициентами прямых затрат [22].

Следует подчеркнуть, что идея о применении линейных неортогональных преобразований пространства для улучшения свойств оптимизируемой функции в преобразованном пространстве переменных оказалась эффективной в выпуклой оптимизации. Модификации r -алгоритмов, разработанные на основе этой идеи Н.З. Шором и его школой, являются эффективными для минимизации негладких функций. Мировую известность получил метод эллипсоидов – частный случай субградиентных методов с растяжением пространства в направлении суб-

градиента. Конгресс по математическому программированию, который проводился в Бонне в 1982 г., был посвящен методу эллипсоидов и его приложениям. В избранных трудах конгресса [23] опубликован обзорный доклад Н.З. Шора „Generalized gradient methods of nondifferentiable optimization employing space dilatation operations“, о методах негладкой оптимизации, разработанных в Институте кибернетики. Участие Н.З. Шора в работе этого конгресса во многом определило его третью идею, которая будет изложена в следующем разделе.

В рамках идеи Шора (1969) в форме метода (9) естественно ожидать новых семейств субградиентных методов с преобразованием пространства для минимизации негладких выпуклых функций. Так, например, в [24] построено и обосновано ряд эффективных субградиентных методов с преобразованием пространства для нахождения точки минимума выпуклой функции при известном оптимальном значении функции. Они используют шаг Поляка в преобразованном пространстве переменных и гарантируют монотонное уменьшение расстояния до множества минимумов в последовательно преобразованных пространствах переменных. Преобразования пространства реализуются с помощью однорангового эллипсоидального оператора и доортогонализирующего однорангового оператора, и цена одной итерации этих методов такая же, как и у r -алгоритмов.

3. Двойственные оценки в экстремальных квадратичных задачах

Многие задачи булевого линейного программирования могут быть переформулированы как экстремальные квадратичные задачи с булевыми переменными. Так, например, условие булевости переменной $x \in \{0, 1\}$ представляется квадратичным равенством $x^2 - x = 0$. Если две булевы переменные x_i и x_j не могут одновременно принимать значение, равное единице, то это условие может быть записано в виде равенства $x_i x_j = 0$ (равносильно линейному неравенству $x_i + x_j \leq 1$). Для экстремальных квадратичных задач Н.З. Шор предложил использовать двойственный подход к получению и уточнению оценок целевой функции (третья важная идея). В квадратичных задачах на минимум эти оценки будут границами снизу для минимального значения целевой функции, а в квадратичных задачах на максимум – границами сверху для максимального значения целевой функции.

Третья идея Н.З. Шора включает алгоритмы нахождения двойствен-

ных оценок на основе методов недифференцируемой оптимизации и использование функционально избыточных квадратичных ограничений (их добавление не изменяет множества допустимых решений исходной квадратичной задачи) для улучшения точности двойственных оценок. Впервые двойственный подход к получению оценок предложен в работе Н.З. Шора и А.С. Давыдова (1985) для задач булевого программирования [25]. В 1986–1987 годах он был дополнен процедурами уточнения оценок и применен к другим многоэкстремальным задачам, в том числе и к задаче нахождения глобального минимума полинома [26]–[28]. Наиболее полное изложение результатов Н.З. Шора можно найти в его англоязычной монографии [3], менее полное – в русскоязычной монографии [18].

Опишем суть двойственного подхода для получения и уточнения оценок в квадратичных экстремальных задачах на примере экстремальной квадратичной задачи с ограничениями-равенствами: найти

$$Q^* = \max_{x \in E^n} Q_0(x) \quad (18)$$

при ограничениях

$$Q_i(x) = 0, \quad i = 1, \dots, m. \quad (19)$$

Здесь квадратичные функции имеют вид

$$Q_i(x) = (K_i x, x) + (b_i, x) + c_i,$$

где K_i – симметричные вещественные $n \times n$ -матрицы, b_i – n -мерные векторы из E^n , c_i – вещественные числа, $i = 0, \dots, m$. Некоторые из функций $Q_i(x)$, $i = 0, 1, \dots, m$ могут быть также линейными.

В общем случае задача (18), (19) многоэкстремальна и относится к классу NP -трудных задач. Оценки сверху для Q^* можно получить путем следующей лагранжевой релаксации. Пусть $u \in E^m$ – вектор множителей Лагранжа, соответствующий ограничениям (19). Функция Лагранжа для задачи (18), (19) имеет вид

$$L(x, u) = Q_0(x) + \sum_{i=1}^m u_i Q_i(x) \equiv (K(u)x, x) + (b(u), x) + c(u),$$

где

$$K(u) = K_0 + \sum_{i=1}^m u_i K_i, \quad b(u) = b_0 + \sum_{i=1}^m u_i b_i, \quad c(u) = c_0 + \sum_{i=1}^m u_i c_i.$$

Рассмотрим функцию

$$\Psi(u) = \max_{x \in R^n} L(x, u) \equiv \max_{x \in R^n} \left[\left(K(u)x, x \right) + \left(b(u), x \right) + c(u) \right].$$

Пусть $\lambda_{max}(K)$ – максимальное собственное число симметричной $n \times n$ -матрицы K .

Функция $\Psi(u)$ является выпуклой функцией от переменных u (как результат взятия операции максимума по переменным x для семейства линейных по переменным u функций). Область определения функции $\Psi(u)$ обозначим $\text{dom } \Psi$. Она состоит из

$$\Omega^- = \left\{ u \in E^m : \lambda_{max}(K(u)) < 0 \right\}$$

(подмножество $u \in E^m$, для которых матрица $K(u)$ отрицательно определена) и подмножества тех точек

$$u \in \Omega^0 = \left\{ u \in E^m : \lambda_{max}(K(u)) = 0 \right\},$$

для которых система линейных уравнений

$$2K(u)x + b(u) = 0 \tag{20}$$

имеет решение. Для всех других точек $\Psi(u) = +\infty$.

Если $\text{dom } \Psi \neq \emptyset$, то для любого $u \in \text{dom } \Psi$ значение функции $\Psi(u)$ является нетривиальной (т. е. неравной $+\infty$) оценкой сверху для Q^* – оптимального значения целевой функции в задаче (18), (19). Наилучшая оценка сверху для Q^* в классе лагранжевых оценок вида $\Psi(u)$ связана с решением следующей задачи негладкой оптимизации:

$$\Psi^* = \min_{u \in \text{dom } \Psi} \Psi(u). \tag{21}$$

Точками негладкости функции $\Psi(u)$ есть точки границы множества Ω^- , где система линейных уравнений (20) имеет неединственное решение.

Оценку Ψ^* с любой заданной точностью можно найти за полиномиальное время методом эллипсоидов; для одной его итерации требуется $O(m^2) + O(n^3)$ арифметических операций. Из них $O(m^2)$ операций связано с растяжением пространства двойственных переменных (множителей Лагранжа), а $O(n^3)$ операций требуются для вычисления вектора, определяющего полупространство локализации оптимальных множителей Лагранжа (при фиксированных значениях множителей Лагранжа

решается система линейных уравнений с симметричной $n \times n$ -матрицей, определяется максимальное собственное число симметричной $n \times n$ -матрицы и соответствующий этому числу собственный вектор).

Оценка Ψ^* обладает следующими свойствами. Если минимум в (21) достигается на $u^* \in \Omega^-$, то $\Psi^* = Q^*$ (т. е. оценка точная, см. лемму 4.1 из [[18], с. 90]). При этом находится и точка глобального минимума $x^* = x(u^*)$, где $x(u^*)$ – решение системы (20) при $u = u^*$. Если же минимум в (21) достигается на границе области Ω^- , то может существовать так называемый "разрыв двойственности" $\Delta^* = \Psi^* - Q^* > 0$. Один из способов уменьшения Δ^* связан с введением функционально избыточных ограничений (при этом может увеличиться и количество переменных в задаче).

Функционально избыточные ограничения – это ограничения, добавление которых не изменяет множества допустимых решений начальной квадратичной задачи. Однако при этом изменяется функция Лагранжа, что в некоторых случаях позволяет уменьшить Δ^* . Если к исходной задаче (18), (19) прибавить r функционально избыточных квадратичных ограничений $Q_{m+1}(x) = 0, \dots, Q_{m+r}(x) = 0$, $r \geq 1$, то новая квадратичная задача примет вид: найти

$$Q^* = \max_{x \in R^n} Q_0(x) \quad (22)$$

при ограничениях

$$Q_i(x) = 0, \quad i = 1, \dots, m+r. \quad (23)$$

Теорема 6. Если Ψ_1^* – оценка вида Ψ^* для задачи (22), (23), то $\Psi_1^* \leq \Psi^*$.

Доказательство. Задаче (22), (23) соответствует вектор множителей Лагранжа $U \in E^{m+r}$, и функция Лагранжа для нее имеет вид:

$$L_1(x, U) = Q_0(x) + \sum_{i=1}^{m+r} u_i Q_i(x) = L(x, u) + \sum_{i=m+1}^{m+r} u_i Q_i(x).$$

Поскольку $L_1(x, (\{u\}, 0, \dots, 0)) = L(x, u)$ и $\Psi_1(\{u\}, 0, \dots, 0) = \Psi(u)$, то

$$\Psi_1^* = \min_{U \in \text{dom } \Psi_1} \Psi_1(U) \leq \min_{u \in \text{dom } \Psi} \Psi(u) = \Psi^*.$$

Теорема 6 не только констатирует, что функционально избыточные ограничения могут улучшить точность двойственной оценки, но и поясняет, что это улучшение есть следствием увеличения количества множителей Лагранжа [3]. Ограничения, которые являются линейными комбинациями уже существующих ограничений, не отражаются на точности

двойственной оценки, т. е. $\Psi_1^* = \Psi^*$. Вклад таких ограничений в функцию Лагранжа эквивалентен лишь определенному изменению множителей Лагранжа при существующих ограничениях. Однако, добавление функционально избыточных ограничений, которые являются нетривиальными следствиями из условий задачи, в ряде случаев приводит к тому, что двойственная оценка Ψ_1^* может стать точной для Q^* . Последнее означает, что оптимальное значение Q^* со сколь угодно большой точностью можно найти за полиномиальное время, которое зависит от числа переменных и числа ограничений в квадратичной задаче.

