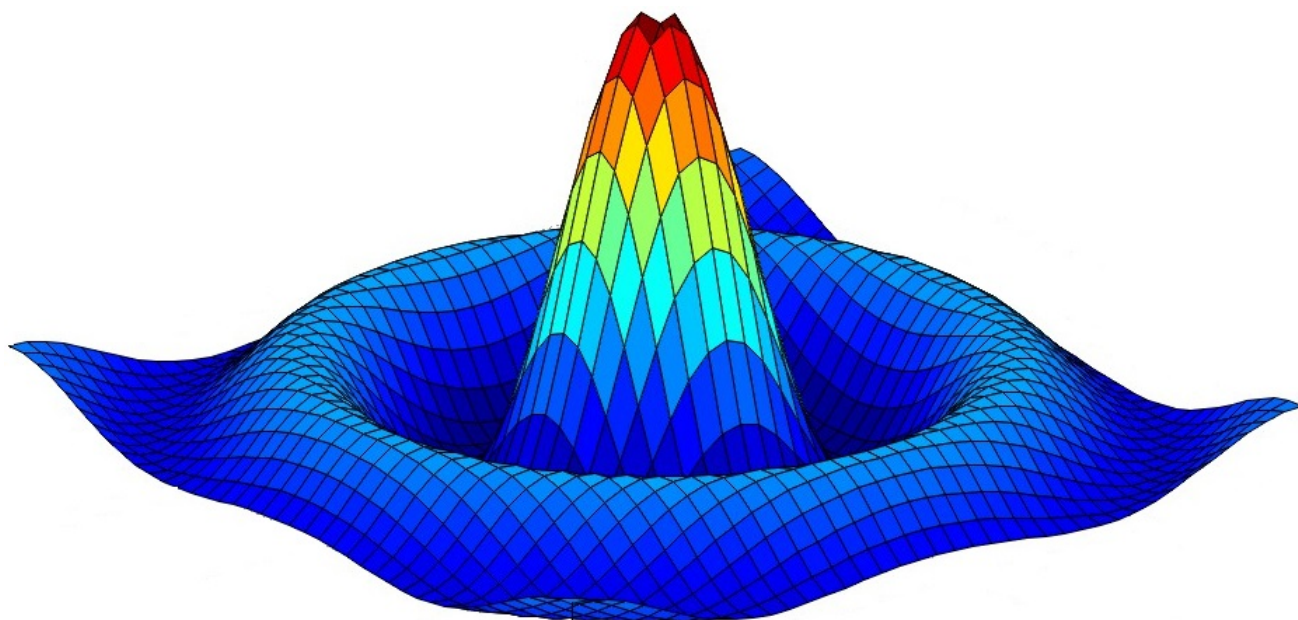


ОПТИМИЗАЦИЯ ПСЕВДООБРАЩЕНИЕ ИТЕРАЦИИ И РЕКУРСИИ

УЧЕБНОЕ ПОСОБИЕ



ЛИПЕЦК

ЛИПЕЦКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

2015

Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«Липецкий государственный технический университет»

**ОПТИМИЗАЦИЯ
ПСЕВДООБРАЩЕНИЕ
ИТЕРАЦИИ И РЕКУРСИИ**

Учебное пособие

Липецк
Липецкий государственный технический университет
2015

УДК 519.863

П433

Рецензенты:

кафедра «Информационные системы и защита информации»

Тамбовского государственного технического университета

(зав. кафедрой д-р техн. наук, проф. Громов Ю.Ю.);

Рубан А.И., д-р техн. наук, проф., заслуженный деятель науки РФ,
зав. кафедрой информатики Института космических и информационных технологий
Сибирского федерального университета

Погодаев, А.К.

П433 Оптимизация. Псевдообращение. Итерации и рекурсии [Текст]: учебное пособие / А.К. Погодаев, С.Л. Блюмин, С.П. Миловидов, А.С. Сысоев. – Липецк: Изд-во Липецкого государственного технического университета, 2015. – 196 с.

ISBN 978-5-88247-741-6

В пособии систематически описаны элементы теории математического программирования, определение, примеры, свойства и алгоритмы псевдообращения, а также постановка и решение нелинейной задачи о наименьших квадратах, что приводит к рекуррентно-итерационным алгоритмам. Предназначено для студентов направлений, получающих углублённую математическую подготовку, и связано с решением широкого круга задач. Включённый в пособие материал будет полезен также инженерам, аспирантам, научным работникам, применяющим в расчётах математические методы; для них пособие может служить и в качестве справочника. В доступной для начинающих форме изложены важнейшие, наиболее часто используемые определения, свойства и примеры задач оптимизации, удобные в вычислительном отношении, пригодные для непосредственной реализации алгоритмы.

Рекомендовано УМС ЛГТУ в качестве учебного пособия для бакалавров, обучающихся по направлению подготовки ВПО 01.03.04 «Прикладная математика», магистров по направлениям 01.04.04 «Прикладная математика», 09.04.01 «Информатика и вычислительная техника».

УДК 519.863

П433

ISBN 978-5-88247-741-6

© ФГБОУ ВПО «Липецкий государственный технический университет», 2015

© Погодаев А.К., Блюмин С.Л., Миловидов С.П.,
Сысоев А.С., 2015

Оглавление

Введение	4
Глава I. Основы оптимизации	7
§ 1. Задачи математического программирования	7
§ 2. Безусловная оптимизация.....	14
§ 3. Классическая задача математического программирования.....	19
§ 4. Нелинейное программирование	30
§ 5. Пример численного решения задачи многомерной нелинейной оптимизации.....	49
Глава II. Алгоритмы псевдообращения	65
§ 6. Определение псевдообратной матрицы.....	70
§ 7. Свойства псевдообратных матриц.....	74
§ 8. Рекуррентные алгоритмы псевдообращения	79
§ 9. Примеры нахождения псевдообратных матриц	81
§ 10. Системы линейных уравнений и метод наименьших квадратов ..	89
Глава III. Итерации и рекурсии	100
§ 11. Общая классификация алгоритмов оптимизации	113
§ 12. Нелинейная оптимизация: итерационные процедуры	123
§ 13. Линейный метод наименьших квадратов: рекуррентные процедуры	132
§ 14. Нелинейный метод наименьших квадратов: рекуррентно-итерационные процедуры	143
Задания и упражнения	159
Заключение	178
Библиографический список	180

Введение

Многие практические задачи различных сфер человеческой деятельности связаны с проблемами определения наилучшего, оптимального варианта решения. Таковы, например, задачи выбора оптимальной производственной программы предприятия, задача оптимального распределения ресурсов, транспортные задачи и так далее. Для решения задач такого рода в математической науке созданы и стремительно развиваются оптимизационные процедуры. Существует огромное количество работ, посвящённых различным сторонам математического программирования и теории оптимизации, прикладным аспектам, методам численного решения. Изучение методов оптимизации становится необходимым для практической работы специалиста любой области – будь то экономика или техника.

Название пособия как бы разделяет его на три основные части: введение в понятия оптимизации, понятие и алгоритмы псевдообращения матриц и рекуррентно-итерационные процедуры решения задач о наименьших квадратах.

В первой главе отражается стремление дать читателю развернутое, чёткое и логически связанное представление об основных понятиях теории оптимизации. Она начинается с постановки задачи оптимизации и подробно рассматривает решение её в линейном виде (задача линейного программирования). Излагаются известные классические методы отыскания экстремумов дифференцируемых функций. Переход к задаче нелинейного программирования (нелинейной оптимизации) показывает, что линейная задача является частным случаем нелинейной. Основные теоремы доказываются с помощью теоремы Куна-Таккера. Усвоение материала облегчается последовательным применением множителей Лагранжа и их содержательной интерпретацией.

Теория оптимизации, рассмотренная в начальных параграфах, является математической базой для постановки ряда практических задач. Ознакомительно представлено также такое направление современной оптимизации, как численное программирование (численная оптимизация). Решение одной из практических задач – поиска оптимального распределения времени зеленого сигнала по фазам светофорного регулирования – показывает специфику задач (и, в первую очередь, целевых функций), при которых такие механизмы решения могут быть применены.

Вторая глава полностью посвящена обобщению понятия обращения матриц – псевдообращению. Аппарат псевдообращения широко применяется при точном или приближённом исследовании и решении систем линейных алгебраических уравнений и приводящихся к ним разнообразных прикладных задач из широкого спектра областей, и, в первую очередь, математического программирования и задач теории оптимизации. Подробно описаны свойства псевдообратных матриц и большое внимание уделено практическому вопросу – алгоритмам нахождения такого рода матриц. Первый рекуррентный алгоритм псевдообращения, предложенный Гревилем, обладает преимуществом по сравнению с другими как вычислительный метод, наиболее подходящий для компьютерной реализации; он не требует обращения и факторизации матриц и других сложных матричных операций. На базе известного метода Фаддеева одновременного вычисления скалярных коэффициентов характеристического многочлена и матричных коэффициентов присоединённой характеристической матрицы может быть построен другой рекуррентный алгоритм псевдообращения. Эти два алгоритма приняты как основные для рассмотрения в пособии. Показана возможность применения современных прикладных математических программ для облегчения нахождения псевдообратных матриц.

В третьей главе представлены рекуррентно-итерационные процедуры решения задач о наименьших квадратах. Одной из практических задач оптимизации является задача о наименьших квадратах, существующая (в зависимости от вида целевой функции) в двух формах – линейной и нелиней-

ной. Нелинейный метод наименьших квадратов служит для решения нелинейной задачи о наименьших квадратах, которая относится к классу задач нелинейной оптимизации или нелинейного программирования, однако характеризуются в этом классе спецификой, оправдывающей её самостоятельное исследование. Классификации и исследованию этих задач и их разновидностей, разработке методов и алгоритмов их решения посвящено большое число работ, которые по различным причинам не всегда доступны студенту или инженеру и не всегда пригодны для первого знакомства с предметом. В главе изложены сведения по этим вопросам в виде не только удобном для последующего использования, но и позволяющем читателю, впервые знакомящемуся с нелинейным программированием, сориентироваться в этой области.

Учебное пособие является продолжением исследований авторов, заложенных ещё в работах [15, 17, 22, 23, 78, 132] и получивших развитие в [18, 20, 21, 24, 26, 28, 90, 91, 130, 131, 133, 134]. При подготовке пособия за основу взят материал, многие годы используемый при чтении лекций в Липецком государственном техническом университете, находивший уже отражение и в более ранних работах [16, 19, 25, 27, 88, 89, 92–96, 144].

Предназначено для студентов направлений, получающих углублённую математическую подготовку, и связано с решением широкого круга задач. Включённый в пособие материал будет полезен также инженерам, аспирантам, научным работникам, применяющим в расчётах математические методы; для них пособие может служить и в качестве справочника. В доступной для начинающих форме изложены важнейшие, наиболее часто используемые определения, свойства и примеры задач оптимизации, удобные в вычислительном отношении, пригодные для непосредственной реализации алгоритмы. Доказательства и обоснования элементарны, опираются на стандартный вузовский курс линейной алгебры и математического анализа.

ГЛАВА I

Основы оптимизации

Математические основы оптимизации изложены в большом числе отечественных и зарубежных монографий, учебников и пособий. Однако далеко не все из них легко доступны различным категориям читателей. Это является одной из причин включения в пособие материала данной главы. Соглашаясь с автором [81], мы считаем, что оптимизации следует обучать столь же основательно, как и другим разделам математики. За основу изложения в этой главе принят материал [62], сочетающий в себе достаточную полноту, общность и доступность.

§ 1. Задачи математического программирования

Статическая задача рациональной деятельности связана с распределением ограниченных ресурсов на различные цели в определённый момент времени. В математической форме задача состоит в нахождении значений переменных, максимизирующих заданную функцию и удовлетворяющих системе ограничений. В такой форме задача статической оптимизации часто называется *задачей математического программирования* [62].

Определённость матриц

Скалярная функция n переменных называется *квадратичной формой* [105], если $\varphi(x) = x^T Q x$.

Матрица Q является *положительно определённой* тогда и только тогда, когда значения квадратичной формы $x^T Q x > 0$ для всех $x \neq 0$.

Рассмотрим пример.

$$\begin{aligned} Q &= \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}, \quad \varphi(x) = (x_1, x_2) \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \\ &= (x_1, x_2) \begin{pmatrix} 2x_1 - x_2 \\ -x_1 + 2x_2 \end{pmatrix} = x_1(2x_1 - x_2) + x_2(-x_1 + 2x_2) = \\ &= 2x_1^2 - 2x_1x_2 + 2x_2^2 > 0, \quad \forall x_{1,2} \neq 0. \end{aligned}$$

Матрица Q является *положительно полуопределённой*, если значения квадратичной формы $x^T Q x > 0 \quad \forall x$ и существует вектор $x \neq 0$, такой, что $x^T Q x = 0$.

Рассмотрим пример.

$$\begin{aligned} Q &= \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}, \quad \varphi(x) = (x_1, x_2) \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \\ &= x_1(x_1 - x_2) + x_2(-x_1 + x_2) = x_1^2 - 2x_1x_2 + x_2^2 = (x_1 - x_2)^2. \end{aligned}$$

Матрица Q является *отрицательно определённой*, если $-Q$ есть положительно определённая матрица.

Матрица Q является *отрицательно полуопределённой*, если $-Q$ есть положительно полуопределённая матрица.

Матрица Q является *неопределённой*, если квадратичная форма $x^T Q x$ может принимать как положительные, так и отрицательные значения.

Рассмотрим пример.

$$\begin{aligned} Q &= \begin{pmatrix} 1 & -1 \\ 1 & -2 \end{pmatrix}, \quad \varphi(x) = (x_1, x_2) \begin{pmatrix} 1 & -1 \\ 1 & -2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \\ &= (x_1, x_2) \begin{pmatrix} x_1 - x_2 \\ x_1 - 2x_2 \end{pmatrix} = x_1(x_1 - x_2) + x_2(x_1 - 2x_2) = \\ &= x_1^2 - x_1x_2 + 2x_2^2 = x_1^2 - 2x_2^2. \end{aligned}$$

Для проверки определённости (полуопределённости) матрицы служат *критерии Сильвестра*:

1. Матрица положительно определена, если:

- а) все диагональные элементы положительны;
- б) все угловые миноры матрицы положительны.

2. Матрица отрицательно определена, если:
- а) все диагональные элементы отрицательны;
 - б) все угловые миноры матрицы имеют чередующиеся знаки, начиная со знака «-».
3. Матрица положительно полуопределена, если значения диагональных элементов и главных миноров матрицы неотрицательны.

Постановка задач оптимизации

При формальной постановке задачи математического программирования основными являются понятия «инструментальных» переменных, допустимого множества и целевой функции. Задача заключается в нахождении значений n переменных x_1, \dots, x_n , которые называются здесь «инструментами». Записанные в виде вектор-столбца

$$x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = (x_1, \dots, x_n)',$$

они составляют вектор инструментальных переменных в n -мерном евклидовом пространстве \mathbb{E}^n .

Если вектор инструментальных переменных x удовлетворяет ограничениям задачи, он называется *допустимым*, а множество всех допустимых векторов образует *множество возможностей* X . Допустимое множество является подмножеством \mathbb{E}^n . Так как задача заключается в выборе вектора инструментальных переменных из допустимого множества, то в любой нетривиальной задаче оно является непустым (то есть, система ограничений совместна) и содержит по крайней мере две различные точки.

Целевая функция – это краткое математическое изложение цели данной задачи. Обычно она представляет собой действительную непрерывно дифференцируемую функцию вектора инструментальных переменных:

$$F = F(x) = F(x_1, \dots, x_n).$$

Общая задача математического программирования состоит в выборе вектора инструментальных переменных из множества возможностей, максимизирующего значение целевой функции

$$\max_x F(x) \quad \text{при условии, что } x \in X,$$

где X – подмножество n -мерного евклидова пространства.

Учитывая, что максимизация функции $F(x)$ эквивалентна максимизации $a + bF(x)$, $b > 0$, можно сделать вывод, что введение дополнительного слагаемого или положительного множителя в целевую функцию не изменяет задачи, в то время как отрицательный множитель может быть использован для преобразования задачи максимизации в задачу минимизации и наоборот (например, с помощью умножения $F(x)$ на -1).

Выделяются два основных вида общей задачи математического программирования: классическая задача математического программирования и задача нелинейного программирования.

В классической задаче математического программирования ограничения представляют собой равенства:

$$\begin{aligned} g_1(x) &= g_1(x_1, \dots, x_n) = b_1, \\ &\vdots \\ g_m(x) &= g_m(x_1, \dots, x_n) = b_m. \end{aligned}$$

Функции $g_1(x), \dots, g_m(x)$ – известные непрерывно дифференцируемые функции, называемые функциями ограничений; параметры b_1, \dots, b_m – заданные действительные числа, называемые константами ограничений. В векторной форме система ограничений записывается в виде $g(x) = b$, где $g(x)$ и b – m -мерные вектор-столбцы:

$$g(x) = \begin{pmatrix} g_1(x_1, \dots, x_n) \\ \vdots \\ g_m(x_1, \dots, x_n) \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix}.$$

Задача классического программирования заключается в максимизации целевой функции при заданных ограничениях:

$$\max_x F(x) \quad \text{при условии, что} \quad g(x) = b.$$

В нелинейном программировании система ограничений состоит из условий двух типов:

– условий неотрицательности

$$x_1 \geq 0, \dots, x_n \geq 0,$$

– ограничений в виде неравенств

$$\begin{aligned} g_1(x) = g_1(x_1, \dots, x_n) &\leq b_1, \\ &\vdots \\ g_m(x) = g_m(x_1, \dots, x_n) &\leq b_m. \end{aligned} \tag{1.1}$$

Ограничения в векторной форме $x \geq 0, \quad g(x) \leq b$.

В приведённой выше записи предполагается, что функции ограничений $g_1(x), \dots, g_m(x)$ непрерывно дифференцируемые; константы ограничений b_1, \dots, b_m , как и раньше, заданные действительные числа; 0 – вектор-столбец, состоящий из нулей.

Задача нелинейного программирования заключается в нахождении неотрицательных значений переменных, удовлетворяющих условиям (1.1) и максимизирующих заданную функцию

$$\max_x F(x) \quad \text{при условии, что} \quad x \geq 0, \quad g(x) \leq b. \tag{1.2}$$

Типы оптимумов, теоремы существования

В общей задаче математического программирования вектор инструментальных переменных x^* является точкой глобального максимума (или решением), если он принадлежит допустимому множеству, и целевая функция принимает на этом векторе значение не меньшее, чем в любой другой допустимой точке: $x^* \in X$ и $F(x^*) \geq F(x) \quad \forall x \in X$.

Будем считать глобальный максимум строгим (сильным), если значение целевой функции при $x = x^*$ строго больше любого другого значения функции на допустимом множестве, то есть $F(x^*) > F(x) \quad \forall x \in X, \quad x \neq x^*$.

Очевидно, что строгий глобальный максимум всегда единственен, так как если мы предположим противное и допустим, что x^* и x^{**} являются различными точками сильного глобального максимума, то из этого будет следовать, что $F(x^*) > F(x^{**})$ и $F(x^{**}) > F(x^*)$, а эти неравенства одновременно выполняться не могут.

Фундаментальная теорема математического программирования – теорема Вейерштрасса – формулирует условия существования глобального максимума.

Теорема (Вейерштрасса). Пусть допустимое множество X является компактным (то есть ограниченным и замкнутым) и непустым. Тогда непрерывная целевая функция $F(x)$, определённая на этом множестве, достигает глобального максимума на внутренней или граничной точке множества X .

Доказательство этого утверждения следует из того факта, что образ непрерывной функции, определённой на компактном множестве, тоже является компактным. Тогда мы имеем множество действительных чисел, удовлетворяющих условию $F(X) = \{z \in \mathbb{E} \mid z = F(x) \quad \forall x \in X\}$, которое является компактным. Всякое компактное множество действительных чисел имеет верхнюю грань. Таким образом, если F^* является верхней гранью $F(X)$, то существует $x^* \in X$, удовлетворяющее условию $F(x^*) = F^*$, а так как $F(x) < F(x^*) \quad \forall \quad x \in X$, то точка x^* является глобальным максимумом.

Примером одномерной задачи, не имеющей решения, может служить задача максимизации функции $F(x) = x^2$ при условии $x \geq 0$. Здесь решения нет, так как целевая функция возрастает с увеличением x , а верхнего предела по переменной x нет (допустимое множество не ограничено). Ещё одна задача того же типа – максимизация функции $F(x) = 10x$ при условии $0 \leq x < 1$. Здесь нет решения, так как целевая функция возрастает с увеличением x , а верхняя грань в точке $x = 1$ не достигается (допустимое

множество не замкнуто). Однако надо отметить, что условия теоремы Вейерштрасса являются достаточными, но не необходимыми. Например, задача максимизации функции $F(x) = x^3$ при условии $0 < x \leq 1$ (или $x \leq 1$) имеет решение в точке $x = 1$, хотя допустимое множество не является компактным.

Вектор инструментальных переменных x^* есть точка *локального максимума*, если он принадлежит допустимому множеству и на нём достигается значение целевой функции большее или равное значениям функции в некоторой малой окрестности этого вектора: $x^* \in X$ и $F(x^*) \geq F(x) \quad \forall x \in X \cap N_\varepsilon(x^*)$, где $N_\varepsilon(x^*)$ есть ε -окрестность вектора x^* , в данном случае множество точек x , удовлетворяющих условию $|x - x^*| = \sqrt{\sum_{j=1}^n (x_j - x_j^*)^2} < \varepsilon$ для сколь угодно малого положительного числа ε .

Локальный максимум в x^* является *строгим*, если значение целевой функции в точке x^* является наибольшим значением, достигаемым целевой функцией в некоторой малой окрестности: $F(x^*) > F(x) \quad \forall x \in X \cap N_\varepsilon(x^*), \quad x \neq x^*$. Очевидно, глобальный максимум является локальным; обратное утверждение не верно, так как могут существовать другие локальные максимумы, на которых целевая функция принимает большие значения.

Второй основной теоремой математического программирования является «локально-глобальная» теорема, дающая достаточные условия, при которых локальный максимум является глобальным.

Теорема (достаточные условия глобального максимума). Пусть допустимое множество не пусто и является компактным и выпуклым, а непрерывная функция $F(x)$ вогнута на X . Тогда:

- 1) локальный максимум является глобальным;
- 2) множество точек, на котором достигается максимум, выпукло.

Если предположить, что функция $F(x)$ строго вогнута, то решение единственно, то есть существует единственный глобальный максимум.

В одномерном случае, если $n = 1$, задаче математического программирования можно дать наглядную геометрическую иллюстрацию. При $n = 2$ мы непосредственно определяем только допустимое множество, откладывая

инструментальные переменные по осям x_1 и x_2 , а целевую функцию $F(x)$ определяем через *линии уровня* и *направление скорейшего роста*.

Поверхность (линия) уровня целевой функции – множество точек евклидова пространства, для которых значения функции одинаковы, то есть

$$\{x \in \mathbb{E}^n \mid F(x) = \text{const}\}, \quad (1.3)$$

причём различные константы порождают различные линии уровня. Изменяя константы в (1.3), получим множество линий уровня, которое называется *картой линий уровня*.

Направление, вдоль которого скорость увеличения целевой функции (в (1.3) – постоянной) максимальна, называется «*предпочтительным*» *направлением*, или *направлением наискорейшего роста*. Оно задаётся вектором, который составлен из первых частных производных целевой функции

$$\frac{\partial F}{\partial x}(x) = \left(\frac{\partial F}{\partial x_1}(x), \dots, \frac{\partial F}{\partial x_n}(x) \right).$$

Вектор частных производных, называемый *вектором-градиентом*, определяет направление скорейшего возрастания функции $F(x)$.

Геометрически в задаче математического программирования мы ищем такую точку или набор точек из допустимого множества, на которой достигается самая верхняя линия уровня, расположенная дальше остальных в направлении скорейшего роста. Этот способ используется при иллюстрации многих задач статической оптимизации в двумерном случае. В задаче нелинейного программирования (1.2) могут существовать два типа решений: как на границе, так и во внутренней точке.

§ 2. Безусловная оптимизация

Одномерные задачи оптимизации при отсутствии ограничений

В случае, когда $m = 0$, а функция зависит от скалярного аргумента ($n = 1$), задача состоит в отыскании вещественного числа x , максимизирующего $F(x)$. Если в такой задаче внутренний локальный максимум дости-

гается в точке x^* , это значит, что для всех точек $x^* + \Delta x$, близких к x^* , выполняется

$$F(x^*) \geq F(x^* + \Delta x), \quad (2.1)$$

где Δx – произвольное малое приращение аргумента.

Пусть функция $F(x)$ имеет непрерывные и конечные производные первого и второго порядков. Тогда в окрестности точки x^* можно разложить функцию, стоящую в правой части неравенства (2.1), по формуле Тейлора (с остаточным членом):

$$F(x^* + \Delta x) = F(x^*) + \frac{dF}{dx}(x^*)\Delta x + \frac{1}{2!} \frac{d^2F}{dx^2}(x^* + \theta\Delta x)\Delta x^2, \quad (2.2)$$

где $0 < \theta < 1$.

Подставив это разложение функции в (2.1), придём к *основному неравенству*:

$$\frac{dF}{dx}(x^*)\Delta x + \frac{1}{2!} \frac{d^2F}{dx^2}(x^* + \theta\Delta x)\Delta x^2 \leq 0,$$

которое должно выполняться для любого произвольно малого приращения Δx . Если Δx больше нуля, то, разделив обе части основного неравенства на Δx и переходя к пределу при Δx стремящемся к 0, получим, что $\frac{dF}{dx}(x^*) \leq 0$.

Но аналогичное рассуждение при отрицательных Δx приводит к неравенству $\frac{dF}{dx}(x^*) \geq 0$.

Таким образом, из основного неравенства получаем в качестве необходимого условия первого порядка, что первая производная в точке локального максимума обращается в нуль:

$$\frac{dF}{dx}(x^*) = 0. \quad (2.3)$$

Если условия первого порядка выполнены, то из основного неравенства следует, что

$$\frac{d^2F}{dx^2}(x^* + \theta\Delta x) \leq 0, \quad (2.4)$$

так как Δx^2 всегда больше нуля. Поскольку (2.4) выполняется для всех Δx , а вторая производная по предположению непрерывна, приходим к необходи-

мому условию второго порядка:

$$\frac{d^2 F}{dx^2}(x^*) \leq 0, \quad (2.5)$$

то есть в точке локального максимума производная второго порядка отрицательна или равна нулю.

Итак, условия (2.3) и (2.5) представляют собой соответственно условия первого и второго порядков, необходимые для существования локального максимума в x^* .

Достаточные условия наличия в x^* строго локального максимума состоят в том, что первая производная в этой точке обращается в нуль, а вторая производная строго меньше нуля. Иначе говоря, если выполнены условия $\frac{dF}{dx}(x^*) = 0$, $\frac{d^2 F}{dx^2}(x^*) < 0$, то x^* есть точка строго локального максимума: $F(x^*) > F(x^* + \Delta x)$.

Достаточность этих условий можно доказать с помощью основного неравенства либо более простым способом, используя теорему о среднем значении. По теореме о среднем значении

$$F(x^* + \Delta x) = F(x^*) + \frac{dF}{dx}(x^* + \theta \Delta x) \Delta x,$$

где $0 < \theta < 1$. Функция $F(x)$ непрерывно дифференцируема, а её первая производная равна нулю и строго убывает при x^* . Следовательно, если $\Delta x > 0$, то $\frac{dF}{dx}(x^* + \theta \Delta x) < 0$, тогда как если $\Delta x < 0$, то $\frac{dF}{dx}(x^* + \theta \Delta x) > 0$. В любом случае $\frac{dF}{dx}(x^* + \theta \Delta x) \Delta x < 0$.

Хотя условия первого (2.3) и второго (2.5) порядков и являются необходимыми, их выполнения недостаточно для существования максимума. Примером этого может служить $F(x) = x^3$ при $x = 0$.

Задачу оптимизации при отсутствии ограничений для функции векторного аргумента $m = 0$, $n > 1$ можно исследовать способом, подобным изложенному выше.

Многомерные задачи оптимизации при отсутствии ограничений

Особый класс задач составляют задачи оптимизации при отсутствии ограничений ($m = 0$). Формулировка задачи: найти $\max_x F(x) = F(x_1, \dots, x_n)$.

Допустим, что локальный максимум существует в x^* :

$$F(x^*) \geq F(x^* + h\Delta x). \quad (2.6)$$

Это означает, что $F(x_1^*, \dots, x_n^*) \geq F(x_1^* + h\Delta x_1, \dots, x_n^* + h\Delta x_n)$, где h – произвольное малое положительное число, Δx_j – произвольное изменение x_j ($j = 1, \dots, n$), а $\Delta x = (\Delta x_1, \dots, \Delta x_n)'$ определяет некоторое направление в \mathbb{E}^n . Функцию в правой части (2.6) можно рассматривать как функцию h . Разложение по формуле Тейлора в окрестности точки $h = 0$ даёт

$$F(x^* + h\Delta x) = F(x^*) + h \frac{\partial F}{\partial x}(x^*) \Delta x + \frac{1}{2!} h^2 (\Delta x)' \frac{\partial^2 F}{\partial x^2}(x^* + \theta h \Delta x) (\Delta x), \quad (2.7)$$

где $0 < \theta < 1$. Вектор $\frac{\partial F}{\partial x}$ в (2.7) есть градиент функции, а матрица $\frac{\partial^2 F}{\partial x^2}$ – матрица Гессе:

$$\frac{\partial F}{\partial x}(x) = \left(\frac{\partial F}{\partial x_1}(x), \dots, \frac{\partial F}{\partial x_n}(x) \right), \quad \frac{\partial^2 F}{\partial x^2} = \begin{pmatrix} \frac{\partial^2 F}{\partial x_1^2}(x) & \dots & \frac{\partial^2 F}{\partial x_1 \partial x_n}(x) \\ \vdots & & \\ \frac{\partial^2 F}{\partial x_n \partial x_1}(x) & \dots & \frac{\partial^2 F}{\partial x_n^2}(x) \end{pmatrix}.$$

Выпишем (2.7) в развёрнутом виде:

$$\begin{aligned} F(x_1^* + h\Delta x_1, \dots, x_n^* + h\Delta x_n) &= F(x_1^*, \dots, x_n^*) + h \sum_{j=1}^n \frac{\partial F}{\partial x_j}(x_1^*, \dots, x_n^*) \Delta x_j + \\ &+ \frac{1}{2!} h^2 \sum_{j=1}^n \sum_{k=1}^n \frac{\partial^2 F}{\partial x_j \partial x_k}(x_1^* + \theta h \Delta x_1, \dots, x_n^* + \theta h \Delta x_n) \Delta x_j \Delta x_k. \end{aligned}$$

Сопоставляя выражения (2.6) и (2.7), придём к основному неравенству

$$h \frac{\partial F}{\partial x}(x^*) \Delta x + \frac{1}{2!} h^2 (\Delta x)' \frac{\partial^2 F}{\partial x^2}(x^* + \theta h \Delta x) (\Delta x) \leq 0, \quad (2.8)$$

которое должно выполняться для всех направлений и для всех малых положительных чисел h . Чтобы вывести из основного неравенства необходимое условие первого порядка, разделим обе части (2.8) на h и перейдём к пределу при h , стремящемся к 0. В результате получается, что в точке локального максимума градиент функции равен нулевому вектору: $\frac{\partial F}{\partial x}(x^*) = 0$.

Точка, в которой все частные производные первого порядка равны нулю, называется *стационарной*. Следовательно, точка локального максимума всегда является стационарной. В качестве ещё одного следствия из основного неравенства получаем необходимое условие второго порядка: в точке локального максимума матрица Гессе отрицательно определена или отрицательно полуопределена: $(\Delta x)' \frac{\partial^2 F}{\partial x^2}(x^*)(\Delta x) \leq 0 \quad \forall \Delta x$.

Для того чтобы строгий локальный максимум достигался в x^* достаточно, чтобы точка x^* была стационарной точкой, в которой матрица Гессе отрицательно определена.

Иначе говоря, если выполнены условия $\frac{\partial F}{\partial x}(x^*) = 0$, $(\Delta x)' \frac{\partial^2 F}{\partial x^2}(x^*)(\Delta x) < 0$, то x^* является точкой строгого локального максимума $F(x^*) > F(x^* + \Delta x)$.

Условия первого порядка для существования локального максимума функции двух переменных при отсутствии ограничений, то есть когда требуется найти

$$\max_{x_1, x_2} F(x_1, x_2),$$

состоят в том, чтобы точка $x^* = (x_1^*, x_2^*)'$ была стационарной точкой:

$$\frac{\partial F}{\partial x_1}(x^*) = 0, \quad \frac{\partial F}{\partial x_2}(x^*) = 0.$$

Необходимые условия второго порядка в данном случае заключаются в том, что матрица Гессе отрицательно определена или отрицательно полуопределена, а это эквивалентно выполнению следующих условий, налагаемых на главные миноры матрицы:

$$\frac{\partial^2 F}{\partial x_1^2}(x^*) \leq 0,$$

$$\det \begin{pmatrix} \frac{\partial^2 F}{\partial x_1^2}(x^*) & \frac{\partial^2 F}{\partial x_1 \partial x_2}(x^*) \\ \frac{\partial^2 F}{\partial x_2 \partial x_1}(x^*) & \frac{\partial^2 F}{\partial x_2^2}(x^*) \end{pmatrix} \geq 0. \quad (2.9)$$

В завершение анализа данного случая укажем, что если в стационарной точке

$$\frac{\partial^2 F}{\partial x_1^2}(x^*) \geq 0, \quad (2.10)$$

и если выполнено (2.9), то x^* будет точкой локального минимума, если же (2.9) не выполнено, то x^* – седловая точка.

§ 3. Классическая задача математического программирования

Постановка задачи

Классическая задача математического программирования – это задача выбора таких значений некоторых переменных, подчинённых системе ограничений в форме равенств, при которых достигается максимум или минимум данной функции. В обозначениях, введённых выше, классическая задача на максимум состоит в следующем: требуется найти

$$\max_x F(x)$$

при условии, что $g(x) = b$ или (в развёрнутом виде) найти

$$\max_{x_1, \dots, x_n} F(x_1, \dots, x_n)$$

при условии, что

$$\begin{aligned} g_1(x_1, \dots, x_n) &= b_1, \\ &\vdots \\ g_m(x_1, \dots, x_n) &= b_m. \end{aligned} \quad (3.1)$$

Переменные x_1, \dots, x_n – составляющие n -мерного вектора столбца x – представляют собой средства, *инструментальные величины*. Функция $F(\cdot)$ – это *целевая функция*, а функции ограничений $g_1(\cdot), \dots, g_m(\cdot)$ есть составля-

ющие m -мерного столбца g . Вектор-столбец b содержит константы ограничений b_1, \dots, b_m .

Пусть число переменных n и число ограничений m есть конечные числа и $n > m$. Разность $n - m$ называется числом *степеней свободы* задачи. Предположим, что заданы $m + 1$ непрерывно дифференцируемых и не содержащих случайных элементов функций $F(\cdot), g_1(\cdot), \dots, g_m(\cdot)$. Вектор b состоит из фиксированных действительных чисел. Вектор x – любой вектор с вещественными компонентами, удовлетворяющий m ограничениям из (3.1).

Геометрически каждое из равенств, входящих в систему ограничений $g_i(x_1, \dots, x_n) = b_i, i = 1, \dots, m$, определяет множество точек в n -мерном евклидовом пространстве, а пересечение всех m множеств представляет собой *допустимое* множество $X = \{x \in \mathbb{E}^n \mid g(x) = b\}$.

Как и ранее, определим поверхности (линии) уровня целевой функции и направление скорейшего роста. Геометрическое истолкование задачи: в допустимом множестве отыскивается точка (или множество точек), где достигается поверхность наибольшего уровня целевой функции. Так как целевая функция непрерывна, а допустимое множество замкнуто, то, по теореме Вейерштрасса, решение существует в том случае, если допустимое множество будет непустым и ограниченным.

Метод множителей Лагранжа

Одним из наиболее эффективных методов решения классических задач программирования является метод множителей Лагранжа. Этому методу придается особое значение, поскольку он неоднократно используется в качестве основного подхода к решению почти всех видов задач оптимизации; кроме того, с его помощью можно узнать, в какой степени оптимальное значение целевой функции чувствительно к изменениям констант ограничений. Последнее обстоятельство позволяет придавать множителям Лагранжа важный экономический смысл в задачах рациональной экономической деятельности.

Для первоначального ознакомления с указанным методом рассмотрим задачу на максимум с одной степенью свободы, в которой $n = 2, m = 1$:

найти

$$\max_{x_1, x_2} F(x_1, x_2)$$

при условии, что $g(x_1, x_2) = b$.

Допустим, что в точке $x^* = (x_1^*, x_2^*)'$ существует локальное решение задачи и что в этой точке одна из частных производных функции ограничений не равна нулю. Пусть переменные занумерованы так, что $\frac{\partial g}{\partial x_2}(x^*) \neq 0$.

Тогда выражение для полного дифференциала

$$dg = \frac{\partial g}{\partial x_1} dx_1 + \frac{\partial g}{\partial x_2} dx_2 = 0 \quad (3.2)$$

можно в окрестности точки x^* записать в виде

$$\frac{\partial x_2}{\partial x_1} = -\frac{\partial g / \partial x_1}{\partial g / \partial x_2}. \quad (3.3)$$

Решая это уравнение, представим x_2 как функцию x_1 :

$$x_2 = h(x_1), \quad \text{где} \quad \frac{dh}{dx_1} = -\frac{\partial g / \partial x_1}{\partial g / \partial x_2}. \quad (3.4)$$

Теперь исходную задачу можно представить как задачу оптимизации функции одной переменной при отсутствии ограничений: найти

$$\max_{x_1} H(x_1) = F(x_1, h(x_1)).$$

Согласно результатам предыдущего раздела, условие первого порядка для существования локального максимума функции состоит в том, что

$$\frac{dH}{dx_1} = \frac{\partial F}{\partial x_1} + \frac{\partial F}{\partial x_2} \frac{dh}{dx_1} = 0.$$

Используя выражение (3.4), получим

$$\frac{dH}{dx_1} = \frac{\partial F}{\partial x_1} - \left(\frac{\partial F / \partial x_2}{\partial g / \partial x_2} \right) \frac{\partial g}{\partial x_1} = 0.$$

Очевидно, что также справедливо

$$\frac{\partial F}{\partial x_2} - \left(\frac{\partial F / \partial x_2}{\partial g / \partial x_2} \right) \frac{\partial g}{\partial x_2} = 0.$$

Введём новую переменную $y = \frac{\partial F / \partial x_2}{\partial g / \partial x_2}$.

Из условия существования локального максимума следует, что

$$\frac{\partial F}{\partial x_j} - y \frac{\partial g}{\partial x_j} = 0, \quad j = 1, 2, \quad (3.5)$$

или, исключая y ,

$$\frac{\partial F / \partial x_1}{\partial F / \partial x_2} = \frac{\partial g / \partial x_1}{\partial g / \partial x_2}. \quad (3.6)$$

Геометрическая интерпретация решения состоит в том, что полный дифференциал функции при перемещении вдоль линии постоянного уровня $F(x_1, x_2) = \text{const}$, то есть $dF = \frac{\partial F}{\partial x_1} dx_1 + \frac{\partial F}{\partial x_2} dx_2 = 0$.

Из последнего выражения вытекает следующее свойство углового коэффициента к кривой ограничений:

$$\left. \frac{dx_2}{dx_1} \right|_{\text{линия уровня}} = - \frac{\partial F / \partial x_1}{\partial F / \partial x_2}.$$

Однако, согласно (3.3), оказывается, что

$$\left. \frac{dx_2}{dx_1} \right|_{\text{кривая ограничений}} = - \frac{\partial g / \partial x_1}{\partial g / \partial x_2}.$$

Следовательно, из условия первого порядка для максимума (3.6) видно, что решение достигается в точке касания линии уровня и кривой ограничений, иначе говоря, там, где угловые коэффициенты касательных к этим кривым равны:

$$\left. \frac{dx_2}{dx_1} \right|_{\text{линия уровня}} = \left. \frac{dx_2}{dx_1} \right|_{\text{кривая ограничений}}.$$

Отметим теперь одно весьма существенное обстоятельство, а именно: необходимые условия (3.5) и исходное ограничение можно было бы получить

как условия стационарности некоторой точки функции

$$L(x_1, x_2, y) = F(x_1, x_2) + y(b - g(x_1, x_2)).$$

Эти условия состоят в том, что

$$\frac{\partial L}{\partial x_j} = \frac{\partial F}{\partial x_j} - y \frac{\partial g}{\partial x_j} = 0, \quad j = 1, 2,$$

$$\frac{\partial L}{\partial y} = b - g(x_1, x_2) = 0.$$

Переменную y называют *множителем Лагранжа*, а функцию $L(\cdot)$ – *функцией Лагранжа* (лагранжианом).