Функционально избыточными могут быть следующие ограничения:

- а) квадратичные следствия линейных ограничений: например, квадратичное ограничение в форме $(b_i^T x + c_i)(b_j^T x + c_j) \geq 0$ является следствием из двух линейных ограничений-неравенств: $b_i^T x + c_i \geq 0$ и $b_j^T x + c_j \geq 0$;
- б) квадратичные ограничения, которые характеризуют неоднозначность представления произведения трех либо большего числа переменных задачи. Как правило, они имеют место при сведении полиномиальной задачи к квадратичной. Например, имеются переменные x_1 , $x_2 = x_1^2$ и $x_3 = x_1^3$. Тогда квадратичное ограничение $x_2^2 - x_1 x_3 = 0$ есть следствием неоднозначного представления x_1^4 , а именно $x_1^4 = (x_1^2)^2 = (x_1^3)(x_1)$;
- в) квадратичные ограничения, которые являются следствиями булевости или бинарности переменных задачи. Например, для бинарных переменных $x_i^2 = 1$, $x_j^2 = 1$, $x_k^2 = 1$ всегда справедливо квадратичное неравенство $x_i x_j + x_i x_k + x_j x_k \geq -1$.

Более детальную информацию о семействах функционально избыточных ограничений, их использовании для нахождения глобального минимума полинома и в экстремальных задачах на графах (максимальное устойчивое множество вершин графа, максимальный разрез графа и др.) можно найти в монографиях [3], [18].

Чрезвычайно интересными оказались результаты, полученные Н.З. Шором для задачи нахождения взвешенного максимального независимого множества вершин графа [3]. Эта задача играет большую роль в многочисленных приложениях: теории информации и кодировании, проектировании различных устройств при определенных условиях несовместности; она тесно связана с известными задачами выбора, разбиения множеств, раскраски графов и другими комбинаторными задачами, имеющими массу приложений. В то же время, она принадлежит

к классу NP -полных задач (при целочисленных весах вершин). Двойственные оценки Шора для этой задачи тесно связаны с известными числами Ловаса $\theta(G, w)$ и $\theta'(G, w)$ [15], которые играют большую роль при обосновании результатов о полиномиальной разрешимости ряда задач в теории графов.

Пусть $G = (V, E)$ – взвешенный неориентированный граф с множеством вершин V и множеством ребер E , вес каждой вершины $i \in V$ задан положительным целым числом w_i . Подмножество вершин $S \subseteq V$ называется устойчивым (или независимым) множеством графа G , если для любых $i, j \in S$ ребро (i, j) не принадлежит E . Взвешенное число устойчивости графа G определяется как $\alpha(G, w) = \max \sum_{i \in S} w_i$, где $S \subseteq V$ – устойчивое множество. Множество S^* , на котором достигается $\alpha(G, w)$, называется максимальным взвешенным устойчивым (или независимым) множеством. Задача нахождения $\alpha(G, w)$ является NP -трудной даже в частном случае, когда все веса равны единице [15]. Поэтому вычисление верхних оценок, достаточно хорошо аппроксимирующих сверху $\alpha(G, w)$, имеет как практический, так и теоретический интерес. Н.З. Шор предложил три верхних оценки для $\alpha(G, w)$.

Первую оценку условимся обозначать $\Psi_1^*(G, w)$. Она связана с формулировкой задачи о максимальном взвешенном устойчивом множестве графа в виде следующей квадратичной булевой задачи:

$$\alpha(G, w) = \max \sum_{i \in V} w_i x_i \quad (24)$$

при ограничениях

$$x_i x_j = 0 \quad \forall (i, j) \in E, \quad (25)$$

$$x_i^2 - x_i = 0 \quad \forall i \in V. \quad (26)$$

Здесь булева переменная $x_i \in \{0, 1\}$ равна единице, если вершина $i \in V$ включается в устойчивое множество, и равна нулю в противном случае. Булевы переменные для всех вершин описаны квадратичными ограничениями – равенствами (26). Квадратичные ограничения (25) означают, что если две вершины связаны ребром в графе G , то они обе не могут одновременно принадлежать устойчивому множеству.

Вторая оценка (обозначим ее $\Psi_2^*(G, w)$) соответствует квадратичной задаче, в которой к ограничениям (25), (26) добавлено следующее семейство функционально избыточных ограничений:

$$x_i x_j \geq 0 \quad \forall (i, j) \notin E. \quad (27)$$

Они являются следствиями неравенств $x_i = x_i^2 \geq 0$ для всех $i \in V$.

Третья оценка $\Psi_3^*(G, w)$ связана с квадратичной задачей, в которой к ограничениям (25)–(27) добавлено семейство функционально избыточных ограничений

$$x_i x_k + x_j x_k \leq x_k \quad \forall (i, j) \in E, \quad k \neq i, j. \quad (28)$$

Квадратичные ограничения (28) в форме неравенств содержат все возможные квадратичные ограничения, построенные по следующему правилу. Для булевых переменных x_i и x_j , когда $(i, j) \in E$ и вершины i и j не могут одновременно включаться в устойчивое множество, всегда справедливо линейное неравенство $x_i + x_j \leq 1$. Если его умножить на переменную x_k такую, что $k \neq i, j$, то знак неравенства не изменится в силу того, что $x_k = x_k^2 \geq 0$. В результате получаем квадратичное неравенство $x_i x_k + x_j x_k \leq x_k$, справедливое для $(i, j) \in E$ и $k \neq i, j$.

Результаты о точности всех трех оценок Шора и их связь с числами Ловаса содержатся в следующей теореме [3].

Теорема 7. *Для оценок $\Psi_1^*(G, w)$, $\Psi_2^*(G, w)$ и $\Psi_3^*(G, w)$ справедливо неравенство*

$$\Psi_1^*(G, w) = \theta(G, w) \geq \Psi_2^*(G, w) = \theta'(G, w) \geq \Psi_3^*(G, w) \geq \alpha(G, w).$$

Следовательно, оценки Шора $\Psi_1^*(G, w)$ и $\Psi_2^*(G, w)$ обладают такими же свойствами, как и числа Ловаса $\theta(G, w)$ и $\theta'(G, w)$, например, $\Psi_1^*(G, w) = \alpha(G, w)$, если граф G принадлежит семейству совершенных графов. Наиболее интересной оказалась третья оценка Шора $\Psi_3^*(G, w)$. Она всегда не хуже, чем оценка $\Psi_2^*(G, w)$, а наличие ограничений (28) придает ей ряд замечательных свойств, которые связаны со специальными семействами графов. Так, например, в [3] показано, что оценка $\Psi_3^*(G, w)$ является точной для $\alpha(G, w)$, когда граф G есть t -совершенным, либо h -совершенным.

Функционально избыточные ограничения сыграли огромную роль при получении эффективных оценок снизу для целевой функции в задачах нахождения глобального минимума полиномиальной функции $P(x)$ от одной или нескольких переменных. Эти задачи специальным образом сводятся к многоэкстремальным квадратичным задачам (на минимум) с определенными семействами функционально избыточных квадратичных ограничений. В [3] доказано, что двойственная оценка для таких квадратичных задач совпадает со значением p^* полинома $P(x)$ в точке глобального минимума тогда и только тогда, когда полином $\bar{P}(x) = P(x) - p^*$ может быть представлен как сумма квадратов других

полиномов. Эти результаты имеют отношение к классическим работам Д. Гильберта по разложению неотрицательных полиномиальных форм в сумму квадратов. Разработанный Шором метод дает возможность не только доказать существование такой декомпозиции (если она существует), но и найти одно из возможных представлений полинома $\overline{P}(x)$ в виде суммы квадратов других полиномов. Более того, этим методом можно определить значение глобального минимума полинома $P(x)$.

Итак, третья идея Шора (1985), связанная с двойственным подходом для получения и уточнения оценок в квадратичных экстремальных задачах, может быть использована при создании эффективных методов решения тех многоэкстремальных задач, которые можно описать с помощью квадратичных моделей (такие модели встречаются во многих приложениях). Эффективность таких методов обеспечивается двумя моментами. Во-первых, двойственные оценки для квадратичных моделей являются более точными, чем оценки в линеаризованных аналогах этих моделей, и целесообразно их использовать в сочетании с методом ветвей и границ. Во-вторых, использование функционально избыточных ограничений позволяет выделить среди многоэкстремальных задач такие их подклассы, которые разрешимы за полиномиальное время. При этом можно даже указать верхнюю границу сложности такого класса задач, которая зависит от количества квадратичных ограничений.

Особого внимания заслуживают и алгоритмы нахождения двойственных оценок в квадратичных моделях. Их можно считать альтернативой использованию методов внутренних точек для решения задач полуопределенного программирования (semidefinite programming), т. е. задач оптимизации, в которых в качестве ограничения фигурирует требование неотрицательной определенности некоторых матриц. Действительно, многие задачи полуопределенного программирования целесообразно рассматривать как задачи недифференцируемой оптимизации и применять для их решения эффективные методы минимизации негладких выпуклых функций. Условие неотрицательной определенности симметричной $(n \times n)$ -матрицы X (принято обозначать $X \succeq 0$) эквивалентно тому, что минимальное собственное число матрицы X не отрицательно: $\lambda_{\min}(X) \geq 0$. Но $\lambda_{\min}(X)$ – вогнутая недифференцируемая функция элементов матрицы, т. е. если элементы матрицы $X(u) = \{x_{ij}\}_{i,j=1}^n$ являются линейными функциями от вектора варьируемых параметров $u \in R^m$, то условие $X(u) \succeq 0$ эквивалентно выпуклому негладкому ограничению $\varphi(u) = -\lambda_{\min}(X(u)) \leq 0$.

Исследования в данных направлениях активно продолжаются. Так,

например, недавно выявлены новые свойства для оценки $\Psi_3^*(G, w)$ [29]. Они связаны с тем, что из квадратичных ограничений (25), (26) и (28) вытекают линейные неравенства, справедливые для ряда других известных подструктур в графе, в том числе и для p -колеса (английский термин p -wheel). В графе p -колесо состоит из клики с количеством вершин $p \geq 1$ и нечетного цикла с количеством вершин $(2k + 1)$, где $k \geq 2$, причем вершины клики и нечетного цикла не пересекаются, а каждая вершина клики связана со всеми вершинами нечетного цикла. Это дает возможность существенно расширить семейство графов, для которых задача нахождения $\alpha(G, w)$ не является NP-трудной, а может быть решена за полиномиальное время. Как частный случай, сюда относится известное семейство W -совершенных графов; для него доказательство основано на использовании метода эллипсоидов [15]. W -совершенному графу соответствует случай, когда $p = 1$, т. е. в p -колесо входит клика только с одной вершиной.

Заключение

Научное наследие Наума Зуселевича Шора намного богаче, чем нам удалось изложить в данной статье. Здесь отмечены только центральные вехи в его творчестве, сделан акцент на субградиентные методы недифференцируемой оптимизации, разработанные в Институте кибернетики им. В. М. Глушкова НАН Украины. Эти методы оказали огромное влияние на развитие теории и практики для многих направлений математического программирования. Книги Н.З. Шора стали настольными для ведущих отечественных и зарубежных специалистов в области математического программирования. Яркой характеристикой этому может служить приведенное ниже письмо профессора С. Бойда (Stephen Boyd) из Стэнфордского университета.