Аналогичный подход можно применить к классической задаче математического программирования в общей форме: найти

$$\max_x F(x) \tag{3.7}$$

при условии, что $g(x) = b$.

Предположим, что задача имеет локальное решение в точке x^* и что функции ограничений удовлетворяют *условию Якоби*, то есть в точке x^* ранг ρ матрицы Якоби – матрицы частных производных функции ограничений – совпадает с числом строк матрицы:

$$\rho \left(\frac{\partial g}{\partial x}(x^*) \right) = \rho \begin{pmatrix} \frac{\partial g_1}{\partial x_1}(x^*) & \dots & \frac{\partial g_1}{\partial x_n}(x^*) \\ \vdots & & \vdots \\ \frac{\partial g_m}{\partial x_1}(x^*) & \dots & \frac{\partial g_m}{\partial x_n}(x^*) \end{pmatrix} = m. \tag{3.8}$$

Отметим, что в приведённой выше задаче с одной степенью свободы матрица Якоби есть вектор-строка. Ранг этой матрицы равен единице тогда и только тогда, когда хотя бы одна из частных производных функций ограничений не обращается в нуль. При необходимости можно перенумеровать переменные таким образом, чтобы определитель матрицы, составленной из последних столбцов матрицы Якоби, не был бы равен нулю и чтобы вектор

инструментальных переменных можно было представить в виде $x = (x^1, x^2)'$, где вектор x^1 состоит из $n - m$ переменных, а вектор x^2 из m переменных. Поскольку условие Якоби выполнено, то по теореме о неявной функции в окрестности x^* можно разрешить систему ограничений, представив x^2 в виде функции x^1 : $x^2 = h(x^1)$, где h – вектор-столбец, состоящий из m функций. Теперь задачу (3.7) можно свести к задаче оптимизации при отсутствии ограничений:

$$\max_{x^1} H(x^1) = F(x^1, h(x^1)).$$

Согласно результатам предыдущего параграфа, необходимое условие для существования локального максимума состоит в том, что

$$\frac{\partial H}{\partial x^1} = \frac{\partial F}{\partial x^1} + \frac{\partial F}{\partial x^2} \frac{\partial h}{\partial x^1} = 0, \quad (3.9)$$

где $\frac{\partial H}{\partial x^1}$ представляет собой вектор-строку, состоящую из $n - m$ элементов, а $\frac{\partial h}{\partial x^1}$ – матрицу размерности $m \times (n - m)$. Так как ограничения можно записать в виде тождества $g(x^1, h(x^1)) \equiv b$, то после дифференцирования получим:

$$\frac{\partial g}{\partial x^1} + \frac{\partial g}{\partial x^2} \frac{\partial h}{\partial x^1} = 0.$$

Матрица $\frac{\partial g}{\partial x^2}$ размерности $m \times m$ – невырожденная, так как выполняется условие Якоби. Следовательно,

$$\frac{\partial h}{\partial x^1} = - \left(\frac{\partial g}{\partial x^2} \right)^{-1} \frac{\partial g}{\partial x^1}$$

и условия (3.9) можно записать следующим образом:

$$\frac{\partial F}{\partial x^1} - \left(\frac{\partial F}{\partial x^2} \right) \left(\frac{\partial g}{\partial x^2} \right)^{-1} \left(\frac{\partial g}{\partial x^1} \right) = 0. \quad (3.10)$$

Очевидно также, что

$$\frac{\partial F}{\partial x^2} - \left(\frac{\partial F}{\partial x^1} \right) \left(\frac{\partial g}{\partial x^1} \right)^{-1} \left(\frac{\partial g}{\partial x^2} \right) = 0. \quad (3.11)$$

Полагая теперь

$$y = \left(\frac{\partial F}{\partial x^2} \right) \left(\frac{\partial g}{\partial x^2} \right)^{-1} = (y_1, \dots, y_m),$$

можно представить необходимые условия (3.10) и (3.11) в форме:

$$\frac{\partial F}{\partial x} - y \frac{\partial g}{\partial x} = 0. \quad (3.12)$$

Эти необходимые условия наряду с исходными ограничениями можно получить, дифференцируя по x и по y функцию $F(x) + y(b - g(x))$.

Перечислим, из каких основных этапов состоит решение по методу Лагранжа классической задачи математического программирования: найти $\max_x F(x)$ при условии, что $g(x) = b$.

На первом этапе вводится вектор-строка из m новых переменных $y = (y_1, \dots, y_m)$, называемых *множителями Лагранжа*. На втором этапе определяется *функция Лагранжа* как сумма целевой функции и скалярного произведения вектора множителей Лагранжа и вектора разности между постоянными ограничениями и функциями ограничений $L(x, y) = F(x) + y(b - g(x))$ или в развёрнутом виде:

$$L(x_1, \dots, x_n; y_1, \dots, y_m) = F(x_1, \dots, x_n) + \sum_{i=1}^m y_i(b_i - g_i(x_1, \dots, x_n)).$$

Последним этапом является отыскание точки (x^*, y^*) , в которой все частные производные первого порядка функции Лагранжа обращаются в нуль:

$$\frac{\partial L}{\partial x}(x^*, y^*) = \frac{\partial F}{\partial x}(x^*) - y^* \frac{\partial g}{\partial x}(x^*), \quad (3.13)$$

$$\frac{\partial L}{\partial y}(x^*, y^*) = b - g(x^*) = 0.$$

Первые n из полученных соотношений, совпадающие с условиями (3.12), показывают, что градиент целевой функции должен равняться вектору множителей Лагранжа, умноженному на матрицу Якоби для функций ограниче-

ний, то есть

$$\frac{\partial F}{\partial x}(x^*) = y^* \frac{\partial g}{\partial x}(x^*), \quad \frac{\partial F}{\partial x_j}(x_1^*, \dots, x_n^*) = \sum_{i=1}^m y_i^* \frac{\partial g_i}{\partial x_j}(x_1^*, \dots, x_n^*), \quad j = 1, 2, \dots, n.$$

Остальные m условий представляют собой систему ограничений $g(x^*) = b$.

Решая совместно $m + n$ уравнений (3.13), получим значения $m + n$ неизвестных: инструментальных переменных $x^* = (x_1^*, \dots, x_n^*)'$ и множителей Лагранжа $y^* = (y_1^*, \dots, y_m^*)$. Если выполнены некоторые достаточные условия, указанные ниже, то значения переменных x^* дают локальное решение классической задачи математического программирования. Этот вывод достаточно очевиден, если учесть тот факт, что x^* удовлетворяет ограничениям и максимизирует функцию Лагранжа, которая в точке (x^*, y^*) просто совпадает со значением целевой функции,

$$L(x^*, y^*) = F(x^*) = F^*. \quad (3.14)$$

Дадим геометрическую интерпретацию $m + n$ условий первого порядка:

$$g(x^*) = b, \quad \frac{\partial F}{\partial x}(x^*) = y^* \frac{\partial g}{\partial x}(x^*). \quad (3.15)$$

Для этого заметим, что если определить i -ю кривую ограничений как

$$\{x \in \mathbb{E}^n \mid g_i(x) = b_i\},$$

то вектор градиента i -й функции ограничений

$$\frac{\partial g_i}{\partial x} = \left(\frac{\partial g_i}{\partial x_1}, \frac{\partial g_i}{\partial x_2}, \dots, \frac{\partial g_i}{\partial x_n} \right),$$

являющийся не чем иным, как i -й строкой матрицы Якоби, будет ортогональным (направленным по нормали) к этой кривой, так как

$$dg_i(x) = \frac{\partial g_i}{\partial x}(x)dx = 0, \quad i = 1, 2, \dots, m.$$

Таким образом, условия (3.15) показывают, что x^* лежит в допустимом множестве X и что направление скорейшего роста в x^* (вектор градиента

целевой функции) представляет собой линейную комбинацию (взвешенную сумму) нормалей к кривым ограничений (градиентов к этим кривым), взвешивающими коэффициентами в которой являются множители Лагранжа y^* .

Необходимые условия второго порядка состоят в том, что матрица Гессе

$$\frac{\partial^2 L}{\partial x^2}(x) = \begin{pmatrix} \frac{\partial^2 L}{\partial x_1^2}(x) & \dots & \frac{\partial^2 L}{\partial x_1 \partial x_n}(x) \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 L}{\partial x_n \partial x_1}(x) & \dots & \frac{\partial^2 L}{\partial x_n^2}(x) \end{pmatrix}, \quad (3.16)$$

составленная из вторых частных производных лагранжиана по инструментальным переменным, должна быть отрицательно определённой или отрицательно полуопределённой в точке локального максимума (x^*, y^*) при том условии, что

$$dg = \frac{\partial g}{\partial x}(x^*)dx = 0. \quad (3.17)$$

Если при указанных условиях матрица Гессе является отрицательно определённой, то условия первого порядка (3.15) являются достаточными для существования локального максимума. Условие, что матрица Гессе (3.16) является отрицательно определённой при ограничениях (3.17), может быть представлено в форме $n - m$ условий, налагаемых на знаки некоторых миноров матрицы размерности $(m + n) \times (m + n)$

$$\begin{pmatrix} 0 & \frac{\partial g}{\partial x} \\ \frac{\partial g'}{\partial x} & \frac{\partial^2 L}{\partial x^2} \end{pmatrix} = \begin{pmatrix} 0 & \dots & 0 & \frac{\partial g_1}{\partial x_1} & \frac{\partial g_1}{\partial x_2} & \dots & \frac{\partial g_1}{\partial x_n} \\ \vdots & & & \vdots & & & \\ 0 & \dots & 0 & \frac{\partial g_m}{\partial x_1} & \frac{\partial g_m}{\partial x_2} & \dots & \frac{\partial g_m}{\partial x_n} \\ \frac{\partial g_1}{\partial x_1} & \dots & \frac{\partial g_m}{\partial x_1} & \frac{\partial^2 L}{\partial x_1^2} & \frac{\partial^2 L}{\partial x_1 \partial x_2} & \dots & \frac{\partial^2 L}{\partial x_1 \partial x_n} \\ \vdots & & & \vdots & & & \\ \frac{\partial g_1}{\partial x_n} & \dots & \frac{\partial g_m}{\partial x_n} & \frac{\partial^2 L}{\partial x_n \partial x_1} & \frac{\partial^2 L}{\partial x_n \partial x_2} & \dots & \frac{\partial^2 L}{\partial x_n^2} \end{pmatrix}, \quad (3.18)$$

полученной в результате окаймления матрицы Гессе матрицами Якоби для функций ограничений. Условия локального максимума в такой форме заключаются в том, что последние $n - m$ миноров полученной матрицы имеют чередующиеся знаки, и знак первого из них совпадает со знаком $(-1)^{m+1}$.

Интерпретация множителей Лагранжа

Решение системы уравнений (3.15), выражающих условия первого порядка, содержит, кроме вектора локального оптимума x^* , еще и вектор множителей Лагранжа y^* . Если выполнено условие Якоби, то существует единственный вектор y^* , соответствующий локальному решению x^* . Знание значений множителя Лагранжа не является излишним; напротив, с их помощью можно получить ценную информацию о задаче. Широкое практическое использование метода множителей Лагранжа во многом объясняется именно последним обстоятельством. Множители Лагранжа, соответствующие решению задачи, измеряют чувствительность оптимального значения целевой функции $F^* = F(x^*)$ к изменениям констант ограничений b :

$$y^* = \frac{\partial F^*}{\partial b}, \quad (3.19)$$

то есть $y_i^* = \frac{\partial F^*}{\partial b_i}$, $i = 1, 2, \dots, m$.

Для доказательства соотношений (3.19) необходимо сначала показать, что если величины b_i рассматриваются как переменные, то переменные x_j и y_i ($j = 1, \dots, n$; $i = 1, \dots, m$) можно представить в виде функций b_i . Рассмотрим с этой целью условия первого порядка (3.15), которые можно записать в виде системы $m + n$ уравнений с $2m + n$ неизвестными (b, y, x) , если считать ($i = 1, \dots, m$) переменными величинами:

$$\Psi^1(b, y, x) \equiv b - g(x) = 0, \quad \Psi^2(b, y, x) \equiv \frac{\partial F}{\partial x}(x) - y \frac{\partial g}{\partial x}(x) = 0.$$

Запишем матрицу Якоби этой системы уравнений в следующем виде:

$$\begin{pmatrix} I & 0 & -\left(\frac{\partial g}{\partial x}\right) \\ 0 & -\left(\frac{\partial g}{\partial x}\right)' & \frac{\partial^2 L}{\partial x^2} \end{pmatrix},$$

где I – единичная матрица порядка m . Ранг матрицы Якоби равен m , если выполнены достаточные условия, налагаемые на окаймлённую матрицу Гессе (3.18). Следовательно, по теореме о неявной функции, решая систему уравнений, составленную из $m + n$ условий первого порядка, можно представить инструментальные переменные в виде функций от постоянных ограничений b : $y = y(b)$, $x = x(b)$.

Рассмотрим теперь функцию Лагранжа как функцию, зависящую от постоянных ограничений: $L(b) = F(x(b)) + y(b)[b - g(x(b))]$.

Дифференцирование по b даёт:

$$\begin{aligned} \frac{\partial L}{\partial b} &= \frac{\partial F}{\partial x} \frac{\partial x}{\partial b} - y \frac{\partial g}{\partial x} \frac{\partial x}{\partial b} + (b - g(x))' \frac{\partial y'}{\partial b} + y = \\ &= \left(\frac{\partial F}{\partial x} - y \frac{\partial g}{\partial x} \right) \frac{\partial x}{\partial b} + (b - g(x))' \frac{\partial y'}{\partial b} + y. \end{aligned}$$

Первые два члена этого выражения в точке (x^*, y^*) обращаются в нуль из-за условий первого порядка (3.13), так что $\frac{\partial L}{\partial b}$ равно вектору множителей Лагранжа в точке (x^*, y^*) – это и есть оптимальное значение целевой функции (3.14). Следовательно, $\frac{\partial L}{\partial b}(x^*, y^*) = \frac{\partial F^*}{\partial b} = y^*$, что и утверждалось ранее. Таким образом, помимо того, что метод множителей Лагранжа даёт решение классической задачи максимизации, он позволяет также проанализировать, насколько оптимальное значение целевой функции чувствительно к изменениям констант ограничений. Например, если какой-то множитель Лагранжа равен нулю, то малые изменения соответствующей константы ограничений не окажут никакого влияния на оптимальное значение целевой функции.

§ 4. Нелинейное программирование

Постановка задачи ограничения неотрицательности

Задача нелинейного программирования – это задача выбора таких неотрицательных значений некоторых переменных, подчинённых системе ограничений в форме неравенств, при которых достигается минимум или максимум данной функции. Задача нелинейного программирования на максимум состоит в следующем: требуется найти

$$\max_x F(x)$$

при условии, что $g(x) \leq b$, $x \geq 0$, или в развёрнутом виде: найти

$$\max_{x_1, \dots, x_n} F(x_1, \dots, x_n)$$

при условии, что

$$g_1(x_1, \dots, x_n) \leq b_1,$$

$$\vdots$$

$$g_m(x_1, \dots, x_n) \leq b_m,$$

$$x_1 \geq 0, \dots, x_n \geq 0.$$

Величины x_1, \dots, x_n – составляющие n -мерного вектора столбца x – представляют собой инструментальные переменные. Функция $F(\cdot)$ – это целевая функция, а функции ограничений $g_1(\cdot), \dots, g_m(\cdot)$ – составляющие m -мерного столбца $g(\cdot)$. Вектор b содержит константы ограничений b_1, \dots, b_m . Предположим, что n и m – конечные числа; заданы $m+1$ непрерывно дифференцируемых и не содержащих случайных элементов функций $F(\cdot), g_1(\cdot), \dots, g_m(\cdot)$; вектор b состоит из заданных вещественных чисел; x – любой вектор с вещественными компонентами, удовлетворяющий $m+n$ ограничениям.

Сделаем несколько замечаний относительно задачи нелинейного программирования. Во-первых, отметим, что, в отличие от задачи классического программирования, величины m и n не связаны никакими ограничениями. Во-вторых, выбор знаков неравенств \leq совершенно условен. Например, умножая неравенство $x_1 - 2x_2 \geq 7$ на -1 , можно превратить его в $-x_1 + 2x_2 \leq -7$,

изменив тем самым знак неравенства на противоположный. В-третьих, отметим, что ограничение в форме равенства, например $x_3 + 8x_7 = 12$, можно заменить парой ограничений неравенств $x_3 + 8x_7 \leq 12$ и $-x_3 - 8x_7 \geq -12$. В-четвертых, условие неотрицательности всех инструментальных переменных x_1, x_2, \dots, x_n не является обязательным. Если на значения некоторой переменной, например x_9 , ограничения не налагаются (то есть она может принимать отрицательные, положительные и нулевые значения), то эту переменную можно заменить двумя переменными: $x'_9 \geq 0$ и $x''_9 \geq 0$, полагая $x_9 = x'_9 - x''_9$, так что новая формулировка задачи не будет включать переменную x_9 . Таким образом, классическую задачу математического программирования можно рассматривать как частную задачу нелинейного программирования, в которой условие неотрицательности переменных отсутствует, а ограничения неравенства заменены ограничениями в форме равенств.

Геометрически каждое из n условий неотрицательности $x_j \geq 0$, $j = 1, 2, \dots, n$ определяет полупространство неотрицательных значений независимых переменных, а пересечение таких полупространств представляет собой подмножество n -мерного евклидова пространства, называемое *неотрицательным ортантом*. Например, в \mathbb{E}^2 неотрицательный ортант – это неотрицательный квадрант, то есть первый квадрант плюс соответствующие полуоси. Каждое из m ограничений-неравенств $g_i(x_1, x_2, \dots, x_n) \leq b_i$, $j = 1, 2, \dots, n$ также определяет множество точек в n -мерном евклидовом пространстве, а пересечение этих m множеств с неотрицательным ортантом составляет *допустимое множество* $X = \{x \in \mathbb{E}^n \mid g(x) \leq b, \ x \geq 0\}$. Поверхности (линии) постоянного уровня и направления скорейшего роста, дающие геометрическое представление о целевой функции, рассмотрены ранее. Геометрически задача нелинейного программирования состоит в отыскании точки или множества точек из допустимого множества, где достигается поверхность наибольшего уровня. Так как по предположению целевая функция непрерывна и допустимое множество замкнуто, то, согласно теореме Вейерштрасса, решение (глобальный максимум) задачи существует, если до-

пустимое множество непустое и ограниченное. Решение может принадлежать либо границе, либо внутренней части допустимого множества.

Важную роль в задачах нелинейного программирования играют условия выпуклости. Задачей выпуклого программирования часто называют задачу, в которой функции ограничений выпуклые, а целевая функция вогнутая. В этом случае локальный максимум целевой функции, находящийся внутри допустимого множества или на его границе, является глобальным максимумом, а множество точек, на которых достигается глобальный максимум, выпукло. Если дополнительно предполагается, что целевая функция строго вогнута, то задача имеет единственное решение.

Если на переменные налагаются только условия неотрицательности ($m = 0$), то основная задача сводится к выбору неотрицательных значений инструментальных переменных, максимизирующих функцию, то есть к задаче вида: найти

$$\max_x F(x) \quad (4.1)$$

при условии, что $x \geq 0$.

Одним из подходов к решению этой задачи является разложение в ряд Тейлора, использованное нами ранее для решения классической задачи математического программирования при отсутствии ограничений. Предположим, что в x^* существует локальный максимум задачи (4.1). Тогда в точках, близких к x^*

$$F(x^*) \geq F(x^* + h\Delta x), \quad (4.2)$$

где Δx определяет направление в \mathbb{E}^n , а h – любое произвольное малое положительное число. Пусть $F(x)$ – дважды непрерывно дифференцируемая функция, тогда функцию в правой части (4.2) можно разложить по формуле Тейлора в окрестности x^* :

$$F(x^* + h\Delta x) = F(x^*) + h \frac{\partial F}{\partial x}(x^*) \Delta x + \frac{1}{2!} h^2 (\Delta x)' \frac{\partial^2 F}{\partial x^2}(x^* + \theta h \Delta x) \Delta x,$$

где $0 < \theta < 1$.