Уважаемый профессор Шор!

Мы никогда не встречались, но ваши работы оказали на меня огромное влияние. Я начал с вашей небольшой книги по субградиентным методам (1985), я тогда был еще аспирантом. А сейчас я читаю вашу новую книгу о недифференцируемой оптимизации (1998) и просто наслаждаюсь ею.

Я высылаю Вам три написанных мною книги. Первая – о проектировании линейных контроллеров с помощью выпуклой оптимизации, вторая – о матричных неравенствах, а третья – учебник по выпуклой оптимизации.

[...]

*Надеюсь, Вы увидите ваше сильное влияние во всех этих книгах.**С уважением,**Стефан П. Бойд, 15.04.2005*

В письме упоминаются обе англоязычные монографии Н.З. Шора. Первая книга [2] была переведена польскими учеными К. Кивелом и А. Рущинским и опубликована издательством Spriger-Verlag в 1985 г. Вторая монография [3], опубликованная издательством Kluwer Academic Publishers, не имеет русскоязычного эквивалента. Ее основой стала русскоязычная монография Н.З. Шора и С.И. Стеценко 1989 года [18], которая была дополнена рядом новых результатов по полиномиальным, матричным и булевым задачам. Центральные из этих результатов отображены в двух сборниках избранных трудов Н.З. Шора, изданных в 2008 и 2009 годах [30], [31]. Они подготовлены к опубликованию в тесном сотрудничестве с Академией транспорта, информатики и коммуникаций (г. Кишинэу, Республика Молдова) и используются в материалах учебных курсов по методам недифференцируемой оптимизации и их приложениям в сложных задачах математического программирования.

Англоязычная статья [32], опубликованная в 2002 г., заканчивается словами: „We want to emphasize that methods of nondifferentiable optimization must become necessary part of the courses of applied mathematics for mathematical and technical education.“ Это утверждение Н.З. Шор постоянно доказывал на практике, читая курсы по методам недифференцируемой оптимизации и их приложениям в ведущих вузах Украины: Киевском отделении Московского физико-технического института, Национальном техническом университете Украины „КПИ“, Киевском национальном университете имени Тараса Шевченко, Соломоновом университете. Он всегда помогал молодым ученым внимательными и доброжелательными научными консультациями, рекомендациями.

Наума Зуселевича любили и уважали все, кто имел счастье работать с ним на протяжении сорока восьми лет его научной деятельности.

Литература

1. Шор Н.З. Применение метода градиентного спуска для решения сетевой транспортной задачи. Материалы науч. семинара по теорет. и прикл. вопр. кибернетики и исследования операций: Науч. совет по кибернетики АН УССР. – Киев. – 1962. – Вып. 1. – С. 9–17.

2. Шор Н.З. Методы минимизации недифференцируемых функций и их приложения. – К.: Наук. думка, 1979. – 199 с.
3. Shor N.Z. Nondifferentiable optimization and polynomial problems. – Boston; Dordrecht; London: Kluwer Academic Publishers, 1998. – 412 p.
4. Сергієнко І. В. Методи оптимізації та системного аналізу для задач трансобчислювальної складності. – К.: Академперіодика, 2010. – 296 с.
5. Поляк Б.Т. Введение в оптимизацию. – М.: Наука, 1983. – 384 с.
6. Еремін І. І. Обобщение релаксационного метода Агмона-Моцкина // УМН. – 1965. – т. XX. – Вып. 2(122). – С. 183-187.
7. Васин В. В., Еремін І. І. Операторы и итерационные процессы фейеровского типа (теория и приложения). – Москва; Ижевск: Институт компьютерных исследований, НИЦ "Регулярная и хаотическая динамика", 2005. – 200 с.
8. Нурминский Е. А. Фейеровские алгоритмы с адаптивным шагом // Журн. вычисл. математики и мат. физики. – 2011. – 51. – Вып. 5. – С. 791–801.
9. Balinski M. L., Wolfe P. (eds.) Nondifferentiable optimization. Math. Programming Study, 3. – Amsterdam: North-Holland, 1975. – 178 p.
10. Nesterov Y. Efficiency of coordinate descent methods on huge-scale optimization problems // CORE Discussion Paper #2010/2, 2010. – 23 p.
11. Шор Н. З., Билецкий В. И. Метод растяжения пространства для ускорения сходимости в задачах овражного типа // Тр. семинара Науч. совета АН УССР по кибернетике "Теория оптимальных решений". – Киев. – 1969. – № 2. – С. 3–18.
12. Шор Н. З. Использование операций растяжения пространства в задачах минимизации выпуклых функций // Кибернетика. – 1970. – № 1. – С. 6–12.
13. Шор Н.З. Метод отсечения с растяжением пространства для решения задач выпуклого программирования // Там же. – 1977. – № 1. – С. 94–95.

14. Юдин Д. Б., Немировский А. С. Информационная сложность и эффективные методы решения выпуклых экстремальных задач // Экономика и мат. методы. – 1976. – Вып. 2. – С. 357–359.
15. Grötschel M., Lovász L., Schrijver A. Geometric algorithms and combinatorial optimization. – Berlin: Springer-Verlag, 1988. – 362 p.
16. Шор Н.З., Журбенко Н.Г. Метод минимизации, использующий операцию растяжения пространства в направлении разности двух последовательных градиентов // Кибернетика. – 1971. – № 3. – С. 51–59.
17. Стецюк П. И. r -алгоритмы и эллипсоиды // Кибернетика и системный анализ. – 1996. – № 1. – С. 113–134
18. Шор Н.З., Стеценко С.И. Квадратичные экстремальные задачи и недифференцируемая оптимизация. – Киев: Наук. думка, 1989. – 208 с.
19. Шор Н.З., Стецюк П.И. Использование модификации r -алгоритма для нахождения глобального минимума полиномиальных функций // Кибернетика и системный анализ. – 1997. – № 4. – С. 28–49.
20. Kappel F., Kuntsevich A. V. An implementation of Shor's r -algorithm // Computational Optimization and Applications. – 2000. – **15**. – P. 193–205.
21. Шор Н.З., Журбенко Н. Г., Лиховид А. П., Стецюк П. И. Развитие алгоритмов недифференцируемой оптимизации и их приложения // Кибернетика и системный анализ. – 2003. – № 4. – С. 80–94.
22. Сергиенко И. В., Михалевич М. В., Стецюк П. И., Кошлай Л. В. Модели и информационные технологии для поддержки принятия решений при проведении структурно-технологических преобразований // Там же. – 2009. – № 2. – С. 26–49.
23. Bachem A., Grötschel M., Korte B. (eds.) Mathematical Programming: the state of art, Bonn, 1982. Berlin: Springer-Verlag, 1983. – 655 p.
24. Стецюк П. И. Ортогонализирующие линейные операторы в выпуклом программировании // Кибернетика и системный анализ. – 1997. – 3. – С. 97–119 (Ч. I). – 1997. – № 5. – С. 111–124 (Ч. II).

25. Шор Н.З., Давыдов А. С. О методе получения оценок в квадратичных экстремальных задачах с булевыми переменными // Там же. – 1985. – № 2. – С. 48–50.
26. Шор Н.З. Квадратичные оптимизационные задачи. – Известия АН СССР. "Техническая кибернетика". – 1987. – № 1. – С 128–139.
27. Шор Н.З. Об одном подходе к получению глобальных экстремумов в полиномиальных задачах математического программирования // Кибернетика. – 1987. – № 5. – С. 102–106.
28. Шор Н.З. Об одном классе оценок глобального минимума полиномиальных функций // Там же. – 1987. – № 6. – С. 9–11.
29. Стецюк П.И. О новых свойствах оценок Шора для взвешенного числа устойчивости графа // Праці міжнарод. конф. "50 років Інституту кібернетики ім. В.М. Глушкова НАН України". – К.: Ін-т кібернетики ім. В.М. Глушкова НАН України, 2008. – С. 164–173.
30. Шор Н.З. Методы недифференцируемой оптимизации и сложные экстремальные задачи: Сб. избр. тр. – Кишинэу: Эврика, 2008. – 270 с.
31. Шор Н.З. Методы минимизации негладких функций и матричные задачи оптимизации: Сб. избр. тр. – Кишинэу: Эврика, 2009. – 240 с.
32. Shor N.Z., Stetsyuk P.I. Lagrangian bounds in multiextremal polynomial and discrete optimization problems // J. of Global Optimization. – 2002 – **23**. – P. 1–41.

Список научных трудов (*Стецюк, 1982–2013*)

1982–1989 годы

1. Юн Г.Н., Андрусенко С.К., Стецюк П.И., Дрюченков В.С. Опыт применения экономико-математического моделирования при обосновании уровня лимитных цен на новые воздушные суда ГА // В сб. "Проблемы исследования перспектив гражданской авиации" – М: ГосНИИ ГА, 1982.
2. Нурминский Е.А., Андрусенко С.К., Стецюк П.И. О новом полиномиальном алгоритме линейного программирования // Кибернетика. – 1985. – № 4. – С. 118–120.
3. Андрусенко С.К., Ивличев В.П., Стецюк П.И. и др. „ПЕ-ГАС“ – проблемно-ориентированная система перспективного планирования // IV Всесоюзная конференция „Диалог человек–ЭВМ“. Тезисы докладов. – Киев, 1985.
4. Нурминский Е.А., Стецюк П.И. Вопросы реализации полиномиального алгоритма ЛП // Девятый всесоюзный симпозиум "Системы программного обеспечения решения задач оптимального планирования" (г. Минск, 23 февраля–3 марта 1986 г.). Краткие тезисы докладов. – М.: ЦЭМИ АН СССР, 1986. – С. 158–159.
5. Андрусенко С.К., Нурминский Е.А., Стецюк П.И. Численные эксперименты с новым классом алгоритмов в линейном программировании // Журнал вычислительной математики и математической физики. – 1987. – Т. 27, № 3. – С. 349–356.
6. Стецюк П.И. Простые границы в полиномиальном алгоритме ЛП // Системы программного обеспечения решения задач оптимального планирования. 10-й Всесоюзн. симп. (г. Нарва-Йыэссуу). Краткие тез. докл. – М: ЦЭМИ АН СССР, 1988.
7. Стецюк П.И. Об одном подходе к учету простых границ в полиномиальном алгоритме ЛП // Методы решения экстремальных задач и смежные вопросы. – Киев: Ин-т кибернетики им. В.М.Глушкова АН УССР, 1989. – С. 60–65.

8. Ивличев В.П., Лефтеров А.В., Стецюк П.И. Об опыте создания системы поддержки и принятия решений "АНАЛИЗ" // Диалоговая оптимизация планово-управленческих решений и проблемы внедрения ее в практику. Тезисы докладов научно-практического семинара, 3–5 октября 1989 г. – Киев: Общество "Знание" Украинской ССР, 1989. – С. 32–34.

1990–1994 годы

9. ХАБАТОВ Р.Ш., ЖУРБЕНКО Н.Г., МИЩУК С.А., СТЕЦЮК П.И. Методика оптимизации состава МТП // Оптимизация машинно-тракторного парка: Сб. науч. тр. – М.: МСХА, 1990. – С. 10–17.
10. СТЕЦЮК П.И. Об одном алгоритме ЛП // Краткие тезисы докладов. Одиннадцатая всесоюзная школа "Системы программного обеспечения решения задач оптимального планирования" (г. Кострома, 21–29 мая 1990 г.). – Москва, 1990. – С. 77–78.
11. ЖУРБЕНКО Н.Г., СТЕЦЮК П.И. Агрегирование и декомпозиция для одного класса динамических распределительных задач // Исследования методов решения экстремальных задач. – Киев: Ин-т кибернетики им. В.М. Глушкова АН УССР, 1990. – С. 4–9.
12. МИХАЛЕВИЧ В.С., СЕРГИЕНКО И.В., ШОР Н.З., др., СТЕЦЮК П.И. Пакет прикладных программ для решения на ЕС ЭВМ в диалоговом режиме задач нелинейной и дискретной оптимизации (ППП ДИСНЕЛ) // АН УССР, Ин-т кибернетики им. В.М. Глушкова. – Киев, 1990. – Рук. деп. в ГосФАП СССР, ВНИЦентр, Инв. № 029.10 019092, 1991. – 449 с.
13. СТЕЦЮК П.И. Об одном методе решения блочной задачи линейного программирования // Моделирование и оптимизация. – Киев: Ин-т кибернетики им. В.М. Глушкова АН УССР, 1991. – С. 35–40.
14. ЖУРБЕНКО Н.Г., СТЕЦЮК П.И., ДОЛГОВ В.В., ТИХОНОВ О.А. Об одной задаче оптимизации многослойных структур в плоских защитах // Методы решения задач нелинейного и дискретного программирования. – Киев: Ин-т кибернетики им. В.М. Глушкова АН УССР, 1991. – С. 36–41.
15. ЖУРБЕНКО Н.Г., САИМБЕТОВ Д.Х., СТЕЦЮК П.И., ТОРЕБАЕВ О. Задача перспективного планирования структуры машинно-тракторного парка // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 1992. – С. 58–63.

16. ЛАПТИН Ю.П., МЕДВЕДЕВ В.А., ЖУРБЕНКО Н.Г., СТЕЦЮК П.И., КУЗЬМИН А.И. Задача оптимизации многоступенчатых конвективных пароперегревателей // Информационные технологии в задачах управления сложными системами. – Киев: Ин-т кибернетики им. В.М.Глушкова АН Украины, 1992. – С. 39–47.
17. ЖУРБЕНКО Н.Г., ЛАПТИН Ю.П., МЕДВЕДЕВ В.А., СТЕЦЮК П.И., КУЗЬМИН А.И. Об одной задаче оптимизации теплоэнергетических установок // Теория и вычисл. проблемы оптимизации. – Киев: Ин-т кибернетики им. В.М.Глушкова АН Украины, 1993. – С. 36–39.
18. ШОР Н.З., СТЕЦЮК П.И. Кусочно-вогнутая задача рюкзачного типа // Методы исследования экстремальных задач. – Киев: Ин-т кибернетики им. В.М.Глушкова АН Украины, 1994. – С. 21–28.

1995 год

19. СТЕЦЮК П.И. К вопросу сходимости r -алгоритмов // Кибернетика и систем. анализ. – 1995. – № 6. – С. 173–177.
20. СТЕЦЮК П.И. К обоснованию сходимости алгоритмов с растяжением пространства // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М.Глушкова НАН Украины, 1995. – С. 4–8.
21. СТЕЦЮК П.И. Об одной схеме методов отсечений – Киев, 1995. – 34с. – (Препр. / НАН Украины. Ин-т кибернетики им. В.М.Глушкова: 95-25).

1996 год

22. СТЕЦЮК П.И. r -алгоритмы и эллипсоиды // Кибернетика и систем. анализ. – 1996. – № 1. – С. 113–134.
23. СТЕЦЮК П.И. Классические фейеровские методы с преобразованием пространства // Методы решения экстремальных задач. – Киев: Ин-т кибернетики им. В.М.Глушкова НАН Украины, 1996. – С. 3–9.
24. СТЕЦЮК П.И. Квазиньютоновские методы и r -алгоритмы – Киев, 1996. – 21 с. – (Препр. / НАН Украины. Ин-т кибернетики им. В.М.Глушкова: 96-10).

25. СТЕЦЮК П.І. Субградієнтні методи з перетворенням простору для мінімізації негладких опуклих функцій // Автореферат дисертації на здобуття наукового ступеня кандидата фізико-математичних наук. Ін-т кібернетики ім. В.М. Глушкова НАН України. – Київ, 1996. – 16 с.
26. СТЕЦЮК П.І. Субградиентные методы с преобразованием пространства для минимизации негладких выпуклых функций // Рукопись диссертации на соискание научной степени кандидата физико-математических наук. – Киев, 1996. – 135 с.
27. СТЕЦЮК П.І. Метод центров тяжести простых тел // Кибернетика и систем. анализ. – 1996. – № 5. – С. 117–138.

1997 год

28. СТЕЦЮК П.І. Ортогонализующие линейные операторы в выпуклом программировании (Часть I) // Кибернетика и системный анализ. – 1997. – № 3. – С. 97–119.
29. СТЕЦЮК П.І. Ортогонализующие линейные операторы в выпуклом программировании (Часть II) // Кибернетика и системный анализ. – 1997. – № 5. – С. 111–124.
30. СТЕЦЮК П.І. Об одном способе ускорения метода ортогонального субградиентного спуска // Оптимизация и ее приложения. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 1997. – С. 42–49.
31. ШОР Н.З., СТЕЦЮК П.І. Использование модификации r -алгоритма для нахождения глобального минимума полиномиальных функций // Кибернетика и систем. анализ. – 1997. – № 4. – С. 28–49.
32. ШОР Н.З., СТЕЦЮК П.І. Использование r -алгоритма в задачах полуопределенного программирования // Праці міжнародної конференції "Питання оптимізації обчислень" (6–8 жовтня 1997 р., м. Київ). – С. 330–335.
33. BARDADYM T.A., SHOR N.Z., KNOPOV P.S., LIKHOVID A.P., STETSYUK P.I. On the Use of Nonsmooth Optimization Methods in Statistics and Stochastic Programming // Thesis of 16-th International Symposium on Mathematical Programming, Lausanne, 1997.

1998 год

34. СТЕЦЮК П.И. Линейные операторы в квазиньютоновских методах // Теория и приложения методов оптимизации. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 1998. – С. 3–8.
35. СЕРГИЕНКО И.В., МИХАЛЕВИЧ М.В., СТЕЦЮК П.И., КОШЛАЙ Л.Б. Межотраслевая модель планирования структурно-технологических изменений // Кибернетика и систем. анализ. – 1998. – № 3. – С. 3–17.
36. BARDADYU T.A., SHOR N.Z., ZHURBENKO N.G., LIKHOVID A.P., STETSYUK P.I. Solver for Linear Two-Stage Fixed Recourse Problems // Thesis of OR98 International Conference on Operations Research, Zurich, 1998. – P. 65–66.

1999 год

37. СТЕЦЮК П.И. К методам эллипсоидов // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 1999. – С. 27–33.
38. ШОР Н.З., БАРДАДЫМ Т.А., ЖУРБЕНКО Н.Г., ЛИХОВИД А.П., СТЕЦЮК П.И. Использование методов негладкой оптимизации в задачах стохастического программирования // Кибернетика и системный анализ. – 1999. – № 5. – С. 33–47.
39. ШОР Н.З., СТЕЦЮК П.И., КРЫЛОВ С.В. Программы нахождения оптимальных по объему эллипсоидов, описанных вокруг набора точек // Thesis of International Conference "Dynamical Systems Modelling and Stability Investigation" (May 25-29, 1999, Kyiv). – P. 62.

2000 год

40. СТЕЦЮК П.И. Об одном методе для нахождения допустимой точки выпуклого неравенства // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 2000. – С. 3–10.
41. STETSYUK P. Method FEG2P1 to Find ε -Solutions of Convex Inequalities // Thesis of U.S.-Ukrainian Workshop "Recent Advances in Non-Differentiable Optimization" (May 15-18, 2000, Kyiv, Ukraine). – P. 36.

42. СТЕЦЮК П.И., КОЛЕСНИК Ю.Г. К вопросу выбора метода аппроксимации результатов измерений // Интеллектуальные информационно-аналитические системы и комплексы. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 2000. – С. 62–67.
43. СТЕЦЮК П.И., КОЛЕСНИК Ю.Г., ПЕПЕЛЯЕВ В.А. Об оценке коэффициентов производственной функции Кобба-Дугласа при неокласических критериях // Теорія оптимальних рішень. Моделювання та керування в умовах невизначеності. – Київ: Ін-т кібернетики ім. В.М. Глушкова НАН України, 2000. – С. 14–23.
44. ШОР Н.З., СТЕЦЮК П.И., КРЫЛОВ С.В. Нахождение глобальных минимумов полиномиальных функций с использованием двойственных квадратичных оценок // Вестник Международного Соломонова университета. – 2000. – № 4. – С.217–233.
45. СЕРГИЕНКО И.В., ШИЛО В.П., СТЕЦЮК П.И. Приближенные алгоритмы для решения задачи нахождения максимально независимого множества // Компьютерная математика. – Киев: Ин-т кибернетики им. В.М.Глушкова НАН Украины, 2000. – С. 4–20.