Сопоставляя последние два неравенства, приходим к *основному неравенству*

$$h \frac{\partial F}{\partial x}(x^*) \Delta x + \frac{1}{2!} h^2 (\Delta x)' \frac{\partial^2 F}{\partial x^2}(x^* + \theta h \Delta x) \Delta x \leq 0,$$

являющемуся необходимым условием существования локального максимума в x^* . Если $x^* > 0$, то есть x^* – внутренняя точка, то основное неравенство выполняется для всех направлений Δx . Поэтому, как и в классических задачах математического программирования, условием первого порядка является обращение в нуль первых производных. Допустим теперь, что одна из инструментальных переменных принимает граничное значение $x_j^* = 0$. Пусть приращения всех остальных переменных равны нулю. Тогда, поскольку при $x_j^* = 0$ единственным допустимым направлением является такое, при котором $\Delta x_j \geq 0$, то из основного неравенства следует, что

$$\frac{\partial F}{\partial x_j}(x^*) \Delta x_j \leq 0.$$

Следовательно, условие первого порядка состоит в том, что

$$\frac{\partial F}{\partial x_j}(x^*) \leq 0, \quad \text{если } x_j^* = 0.$$

Итак, в точке внутреннего максимума ($x_j^* > 0$) производная по x_j обращается в 0, а если максимум достигается на границе ($x_j^* = 0$), то первая производная меньше или равна 0. Но так как либо производная равна нулю (во внутренней точке), либо соответствующая переменная принимает нулевое значение (в граничной точке), то их произведение в точке максимума всегда равно нулю:

$$\frac{\partial F}{\partial x_j}(x^*) x_j^* = 0. \tag{4.3}$$

Сумма этих произведений также равна нулю:

$$\frac{\partial F}{\partial x}(x^*) x^* = \sum_{j=1}^n \frac{\partial F}{\partial x_j}(x^*) x_j^* = 0.$$

Условие обращения в нуль суммы указанных произведений на самом деле требует, чтобы каждое слагаемое равнялось нулю (то есть соотношение (4.3) должно выполняться для всех j). Вышесказанное следует из того, что все переменные неотрицательны, а частные производные первого порядка не превышают нуля. Таким образом, локальный максимум в x^* определяется с помощью $2n + 1$ условий первого порядка:

$$\frac{\partial F}{\partial x}(x^*) \leq 0, \quad \frac{\partial F}{\partial x}(x^*)x^* = 0, \quad x^* \geq 0. \quad (4.4)$$

Из этих условий следует приведённый выше результат: в точке максимума частная производная либо равна нулю, если соответствующая переменная положительна, либо меньше или равна нулю, если соответствующая переменная обращается в нуль:

$$\begin{aligned} \frac{\partial F}{\partial x_j}(x^*) &= 0, \quad \text{если } x_j^* > 0, \\ \frac{\partial F}{\partial x_j}(x^*) &\leq 0, \quad \text{если } x_j^* = 0, \end{aligned} \quad j = 1, 2, \dots, n.$$

Условия Куна-Таккера

Используя результаты предыдущего параграфа, можно проанализировать общую задачу нелинейного программирования: найти

$$\max_x F(x) \quad \text{при условии} \quad g(x) \leq b, \quad x \geq 0. \quad (4.5)$$

Для того, чтобы преобразовать ограничения-неравенства в ограничения в форме равенств, введём вектор, состоящий из m вспомогательных («свободных») переменных $s \equiv b - g(x) = (s_1, s_2, \dots, s_m)'$. Теперь задача состоит в отыскании

$$\max_{x,s} F(x) \quad \text{при условии} \quad g(x) + s = b, \quad x \geq 0, \quad s \geq 0, \quad (4.6)$$

причём неотрицательность вспомогательных переменных обеспечивает выполнение ограничений-неравенств.

Если бы выражения (4.6) не включали $m + n$ ограничений-неравенств, то наша задача являлась бы классической задачей математического программирования. Функция Лагранжа такой задачи имела бы вид $L' = F(x) + y(b - g(x) - s)$, где $y = (y_1, y_2, \dots, y_m)$ – вектор множителей Лагранжа.

Необходимые условия первого порядка состояли бы в том, что все первые частные производные функции L' по x , y и s должны обращаться в нуль. Но так как x и s неотрицательны, то соответствующие условия на значения первых производных по этим $m+n$ переменным заменяются на условия предыдущего параграфа. В соответствии с этим условия первого порядка для существования локального максимума задачи (4.6) состоят в следующем:

$$\begin{aligned} \frac{\partial L'}{\partial x} = \frac{\partial F}{\partial x} - y \frac{\partial g}{\partial x} &\leq 0, & \frac{\partial L'}{\partial x} x &= \left(\frac{\partial F}{\partial x} - y \frac{\partial g}{\partial x} \right) x = 0, \\ x \geq 0, & \frac{\partial L'}{\partial y} = b - g(x) - s = 0, & \frac{\partial L'}{\partial s} = -y &\leq 0, \\ & \frac{\partial L'}{\partial s} s = -ys = 0, & s &\geq 0, \end{aligned} \quad (4.7)$$

где все переменные, функции и производные вычисляются при x^* , y^* и s^* . Заменяя вектор вспомогательных переменных s на $b - g(x)$, приходим к условиям Куна-Таккера:

$$\left(\frac{\partial F}{\partial x} - y \frac{\partial g}{\partial x} \right) \leq 0, \quad \left(\frac{\partial F}{\partial x} - y \frac{\partial g}{\partial x} \right) x = 0, \quad x \geq 0,$$

$$(b - g(x)) \geq 0, \quad y(b - g(x)) = 0, \quad y \geq 0.$$

Те же условия можно получить, если определить функцию Лагранжа для исходной задачи (4.5) как сумму целевой функции и скалярного произведения векторов множителей Лагранжа и разности между константами ограничений $L = L(x, y) = F(x) + y(b - g(x))$, или в развёрнутом виде

$$L(x_1, \dots, x_n; y_1, \dots, y_m) = F(x_1, \dots, x_n) + \sum_{i=1}^m y_i (b_i - g_i(x_1, \dots, x_n)).$$

Условия Куна-Таккера записываются теперь следующим образом:

$$\begin{aligned}
\frac{\partial L}{\partial x}(x^*, y^*) &= \frac{\partial F}{\partial x}(x^*) - y^* \frac{\partial g}{\partial x}(x^*) \leq 0, \\
\frac{\partial L}{\partial x}(x^*, y^*) x^* &= \left(\frac{\partial F}{\partial x}(x^*) - y^* \frac{\partial g}{\partial x}(x^*) \right) x^* = 0, \\
x^* &\geq 0, \\
\frac{\partial L}{\partial y}(x^*, y^*) &= b - g(x^*) \geq 0, \\
y^* \frac{\partial L}{\partial y}(x^*, y^*) &= y^*(b - g(x^*)) = 0, \\
y^* &\geq 0.
\end{aligned}$$

Эти условия являются необходимыми и достаточными для существования локального максимума, если целевая функция (строго) вогнутая, а функции ограничений выпуклые, и если, кроме того, выполнено налагаемое на ограничения некоторое условие регулярности, которое будет введено в следующем параграфе. Условия Куна-Таккера можно записать в развернутом виде как $2m + 2n + 2$ соотношений:

$$\frac{\partial L}{\partial x_j} = \frac{\partial F}{\partial x_j} - \sum_{i=1}^m y_i \frac{\partial g_i}{\partial x_j} \leq 0, \quad j = 1, 2, \dots, n, \quad (4.8)$$

$$\sum_{j=1}^n \frac{\partial L}{\partial x_j} x_j = \sum_{j=1}^n \left(\frac{\partial F}{\partial x_j} - \sum_{i=1}^m y_i \frac{\partial g_i}{\partial x_j} \right) x_j = 0, \quad j = 1, 2, \dots, n, \quad (4.9)$$

$$x_j \geq 0, \quad j = 1, 2, \dots, n, \quad (4.10)$$

$$\frac{\partial L}{\partial y_i} = b_i - g_i(\cdot) \geq 0, \quad i = 1, 2, \dots, m, \quad (4.11)$$

$$\sum_{i=1}^m y_i \frac{\partial L}{\partial y_i} = \sum_{i=1}^m y_i (b_i - g_i(\cdot)) = 0, \quad (4.12)$$

$$y_i \geq 0, \quad i = 1, 2, \dots, m, \quad (4.13)$$

где все переменные, функции и производные вычисляются в (x^*, y^*) .

Чтобы понять важный смысл этих условий, отметим прежде всего, что соотношения (4.10) и (4.11) выражают соответственно ограничения неотрицательности и ограничения-неравенства исходной задачи нелинейного программирования. Отметим, что вследствие неравенств (4.8) и (4.10) каждое из слагаемых суммы (4.9) должно равняться нулю, так что

$$\begin{aligned} \text{либо } \frac{\partial F}{\partial x_j} - \sum_{i=1}^m y_i \left(\frac{\partial g_i}{\partial x_j} \right) &= 0, \\ \text{либо } x_j &= 0, \end{aligned}$$

(либо эти равенства выполняются одновременно),

$$j = 1, 2, \dots, n.$$

Иначе говоря, либо каждое из соотношений между производными выполняется как равенство, либо переменная равна нулю, либо обе эти возможности осуществляются одновременно. Таким образом,

$$\begin{aligned} \frac{\partial F}{\partial x_j} - \sum_{i=1}^m y_i \frac{\partial g_i}{\partial x_j} &\leq 0, \quad \text{но} \\ \frac{\partial F}{\partial x_j} - \sum_{i=1}^m y_i \frac{\partial g_i}{\partial x_j} &= 0, \quad \text{если } x_j^* > 0, \end{aligned} \tag{4.14}$$

$$\begin{aligned} x_j^* \geq 0, \quad \text{но } x_j^* = 0, \quad \text{если } \frac{\partial F}{\partial x_j} - \sum_{i=1}^m y_i \frac{\partial g_i}{\partial x_j} < 0, \\ j = 1, 2, \dots, n. \end{aligned}$$

Каждое из слагаемых суммы (4.12) также равно нулю благодаря неравенствам (4.11) и (4.13), так что либо $y_i = 0$, либо $g_i(x^*) = b_i$ (или же эти неравенства выполняются одновременно), $i = 1, 2, \dots, m$.

Иначе говоря, либо множитель Лагранжа равен нулю, либо соответствующее ограничение-неравенство выполняется как равенство, либо и то и

другое выполняется одновременно. Итак,

$$\begin{aligned} g_i(x^*) \leq b_i, \quad \text{но} \quad g_i(x^*) = b_i, \quad \text{если} \quad y_i^* > 0, \\ y_i^* \geq 0, \quad \text{но} \quad y_i^* = 0, \quad \text{если} \quad g_i(x^*) < b_i, \\ i = 1, 2, \dots, m. \end{aligned} \quad (4.15)$$

Условия (4.14) и (4.15), представляющие собой иной способ формулирования условий Куна-Таккера, называют *условиями дополняющей нежёсткости*. Совершенно ясно, что $L(x^*, y^*) = F(x^*) + y^*(b - g(x^*)) = F(x^*)$, то есть значение функции Лагранжа в точке решения совпадает с оптимальным значением целевой функции, так как согласно (4.12) $y^*(b - g(x^*)) = 0$.

Дадим геометрическую интерпретацию условий Куна-Таккера. Будем рассматривать эти условия в форме (4.7). Если ввести второй вектор дополнительных переменных $r = y \frac{\partial g}{\partial x} - \frac{\partial F}{\partial x} = (r_1, \dots, r_n)$, то (4.7) преобразуются к виду:

$$\frac{\partial F}{\partial x} - y \frac{\partial g}{\partial x} + r = 0,$$

$$rx = 0, \quad r = 0, \quad x \leq 0,$$

$$b - g(x) - s = 0,$$

$$ys = 0, \quad s \geq 0, \quad y \geq 0,$$

где все переменные, функции и производные вычисляются при $x = x^*$, $y = y^*$, $r = r^*$, $s = s^*$. Неотрицательность вспомогательных переменных обеспечивает выполнение соответствующих ограничений-неравенств. Первые n соотношений можно записать в форме

$$\frac{\partial F}{\partial x}(x^*) = y^* \frac{\partial g}{\partial x}(x^*) + r^*(-I), \quad (4.16)$$

где I – единичная матрица. Геометрический смысл (4.16) состоит в том, что в искомой точке градиент целевой функции $\frac{\partial F}{\partial x}$ должен быть линейной комбинацией градиентов ограничивающих гиперповерхностей. Если решение

лежит на границе, то направление скорейшего роста представляет собой линейную комбинацию векторов – нормалей к поверхности, взятых с неотрицательными коэффициентами.

Рассмотрим теперь характерный пример задачи нелинейного программирования: найти

$$\max_{x_1, x_2} F(x_1, x_2) = -8x_1^2 - 10x_2^2 + 12x_1x_2 - 50x_1 + 80x_2 \quad (4.17)$$

при условии, что

$$x_1 + x_2 \leq 1, \quad 8x_1^2 + x_2^2 \leq 2,$$

$$x_1 \geq 0, \quad x_2 \geq 0.$$

Поскольку здесь целевая функция строго вогнутая, а функции ограничений выпуклые, то система неравенств, входящих в условия Куна-Таккера, имеет единственное решение в точке глобального максимума. Функция Лагранжа для этой задачи имеет вид

$$\begin{aligned} L(x_1, x_2, y_1, y_2) = & -8x_1^2 - 10x_2^2 + 12x_1x_2 - 50x_1 + 80x_2 + \\ & + y_1(1 - x_1 - x_2) + y_2(2 - 8x_1^2 - x_2^2), \end{aligned}$$

а условия Куна-Таккера:

$$\frac{\partial L}{\partial x_1} = -16x_1 + 12x_2 - 50 - y_1 - 16y_2x_1 \leq 0,$$

$$\frac{\partial L}{\partial x_2} = -20x_2 + 12x_1 + 80 - y_1 - 2y_2x_2 \leq 0,$$

$$\frac{\partial L}{\partial x_1}x_1 + \frac{\partial L}{\partial x_2}x_2 = (-16x_1 + 12x_2 - 50 - y_1 - 16y_2x_1)x_1 +$$

$$+ (-20x_2 + 12x_1 + 80 - y_1 - 2y_2x_2)x_2 = 0,$$

$$x_1 \geq 0, \quad x_2 \geq 0,$$

$$\frac{\partial L}{\partial y_1} = 1 - x_1 - x_2 \geq 0, \quad \frac{\partial L}{\partial y_2} = 2 - 8x_1^2 - x_2^2 \geq 0,$$

$$y_1 \frac{\partial L}{\partial y_1} + y_2 \frac{\partial L}{\partial y_2} = y_1(1 - x_1 - x_2) + y_2(2 - 8x_1^2 - x_2^2) = 0,$$

$$y_1 \geq 0, \quad y_2 \geq 0.$$

Хотя эти условия полностью характеризуют решение, они малоприспособлены для практического отыскания решений этой задачи. Рассмотрим несколько различных допустимых точек. Точка начала координат $(0, 0)$ не удовлетворяет условиям Куна-Таккера, так как в этой точке $y_1 = 0$, $y_2 = 0$, а $\frac{\partial L}{\partial x_2} = 80$. Точка $\left(\frac{1}{2}, 0\right)$ также не удовлетворяет условиям, так как здесь $y_1 = 0$, а $\frac{\partial L}{\partial x_2} = 86$. Однако точка $(0, 1)$ является решением задачи. В этой точке:

$$x^* = (0, 1)', \quad y^* = (60, 0),$$

$$\frac{\partial L}{\partial x}(x^*, y^*) = (-98, 0), \quad \frac{\partial L}{\partial y}(x^*, y^*) = (0, 1)',$$

$$F(x^*) = 70, \quad \frac{\partial F}{\partial x}(x^*) = (-38, 60).$$

Отметим, что в точке максимума направление скорейшего роста лежит между нормальными (векторы нормалей направлены прочь от допустимого множества). Кроме того, второе ограничение в данном случае оказалось излишним.

Теорема Куна-Таккера

Подход Куна-Таккера к общей задаче нелинейного программирования:
найти

$$\max_x F(x) \quad \text{при условии, что} \quad g(x) \leq b, \quad x \geq 0, \quad (4.18)$$

изложенный в предыдущем параграфе, состоит в том, что вводится вектор-строка множителей Лагранжа $y = (y_1, y_2, \dots, y_m)$, число которых равно числу ограничений-неравенств и определяется функциями Лагранжа

$$L(x, y) = F(x) + y(b - g(x)).$$

В этом случае условия Куна-Таккера записываются как

$$\begin{aligned} \frac{\partial L}{\partial x}(x^*, y^*) &\leq 0, & \frac{\partial L}{\partial y}(x^*, y^*) &\geq 0, \\ \frac{\partial L}{\partial x}(x^*, y^*)x^* &= 0, & y^* \frac{\partial L}{\partial y}(x^*, y^*) &= 0, \\ x^* &\geq 0, & y^* &\geq 0. \end{aligned} \tag{4.19}$$

Из сопоставления этих неравенств с условиями существования максимума (4.4) видно, что (x^*, y^*) является седловой точкой функции Лагранжа, то есть точкой, максимизирующей функцию по совокупности всех неотрицательных переменных x и минимизирующей её по совокупности всех неотрицательных множителей Лагранжа y :

$$L(x, y^*) \leq L(x^*, y^*) \leq L(x^*, y) \quad \forall \quad x \geq 0, y \geq 0. \tag{4.20}$$

Задача отыскания неотрицательных векторов (x^*, y^*) , удовлетворяющих (4.20), называется *задачей о седловой точке*.

По теореме Куна-Таккера x^* является решением задачи нелинейного программирования, если (x^*, y^*) является решением задачи о седловой точке, и при некоторых условиях x^* является решением задачи нелинейного программирования лишь в том случае, когда существует такой вектор y^* , что (x^*, y^*) является решением задачи о седловой точке.

В первой части теоремы утверждается, что если (x^*, y^*) представляет собой седловую точку, как в (4.20), то x^* есть решение задачи нелинейного программирования. Пусть (x^*, y^*) – седловая точка, тогда x^* максимизирует

функцию Лагранжа (относительно всех $x \geq 0$), а y^* минимизирует её:

$$F(x) + y^*(b - g(x)) \leq F(x^*) + y^*(b - g(x^*)), \quad (4.21)$$

$$F(x^*) + y^*(b - g(x^*)) \leq F(x^*) + y(b - g(x^*)).$$

Запишем последнее неравенство в следующем виде:

$$(y - y^*)(b - g(x^*)) \geq 0, \quad y \geq 0. \quad (4.22)$$

Поскольку компоненты могут быть сколько угодно большими, x^* должно удовлетворять ограничениям-неравенствам $g(x^*) \leq b$. С другой стороны, если положить в (4.22) $y = 0$, то

$$y^*(b - g(x^*)) = 0, \quad (4.23)$$

так как $y^* \geq 0$, и $b - g(x^*) \geq 0$. Используя (4.23), неравенство (4.21) можно записать как

$$F(x^*) \geq F(x) + y^*(b - g(x)), \quad x \geq 0. \quad (4.24)$$

Если x – допустимый вектор, то $F(x^*) \geq F(x)$, так как y^* неотрицателен. Следовательно, вектор x^* максимизирует $F(\cdot)$ в классе всех допустимых значений x , являясь тем самым решением задачи нелинейного программирования. Следует заметить, что при доказательстве первой части (достаточности) теоремы Куна-Таккера не потребовалось никаких специальных предположений относительно функций $F(\cdot)$ и $g(\cdot)$.

Доказательство второй части (необходимости) теоремы Куна-Таккера существенно опирается на некоторые предположения о функциях $F(\cdot)$ и $g(\cdot)$. Эта часть теоремы справедлива, если $F(\cdot)$ – вогнутая, функции $g(\cdot)$ – выпуклые и если выполнено специальное условие регулярности ограничений, состоящее в том, что в некоторой точке допустимого множества все ограничения-неравенства выполняются как строгие неравенства, то есть существует такой вектор x^0 , что $x^0 \geq 0$ и $g(x^0) < b$. Пусть при этих предположениях x^* является решением задачи нелинейного программирования, то есть $x^* \geq 0$, $g(x^*) \leq b$, $F(x^*) \geq F(x)$ при всех x таких, что $x \geq 0$, $g(x) \leq b$.