2001 год

46. СТЕЦЮК П.И. К методам решения плохообусловленных систем линейных уравнений // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М.Глушкова НАН Украины, 2001. – С. 9–15.
47. СТЕЦЮК П.И. Об ускорении сходимости методов эллипсоидов // Труды XII Байкальской международной конференции (Иркутск, Байкал, 24 июня – 1 июля 2001 г.). Том 1. Математическое программирование. – Иркутск, 2001. – С. 61–66.
48. PARDALOS P., BUTENKO S., SERGIENKO I.V., SHYLO V.P., STETSYUK P.I. Approximate and Exact Algorithms for Determination of a Maximum Independent Set // Thesis of WISTCIS Kick-Off Conference "Teleworking in Research, Medicine and Business"(April 19-21, 2001, Kiev, Ukraine). – P. 18.
49. СТЕЦЮК П.И. Линейная модель для нахождения пропускных способностей компонент надежной сети // Збірник наукових праць „Комп'ютерна математика. Оптимізація обчислень.“ – Київ: Ін-т кібернетики ім. В.М. Глушкова НАН України, 2001. – Т. 1. – С. 376–384.

50. SHOR N.Z. AND STETSYUK P.I. Dual Solution of Quadratic-Type Problems by r -algorithm (subroutine DSQTP_r) // Abstracts of Second International Workshop „Recent Advances in Non-Differentiable Optimization“ (October 1-4, 2001, Kyiv, Ukraine). – P. 36.
51. SHYLO V.P. AND STETSYUK P.I. Using GES and r -algorithm for Solving Multidimensional Knapsak Problems // Abstracts of Second International Workshop „Recent Advances in Non-Differentiable Optimization“ (October 1-4, 2001, Kyiv, Ukraine). – P. 38.
52. BUTENKO S., PARDALOS P.M., SERGIENKO I.V., SHYLO V., STETSYUK P. Approximate and Exact Algorithms for the Maximum Independent Set // "MaxClique '01" Conference, University of Klagenfurt, Austria, May 31 – June 2, 2001.

2002 год

53. SHOR N.Z. AND STETSYUK P.I. Constructing Utility Functions by Methods of Nondifferentiable Optimization // in: A.S. Tangian, AND J. Gruber (Eds.) *Constructing and Applying Objective Functions*, Lecture Notes in Economics and Mathematical Systems. – Vol. 510. – Berlin.: Springer-Verlag. – 2002. – P. 215–232.
54. SHOR N.Z. AND STETSYUK P.I. Lagrangian bounds in multiextremal polynomial and discrete optimization problems // Journal of Global Optimization. – 2002. – **23**. – P. 1–41.
55. ШОР Н.З., СТЕЦЮК П.И., ЛЕЙБОВИЧ М.М. К выбору метода оценки параметров линейной модели // Матеріали ІХ-ої Міжнародної наукової конференції ім. академіка М.Кравчука (16-19 травня 2002 р., Київ). – К.: НТТУ "КПІ". – 2002. – С. 399.
56. STETSYUK P.I. On one interesting ellipsoid // Международная конференция, посвященная 65-летию со дня рождения Б.Н.Пшеничного. Тезисы докладов. – Киев: Издательство НТТУ "КПИ", 2002. – С. 53-54.
57. СТЕЦЮК П.І. Про одну модифікацію методу еліпсоїдів // Міжнародна школа-семінар "Теорія прийняття рішень", Ужгород, 7–12 жовтня 2002. Праці школи семінару. – С. 64.

58. СТЕЦЮК П.И., БУХАНЦОВ Д.М. К ускорению метода эллипсоидов с помощью использования шарового слоя // Збірка тез міжнародної конференції "Обчислювальна та прикладна математика" присвяченої 80-річчю академіка І.І. Ляшка (9.09.–10.09.2002) – Київ, 2002 – С. 94.
59. СТЕЦЮК П.И., БУХАНЦОВ Д.М. К ускорению метода эллипсоидов с помощью использования шарового слоя // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 2002. – С. 63–70.
60. BUTENKO S., PARDALOS P.M., SERGIENKO I.V., SHYLO V., STETSYUK P. Finding Maximum Independent Sets in Graphs Arising from Coding Theory // Proceedings of the 17th Symposium on Applied Computing. – ACM Press, 2002. – P. 542–546.
61. СТЕЦЮК П.И., КОЛЕСНИК Ю.С., ЛЕЙБОВИЧ М.М. О робастности метода наименьших модулей // Компьютерная математика. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 2002. – Выпуск 2. – С. 114–123.

2003 год

62. СТЕЦЮК П.И. Модификация метода эллипсоидов // Информационный бюллетень Ассоциации математического программирования. №10. Научное издание. – Екатеринбург: УрО РАН, 2003. – С. 216–217.
63. СТЕЦЮК П.И. Приближенный метод эллипсоидов // Кибернетика и системный анализ. – 2003. – № 3. – С. 141–146.
64. СТЕЦЮК П.И. Об одном эллипсоиде для внешней аппроксимации n -мерного полушара // Компьютерная математика. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 2003. – Выпуск 2. – С. 144–151.
65. СТЕЦЮК П.И., БУХАНЦОВ Д.М. Нахождение квадратичных функций Ляпунова для наилучших оценок решения линейной системы // Міжнародна наукова конференція "ШОСТІ БОГОЛЮБОВСЬКІ ЧИТАННЯ", м. Чернівці, Україна, 26-30 серпня 2003 р.

66. СТЕЦЮК П.И., КОЛЕСНИК Ю.С., БЕРЕЗОВСКИЙ О.А. Об одном методе нахождения L_p -решения системы линейных уравнений // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М.Глушкова НАН Украины, 2003. – С. 83–90.
67. ШОРН.З., ЖУРБЕНКО Н.Г., ЛИХОВИД А.П., СТЕЦЮК П.И. Развитие алгоритмов недифференцируемой оптимизации и их приложения // Кибернетика и системный анализ. – 2003. – № 4. – С. 80–94.
68. МІЦА О.В., СТЕЦЮК П.І. Задача знаходження оптимальних параметрів однорідного оптичного покриття // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М.Глушкова НАН Украины, 2003. – С. 127–134.

2004 год

69. БАБАК О.В., ГАСАНОВ А.С., ЛЕЙБОВИЧ М.М., СТЕЦЮК П.И. Об одном подходе к использованию метода наименьших модулей при построении линейных моделей // Кибернетика и системный анализ. – 2004. – №3. – С. 109–115.
70. GOLDENGORIN B., KUZ'MENKO V., STETSYUK P., TSO M. Pricing by an Allocation Model with Different Types of Discount // Book of Abstracts, (CO 2004, 28–31 March 2004). – Lancaster University, Lankaster, UK. – P. 31.
71. ШОРН.З., СТЕЦЮК П.И., БЕРЕЗОВСКИЙ О.А. Двойственные оценки для специальной оптимизационной задачи квадратичного типа на многообразии Штиффеля // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 2004. – С. 3–10.
72. ШОРН.З., СТЕЦЮК П.И., БЕРЕЗОВСКИЙ О.А. Нахождение оценок глобального экстремума в задачах на многообразии Штиффеля // Автоматика-2004. Материалы 11-й международной конференции по автоматическому управлению, 27–30 сентября 2004, Национальный университет пищевых технологий. – Киев, 2004. – Т. 1. – С. 47.
73. КУЗЬМЕНКО В.Н., ГОЛЬДЕНГОРИН Б.И., ТСО М., СТЕЦЮК П.И. Сравнение двух субградиентных методов при нахождении оценок

- для задач размещения // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М.Глушкова НАН Украины, 2004. – С. 111–119.
74. СТЕЦЮК П.І., МИЦА О.В. До використання методів першого порядку в задачах оптимізації багатошарових оптичних плівок // II-а Міжнародна школа-семінар "Теорія прийняття рішень", Ужгород, 27 вересня – 2 жовтня 2004 р. Праці школи-семінару. – С. 88.

2005 год

75. СТЕЦЮК П.І., МИЦА А.В. О задачах оптимизации параметров для многослойных оптических покрытий // Кибернетика и системный анализ. – 2005. – № 4. – С. 107–115.
76. СТЕЦЮК П.І., МИЦА А.В. О вычислении градиента в задаче синтеза оптических покрытий // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М.Глушкова НАН Украины, 2005. – С. 127–133.
77. SHOR N. AND STETSYUK P. Quadratic-type models and upper bounds for the problems of finding the maximum weighted independent set in graphs // Conference on Systems Analysis, Data Mining and Optimization in Biomedicine, February 2-4, 2005, University of Florida, Gainesville, FL.
78. СТЕЦЮК П.І. О функционально избыточных ограничениях для булевых оптимизационных задач квадратичного типа // Кибернетика и системный анализ. – 2005. – № 6. – С. 168–172.
79. Задачі оптимального проектування надійних мереж / Шор Н.З., Сергієнко І.В., Шило В.П., Стецюк П.І., Парасюк І.М., Лебедєва Т.Т., Лаптін Ю.П., Журбенко М.Г., Бардадим Т.О., Шаріфов Ф.А., Лиховид О.П., Березовський О.А., Мірошніченко В.М. – Київ, Наукова думка, 2005. – 230 с.

2006 год

80. СТЕЦЮК П.І. Новые модели квадратичного типа для задачи о максимальном взвешенном разрезе графа // Кибернетика и системный анализ. – 2006. – №1. – С. 63–75.

81. СТЕЦЮК П.И., БЕРЕЗОВСКИЙ О.А. Лагранжевые оценки для максимального разреза графа с заданными количествами вершин в обоих подмножествах разбиения // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 2006. – С. 31-38.
82. СТЕЦЮК П.И., ПАРДАЛОС П.М. Об уточнении лагранжевых двойственных оценок в бинарных и булевых квадратичных задачах // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М.Глушкова НАН Украины, 2006. – С. 145–153.
83. STETSYUK P.I., BARDADYM T.O. The Use of Odd Sets of Variables to Improve Bounds in Boolean Quadratic Problems // Abstracts of International Conference in Honor of the 70th Birthday of Academician I.V.Sergienko "Applied Optimization and Metaheuristic Innovations"(July 19-1, 2006, Yalta, Ukraine)
84. МІЦА О.В., СТЕЦЮК П.І. Дослідження оптимізації спектральних характеристик семишарового оптичного покриття з чергуючими-ся шарами // III-а Міжнародна школа-семінар "Теорія прийняття рішень", Ужгород, 2 – 7 жовтня 2006 р. Праці школи-семінару. – С. 73–74.
85. П.И.СТЕЦЮК П.И., ПАРДАЛОС П.М., КРОШКО Д.Л. О новых лагранжевых двойственных оценках для числа устойчивости графа // Компьютерная математика. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 2006. – Выпуск 3. – С. 149–158.

2007 год

86. СТЕЦЮК П.И. Об уточнении верхних оценок Н.З.Шора для числа независимости неориентированного графа // Информационный бюллетень Ассоциации математического программирования. № 11. Научное издание. – Екатеринбург: УрО РАН, 2007. – С. 80–81.
87. СТЕЦЮК П.И., ЧУМАКОВ Б.М. О свойствах одной верхней оценки Н.З.Шора для взвешенного числа устойчивости графа // Праці міжнародного симпозіума "Питання оптимізації обчислень (ПОО–XXXIII)". – К.: Ін-т кібернетики ім. В.М.Глушкова НАН України, 2007. – С. 271-272.

88. СТЕЦЮК П.И., БУТЕНКО С.И., БЕРЕЗОВСКИЙ О.А. Об одной верхней оценке для взвешенного числа устойчивости графа // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М.Глушкова НАН Украины, 2007. – № 6. – С. 80–89.

2008 год

89. БЕРЕЗОВСКИЙ О.А., СТЕЦЮК П.И. Об одном способе нахождения двойственных квадратичных оценок Шора // Кибернетика и системный анализ. – 2008. – № 2. – С. 89–99.
90. СТЕЦЮК П.И. Оценка Шора для взвешенного числа устойчивости W_p -совершенных графов // Материалы международной научной конференции "Математическое моделирование, оптимизация и информационные технологии", 19–21 марта 2008, Кишинэу, Академия транспорта, информатики и коммуникаций. – С. 112–121.
91. СТЕЦЮК П.И. О новых свойствах оценок Шора для взвешенного числа устойчивости графа // Праці міжнародної конференції "50 років Інституту кібернетики ім. В.М. Глушкова НАН України". – К.: Інститут кібернетики імені В.М. Глушкова НАН України. – 2008. – С. 164–173.
92. СТЕЦЮК П.И., БУТЕНКО С.И., ЛИХОВИД А.П. ЛП-ориентированная верхняя оценка для числа устойчивости графа на основе p -колес // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М.Глушкова НАН Украины, 2008. – № 7. – С. 34–44.
93. STETSYUK P. On interesting properties of Shor's upper bounds for weighted stability number of a graph // Abstracts of International Conference "Discrete and Global Optimization"(July 31 - August 2, 2008, Yalta, Ukraine).
94. ЛЫКНОВЫД О., STETSYUK P. On numerical experiments with two LP-oriented upper bounds for the stability number of a graph // Abstracts of International Conference "Discrete and Global Optimization"(July 31 - August 2, 2008, Yalta, Ukraine).
95. СТЕЦЮК П.И., ПИЛИПОВСКИЙ А.В. Математическая модель оптимальной загрузки мощностей энергосистемы с учетом их маневренности // Праці IV міжнародної школи-семінару "Теорія прийняття рішень". – Ужгород: УжНУ, 2008. – С. 159.

2009 год

96. Стецюк П.И., Лиховид А.П. Об ЛП-ориентированных верхних оценках для взвешенного числа устойчивости графа // Кибернетика и системный анализ. – 2009. – № 1. – С. 157–170.
97. СЕРГИЕНКО И.В., МИХАЛЕВИЧ М.В., СТЕЦЮК П.И., КОШЛАЙ Л.Б. Модели и информационные технологии для поддержки принятия решений при проведении структурно-технологических преобразований // Кибернетика и систем. анализ. – 2009. – № 2. – С. 26–49.
98. Стецюк П.И., Михалевич М.В., Пилиповский А.В. Применение методов негладкой оптимизации для планирования структурно-технологических изменений // Матеріали ХІ міжнародної науково-технічної конференції "Системний аналіз та інформаційні технології", 26-30 травня 2009 р. – НТТУ КПІ, Київ. – С. 137.
99. Методи негладкої оптимізації у спеціальних задачах класифікації / П.І. Стецюк, О.А. Березовський, М.Г. Журбенко, Д.О. Кропотов. – Київ, 2009. – 28 с. – (Препр./НАН України. Ін-т кібернетики ім.В.М.Глушкова; 2009–1).
100. ЖУРБЕНКО М.Г., СТЕЦЮК П.І. Субградієнтні методи змінної метрики для розв'язування яружних задач оптимізації – Київ, 2009. – 27 с. – (Препр. / НАН України. Ін-т кібернетики імені В.М.Глушкова; 2009-3).
101. Стецюк П.И., Лиховид А.П., Пилиповский А.В. О решении одного класса оптимизационных задач нахождения нагрузок энергетических объектов с помощью NEOS-программ // Праці міжнародного симпозіуму "Питання оптимізації обчислень (ПОО-XXXV)". – Київ: Ін-т кібернетики ім. В.М. Глушкова НАН України, 2009. – Т.2. – С. 350–354.
102. Стецюк П.И. Субградиентные методы переменной метрики, использующие шаг Агмона-Мощкина и одноранговый эллипсоидальный оператор // Труды АТИК – 2007-2008. – Кишинэу: Эврика, 2009. – Т. I (XII). – С. 16–25.

103. СТЕЦЮК П.И., ЛИХОВИД А.П., ПИЛИПОВСКИЙ А.В. Задачи оптимизации для выбора электрических нагрузок в энергосистеме // Теорія оптимальних рішень. – Київ: Ін-т кібернетики ім. В.М. Глушкова НАН України, 2009. – № 8. – С. 136–141.
104. BUTENKO S., PARDALOS P.M., SERGIENKO I.V., SHYLO V., STETSYUK P. Estimating the size of Correcting Codes using Extremal Graph Problems // In C. Pearce and E. Hunt, editors, Optimization: Structure and Applications. – Springer, 2009. – P. 227–243.

2010 год

105. СТЕЦЮК П.И., НУРМИНСКИЙ Е.А. Негладкий штраф и субградиентные алгоритмы для решения задачи проекции на политоп // Кибернетика и системный анализ. – 2010. – №1. – С. 59-63.
106. СТЕЦЮК П.И. Оценки Шора в квадратичных экстремальных задачах // Материалы международной научной конференции "Математическое моделирование, оптимизация и информационные технологии", 24–26 марта 2010, Кишинэу, Академия транспорта, информатики и коммуникаций. – С. 19–25.
107. STETSYUK P., NURMINSKI E. On a modification of ellipsoid method // Материалы международной научной конференции "Математическое моделирование, оптимизация и информационные технологии", 24–26 марта 2010, Кишинэу, Академия транспорта, информатики и коммуникаций. – С. 37-42.
108. МИЦА А.В., СТЕЦЮК П.И. Использование г-алгоритма для оптимизации трех оптических многослойных покрытий // Материалы международной научной конференции "Математическое моделирование, оптимизация и информационные технологии", 24–26 марта 2010, Кишинэу, Академия транспорта, информатики и коммуникаций. – С. 100–108.
109. СТЕЦЮК П.И., КОШЛАЙ Л.Б., ПИЛИПОВСКИЙ А.В. О задаче оптимального соотношения между спросом и добавленной стоимостью в моделях Леонтьева // Теорія оптимальних рішень. – 2010. – № 9. – С. 136–143.
110. СТЕЦЮК П.И., КОШЛАЙ Л.Б. Оптимальная нормированная структура спроса и добавленной стоимости в продуктивной

- модели Леонтьева // Кибернетика и систем. анализ. – 2010. – № 5. – С. 51–59.
111. СТЕЦЮК П.И., ЛЯШКО В.И., НУРМИНСКИЙ Е.А. Точная ЛП-оценка для взвешенного числа устойчивости t -совершенных графов // Журнал обчислювальної та прикладної математики. – 2009. – № 3(99). – С. 106–115.
 112. СТЕЦЮК П.И., ЗОЛОТЫХ Н.Ю. Бинарный квадратичный многогранник и его аппроксимации // Журнал обчислювальної та прикладної математики. – 2010. – № 2(101). – С. 76–86.
 113. СТЕЦЮК П.И. Оценки Н.З.Шора в квадратичных экстремальных задачах и их применение в комбинаторной оптимизации // Российская конференция "Дискретная оптимизация и исследование операций": Материалы конференции (Алтай, 27-июня - 3 июля 2010). – Новосибирск: Изд-во Ин-та математики, 2010. – С. 60–64.
 114. БЕРЕЗОВСКИЙ О.А., СТЕЦЮК П.И. Задачи негладкой безусловной оптимизации для линейного и квадратичного классификаторов с максимальным зазором // V-а Міжнародна школа-семінар "Теорія прийняття рішень", Ужгород, 27 вересня – 1 жовтня 2010. Праці школи-семінару. – С. 15–16.
 115. ЗОЛОТЫХ Н.Ю., СТЕЦЮК П.И. О внешней аппроксимации бинарного и булевого квадратичных многогранников // V-а Міжнародна школа-семінар "Теорія прийняття рішень", Ужгород, 27 вересня – 1 жовтня 2010. Праці школи-семінару. – С. 103–104.
 116. KOSHLAI L., STETSYUK P. Application of Nonlinear Optimization to the Problems of Structural Changes during Transition // 8th EUROPT Workshop on Advances in Continuous Optimization, Aveiro, Portugal, July 9-10, 2010. – P. 28.
 117. STETSYUK P., KOSHLAI L., PYLYPOVSKYI O. Decision Support System for Planning Structural and Technological Changes // 8th EUROPT Workshop on Advances in Continuous Optimization, Aveiro, Portugal, July 9-10, 2010. – P. 29.
 118. П.И.СТЕЦЮК, Л.Б.КОШЛАЙ Об одной экстремальной задаче для связи прямой и двойственной моделей Леонтьева // Крымская осенняя математическая школа симпозиум (КРОМШ), 18-29 сентября 2010 года. – Севастополь: КНЦ НАНУ, 2010. – С. 56.

119. СТЕЦЮК П.І. Алгоритми недиференційовної оптимізації та їх застосування // В монографії „Стан та перспективи розвитку інформатики в Україні“ – Київ: Наукова думка, 2010. – С.183–208.