Определим два множества в $(m + 1)$ -мерном пространстве:

$$A = \left\{ \begin{pmatrix} a_0 \\ a \end{pmatrix} \mid \begin{pmatrix} a_0 \\ a \end{pmatrix} \leq \begin{pmatrix} F(x) \\ b - g(x) \end{pmatrix} \right\} \quad \text{при некоторых } x \geq 0,$$

$$B = \left\{ \begin{pmatrix} b_0 \\ b \end{pmatrix} \mid \begin{pmatrix} b_0 \\ b \end{pmatrix} > \begin{pmatrix} F(x^*) \\ 0 \end{pmatrix} \right\},$$

где a_0 и b_0 – скаляры, a и b – m -мерные векторы-строки. Так как $F(\cdot)$ – вогнутая функция, а $g(\cdot)$ – выпуклая, то множество A выпукло. Множество B , как внутренняя часть ортанта, также выпукло.

Поскольку x^* является решением задачи нелинейного программирования, то эти два множества не имеют общих точек. Следовательно, по теореме о существовании разделяющей гиперплоскости для выпуклых непересекающихся множеств существует ненулевой вектор-строка (y_0, y) , где y_0 – скаляр, а y – m -мерный вектор, такой, что для всех

$$\begin{pmatrix} a_0 \\ a \end{pmatrix} \in A \quad \text{и} \quad \begin{pmatrix} b_0 \\ b \end{pmatrix} \in B \quad (y_0, y) \begin{pmatrix} a_0 \\ a \end{pmatrix} \leq (y_0, y) \begin{pmatrix} b_0 \\ b \end{pmatrix}.$$

Так как вектор (y_0, y) неотрицателен по самому определению множества B и так как точка $(F(x^*), 0)'$ лежит на границе B , то для всех $x \geq 0$

$$y_0 F(x) + y(b - g(x)) \leq y_0 F(x^*). \quad (4.25)$$

Покажем, что $y_0 > 0$. Так как, согласно условию регулярности ограничений, в некоторой точке допустимого множества все ограничения-неравенства выполняются как строгие неравенства, то из (4.25) следует, что $y(b - g(x)) \leq 0$ при всех $x \geq 0$.

Если $y_0 = 0$, то из (4.25) следует, что $y(b - g(x)) \leq 0$ при любом $x \geq 0$. Но так как y – неотрицательный вектор, то последнее противоречит предположению о том, что существует $x^0 \geq 0$, такой, что $g(x^0) < b$.

Поскольку $y_0 > 0$, то обе части неравенства (4.25) можно разделить на y_0 , так что при любом $x \geq 0$

$$F(x) + y^*(b - g(x)) \leq F(x^*), \quad (4.26)$$

где $y^* = \left(\frac{1}{y_0}\right)$, $y \geq 0$. В частности, если $x = x^*$, то $y^*(b - g(x)) \leq 0$, а так как $g(x^*) \leq b$ и $y^* \geq 0$, то

$$y^*(b - g(x)) = 0. \quad (4.27)$$

Следовательно, если функция Лагранжа определяется как

$$L(x, y) = F(x) + y(b - g(x)),$$

то из (4.26), (4.27) и неотрицательности y следует, что (x^*, y^*) представляет собой седловую точку для $L(x, y)$ при $x \geq 0$, $y \geq 0$. Таким образом доказана вторая часть теоремы. Итак, при указанных условиях x^* будет решением задачи нелинейного программирования (4.18) тогда и только тогда, когда существует вектор y^* , такой, что (x^*, y^*) является решением задачи о седловой точке (4.20).

Рассмотрим теперь задачу о седловой точке при дополнительном предположении о дифференцируемости функций $F(x)$ и $g(x)$, которое не использовалось при доказательстве. Первая часть задачи о седловой точке заключается в отыскании максимума по неотрицательным переменным x . Применяя (4.4), получаем следующие условия:

$$\frac{\partial L}{\partial x}(x^*, y^*) \leq 0, \quad \frac{\partial L}{\partial x}(x^*, y^*)x^* = 0, \quad x^* \geq 0.$$

Для решения второй части задачи о седловой точке – минимизации $L(x^*, y^*)$ по неотрицательным множителям Лагранжа y требуется выполнение условий:

$$\frac{\partial L}{\partial y}(x^*, y^*) \geq 0, \quad y^* \frac{\partial L}{\partial y}(x^*, y^*) = 0, \quad y^* \geq 0.$$

Полученные соотношения есть условия Куна-Таккера (4.19).

Интерпретация множителей Лагранжа

Как и ранее, множители Лагранжа можно истолковывать как характеристики изменений оптимального значения целевой функции при изменениях констант ограничений:

$$y^* = \frac{\partial F^*}{\partial b}. \quad (4.28)$$

Для доказательства следует сначала показать, что x^* и y^* можно представить в виде функций от констант-ограничений, а затем продифференцировать функцию Лагранжа по этим константам.

Условия Куна-Таккера можно было бы записать в виде равенств, если бы было известно, какие именно ограничения выполняются как равенства, а какие как неравенства, а также то, какие именно переменные равны нулю, а какие положительны. Допустим, что эти соотношения и переменные переименованы так, что в точке оптимума первые m_1 из них выполняются как равенства, а остальные $m - m_1$ — как неравенства ($0 \leq m_1 \leq m$) и что первые n_1 переменные положительны, а остальные $n - n_1$ равны нулю ($0 \leq n_1 \leq n$). Векторы можно расчленить следующим образом:

$$g(x) = \begin{pmatrix} g^1(x) \\ g^2(x) \end{pmatrix}, \quad b = \begin{pmatrix} b^1 \\ b^2 \end{pmatrix}, \quad y = (y^1, y^2), \quad x = \begin{pmatrix} x^1 \\ x^2 \end{pmatrix},$$

где $g^1(x)$, b^1 и y^1 состоят из первых элементов векторов $g(x)$, b и y соответственно, а x^1 — из первых n_1 элементов вектора x . Тогда условия Куна-Таккера можно записать как:

$$\begin{aligned} \frac{\partial L}{\partial x^1} &= \frac{\partial F}{\partial x^1}(x) - y^1 \frac{\partial g^1}{\partial x^1}(x) = 0, \\ x^2 &= 0, \\ \frac{\partial L}{\partial y^1} &= b^1 - g^1(x) = 0, \\ y^2 &= 0. \end{aligned}$$

Очевидно, что (4.28) выполняется для последних $m - m_1$ множителей Лагранжа, равных нулю; следовательно,

$$y_i^* = \frac{\partial F^*}{\partial b_i} = 0, \quad i = m_1 + 1, m_1 + 2, \dots, m.$$

Последние $m - m_1$ ограничений выполняются как неравенства, так что малые изменения соответствующих констант ограничений не могли бы изменить оптимального значения целевой функции. Что касается первых m_1 множителей Лагранжа, то заметим, что задача приведена к форме классической задачи математического программирования: найти

$$\max_{x^1} F(x^1, 0) \quad \text{при условии, что} \quad g^1(x^1, 0) = b^1.$$

Оказывается возможным представить x^1 и y^1 как функции от b^1 и продифференцировать функцию Лагранжа по b^1 . В результате получим, что

$$y_i^* = \frac{\partial F^*}{\partial b_i} \geq 0, \quad i = 1, 2, \dots, m_1.$$

В задаче (4.17), где $y^* = (60, 0)$, при малом увеличении константы первого ограничения до $1 + \Delta b^1$ оптимальное значение возросло бы до $70 + 60\Delta b^1$, тогда как малое увеличение константы второго ограничения никак не повлияло бы на оптимальное значение целевой функции, поскольку это ограничение оказалось нежёстким.

Алгоритмы решения

Условия Куна-Таккера хоть и дают полную характеристику решения, но не содержат конструктивного метода отыскания решения. Под алгоритмом решения понимается конструктивный метод, с помощью которого можно найти численное решение. Алгоритмы решения обычно позволяют определить, как изменяются во времени инструментальные переменные, то есть эти переменные отыскиваются в виде функций времени:

$$x(t) = (x_1(t), x_2(t), \dots, x_n(t))'.$$

Эти функции часто определяют с помощью дифференциальных уравнений, дающих выражение для скорости изменения переменных $x(t)$:

$$x'(t) = \left(\frac{dx_1(t)}{dt}, \frac{dx_2(t)}{dt}, \dots, \frac{dx_n(t)}{dt} \right)' = (\dot{x}_1(t), \dot{x}_2(t), \dots, \dot{x}_n(t))'.$$

Если значение вектора инструментальных переменных в начальный момент $x(0) = (x_1(0), x_2(0), \dots, x_n(0))$ является фиксированным, то решение дифференциального уравнения сводится к решению задачи нелинейного программирования $\lim_{t \rightarrow \infty} x(t) = x^*$.

Можно дать классификацию этих алгоритмов, основываясь на свойствах начальной точки $x(0)$. Если начальная точка не является допустимой, то дифференциальные уравнения смещают её в пределы допустимого множества, а затем приводят к решению задачи. Этот подход к решению назовём *выбором начальной точки без учёта ограничений*. При другом подходе – *выборе начальной точки с учётом ограничений* – начальная точка должна принадлежать допустимому множеству, а дифференциальные уравнения перемещают точку по линиям уровня, всё увеличивая значение функции и постепенно приводя к решению.

Многие алгоритмы решения задач нелинейного программирования представляют собой градиентные методы. Эти методы основаны на том, что градиент, то есть вектор $\frac{\partial F}{\partial x}(x) = \left(\frac{\partial F}{\partial x_1}(x), \dots, \frac{\partial F}{\partial x_n}(x) \right)$, состоящий из частных производных первого порядка целевой функции, направлен в каждой точке в сторону наибольшего роста целевой функции. Поэтому перемещение в направлении градиента приводит к наибольшему увеличению целевой функции. Решая градиентным методом задачу оптимизации при отсутствии ограничений, мы меняем каждую переменную в любой точке в соответствии с величиной её частной производной в этой точке $\dot{x}_i(t) = \frac{\partial F}{\partial x_i}(x(t))$, $i = 1, 2, \dots, n$.

Если предположить, что $F(\cdot)$ выпуклая функция, то с помощью этого метода можно достичь точки максимума.

Для задачи оптимизации при наличии ограничений необходимо модифицировать градиентный метод. Примером такого модифицированного метода является *градиентный проективный метод*.

При этом методе решения выбор начальной точки осуществляется с учетом ограничений, так что эта точка может быть допустимой. Перемещение точки происходит в направлении градиента, за исключением тех случаев, когда движение в этом направлении может вывести за пределы допустимого множества. В последнем случае точка смещается в направлении проекции градиента на плоскость, касательную к границе. Перемещение по указанному принципу не выводит точку из допустимого множества и постоянно увеличивает значение целевой функции. Если целевая функция вогнутая, а допустимое множество выпуклое, то указанное перемещение точки в конце концов приводит к решению.

Другим примером градиентного метода в нелинейном программировании является дифференциальный градиентный метод, основанный на условиях Куна-Таккера для функции Лагранжа. Выбор начальной точки в этом методе осуществляется без учёта ограничений. Дифференциальные уравнения для переменных $x(t)$ и для множителей Лагранжа $y(t)$, рассматриваемых как функции времени, имеют следующий вид

$$\dot{x}_j(t) = \begin{cases} 0, & \text{если } x_j = 0 \text{ и } \frac{\partial L}{\partial x_j}(x(t), y(t)) < 0, \\ \frac{\partial L}{\partial x_j}(x(t), y(t)) = \frac{\partial F}{\partial x_j}(x(t)) - \sum_{i=1}^m y_i \frac{\partial g_i}{\partial x_j}(x(t)), & \text{иначе} \end{cases},$$

$$j = 1, 2, \dots, n,$$

$$\dot{y}_i(t) = \begin{cases} 0, & \text{если } y_i = 0 \text{ и } \frac{\partial L}{\partial y_i}(x(t), y(t)) < 0, \\ -\frac{\partial L}{\partial y_i}(x(t), y(t)) = -[b_i - g_i(x(t))], & \text{иначе} \end{cases},$$

$$i = 1, 2, \dots, m.$$

Данный способ решения является градиентным методом, так как скорость изменения каждой из переменных равна соответствующей частной производной первого порядка целевой функции, изменённой на некоторую величину для учёта наложенных ограничений.

§ 5. Пример численного решения задачи многомерной нелинейной оптимизации

Постановка задачи численной оптимизации

Оценка значения функционала качества системы может дать ответ на вопрос о приемлемости уровня её функционирования. Для того, чтобы добиться наилучшего функционирования системы решается оптимизационная задача, где в качестве целевой функции выступает функционал качества системы, а в качестве ограничений – условия, диктуемые предметной областью.

Пусть решается задача: найти

$$\min_{x \in D} f(x), \quad \text{при условии, что} \quad D = \{x \in \mathbb{R}^n \mid F(x) = 0, G(x) \leq 0\}. \quad (5.1)$$

В предыдущих параграфах подробно описаны подходы к решению оптимизационных задач с использованием стандартных подходов, основанных на использовании аналитических методов. Однако в большинстве случаев приходится решать поставленную задачу (5.1) при заданных ограничениях численно. Это связано с тем, что целевые функции реальных оптимизируемых процессов и систем далеко не всегда имеют пригодный для аналитической работы с ними вид.

Работа любого алгоритма численного математического программирования (численной оптимизации) состоит из двух этапов. На первом вычисляются предусмотренные алгоритмом характеристики, на втором – по полученной информации строится приближение к решению.

Теоретические основы и рекомендации по практическому применению методов численного математического программирования приводятся в [3, 7, 9, 33, 42, 53, 66, 100, 104, 120, 135, 149].

Пример численного решения задачи многомерной оптимизации

Примером содержательной задачи является проблема поиска оптимальной длительности цикла светофорного регулирования и оптимального распределения её по фазам.

Различные исследования структуры и характеристик транспортного потока определили многочисленные критерии для его описания и анализа [36, 46, 51, 56, 61, 63, 65, 67, 68, 71, 77, 122, 126]. К основным показателям транспортного потока относят интенсивность движения, скорость, плотность потока. Также важными характеристиками, особенно для описания и анализа регулируемых пересечений городских магистралей, являются состав транспортного потока, степень насыщения группы движения. Особую важность также представляет пропускная способность полосы движения. Интенсивность движения (интенсивность транспортного потока) – это число транспортных средств, проезжающих через сечение дороги за единицу времени [65]. В качестве расчётного периода времени для определения интенсивности движения можно принять различные интервалы – год, месяц, сутки, час и более короткие промежутки времени (минуты, секунды) в зависимости от поставленной задачи наблюдения и средств измерения. Для характеристики количества автомобилей, которые дорога может пропустить, фактическую интенсивность движения обычно приводят к эквивалентному количеству легковых автомобилей. Для этого вводят коэффициенты приведения [71], характеризующие, сколько легковых автомобилей могло бы проехать по участку дороги за время проезда одного грузового автомобиля или автопоезда. На эти коэффициенты умножают число автомобилей каждого типа. Пропускная способность – максимальное число автомобилей, которое может пропустить участок в единицу времени в одном или двух направлениях в рассматриваемых дорожных и погодно-климатических условиях [140]. Степень насыщения – это отношение интенсивности обслуживания транспортных средств в очереди к интенсивности их поступления в очередь [140]:

$$x_i = \frac{v_i}{c_i},$$

где x_i – степень насыщения направления i , v_i – интенсивность обслуживания транспортных средств на направлении i , c_i – пропускная способность направления i .

В теории и практике регулирования уличного движения сложились два направления работ, отражающих структуру дорожно-транспортной сети городов, различаются которые изучаемым объектом. Это может быть либо отдельный перекрёсток, либо система перекрёстков. При решении задачи рассматривался первый тип объекта, а именно отдельный (изолированный) регулируемый перекрёсток. Регулируемый перекрёсток является изолированным, если процесс его функционирования и режим работы светофорной сигнализации не зависят от работы светофорной сигнализации на смежных перекрёстках [68].



Рис. 1. Пример программы светофорного регулирования

Для характеристики и основания для функционирования регулируемого перекрёстка существуют понятия такта, фазы и цикла светофорного регулирования. Тактом светофорного регулирования называется промежуток времени, в течение которого разрешено движение для одного из конкурирующих направлений (или запрещено для другого). Такты бывают основными и промежуточными.

Движение в промежуточные такты запрещено для всех транспортных средств, кроме заехавших на перекрёсток в период окончания основного такта. Совокупность основного такта и следующего за ним промежуточного называется фазой светофорного регулирования. Периодически повторяющаяся последовательность фаз называется циклом светофорного регулирования. Пример программы светофорного регулирования для двухфазного перекрёстка представлен на рис. 1.

Пофазный разъезд транспортных средств обеспечивает разделение конфликтующих потоков по времени. Число фаз, а, следовательно, и выделен-

ных групп транспортных и пешеходных потоков в соответствующих фазах зависит от характера конфликтных точек на перекрёстке и интенсивности движения в каждом направлении [41].

Все критерии оценки функционирования регулируемых пересечений (равно как и нерегулируемых, а также пешеходных потоков) делятся на количественные и качественные. Качественная оценка строится на отнесении объекта управления к одной из групп (в зависимости от различных его характеристик). В [140] выделяют 6 уровней качества функционирования регулируемого перекрёстка (на основании задержки светофорного регулирования).

Количественные оценки в основном подразумевают вычисление задержки светофорного регулирования. Транспортная задержка (задержка регулирования) является показателем, определяющим, какое количество времени транспортные средства должны, стоя на подходе к перекрёстку, ждать своего обслуживания. Задержка регулирования тесно связана с длиной очереди, интенсивностью движения, пропускной способностью, параметрами режима регулирования и влияет на такие показатели, как скорость сообщения, количество торможений и троганий в расчёте на единицу длины.

В качестве критерия оптимизации управления регулируемым перекрёстком получила большое распространение величина средней транспортной задержки регулирования [32, 44].

Модели задержек, основанные на статистических распределениях процессов прибытия и разъезда, называются «точными» моделями. Так как эти модели получены исключительно теоретическим путем, то для того, чтобы эти модели были адекватными, необходимо накладывать строгие ограничения, что существенно усложняет решение задачи.

В 1956 году Бекман (см., например, [63]) на основе биномиального процесса прибытия и детерминированного процесса обслуживания впервые определил ожидаемую задержку:

$$d = \frac{T - g}{T(1 - N/N_{C1})} \left[\frac{q_0}{N} + \frac{T - g + 1}{2} \right], \quad (5.2)$$

где d – ожидаемая задержка регулирования, с; T – длительность цикла регулирования, с; g – длительность разрешающего (зеленого) сигнала, с; N – количество транспортных средств в очереди на текущий момент; N_{C1} – количество убывших транспортных средств за предыдущий цикл; q_0 – ожидаемая остаточная очередь от предыдущего цикла.

Ожидаемая остаточная длина очереди и ограничения, накладываемые на биномиальный процесс прибытия, значительно уменьшают практическую значимость формулы (5.2).

В 1968 году Мак Нейл предложил формулу ожидаемой задержки транспортных средств на основе нормального процесса прибытия и постоянного времени разъезда. Согласно его работе, общая задержка за цикл складывается из двух составляющих:

$$d = d_1 + d_2, \quad (5.3)$$

где d_1 – общая задержка фазы горения красного сигнала светофора, с; d_2 – общая задержка фазы горения зеленого сигнала светофора, с.

Таким образом, формула для математического ожидания (5.3) имеет вид:

$$E(d) = \frac{T - g}{T(1 - N/N_{C1})} \left[\frac{q_0}{N} + \left(\frac{T - g}{2} \right) TN + \frac{1}{N_{C1}} \left(1 + \frac{I}{1 - N/N_{C1}} \right) \right],$$

где $E(d)$ – математическое ожидание d ; I – среднеквадратическое отклонение интенсивности прибытия транспортных средств.

Средняя задержка одного автомобиля d получается при делении математического ожидания $E(d)$ на среднее число прибывших транспортных средств за цикл:

$$d = \frac{T - g}{2T(1 - N/N_{C1})} \left[\frac{2q_0}{N} + (T - g) + \frac{1}{N_{C1}} \left(1 + \frac{I}{1 - N/N_{C1}} \right) \right]. \quad (5.4)$$

Для биномиального закона распределения $I = 1 - N/N_{C1}$ и выражение (5.4) становится идентичным выражению, полученному Бекманом (5.2).

Мак Нейл (см., например, [63]) в 1974 году, рассмотрев случай сложного пуассоновского процесса прибытия и нормального процесса обслуживания, получил следующую формулу:

$$d = \frac{T - g}{2T \left(1 - \frac{N}{N_{C1}}\right)} \left[(T - g) + \frac{2}{N} \left[1 + \frac{\left(1 - \frac{N}{N_{C1}}\right) (1 - B^2)}{2N_{C1}} \right] q_0 + \right. \\ \left. + \frac{1}{N_{C1}} \left(1 + \frac{I + B^2 \frac{N}{N_{C1}}}{1 - \frac{N}{N_{C1}}} \right) \right], \quad (5.5)$$

где B – среднеквадратическое отклонение интенсивности убывания транспортных средств.