2011 год

120. СТЕЦЮК П.І. Метод `amsq2p` для овражных выпуклых функций // Информационный бюллетень Ассоциации математического программирования. № 12. Научное издание. – Екатеринбург: УрО РАН, 2011. – С. 57–58.
121. СТЕЦЮК П.І. Релаксационный субградиентный метод минимизации овражных выпуклых функций // Проблемы теоретической кибернетики. Материалы XVI Международной конференции (Нижний Новгород, 20–25 июня 2011 г.) / Под ред. Ю.И.Журавлева. – Нижний Новгород: Изд-во Нижегородского госуниверситета, 2011. – С. 449–453.
122. СТЕЦЮК П.І. Субградиентные методы с преобразованием пространства для минимизации овражных выпуклых функций // Международная конференция „Современные проблемы прикладной математики и механики: теория, эксперимент и практика“, посвященная 90-летию со дня рождения академика Н.Н. Яненко. Тезисы докладов. – Новосибирск, Академгородок, 2011. – С. 57.
123. СТЕЦЮК П.І. Субградиентные методы с преобразованием пространства для минимизации овражных выпуклых функций // Современные проблемы прикладной математики и механики: теория, эксперимент и практика [Электронный ресурс] / Международная конференция, посвященная 90-летию со дня рождения академика Н.Н. Яненко, Новосибирск, Россия, 30 мая – 4 июня 2011 г., Новосибирск, ИВТ СО РАН, 2011, № гос. регистрации – 0321101160, Режим доступа: <http://conf.nsc.ru/files/conferences/niknik-90/fulltext/37828/53979/Stetsyuk.pdf>, свободный. – Загл. с экрана (дата обращения: 07.11.2011).
124. Віділ А.Ю., Пилиповський О.В., СТЕЦЮК П.І. Паралельний алгоритм мултистарту на основі субградієнтного методу для розв'язування системи нелінійних рівнянь // Праці міжнародної молодіжної математичної школи "Питання оптимізації обчислень (ПОО-XXXVII)". – К.: Ін-т кібернетики ім. В.М.Глушкова НАН України, 2011. – С. 28–29.

125. СТЕЦЮК П.И., БОНДАРЕНКО А.В. О спектральных свойствах модели Леонтьева // Теорія оптимальних рішень. – 2011. – № 10. – С. 84–90.
126. БОНДАРЕНКО А.В., КОШЛАЙ Л.Б., СТЕЦЮК П.И. Об одной задаче нахождения оптимальной структуры спроса и добавленной стоимости // Математичні методи, моделі та інформаційні технології в економіці. – Матеріали II Міжнародної науково-методичної конференції. – Чернівці: ДрукАрт, 2011. – С. 26–27.
127. СТЕЦЮК П.И., КОШЛАЙ Л.Б. Об одной экстремальной задаче для связи прямой и двойственной моделей Леонтьева // Спектральные и эволюционные задачи. – 2011. – Т. 2, № 2. – С. 164–169.
128. П.И.СТЕЦЮК, Л.Б.КОШЛАЙ Об одной оптимизационной задаче для моделей Леонтьева // XXII ежегодная междунар. конферен. (KROMSH- 2011), (Крым, Севастополь, 18-29 сентября 2011), КНЦ НАНУ, 2011. – С. 51.
129. KOSHLAI L., STETSYUK P. Application of Nonlinear Optimization to the Problems of Economic Reforms during Transition // 82nd Annual Scientific Conference of the International Assosiation of Applied Mathematics and Mechanics, Graz University of Technology, Graz, Austria (April 18–21, 2011). – P. 350.
130. СТЕЦЮК П. И. О спектральных свойствах матриц Леонтьева // Статистика. Моделирование. Оптимизация: сборник трудов Всероссийской конференции (Челябинск, 28 ноября – 3 декабря 2011 г.). – Челябинск: Издательский центр ЮУрГУ, 2011. – С. 173–178.
131. NURMINSKI E.A., STETSYUK P.I. Nonsmooth optimization and projection problem // II International Conference "Optimization and Applications (OPTIMA-2011), Abstracts. – Petrovac, Montenegro, September 25 - October 2, 2011. – М.: Вычислительный центр им. А.А. Дородницына РАН, 2011. – С. 167–168.
132. Математичні методи оптимізації та інтелектуальні комп'ютерні технології моделювання складних процесів і систем з урахуванням просторових форм об'єктів/ Грицик В.В., Шевченко А.І., Кісельова О.М., Яковлев С.В., Бідюк П.І., Гіль М.І., Крак Ю.В., Романова Т.Є., Куляс А.І., Стецюк П.І. – Донецьк: ІПП "Наука і освіта", 2011. – 650 с.

133. ГОРБАЧУК В.М., ГАРКУША Н.І., СТЕЦЮК П.І., ЄГОРОВ Ю. До розрахунку транзитних обсягів і ставок // PDMU-2011 (Алупка). – К.: КНУ ім. Т. Шевченка, 2011. – С. 61–62.

2012 год

134. СЕРГИЕНКО И.В., СТЕЦЮК П.И. О трех научных идеях Н.З.Шора // Кибернетика и системный анализ. – 2012. – № 1. – С. 4–22.
135. СТЕЦЮК П.И. О важных датах НДО и ЛП в 2012 году // Материалы международной конференции „Математика, экономика, менеджмент: 100 лет со дня рождения Л. В. Канторовича". – СПб.: ООО „ТАИС“, 2012. – С. 67–69.
136. СЕРГИЕНКО И.В., СТЕЦЮК П.И. Три центральные идеи Н.З.Шора // Математическое моделирование, оптимизация и информационные технологии: материалы 3-й Междунар. науч. конф. (Кишинэу, 19-23 марта 2012 г.). – Кишинэу: Эврика, 2012. – С. 468–475.
137. СТЕЦЮК П.И. Ускоренные по Шору модификации субградиентного метода Поляка // Математическое моделирование, оптимизация и информационные технологии: материалы 3-й Междунар. науч. конф. (Кишинэу, 19-23 марта 2012 г.). – Кишинэу: Эврика, 2012. – С. 509–519.
138. СТЕЦЮК П.И., БАРДАДЫМ Т.А., БЕРЕЗОВСКИЙ О.А., ЖУРБЕНКО Н.Г., ЧУМАКОВ Б.М., ШОР Е.И. К 75-летию со дня рождения выдающегося ученого – академика НАН Украины Наума Зуселевича Шора // Кибертония. – 2012. – № 1. – С. 44–45.
139. СТЕЦЮК П.И. Ускорение субградиентного метода Поляка // Теорія оптимальних рішень. – 2012. – № 11. – С. 151–160.
140. СТЕЦЮК П.И. О решении системы линейных уравнений с двусторонними ограничениями на переменные // Алгебра и линейная оптимизация. Тезисы международной конференции, посвященной 100-летию С.Н.Черникова. Екатеринбург, 14–19 мая 2012 года. – Екатеринбург: Изд-во „УМЦ-УПИ“, 2012. – С. 155–157.
141. СТЕЦЮК П.І., ЖУРБЕНКО М.Г., ЛИХОВИД О.П. Математичні моделі та програмне забезпечення в задачах енергетики. – К.: ПП "Ательє "Поліграфічний комплекс", 2012. – 64 с.

142. СТЕЦЮК П.И. Shor's acceleration for Polyak's subgradient method (in Russian)³ // Конструктивный негладкий анализ и смежные вопросы. Тезисы докладов международной конференции. – СПб.: Изд-во Санкт-Петербургского университета, 2012. – Р. 157–160.
143. STETSYUK P. Polyak's accelerated subgradient method // 10th EUROPT Workshop on Advances in Continuous Optimization, 5–7 Yuly 2012, Siauliai, Lithuania. – P. 53.
144. PYLYPOVSKYI O., KOSHLAI L., STETSYUK P. Model-driven DSS for economic reforms during transition // 10th EUROPT Workshop on Advances in Continuous Optimization, 5–7 Yuly 2012, Siauliai, Lithuania. – P. 43.
145. STETSYUK P. On one nonlinear optimization problem for Leontief models // 25th European Conference on Operational Reserch, 8–11 July 2012. Vilnius, Lithuania. – P. 141.
146. PYLYPOVSKYI O., KOSHLAI L., STETSYUK P. DSS for technological planning // 25th European Conference on Operational Reserch, 8–11 July 2012. Vilnius, Lithuania. – P. 98.
147. СТЕЦЮК П.И. Ускоренные модификации субградиентного метода Поляка для овражных выпуклых функций // В книге „Стохастическое программирование и его приложения“ / П.С. Кнопов, В.И. Зоркальцев, Я.М. Иваньо и др. – Иркутск: Институт систем энергетики им. Л.А. Мелентьева СО РАН, 2012. – С. 160–184.
148. СТЕЦЮК П.И. Оптимальные нормированные векторы конечного продукта и добавленной стоимости в продуктовой модели Леонтьева // В книге „Стохастическое программирование и его приложения“ / П.С. Кнопов, В.И. Зоркальцев, Я.М. Иваньо и др. – Иркутск: Институт систем энергетики им. Л.А. Мелентьева СО РАН, 2012. – С. 483–492.
149. СТЕЦЮК П.И., КОШЛАЙ Л.Б. Ускоренный субградиентный метод Поляка // Моделирование, управление и устойчивость (MCS-2012): межд. конф.; Севастополь, 10–14 сентября 2012 г. / отв. ред. О.В. Анашкин; Таврический нац. ун-т имени В.И. Вернадского. – Симферополь: ДИАЙПИ, 2012. – С. 186–187.

³ Ускоренный по Шору субградиентный метод Поляка

150. СТЕЦЮК П.И., КОШЛАЙ Л.Б. Об одном субградиентном методе с преобразованием пространства // Крымская Осенняя Математическая Школа (КРОМШ-2012). XXIII ежегодная международная конференция. Тезисы докладов. – Симферополь: издательство КНЦ НАНУ, 2012. – С. 65–66.
151. Лиховид А.П., Ляшко В.И., СТЕЦЮК П.И. Задача линейного программирования с булевыми переменными для оптимальной суточной загрузки энергосистемы // VI-а Міжнародна школа-семінар "Теорія прийняття рішень", Ужгород, 1 – 6 жовтня 2012. Праці школи-семінару. – С. 129–130.
152. СТЕЦЮК П.И., ИВЛИЧЕВ А.В. Тестовые эксперименты с r -алгоритмом для мегабайтных систем линейных уравнений с двухсторонними границами на переменные // VI-а Міжнародна школа-семінар "Теорія прийняття рішень", Ужгород, 1 – 6 жовтня 2012. Праці школи-семінару. – С. 186–187.
153. ПАНКРАТОВ А.В., РОМАНОВА Т.Е., СТЕЦЮК П.И. Задача оптимальной кластеризации 2D-объектов в круговом контейнере // Матеріали Х міжнародної науково-практичної конференції "Математичне та програмне забезпечення інтелектуальних систем (MPZIS-2012)", Дніпропетровський національний університет, 21-23 листопада 2012р. – С. 225–226.

2013 год

154. STETSYUK P., ROMANOVA T., PANKRATOV A., KOVALENKO A. Optimal Packing Circular Cylinders into a Cylindrical Container Taking into Account Behavior Constraints // Proc. 10th ESICUP Meeting. – Lille (France). – 2013. – P. 21–22.
155. СТЕЦЮК П.И. Алгоритми недиференційовної оптимізації та лагранжеві двоїсті оцінки в складних екстремальних задачах. – Автореф. дис. докт. фіз.-мат. наук., Ін-т кібернетики ім. В.М.Глушкова НАН України. – Київ, 2013. – 32 с.
156. КОВАЛЕНКО А.А., ПАНКРАТОВ А.В., РОМАНОВА Т.Е., СТЕЦЮК П.И. Упаковка круговых цилиндров в цилиндрический контейнер с учетом специальных ограничений поведения системы // Журнал обчислювальної та прикладної математики. – 2013. – № 1 (111). – С. 126–134.

157. НЕНАХОВ Э.И., РОМАНОВА Т.Е., СТЕЦЮК П.И. Равновесная упаковка кругов в круг минимального радиуса // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 2013. – С. 143–153.
158. СТОЯН Ю.Г., ПАНКРАТОВ А.В., РОМАНОВА Т.Е., СТЕЦЮК П.И. Методи розв'язання задач кластеризації 2D-об'єктів у крузі мінімального радіуса // Доповіді НАН України. – 2013. – № 8. – С. 51–57.
159. ДОЛГИХ В.Н., СТЕЦЮК П.И. О применении методов негладкой оптимизации для исследования эффективности сложных экономических систем // Фінансова система України: проблеми та перспективи розвитку в умовах трансформації соціально-економічних відносин: збірник тез доповідей Міжнародної науково-практичної конференції (16-18 травня 2013 р., м. Севастополь) / Севастопольський інститут банківської справи Української академії банківської справи Національного банку України ; Таврійський національний університет імені В.І. Вернадського. – Сімферополь : ТНУ ім. В.І. Вернадського, 2013. – С. 104–105.
160. СТЕЦЮК П.И., РОМАНОВА Т.Е. Равновесная упаковка шаров в шар минимального радиуса // Материалы международной конференции "Дискретная оптимизация и исследование операций" (Новосибирск, 24-28 июня 2013 г.). – Новосибирск: Издательство Института математики, 2013. – С. 142.
161. СТЕЦЮК П.И. Ускоренные по Шору модификации метода Поляка для овражных функций // Всероссийская молодежная школа-семинар "Дискретные модели и методы принятия решений": Материалы школы-семинара (г. Новосибирск, 21-23 июня 2013). – Новосибирск: Изд-во Ин-та математики, 2013. – С. 142–153.
162. СТЕЦЮК П.И., ПЕТРУХИН В.А., БУГРОВ Н.В., ХРИПКО К.Ю. Метод эллипсоидов и условно-оптимальный маршрут // Міжнародна наукова конференція "Сучасна інформатика: проблеми, досягнення та перспективи розвитку", присвяченої 90-річчю від дня народження академіка В.М. Глушкова (12-13 вересня 2013 року). – Київ: Інститут кибернетики імені В.М. Глушкова НАН України, 2013. – С. 111-113.
163. КОВАЛЕНКО А.А., ПАНКРАТОВ А.В., РОМАНОВА Т.Е., СТЕЦЮК П.И. Упаковка круговых цилиндров в цилиндрический контейнер

- с учетом специальных ограничений // Праці міжнародної наукової конференції "Питання оптимізації обчислень (ПОО-XL)", присвяченої 90-річчю від дня народження академіка В.М. Глушкова. – Київ: Інститут кібернетики імені В.М. Глушкова НАН України, 2013. – С. 123.
164. SERGIENKO I., SHYLO V., STETSYUK P., STOYAN YU., ROMANOVA T. Computer technologies for placement and covering problems // Праці міжнародної наукової конференції "Питання оптимізації обчислень (ПОО-XL)", присвяченої 90-річчю від дня народження академіка В.М. Глушкова. – Київ: Інститут кібернетики імені В.М. Глушкова НАН України, 2013. – С. 240.
 165. СТЕЦЮК П.И. Квадратичная задача для максимального сингулярного числа // Праці міжнародної наукової конференції "Питання оптимізації обчислень (ПОО-XL)", присвяченої 90-річчю від дня народження академіка В.М. Глушкова. – Київ: Інститут кібернетики імені В.М. Глушкова НАН України, 2013. – С. 255.
 166. СТЕЦЮК П.И. r-Алгоритмы: теория и практика // Матеріали ХІ Міжнародної науково-практичної конференції "Математичне та програмне забезпечення інтелектуальних систем (MPZIS-2013)" (20-22 листопада 2013 р., м. Дніпропетровськ). – С. 227–232.
 167. СТЕЦЮК П.И., НУРМИНСКИЙ Е.А., СОЛОМОН Д.И. Транспортная задача и ортогональное проектирование на линейные многообразия // Материалы V-ой международной научной конференции "Транспортные системы и логистика", Кишинэу, 11-13 декабря 2013 года. – Кишинэу: Эврика, 2013. – С. 251–263.
 168. СТЕЦЮК П.И., ЖИДКОВ В.А. О двух задачах оптимизации пропускных способностей дуг отказоустойчивой сети // Материалы V-ой международной научной конференции "Транспортные системы и логистика", Кишинэу, 11-13 декабря 2013 года. – Кишинэу: Эврика, 2013. – С. 300–309.

СОДЕРЖАНИЕ

Предисловие	6
-------------------	---

Р А З Д Е Л I

Метод эллипсоидов и его обобщения

Метод отсечения с растяжением пространства для решения задач выпуклого программирования (<i>Шор, 1977</i>)	14
Об одном семействе алгоритмов для решения задач выпуклого программирования (<i>Шор, Гершович, 1979</i>)	19
Метод эллипсоидов, его обобщения и приложения	30
(<i>Гершович, Шор, 1982</i>)	
Использование операции растяжения пространства в задачах минимизации выпуклых функций (<i>Шор, 1970</i>)	51

Р А З Д Е Л II

Приближенный метод эллипсоидов

Приближенный метод эллипсоидов (<i>Стецюк, 2003</i>)	66
Об одном эллипсоиде для внешней аппроксимации	
n -мерного полушара (<i>Стецюк, 2003</i>)	74
Модификация метода эллипсоидов (<i>Стецюк, 2003</i>)	83
On a modification of ellipsoid method (<i>Stetsyuk, Nurminski, 2010</i>)	86
Об одном методе нахождения L_p -решения системы линейных уравнений (<i>Стецюк, Колесник, Березовский, 2003</i>)	92
К ускорению метода эллипсоидов с помощью использования шарового слоя (<i>Стецюк, Буханцов, 2002</i>)	100

Р А З Д Е Л III

r -алгоритмы и эллипсоиды

К вопросу сходимости $r_\mu(\alpha)$ -алгоритма (<i>Стецюк, 1995</i>)	110
К обоснованию сходимости алгоритмов с растяжением пространства (<i>Стецюк, 1996</i>)	116
r -алгоритмы и эллипсоиды (<i>Стецюк, 1996</i>)	121
Метод центров тяжести простых тел (<i>Стецюк, 1996</i>)	150
К методам эллипсоидов (<i>Стецюк, 1999</i>)	177
Об ускорении сходимости методов эллипсоидов (<i>Стецюк, 2001</i>)	185

Р А З Д Е Л IV

Ортогонализирующие линейные операторы

Ортогонализирующие линейные операторы в выпуклом программировании I (<i>Стецюк, 1997</i>)	192
Ортогонализирующие линейные операторы в выпуклом программировании II (<i>Стецюк, 1997</i>)	220
Классические фейеровские методы с преобразованием пространства (<i>Стецюк, 1996</i>)	239
Об одном способе ускорения метода ортогонального субградиентного спуска (<i>Стецюк, 1998</i>)	245
Об одном методе для нахождения допустимой точки выпуклого неравенства (<i>Стецюк, 2000</i>)	253
К методам решения плохообусловленных систем линейных уравнений (<i>Стецюк, 2001</i>)	261

Р А З Д Е Л V

Одноранговый эллипсоидальный оператор в фейеровских процессах

Субградиентные методы переменной метрики, использующие..... шаг Агмона–Мощкина и одноранговый эллипсоидальный..... оператор (<i>Стецюк, 2009</i>).....	272
Метод <code>amsg2p</code> для овражных выпуклых функций (<i>Стецюк, 2011</i>)..	282
Релаксационный субградиентный метод минимизации овражных..... выпуклых функций (<i>Стецюк, 2011</i>).....	285
Ускоренные по Шору модификации субградиентного метода..... Поляка (<i>Стецюк, 2012</i>).....	290
Ускорение субградиентного метода Поляка (<i>Стецюк, 2012</i>)	299
Shor’s acceleration for Polyak’s subgradient method (in Russian) ⁴ (<i>Stetsyuk, 2012</i>).....	309
Ускоренные модификации субградиентного метода Поляка..... для овражных выпуклых функций (<i>Стецюк, 2012</i>).....	313
Ускоренный субградиентный метод Поляка..... (<i>Стецюк, Кошлай, 2012</i>).....	337
Ускоренные по Шору модификации метода Поляка для овражных.... функций (<i>Стецюк, 2013</i>).....	340

Р А З Д Е Л VI

r -Алгоритмы и квазиньютоновские методы

Квазиньютоновские методы и r -алгоритмы (<i>Стецюк, 1996</i>)	352
Линейные операторы в квазиньютоновских методах (<i>Стецюк, 1998</i>)..	375
Субградиентные методы с преобразованием пространства для..... минимизации овражных выпуклых функций (<i>Стецюк, 2011</i>).....	382
r -Алгоритмы: теория и практика (<i>Стецюк, 2013</i>)	391

⁴Ускоренный по Шору субградиентный метод Поляка

Р А З Д Е Л VII

Юбилейная статья

Об одной схеме отсечений (<i>Стецюк, 1995</i>)	398
О трех научных идеях Н.З.Шора (<i>Сергиенко, Стецюк, 2012</i>).....	433
Список научных трудов (<i>Стецюк, 1982–2013</i>).....	462
Содержание	484

НАЦИОНАЛЬНАЯ АКАДЕМИЯ НАУК УКРАИНЫ
ИНСТИТУТ КИБЕРНЕТИКИ ИМ. В. М. ГЛУШКОВА

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ
РЕСПУБЛИКИ МОЛДОВА

АКАДЕМИЯ ТРАНСПОРТА,
ИНФОРМАТИКИ И КОММУНИКАЦИЙ

П. И. СТЕЦЮК

МЕТОДЫ ЭЛЛИПСОИДОВ
И r -АЛГОРИТМЫ

Издательство “Эврика”, тел. 0 22 384530
мун. Кишинэу, Республика Молдова
Мунчешское шоссе, 121а

Подписано в печать 10.03.14
Формат 60х90/16
Усл. печ. листов 30,5
Тираж 300
Заказ № 3

Типография Академии наук Молдовы
г. Кишинев, ул. П. Мовилэ, 8