В общем случае модель (5.5) требует данных о среднем размере остаточной очереди, что является главным ограничением на практическую возможность использования этой формулы.

В нашей стране основной формулой для расчёта средней задержки регулирования на подходе к регулируемому перекрестку была и остается формула, полученная Ф. Вебстером ещё в 1956 году.

$$d = \frac{T \left(1 - \frac{g}{T}\right)^2}{2 \left[1 - \left(\frac{g}{T}\right) X\right]} + \frac{X^2}{2v(1 - X)} - 0,65 \left(\frac{T}{v^2}\right)^{\frac{1}{2}} X^{2+5\frac{g}{T}}, \quad (5.6)$$

где d – средняя задержка за цикл регулирования, с; T – длительность цикла регулирования, с; g – длительность разрешающего (зеленого) сигнала, с; X – степень насыщения; v – интенсивность прибытия транспортных средств.

Современные модели расчёта задержки регулирования также построены из двух основных составляющих – стандартной (без очереди и предполагается постоянное прибытие) и дополнительной (вносит элемент случайности) задержек регулирования.

Согласно Canadian Capacity Guide for Signalized Intersections 2008 [137] задержка группы движения транспортных средств равна:

$$d = k^f d_1 + d_2, \quad (5.7)$$

где d – задержка за цикл регулирования, с; d_1 – стандартная задержка, с; d_2 – дополнительная задержка, с; k^f – коэффициент, показывающий качество регрессии транспортных средств.

При этом вычисление стандартной задержки модели (5.7) предполагает одинаковое прибытие транспортных средств к пересечению и отсутствие очереди в начале цикла регулирования:

$$d_1 = \frac{T \left(1 - \frac{g}{T}\right)^2}{2 \left(1 - \min \left\{1, X\right\} \frac{g}{T}\right)},$$

где T – длительность цикла регулирования, с; g – продолжительность разрешающего сигнала для данной фазы, с; X – степень насыщения – отношение интенсивности движения к пропускной способности.

Компонент дополнительной задержки в модели (5.7) учитывает непостоянство прибытия автомобилей, а также случайное появление перенасыщения в некоторых циклах:

$$d_2 = 15t_e \left((X - 1) + \sqrt{(X - 1)^2 + 240 \frac{X}{ct_e}} \right),$$

где c – пропускная способность, t_e – длительность периода оценки, мин.

Американское руководство по планированию дорог Highway Capacity Manual 2010 [140] определяет задержку регулирования следующим образом:

$$d = d_1 + d_2 + d_3. \quad (5.8)$$

При этом стандартная задержка d_1 модели (5.8) вычисляется также, как и в Canadian Capacity Guide for Signalized Intersections 2008 (5.7):

$$d_1 = \frac{T \left(1 - \frac{g}{T}\right)^2}{2 \left(1 - \min \left\{1, X\right\} \frac{g}{T}\right)},$$

где T – длительность цикла регулирования, с; g – продолжительность разрешающего сигнала для данной фазы, с; X – степень насыщения – отношение интенсивности движения к пропускной способности.

Дополнительная задержка d_2 модели (5.8) учитывает наличие адаптивного регулирования и степень изолированности перекрёстка (то есть удалённость его от соседних регулируемых перекрёстков):

$$d_2 = 900t_e \left((X - 1) + \sqrt{(X - 1)^2 + \frac{8kIX}{ct_e}} \right),$$

где t_e – длительность оцениваемого периода, мин.; c – пропускная способность; k – коэффициент, показывающий наличие адаптивного регулирования; I – коэффициент учёта удалённости от соседнего регулируемого перекрёстка.

Остаточная задержка d_3 модели (5.8) учитывает задержку с предыдущего цикла.

Приведём алгоритм формирования функции средней задержки на перекрёстке [115, 140].

Шаг 1. Определить критическую группу движения, то есть такую группу в каждой фазе, у которой отношение интенсивности движения к пропускной способности соответствующего направления максимально.

Шаг 2. Построить для выбранной группы функцию транспортной задержки по формуле

$$d = k_f \cdot d_1 + d_2 + d_3,$$

где d – транспортная задержка критической группы в направлении, то есть максимальная задержка в данном направлении, d_1 – стандартная задержка критической группы, d_2 – дополнительная задержка, учитывающая случайный характер прибытия транспортных средств, d_3 – остаточная задержка, k_f – поправочный коэффициент, показывающий качество прогрессии.

Шаг 3. Произвести агрегирование задержки для всего перекрестка по критическим группам:

$$D = \frac{\sum_{i=1}^m (d)_i \cdot v_i}{\sum_{i=1}^m v_i}.$$

Задача оптимального регулирования дорожного движения не является новой. Минимизация величины средней транспортной задержки проводилась различными специалистами этой области. В [106] решается задача оптимизации цикла работы светофора методами линейного программирования, в [63] решается задача нелинейного программирования (определения оптимальной длительности цикла светофорного регулирования и оптимального распределения продолжительностей горения основных тактов) с помощью современного статистического программного обеспечения. В [44] предлагается метод статистической оптимизации циклов светофорного регулирования, основанный не на классически используемых коэффициентах приведения состава транспортного потока к условным автомобилям, а на учёте тяговых свойств транспортных средств. В [32] рассматриваются алгоритмы нечёткой логики для управления транспортными потоками на перекрёстках, а также предлагается использовать изменение дорожных транспортных знаков для перераспределения транспортных потоков на перекрёстке со временем.

Итак, решается оптимизационная задача [29, 116–118, 148]: найти минимум агрегированной задержки на регулируемом перекрёстке, то есть найти оптимальное время цикла и его оптимальное распределение по фазам регулирования.

В строгой постановке: найти точку (T, g_1, \dots, g_m) , такую, что

$$f(T, g_1, \dots, g_m) = \min_X \frac{\sum_{i=1}^m f_i(T, g_i) \cdot v_i}{\sum_{i=1}^m v_i} \quad (5.9)$$

при системе ограничений

$$X = \begin{cases} \sum_{i=1}^m g_i = T, \\ T_{\min} \leq T \leq T_{\max}, \\ (g_i)_{\min} \leq g_i \leq (g_i)_{\max}, \quad i = 1, \dots, m, \\ g_1, \dots, g_m > 0. \end{cases} \quad (5.10)$$

Здесь

$$f_i(T, g_i) = k_f^i \frac{T \left(1 - \frac{g_i}{T}\right)^2}{2 \left(1 - \min\{1, X_i\} \frac{g_i}{T}\right)} +$$

$$+ 15t_e \left((X_i - 1) + \sqrt{(X_i - 1)^2 + 240 \frac{X_i}{t_e \cdot \frac{s_i g_i}{T}}} \right),$$

$$X_i = \frac{v_i T}{s_i g_i},$$

где T – время цикла светофорного регулирования (с); g_i – длительность фаз светофорного регулирования (с); $f(T, g_1, \dots, g_m)$ – значение функции агрегированной задержки (с); k_f^i – коэффициент, определяющий тип прибытия транспортных средств; v_i – интенсивность прибытия транспортных средств на направлении i (прив.авт./ час), s_i – поток насыщения направления i (прив.авт./час); $T_{\min(\max)}$ – ограничения на длительность времени цикла светофорного регулирования (с); $(g_i)_{\min(\max)}$, $i = 1, \dots, m$ – ограничения на длительности фаз светофорного регулирования (с); t_e – прогнозное время вычисления задержки (мин).

Учитывая систему вторых и третьих компонентов системы ограничений (5.10), получаем множество D , на котором будет происходить минимизация.

Задача сводится к задаче условной минимизации с ограничением-равенством. Для перехода к задаче безусловной минимизации построим квадра-

тичную штрафную функцию вида

$$\begin{aligned}\varphi(g, T, \beta) = & \beta \left(\left(\sum_{i=1}^m g_i \right) - T \right)^2 + \sum_{i=1}^m (g_i - (g_i)_{\min})^2 [1 - \operatorname{sgn}(g_i - (g_i)_{\min})] + \\ & + \sum_{i=1}^m ((g_i)_{\max} - g_i)^2 [1 - \operatorname{sgn}((g_i)_{\max} - g_i)] + \\ & + (T - T_{\min})^2 [1 - \operatorname{sgn}(T - T_{\min})] + (T_{\max} - T)^2 [1 - \operatorname{sgn}(T_{\max} - T)],\end{aligned}$$

которая будет стремительно увеличиваться за пределами множества D и обращаться в ноль внутри его.

В результате задача (5.9) принимает вид задачи многомерной безусловной минимизации: найти

$$\min_{x \in \mathbb{R}^{m+1}} \tilde{F} = \min_{x \in \mathbb{R}^{m+1}} F + \varphi(g, T, \beta). \quad (5.11)$$

Примером решения полученной задачи (5.11) в контексте численного математического программирования является синтезируемый далее мульти-стартовой алгоритм безусловной многомерной оптимизации.

Шаг 0. Настройка параметров алгоритма. На этом шаге необходимо задать:

n – число моделируемых равномерно распределенных точек на области оптимизации; $\Delta\beta$ – приращение параметра штрафной функции ($\Delta\beta = \lg(k)$); $\varepsilon_1 > 0$ – точность шага алгоритма оптимизации; $\varepsilon_2 > 0$ – точность алгоритма; $\delta > 0$ – параметр кластеризации.

Шаг 1. В качестве начального значения для множителя штрафной функции положим $\beta = 0$. Тем самым будет найден глобальный оптимум исходной функции без учёта ограничения-равенства.

Шаг 2. Для заданной функции без ограничений необходимо смоделировать на множестве D n равномерно распределённых в \mathbb{R}^{m+1} точек x_1, \dots, x_n .

Шаг 3. Минимизация штрафной функции для конкретного β .

Шаг 3.1. Используя полученные точки в качестве начальных, провести одну или несколько итераций алгоритма статистического градиента с целью

улучшения рабочих точек. В результате получены точки z_1, \dots, z_n (для первой итерации используются точки шага 2, для последующих – точки шага 3.4).

Шаг 3.2. Применить кластеризацию к полученным точкам z_1, \dots, z_n .

Шаг 3.2.1. Пусть все точки принадлежат различным кластерам.

Шаг 3.2.2. Найти ближайшую пару z_i, z_j , такую, что

$$\rho_{ij} = \arg \min_{k,l=1,\dots,N} \rho_{k,l}.$$

Шаг 3.2.3. Если расстояние между ближайшими соседями не превосходит некоторое достаточно малое число $\delta > 0$, то z_i, z_j объединяются в один кластер, и, тем самым, число кластеров сокращается на единицу.

Пусть получено m кластеров. Если $m = 1$, переход к шагу 3.4, иначе – к шагу 3.3.

Шаг 3.3. Выбрать наиболее перспективных представителей от всех кластеров (то есть точку из каждого кластера, доставляющую наилучшее значение целевой функции). Положить $N = m$ и перейти к шагу 3.1.

Шаг 3.4. Положение – в окрестности глобального минимума. Выбрать представителя единственного кластера и использовать его в качестве начальной точки для алгоритма локальной оптимизации (например, алгоритма статистического градиента).

Для окончания шага безусловной оптимизации используется критерий

$$\left| \tilde{f}(z^k, \beta^k) - \tilde{f}(z^{k-1}, \beta^k) \right| \leq \varepsilon_1. \quad (5.12)$$

Если условие (5.12) выполнено, переход на шаг 4.

Шаг 4. Проверка условия окончания алгоритма:

$$\left| \tilde{f}(z^{k+1}, \beta^k) - \tilde{f}(z^k, \beta^{k-1}) \right| \leq \varepsilon_2. \quad (5.13)$$

Если условие (5.13) выполнено – положить $x^* = z^{k+1}$, в противном случае – переход к шагу 5.

Шаг 5. Пересчёт (увеличение) параметра β штрафной функции по заданному правилу: $\beta^{k+1} = \beta^k + \Delta\beta$. Переход к шагу 2.

В приведённом алгоритме в качестве метода локальной оптимизации использован алгоритм статистического градиента, схема которого приведена далее.

Шаг 1. Выбрать начальную точку $x_0 = (x_1, \dots, x_n) \in \mathbb{R}^{m+1}$, пробный шаговый множитель $\gamma > 0$, рабочий шаговый множитель $\rho > 0$, число пробных шагов $p \geq 1$. Положить $t = 0$.

Шаг 2. Вычислить p случайных независимых реализаций $\xi^{t,i}, i = 1, \dots, p$ единичного вектора ξ^t равномерно распределённого по всем направлениям в \mathbb{R}^{m+1} .

Шаг 3. Вычислить приращение функции \tilde{F} :

$$\Delta_i^t = \tilde{F}(x^t + \gamma \xi^{t,i}) - \tilde{F}(x^t), \quad i = 1, \dots, p.$$

Шаг 4. Вычислить вектор

$$\text{grad}^t = \sum_{i=1}^p (\xi^{t,i} \Delta_i^t).$$

Шаг 5. Вычислить вектор движения h^t к следующему приближению x^{t+1} по формуле:

$$h^t = -\frac{\text{grad}^t}{\|\text{grad}^t\|}.$$

Шаг 6. Вычислить следующее приближение по формуле:

$$x^{t+1} = x^t + \rho h^t.$$

Шаг 7. Полагаем $t = t + 1$ и переходим к шагу 2.

Заметим, что при $p \rightarrow \infty, \gamma \rightarrow 0$, grad^t – статистическая оценка градиента функции \tilde{F} в точке x^t .

Произведённый вычислительный эксперимент, направленный на поиск значения параметра кластеризации δ , показал, что приемлемым для рассматриваемой содержательной задачи оптимизации является значение 10, так как при этом значении достигается необходимый уровень точности вычисления

значения функции, но всё же число шагов, необходимых для поиска параметров, возрастает по сравнению со значениями, меньшими 10 (табл. 1).

Таблица 1

Результаты поиска приемлемого параметра кластеризации для алгоритма оптимизации

Значение параметра кластеризации δ	Число шагов алгоритма	Точность
1	12	0,00087
2	14	0,00062
3	15	0,00054
4	19	0,00046
5	21	0,00033
6	23	0,00027
7	32	0,00019
8	33	0,00017
9	36	0,00005
10	39	0

В качестве объекта исследования взят регулируемый перекрёсток, который находится в центральной части города Липецка. Образующие его улицы являются магистральными дорогами общегородского значения, которые обеспечивают транспортную связь в пределах планировочных районов, а также между общественными центрами города и магистральными улицами. На пересечении проспекта Победы и улицы Механизаторов введено жёсткое программное светофорное регулирование, представляющее собой четырёхфазный цикл работы системы светофоров. Схема организации движения транспортных потоков приведена на рис. 2, схемы пофазного разъезда транспортных средств на перекрёстке пр. Победы и ул. Механизаторов приведены на рис. 3. Данные по действующему режиму светофорной сигнализации и интенсивности приведены в табл. 2 и 3 соответственно.

В табл. 4 представлены результаты применения описанного алгоритма для решения задачи оптимизации функционирования регулируемого перекрёстка. Показано, что оптимизация дала 30,1% экономии времени агрегированной задержки по сравнению с существующим режимом.

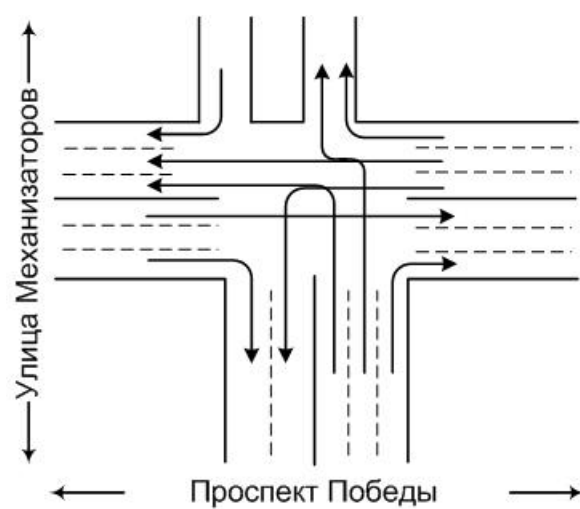


Рис. 2. Организация движения транспортных потоков на пересечении магистралей

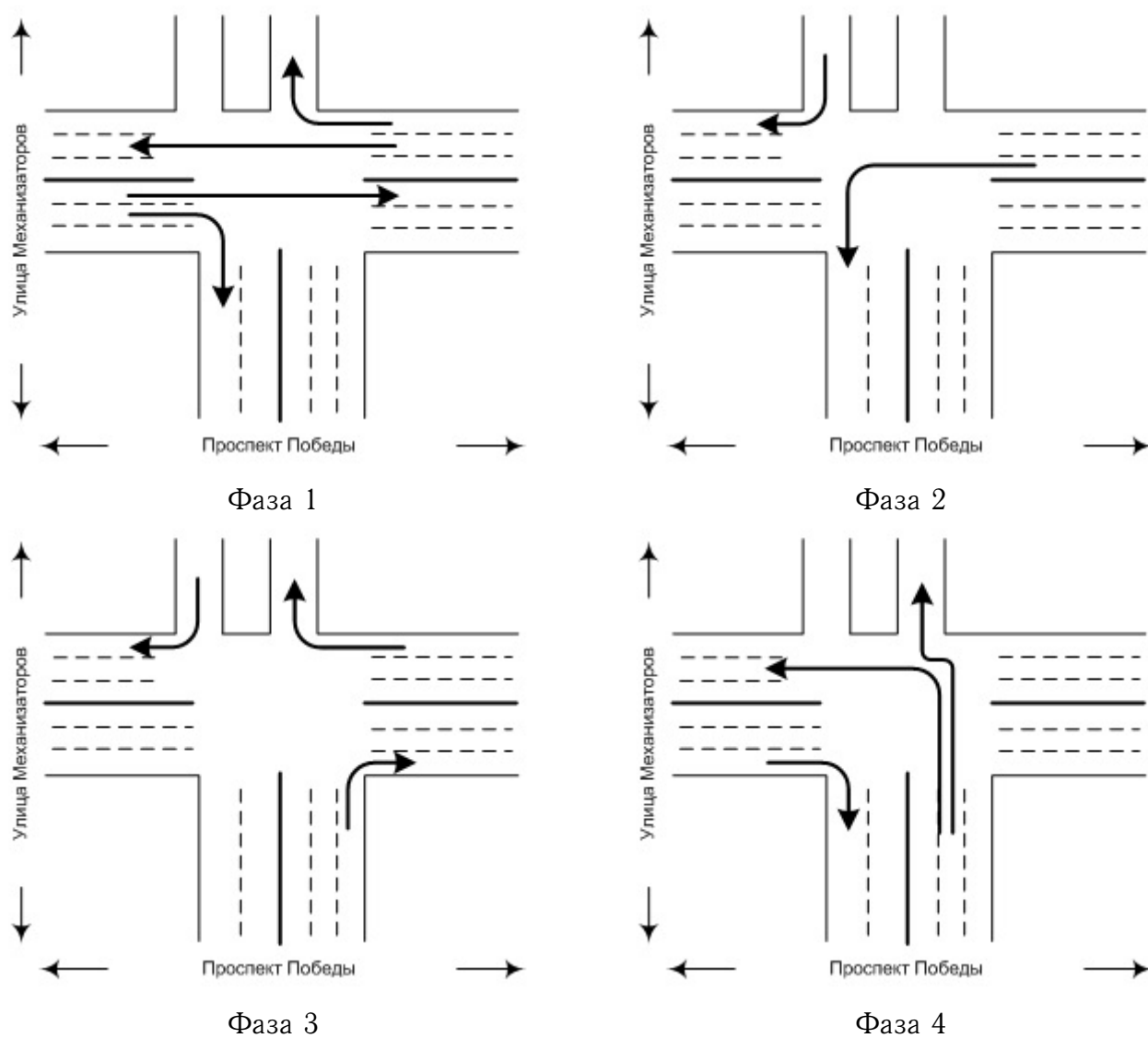


Рис. 3. Фазы цикла работы регулировочных светофоров при четырёхфазной системе

Таблица 2

График светофорной сигнализации на пересечении

Номер фазы	Длительность зелёного сигнала, с	Длительность жёлтого сигнала, с
1	38	3
2	16	3
3	8	3
4	23	3

Таблица 3

Интенсивность движения транспортных потоков на пересечении

№ направления	Интенсивность, ед./час	Проп. способность, ед./час
1	0	0
2	1533	705
3	549	1131
4	1086	426
5	326	426
6	59	148
7	78	296
8	871	705
9	95	853
10	0	0
11	0	0
12	242	444

Таблица 4

Результаты оптимизации процесса функционирования пересечения городских магистралей

		Фаза 1	Фаза 2	Фаза 3	Фаза 4
Интенсивность движения в критической группе, прив.авт./ч		1533	242	242	1086
Поток насыщения в критической группе, прив.авт./ч		1800	1800	1800	1800
Длительность фаз, с	до	38	16	8	23
	после	31,5	9,8	9,9	21,5
v/c-отношение	до	1,905	0,714	1,428	2,229
	после	1,966	0,997	0,987	2,040
Задержка, с	до	212,2	54,75	14,97	647,52
	после	207,09	48,87	11,63	367,28

ГЛАВА II

Алгоритмы псевдообращения

Обобщением понятия обратной матрицы A^{-1} , однозначно определённой для квадратной невырожденной матрицы $A \in \mathbb{R}^{n \times n}$, является понятие *псевдообратной матрицы* A^+ , которая существует и единственна для любой матрицы $A \in \mathbb{R}_r^{m \times n}$ с произвольным числом строк m , столбцов n и произвольного ранга r . Стоит отметить, что псевдообратная совпадает с обратной для квадратной невырожденной матрицы.

Аппарат псевдообращения широко применяется при точном или приближённом исследовании и решении систем линейных алгебраических уравнений и приводящихся к ним разнообразных прикладных задач из широкого спектра областей, таких как математическое программирование, математическая статистика, математическое моделирование систем и управление ими и другие. В качестве конкретных примеров можно указать задачи линейного программирования, оценивания сигналов и идентификации систем, синтеза алгоритмов управления. Применения к нелинейным же задачам опираются на схемы линеаризации.

Среди множества существующей литературы по псевдообращению можно отметить ряд отечественных [30, 38, 40, 72, 102, 128], переводных [2, 70, 73, 103, 112] и зарубежных [129, 146, 150] монографий, обзоров и методических статей, а также учебников, пособий или справочников [8, 35, 37, 43, 59], в которых с большей или меньшей степенью подробности или доступности для начинающего приводятся те или иные разнообразные свойства, методы вычисления и приложения псевдообратных матриц (этот перечень, возможно, не полон; из зарубежных указана одна из первых подробных монографий [146] и два обзора [129, 150], содержащих обширную библиографию).

В 1920 году Мур [142] опубликовал предложенное им обобщение понятия обратной на вырожденные и вообще прямоугольные матрицы; как указано в [129], первые сообщения об этом понятии содержались в лекциях Мура 1903-1906 годов, а истоки понятия прослеживаются в ранних работах Фредгольма, Гильберта, Гурвица и других по интегральным и дифференциальным операторам («псевдообратные преобразования», «обобщённая функция Гоига», «псевдорезольвента ядра интегрального уравнения» и др.).

В 1955 году Пенроуз [143], не зная о работе Мура (и это подчёркивается в [129, 146, 150]), переоткрыл введённое им понятие и показал, что такая обобщённая обратная матрица G к произвольной матрице A однозначно описывается системой соотношений, известных как определяющие *соотношения Мура-Пенроуза*:

$$1) (AG)^T = AG,$$

$$2) (GA)^T = GA,$$

$$3) AGA = A,$$

$$4) GAG = G.$$

В том же 1955 году Рао [150] построил обобщённую обратную к вырожденной матрице, возникающей при решении (переопределённой) системы нормальных уравнений методом наименьших квадратов. Учёный назвал её псевдообратной и показал, что она может служить той же цели, что и обычная обратная к невырожденной матрице. И хотя оказалось, что для матриц, использованных Рао, достаточно выполнения только одного из соотношений, $AGA = A$, они были названы позднее обобщёнными обратными к матрице A , а название «псевдообратная к A » закрепилось за однозначно описываемой всеми четырьмя определяющими соотношениями Мура-Пенроуза матрицей A^+ .

Первый рекуррентный алгоритм псевдообращения предложен Гревилем (см., например [2, 40]). Специалистами, изучавшими вычислительные аспекты псевдообращения, было признано [139], что алгоритм Гревия обладает преимуществом по сравнению с другими как вычислительный метод, наиболее подходящий для компьютерной реализации; он не требует обращения и

факторизации матриц и других сложных матричных операций. Использование только элементарных действий над векторами и матрицами, изучаемых в стандартном курсе линейной алгебры, делает этот метод предпочтительным и в учебных целях. В [2] приведены результаты, показывающие, как на базе известного (см., например [40]) метода Фаддеева одновременного вычисления скалярных коэффициентов характеристического многочлена и матричных коэффициентов присоединенной характеристической матрицы может быть построен другой рекуррентный алгоритм псевдообращения. Эти два алгоритма приняты в качестве основных для рассмотрения далее.

Важным аспектом вычислений вообще и псевдообращения в частности является их точность или безошибочность [45]. При компьютерной реализации безошибочность вычислений тесно связана с их проведением в целочисленной арифметике, когда операция фактического деления исключается из расчётов вплоть до их заключительного шага. Иначе говоря, вычислительные процедуры строятся (или уже имеющиеся перестраиваются) так, что для всех возникающих в процессе счёта дробей числители и знаменатели вычисляются отдельно. Применительно к псевдообращению это означает, в частности, что для целочисленной исходной матрицы, когда псевдообратная, как и обычная обратная, рациональна, возникает проблема отдельного вычисления знаменателя и числителя этой дроби. В случае квадратной невырожденной матрицы A это – её определитель $\det(A)$ и присоединённая (здесь и далее обозначаемая $\langle A \rangle$), так что проблема решается известной формулой

$$A^{-1} = \frac{\langle A \rangle}{\det(A)}.$$

Следует подчеркнуть, что такая дробь может допускать сокращение, так что рациональное представление A^{-1} (как, впрочем, и любого рационального числа) не единственно.

В случае целочисленной матрицы $A \in \mathbb{Z}_r^{m \times n}$ с произвольным числом строк и столбцов и произвольного ранга представления такого рода приводят к понятиям о псевдоопределителе $\det^+(A)$ и псевдоприсоединённой $\langle A \rangle^+$ к

матрице A , с использованием которых можно записать:

$$A^+ = \frac{\langle A \rangle^+}{\det^+(A)}.$$

Вопрос о представлении псевдообратной в виде дроби рассматривался ещё в работе Мура [142] (детерминантное представление; дальнейшие ссылки содержатся в [129]); рациональное представление псевдообратной к целочисленной матрице исследуется в работах [141, 147], где получены выражения для псевдоопределителей и псевдоприсоединённых.

Существенным для целочисленных процедур является необходимость предусматривать возможность гашения быстрого роста разрядности чисел, характерного для целочисленной арифметики и приводящего к переполнению памяти. Одна из реализаций этой возможности – включение в вычислительные алгоритмы операции деления нацело, обозначаемая здесь через «div», обоснование справедливости которой в ряде случаев может представить известные аналитические трудности. Примером такого рода может служить работа [75], где для решения целочисленной определённой системы линейных уравнений установлены целочисленные алгоритмы Гаусса и Крамера, включающие операцию деления нацело и позволяющие находить точные рациональные решения, причём фактическое деление выполняется лишь на заключительном шаге. Алгоритм Фаддеева допускает непосредственную переформулировку с заменой «обычного» деления делением нацело, а для алгоритма Гревия возможность такой замены обоснована в [15], где, кроме того, этот алгоритм модифицирован в направлении, делающем его удобным не только для решения, но и для полного исследования целочисленных систем линейных уравнений общего вида, точного определения рангов основной и расширенной матрицы таких систем.

Системы линейных уравнений – одна из основных областей применения псевдообратных матриц. Для системы $Ax = b$ критерий совместности в терминах псевдообратных формулируется в виде $AA^+b = b$.

При его выполнении – решение системы, а при нарушении – псевдорешение в смысле метода наименьших квадратов (минимизирующее обычную

евклидову норму невязки $Ax - b$) записывается в терминах псевдообратных в виде $x = A^+b + (I - A^+A)y$, y – произвольный вектор.

Совместная система *определённая*, если $A^+A = I$. Нормальное решение или псевдорешение (имеющее минимум обычной евклидовой нормы) определяется первым слагаемым $x_n = A^+b$.

Используется псевдообращение и для решения систем неравенств [34]; в [121] рассматривается точное решение целочисленных неравенств.

Применение псевдообращения к исследованию и решению вырожденных систем линейных обыкновенных дифференциальных уравнений $E\dot{x} = Ax + Bu$, где E – некоторая матрица общего вида, – рассмотрено в работах [30, 125, 136], а к оцениванию сигналов в таких системах – в [138].

По-видимому, первое упоминание о применении псевдообращения в задачах синтеза алгоритмов автоматического управления динамическими системами следует отнести к 1963 году [64]. Подобные применения проще всего охарактеризовать на примере метода синтеза алгоритмов управления движением динамических систем по назначенным траекториям, состоящего в решении уравнения системы относительно управляющих воздействий; постановка задачи соответствует концепции обратных задач динамики в смысле [69], а метод решения с использованием псевдообращения рассмотрен в [12]. Если для системы $\dot{x} = Ax + Bu$ назначена траектория движения x^* , то управление u_* , реализующее движение по ней, находится из уравнения $Bu_* = \dot{x}^* - Ax^*$.

Критерий совместности этого уравнения приводит к критерию осуществимости назначаемой траектории $BB^+(\dot{x}^* - Ax^*) = \dot{x}^* - Ax^*$ или $E\dot{x}^* = EAx^*$, $E = I - BB^+$ – уравнению осуществимых траекторий, которое относится к типу вырожденных однородных линейных дифференциальных уравнений; систематически проводимый с использованием псевдообращения анализ этого уравнения приводит к понятию об ортогональной канонической форме динамической системы [13]. Алгоритм управления, доставляющий решение или псевдорешение такой задачи, синтезируется в виде $u_* = B^+(\dot{x}^* - Ax^*) + (I - B^+B)\nu$, где ν – произвольное воздействие.

Описанный подход предпринят в [14] применительно к динамической транспортной задаче, трактуемой как система управления запасами; построены критерий осуществимости назначаемых траекторий, ортогональная каноническая форма, алгоритм управления; псевдообращение матрицы условий такой задачи подробно проведено в [78].

Данный метод использован в [6] применительно к дискретным распределенным системам (системам с многомерным параметром), где с использованием псевдообращения сформулированы критерии управляемости и наблюдаемости, построены рекуррентные алгоритмы оптимального управления и оптимальной оценки состояний. На широкий класс дискретно-аргументных систем [11] этот подход распространен в [10], где построены критерии осуществимости назначаемых траекторий для различных подклассов дискретно-аргументных систем и, в предположении их выполнимости, синтезированы алгоритмы управления, дающие решение задачи. Нормальные псевдорешения такой задачи построены в [107].

При практическом применении псевдообращения следует иметь в виду, что, как было обнаружено вскоре после создания этого метода, он может проявлять неустойчивость по отношению к ошибкам в исходных данных (см., например, [40, 141]). Одним из путей преодоления этой трудности служит метод регуляризации некорректно поставленных задач, предложенный Тихоновым [119]. Дальнейшему развитию методов регуляризации неустойчивых задач и разработке вычислительных алгоритмов посвящена работа [82].

§ 6. Определение псевдообратной матрицы

Пусть $A \in \mathbb{R}_r^{m \times n}$ – матрица размера $m \times n$ (m строк и n столбцов) ранга r над полем действительных чисел \mathbb{R} , $A^T \in \mathbb{R}_r^{n \times m}$ – транспонированная матрица.

Псевдообратная матрица (обобщённая обратная по Муру-Пенроузу) $A^+ \in \mathbb{R}_r^{n \times m}$ к матрице A определяется соотношениями:

$$(AA^+)^T = AA^+, \quad (\text{I})$$

$$(A^+A)^T = A^+A, \quad (\text{II})$$

$$AA^+A = A, \quad (\text{III})$$

$$A^+AA^+ = A^+. \quad (\text{IV})$$

Для любой прямоугольной матрицы A произвольного ранга существует единственная псевдообратная матрица A^+ .

Непосредственно по определению проверяется, что:

1. Если $A = \alpha \in \mathbb{R}$ — скаляр, то

$$\alpha^+ = \begin{cases} 0, & \alpha = 0, \\ \alpha^{-1}, & \alpha \neq 0. \end{cases} \quad (6.1)$$

2. Если $A = 0 \in \mathbb{R}_0^{m \times n}$ — нулевая матрица, то

$$0^+ = 0^T \in \mathbb{R}_0^{n \times m}. \quad (6.2)$$

3. Если $A = \text{diag} \{ \alpha_1, \dots, \alpha_n \} \in \mathbb{R}_r^{n \times n}$ — диагональная матрица (среди чисел α_i имеется $n - r$ нулей), то

$$A^+ = \text{diag} \{ \alpha_1^+, \dots, \alpha_n^+ \}. \quad (6.3)$$

4. Если $A = b \in \mathbb{R}_1^{m \times 1}$ — ненулевая матрица-столбец, то

$$b^+ = (b^T b)^{-1} b^T, \quad b^+ b = 1, \quad (6.4)$$

где $|b|^2 = b^T b = \sum_{i=1}^m (b_i)^2$.

5. Если $A = c^T \in \mathbb{R}_1^{1 \times n}$ — ненулевая матрица-строка, то

$$(c^T)^+ = c(c^T c)^{-1}, \quad c c^+ = 1. \quad (6.5)$$

6. Если $A = b c^T \in \mathbb{R}_1^{m \times n}$ — диадная матрица (произведение столбца на строку), то

$$A^+ = (c^T)^+ b^+ = \frac{A^T}{|b|^2 |c|^2}. \quad (6.6)$$

7. Если $A \in \mathbb{R}_n^{n \times n}$ — невырожденная матрица, то

$$A^+ = A^{-1}. \quad (6.7)$$

8. Если $B \in \mathbb{R}_r^{n \times r}$ — матрица полного столбцового ранга, то

$$B^+ = (B^T B)^{-1} B^T, \quad B^+ B = I_r, \quad (6.8)$$

I_r — единичная матрица размера $r \times r$.

9. Если $C \in \mathbb{R}_r^{r \times n}$ — матрица полного строчного ранга, то

$$C^+ = C^T (C C^T)^{-1}, \quad C C^+ = I_r. \quad (6.9)$$

10. Если $A = BC \in \mathbb{R}_r^{m \times n}$, $B \in \mathbb{R}_r^{m \times r}$, $C \in \mathbb{R}_r^{r \times n}$ (скелетное разложение матрицы A), то

$$A^+ = C^+ B^+ = C^T (C C^T)^{-1} (B^T B)^{-1} B^T = C^T (B^T A C^T)^{-1} B^T. \quad (6.10)$$

11. Если $A = \begin{pmatrix} A_1 & : & 0 \end{pmatrix}$ — блочная матрица, то

$$A^+ = \begin{pmatrix} A_1^+ \\ \dots \\ 0 \end{pmatrix}. \quad (6.11)$$

12. Если $A = \begin{pmatrix} A_1 & : & 0 \\ \dots & \dots & \dots \\ 0 & : & A_2 \end{pmatrix}$, то

$$A^+ = \begin{pmatrix} A_1^+ & : & 0 \\ \dots & \dots & \dots \\ 0 & : & A_2^+ \end{pmatrix}. \quad (6.12)$$

13. Если $A = \begin{pmatrix} A_1 \\ \dots \\ A_2 \end{pmatrix}$, $A_1 \in \mathbb{R}_r^{r \times r}$, $\text{rg} A = \text{rg} A_1 = r$, то

$$A^+ = (A_1^T A_1 + A_2^T A_2)^{-1} \begin{pmatrix} A_1^T & : & A_2^T \end{pmatrix}. \quad (6.13)$$

14. Если $A = \begin{pmatrix} A_1 & : & A_2 \end{pmatrix}$, $A_1 \in \mathbb{R}_r^{r \times r}$, $\text{rg} A = \text{rg} A_1 = r$, то

$$A^+ = \begin{pmatrix} A_1^T \\ \dots \\ A_2^T \end{pmatrix} (A_1 A_1^T + A_2 A_2^T)^{-1}. \quad (6.14)$$

15. Если $A = \begin{pmatrix} A_{11} & : & A_{12} \\ \dots & \dots & \dots \\ A_{21} & : & A_{22} \end{pmatrix}$, $A_{11} \in \mathbb{R}_r^{r \times r}$, $\text{rg} A = \text{rg} A_{11} = r$, то

$$A_{22} = A_{21} A_{11}^{-1} A_{12}, \quad A = \begin{pmatrix} A_{11} \\ \dots \\ A_{21} \end{pmatrix} A_{11}^{-1} \begin{pmatrix} A_{11} & : & A_{12} \end{pmatrix}, \quad (6.15)$$

$$A^+ = \begin{pmatrix} A_{11} & : & A_{12} \end{pmatrix}^+ A_{11} \begin{pmatrix} A_{11} \\ \dots \\ A_{21} \end{pmatrix}^+ =$$

$$= \begin{pmatrix} A_{11}^T \\ \dots \\ A_{21}^T \end{pmatrix} (A_{11} A_{11}^T + A_{12} A_{12}^T)^{-1} A_{11} (A_{11}^T A_{11} + A_{21}^T A_{21})^{-1} \begin{pmatrix} A_{11}^T & : & A_{21}^T \end{pmatrix}.$$

Соотношения (6.1-6.3) очевидны. Для обоснования (6.4) проверим определяющие соотношения:

$$(bb^+)^T = \left(\frac{bb^T}{|b|^2} \right)^T = \frac{bb^T}{|b|^2} = bb^+,$$

$$(b^+b)^T = 1 = b^+b, \quad bb^+b = b \cdot 1, \quad b^+bb^+ = 1 \cdot b^+.$$

Аналогично обосновываются (6.5) и (6.6); (6.7) очевидно. Для обоснования (6.8) проверим определяющие соотношения:

$$(BB^+)^T = (B(B^T B)^{-1} B^T)^T = B(B^T B)^{-1} B^T = BB^+,$$

$$(B^+ B)^T = ((B^T B)^{-1} B^T B)^T = I_r = B^+ B,$$

$$BB^+ B = BI_r, \quad B^+ BB^+ = I_r B^+.$$

Аналогично обосновываются (6.9) и (6.10); (6.11) и (6.12) очевидны. Соотношения (6.13) и (6.14) непосредственно следуют из (6.8) и (6.9). Для обоснования (6.15) заметим, что в силу сделанных предположений, соотношение

$$A = \begin{pmatrix} A_{11} \\ \dots \\ A_{21} \end{pmatrix} A_{11}^{-1} (A_{11} \quad A_{12})$$

является скелетным разложением, а потому результат следует из (6.10), (6.13) и (6.14).

§ 7. Свойства псевдообратных матриц

1. $(A^+)^+ = A$

Здесь и далее доказательства свойств псевдообратных матриц опираются на непосредственную проверку определяющих соотношений (I-IV) Мура-Пенроуза или на использование скелетного разложения (6.10).

Положим $A^+ = B$; тогда определяющие соотношения для A^+ как псевдообратной к A определяет A как псевдообратную к B :

$$(AB)^T = AB, \quad (BA)^T = BA, \quad ABA = A, \quad BAB = B.$$

Тем самым $A = B^+ = (A^+)^+$. \square

2. $(\alpha A)^+ = \alpha^+ A^+$

Если $\alpha = 0$, то $\alpha A = 0$, $\alpha^+ = 0$, $0^+ = 0$ в силу (6.1)-(6.2), поэтому $(\alpha A)^+ = 0^+ = 0 = 0 \cdot A^+ = \alpha^+ A^+$. Если $\alpha \neq 0$, то с учётом $\alpha^+ = \alpha^{-1}$ (6.1),

определяющие соотношения для $(\alpha A)^+$ снова являются переформулировкой определяющих соотношений для A^+ . \square

$$\mathbf{3.} \quad (A^T)^+ = (A^+)^T$$

Проверяя определяющие соотношения для $(A^T)^+$, убеждаемся, что все они получаются транспонированием определяющих соотношений для A^+ , например:

$$(A^+A)^T = A^+A \Rightarrow (A^T(A^+)^T)^T = A^T(A^+)^T,$$

$$AA^+A = A \Rightarrow A^T(A^+)^TA^T = A^T,$$

откуда и заключаем, что $(A^T)^+ = (A^+)^T$. \square

$$\mathbf{4.} \quad A^T = A \Rightarrow (A^+)^T = A^+, \quad AA^+ = A^+A$$

В силу предыдущего свойства $(A^+)^T = (A^T)^+ = A^+$, если $A^T = A$. \square

$$\mathbf{5.} \quad A^T = A, \quad A \geq 0 \Rightarrow A^+ \geq 0$$

Применяя определение неотрицательно определенной матрицы A при $x = A^+y$ и произвольном y , свойство 4 и определяющее соотношение (IV), получим

$$0 \leq x^T Ax = y^T (A^+)^T AA^+ y = y^T A^+ AA^+ y = y^T A^+ y,$$

что и означает неотрицательную определенность A^+ . \square

$$\mathbf{6.} \quad A^T = A, \quad A \geq 0, \quad A = C^T C \Rightarrow A^+ = C^T (CC^T)^{-2} C$$

При доказательстве данного свойства имеется в виду, что $A \in \mathbb{R}_r^{n \times n}$ и $r < n$ (иначе A невырождена и $A^+ = A^{-1}$); $C \in \mathbb{R}_r^{r \times n}$ – любая матрица, удовлетворяющая разложению $A = C^T C$; в этом случае $CC^T \in \mathbb{R}_r^{r \times r}$ невырождена, а разложение может рассматриваться как скелетное, в силу чего $A^+ = C^+(C^T)^+ = C^T (CC^T)^{-1} ((C^T)^T C^T)^{-1} (C^T)^T = C^T (CC^T)^{-2} C$. \square

$$\mathbf{7.} \quad P^T = P, \quad P^2 = P \Rightarrow P = P^+$$

Непосредственная проверка определяющих соотношений с учётом требований к P даёт: для соотношения (I) $(PP)^T = (P^2)^T = P^T = P = PP$; для соотношения (III) $PPP = P^2P = PP = P^2 = P$.

Соотношения (II) и (IV) проверяются аналогично. \square

$$\mathbf{8.} \quad A^T = A^T AA^+ = A^+ AA^T$$

В силу определяющих соотношений

$$A^T = (AA^+A)^T = A^T(AA^+)^T = A^TAA^+ = (A^+A)^T A^T = A^+AA^T. \quad \square$$

$$\mathbf{9.} \quad (A^T A)^+ = A^+(A^T)^+, \quad (AA^T)^+ = (A^T)^+ A^+$$

Проверим определяющие соотношения для $(A^T A)$:

$$A^T AA^+(A^T)^+ A^T A = A^T AA^+(A^+)^T A^T A = A^T(AA^+)^T A = A^T AA^+ A = A^T A,$$

$$A^T AA^+(A^T)^+ = A^T(A^T)^+ = A^T(A^+)^T = (A^+A)^T = A^+A$$

– симметрична (остальные проверяются аналогично). Второе равенство доказывается аналогично. \square

$$\mathbf{10.} \quad A^+ = A^+(A^+)^T A^T = A^T(A^+)^T A^+ = (A^T A)^+ A^T = A^T(AA^T)^+$$

Два первых равенства доказываются аналогично доказательства свойства 8 с использованием определяющих соотношений, например:

$$A^+ = A^+AA^+ = (A^+A)^T A^+ = A^T(A^+)^T A^+.$$

Для проверки равенства (III) используем (II), а также свойства 3 и 9: $A^+ = A^+(A^+)^T A^T = A^+(A^T)^+ A^T = (A^T A)^+ A^T$. Четвёртое равенство доказывается аналогично. \square

$$\mathbf{11.} \quad (AA^+)^+ = AA^+, \quad (A^+A)^+ = A^+A$$

Равенства доказываются непосредственной проверкой определяющих соотношений для AA^+ , A^+A , например:

$$(AA^+AA^+)^T = (AA^+)^T = AA^+ = AA^+AA^+, \quad AA^+AA^+AA^+ = AA^+. \quad \square$$

$$\mathbf{12.} \quad (AA^+)^2 = AA^+,$$

$$(A^+A)^2 = A^+A,$$

$$(I - AA^+)^2 = I - AA^+,$$

$$(I - A^+A)^2 = I - A^+A.$$

Равенства следуют непосредственно из определяющих соотношений для A^+ , например:

$$\begin{aligned} (AA^+)^2 &= AA^+AA^+ = AA^+, \quad (I - A^+A)^2 = (I - A^+A)(I - A^+A) = \\ &= I - A^+A - A^+A + A^+AA^+A = I - A^+A - A^+A + A^+A = I - A^+A. \end{aligned} \quad \square$$

13. $A = UBV^T$, $U^T U = I$, $V^T V = I$, $A^+ = VB^+U^T$; $\text{rg} A = r$,

$$B = \begin{pmatrix} B_1 & 0 \\ 0 & 0 \end{pmatrix}, \quad B_1 \in \mathbb{R}_r^{r \times r} \Rightarrow A^+ = V \begin{pmatrix} B_1^{-1} & 0 \\ 0 & 0 \end{pmatrix} U^T.$$

Непосредственная проверка определяющих соотношений с учётом требований к A , U , V даёт:

для (I) $(UBV^T VB^+ U^T)^T = (UBB^+ U^T)^T = U(BB^+)^T U^T = UBB^+ U^T$,

для (II) $(VB^+ U^T UBV^T)^T = (VB^+ BV^T)^T = V(B^+ B)^T V^T = VB^+ BV^T$,

для (III) $UBV^T VB^+ U^T UBV^T = UBB^+ BV^T = UBV^T$,

для (IV) $VB^+ U^T UBV^T VB^+ U^T = VB^+ BB^+ U^T = VB^+ U^T$. \square

14. $\det Q \neq 0 \Rightarrow (QA)^+ QA = A^+ A$, $AQ(AQ)^+ = AA^+$;

$Q \in \mathbb{R}_r^{r \times m}$, $A = BC$, $QB \in \mathbb{R}_r^{r \times r} \Rightarrow (QA)^+ QA = C^+ C$,

$Q \in \mathbb{R}_r^{n \times r}$, $A = BC$, $CQ \in \mathbb{R}_r^{r \times r} \Rightarrow AQ(AQ)^+ = BB^+$

Если $A = BC$ – скелетное разложение для A , то $\hat{A} = QA = QBC = \hat{B}C$ – скелетное разложение для \hat{A} (умножение матрицы на невырожденную не меняет ранга исходной); поэтому выражения

$$A^+ A = C^T (CC^T)^{-1} (B^T B)^{-1} B^T BC = C^T (CC^T)^{-1} C,$$

$$\hat{A}^+ \hat{A} = C^T (CC^T)^{-1} (B^T Q^T QB)^{-1} B^T Q^T QBC = C^T (CC^T)^{-1} C$$

совпадают. Аналогично доказывается второе равенство. \square

15. $A \in \mathbb{R}_r^{m \times n}$, $U \in \mathbb{R}_{m-r}^{m \times (m-r)}$, $V \in \mathbb{R}_{n-r}^{m \times (n-r)}$, $A^T U = 0$, $AV = 0$

$$\Rightarrow I - A^+ A = VV^+, \quad I - AA^+ = UU^+,$$

$$\begin{pmatrix} A & : & U \\ \dots & \dots & \dots \\ V^T & : & 0 \end{pmatrix}^{-1} = \begin{pmatrix} A^+ & : & (V^T)^+ \\ \dots & \dots & \dots \\ U^+ & : & 0 \end{pmatrix} = \begin{pmatrix} A^+ & : & V(V^T V)^{-1} \\ \dots & \dots & \dots \\ (U^T U)^{-1} U^T & : & 0 \end{pmatrix}.$$

Как известно, для любой матрицы H матрица HH^+ ортогонально проектирует на образ H , матрица $I - H^+ H$ ортогонально проектирует на ядро H , а матрица $I - HH^+$ ортогонально проектирует на ядро H^T , причём эти ортогональные проекторы единственны.

Так как по условию $AV = 0$, то образ V совпадает с ядром A . Матрица VV^+ ортогонально проектирует на образ V , а значит – на ядро A . Поэтому VV^+ совпадает с $I - A^+ A$, чем доказано первое равенство.

Так как по условию $A^T U = 0$, то образ U совпадает с ядром A^T . Матрица UU^T ортогонально проектирует на образ U , а значит – на ядро A^T . Поэтому UU^T совпадает с $I - AA^+$, чем доказано второе равенство.

Для доказательства третьего равенства проверим сначала, что

$$\begin{pmatrix} A & : & U \\ \dots & \dots & \dots \\ V^T & : & 0 \end{pmatrix} \cdot \begin{pmatrix} A^+ & : & (V^T)^+ \\ \dots & \dots & \dots \\ U^+ & : & 0 \end{pmatrix} = \begin{pmatrix} I & : & 0 \\ \dots & \dots & \dots \\ 0 & : & I \end{pmatrix}.$$

Действительно,

- 1) $AA^+ + UU^+ = I$ в силу второго равенства;
- 2) $A(V^T)^+ = (AV^+)^T = A(V^+(V^+)^T V^T)^T = AVV^+(V^+)^T = 0$ в силу $AV = 0$ (здесь использовано свойство 10);
- 3) $V^T A^+ = V^T A^+ AA = V^T (A^+ A)^T A^+ = V^T A^T (A^+)^T A^+ = 0$ в силу $V^T A^T = 0$;
- 4) $V^T (V^T)^+ = V^T (V^+)^T = (V^+ V)^T = I$, так как V – матрица полного столбцового ранга (см. (6.8)).

Теперь проверим, что

$$\begin{pmatrix} A^+ & : & (V^T)^+ \\ \dots & \dots & \dots \\ U^+ & : & 0 \end{pmatrix} \cdot \begin{pmatrix} A & : & U \\ \dots & \dots & \dots \\ V^T & : & 0 \end{pmatrix} = \begin{pmatrix} I & : & 0 \\ \dots & \dots & \dots \\ 0 & : & I \end{pmatrix}.$$

Действительно,

- 1) $A^+ A + (V^+)^T V^T = A^+ A + (VV^+)^T = A^+ A + VV^+ = I$ в силу первого равенства;
- 2) $A^+ U = A^+ AA^+ U = A^+ (AA^+)^T U = A^+ (A^+)^T A^T U = 0$ в силу $A^T U = 0$;
- 3) $U^+ A = U^+ (U^+)^T U^T A = 0$ в силу $U^T A = 0$ (здесь использовано свойство 10);
- 4) $U^+ U = I$, так как U – матрица полного столбцового ранга. \square

§ 8. Рекуррентные алгоритмы псевдообращения

Алгоритм Фаддеева

Пусть $A \in \mathbb{R}_r^{m \times n}$. Тогда матрица $A^+ \in \mathbb{R}_r^{n \times m}$ может быть определена с помощью следующего алгоритма.

$$\text{Шаг 1.} \quad \Phi_1 = I, \quad \varphi_1 = \text{tr}(A^T A).$$

Пусть после шага $k - 1$ получены матрицы Φ_{k-1} и число φ_{k-1} .

$$\text{Шаг } k. \quad \Phi_k = \varphi_{k-1} I - A^T A \Phi_{k-1},$$

$$\varphi_k = \frac{\text{tr}(A^T A \Phi_k)}{k}, \quad k = 2, \dots, l,$$

где l – наибольшее число, для которого $\varphi_l \neq 0$.

$$\text{Шаг } l. \quad \varphi_l^{-1} \Phi_l A^T = A^+, \quad r = \text{rg} A = l.$$

Целочисленный алгоритм Фаддеева

$$\text{Пусть } A \in \mathbb{Z}_r^{m \times n}, \quad \mathbb{Q}_r^{n \times m} \ni A = \frac{\Phi}{\varphi}, \quad \varphi \in \mathbb{Z}, \quad \Phi \in \mathbb{Z}_r^{n \times m}.$$

Тогда целочисленная матрица Φ и целое число φ могут быть определены с помощью следующего алгоритма.

$$\text{Шаг 1.} \quad \Phi_1 = I, \quad \varphi_1 = \text{tr}(A^T A).$$

Пусть после шага $k - 1$ получены целочисленная матрица Φ_{k-1} и целое число φ_{k-1} .

$$\text{Шаг } k. \quad \Phi_k = \varphi_{k-1} I - A^T A \Phi_{k-1},$$

$$\varphi_k = \text{div} \left(\frac{\text{tr}(A^T A \Phi_k)}{k} \right), \quad k = 2, \dots, l,$$

где l – наибольшее число, для которого $\varphi_l \neq 0$.

$$\text{Шаг } l. \quad \Phi_l A^T = \Phi, \quad \varphi_l = \varphi, \quad r = \text{rg} A = l.$$

Здесь и далее символическое обозначение div определяет операцию деления нацело. В результате подачи на вход алгоритма целочисленной матрицы A получим целочисленную псевдоприсоединённую матрицу Φ и целочисленный псевдоопределитель φ , доставляющие $A^+ = \frac{\Phi}{\varphi}$.

Алгоритм Гревия

Пусть $A \in \mathbb{R}_r^{m \times n}$, a_k — k -й столбец матрицы A , $k = 1, \dots, n$,

$$A_k = \begin{pmatrix} a_1 & a_2 & \dots & a_k \end{pmatrix} = \begin{pmatrix} A_{k-1} & a_k \end{pmatrix}, \quad k = 2, \dots, n, \quad A_1 = a_1, \quad A_n = A.$$

Тогда матрица $A^+ \in \mathbb{R}_r^{n \times m}$ может быть вычислена с помощью следующего алгоритма.

$$\text{Шаг 1.} \quad A_1^+ = \begin{cases} 0, & a_1 = 0, \\ \frac{a_1^T}{|a_1|^2}, & a_1 \neq 0. \end{cases}$$

Пусть после шага $k - 1$ получена матрица A_{k-1}^+ .

$$\text{Шаг } k. \quad A_k^+ = \begin{pmatrix} A_{k-1}^+(I - a_k f_k) \\ \dots \\ f_k \end{pmatrix},$$

$$\text{где} \quad f_k = \begin{cases} \frac{C_k^T}{|C_k|^2}, & C_k = (I - A_{k-1}^+ A_k) a_k \neq 0, \\ \frac{a_k^T (A_{k-1}^+)^T A_{k-1}^+}{d_k}, & C_k = 0, \\ d_k = 1 + |A_{k-1}^+ a_k|^2, & k = 2, \dots, n. \end{cases}$$

$$\text{Шаг } n. \quad A_n^+ = A^+.$$

Целочисленный алгоритм Гревия

Пусть $A \in \mathbb{Z}_r^{m \times n}$, $\mathbb{Q}_r^{n \times m} \ni A^+ = \frac{T}{t}$, $t \in \mathbb{Z}$, $T \in \mathbb{Z}_r^{n \times m}$.

Тогда целочисленная матрица T и целое число t могут быть определены с помощью следующего алгоритма.

$$\text{Шаг 1.} \quad T_1 = a_1^T, \quad t_1 = |a_1|^2, \quad a_1 \neq 0.$$

Пусть после шага $k - 1$ получены целочисленная матрица T_{k-1} и целое число t_{k-1} .

$$\text{Шаг } k. \quad T_k = \begin{pmatrix} \operatorname{div} \left(\frac{T_{k-1}(t_k I - a_k F_k)}{t_{k-1}} \right) \\ \dots \\ F_k \end{pmatrix},$$

где

$$F_k = \begin{cases} C_k^T, & C_k = (t_{k-1}I - A_{k-1}T_{k-1})a_k \neq 0, \\ a_k^T T_{k-1}^T T_{k-1}, & C_k = 0, \end{cases}$$

$$t_k = \begin{cases} t_{k-1}|a_k|^2 - a_k^T A_{k-1} T_{k-1} a_k, & C_k \neq 0, \\ t_{k-1}^2 + |T_{k-1} a_k|^2, & C_k = 0. \end{cases}$$

$k = 2, \dots, n.$

Шаг n. $T_n = T, \quad t_n = t.$

В результате подачи на вход алгоритма целочисленной матрицы A получим целочисленную псевдоприсоединённую матрицу T и целочисленный псевдоопределитель t , доставляющие $A^+ = \frac{T}{t}$.

Основой построения целочисленных алгоритмов служат их обычные аналоги.

§ 9. Примеры нахождения псевдообратных матриц

В качестве примера рассмотрим процедуры псевдообращения матрицы

$$A = \begin{pmatrix} 1 & 2 & 0 & 0 & -1 \\ 1 & 2 & 0 & 1 & -1 \\ 1 & 2 & 0 & 0 & -1 \\ 1 & 2 & 1 & 0 & 0 \end{pmatrix}.$$

Псевдообращение с помощью математических систем

Процедуры нахождения псевдообратных по Муру-Пенроузу матриц с использованием различных алгоритмов реализованы в большинстве существующих на сегодняшний день математических программных сред.

Например, в пакете программ Matlab присутствует функция `pinv`. Приведём синтаксис этой функции [97]:

$$P = \text{pinv}(A).$$

Функция $P = \text{pinv}(A)$ вычисляет матрицу, псевдообратную к матрице A , которая имеет такие же размеры, как и исходная матрица.

Рассмотрим пример работы этой функции.

```
>> A = [1 2 0 0 -1; 1 2 0 1 -1; 1 2 0 0 -1; 1 2 1 0 0]
>> A
A =
1 2 0 0 -1
1 2 0 1 -1
1 2 0 0 -1
1 2 1 0 0
>> A_preuso_inv = pinv(A)
>> A_preuso_inv
0.0455 0.0000 0.0455 0.0909
0.0909 -0.0000 0.0909 0.1818
-0.2273 -0.0000 -0.2273 0.5455
-0.5000 1.0000 -0.5000 0.0000
-0.2727 0.0000 -0.2727 0.4545
```

Аналогичную описанной выше процедуре реализует функция `pseudoinverse` пакета `corpcor` среды R [145]:

```
p <- pseudoinverse(m).
```

Приведём пример формирования матрицы в среде R и поиска псевдообратной к ней.

```
> library("corpcor")
> A <- rbind(c(1,2,0,0,-1), c(1,2,0,1,-1),
+c(1,2,0,0,-1), c(1,2,1,0,0))
> A_preuso_inv <- pseudoinverse(A)
> A_preuso_inv
      [,1]      [,2]      [,3]      [,4]
[1,] 0.04545455 1.850848e-15 0.04545455 9.090909e-02
[2,] 0.09090909 -7.849556e-16 0.09090909 1.818182e-01
[3,] -0.22727273 -3.353166e-16 -0.22727273 5.454545e-01
[4,] -0.50000000 1.000000e+00 -0.50000000 -1.122149e-17
[5,] -0.27272727 1.300907e-16 -0.27272727 4.545455e-01
```

Проверим некоторые из свойств найденной псевдообратной матрицы. Например, $(A^+)^+ = A$ (свойство 1):

```
> pseudoinverse(A_preuso_inv)
      [,1] [,2]      [,3]      [,4]      [,5]
[1,] 1 2 5.551115e-17 -9.992007e-16 -1.000000e+00
```

```
[2,] 1 2 3.608225e-16 1.000000e+00 -1.000000e+00
[3,] 1 2 4.163336e-16 -1.165734e-15 -1.000000e+00
[4,] 1 2 1.000000e+00 -7.216450e-16 1.554312e-15
```

выполняется (с достаточно большой точностью, определяемой настройками используемого в R алгоритма псевдообращения).

Свойство $(A^T)^+ = (A^+)^T$ (свойство 3):

```
> options(digits=3)
> pseudoinverse(t(A))
      [,1]      [,2]      [,3]      [,4]      [,5]
[1,] 4.55e-02  9.09e-02 -2.27e-01 -5.00e-01 -0.273
[2,] 1.04e-16 -3.47e-17 -1.94e-16  1.00e+00  0.000
[3,] 4.55e-02  9.09e-02 -2.27e-01 -5.00e-01 -0.273
[4,] 9.09e-02  1.82e-01  5.45e-01  1.94e-16  0.455

> t(pseudoinverse(A))
      [,1]      [,2]      [,3]      [,4]      [,5]
[1,] 4.55e-02  9.09e-02 -2.27e-01 -5.00e-01 -2.73e-01
[2,] 2.12e-15 -9.71e-16  2.78e-17  1.00e+00 -1.39e-16
[3,] 4.55e-02  9.09e-02 -2.27e-01 -5.00e-01 -2.73e-01
[4,] 9.09e-02  1.82e-01  5.45e-01  8.33e-17  4.55e-01
```

выполняется.

Проверим также выполнимость

$A^+ = A^+(A^+)^T A^T = A^T (A^+)^T A^+ = (A^T A)^+ A^T = A^T (A A^T)^+ \quad (\text{свойство 10}) :$

```
> A_preuso_inv %*% t(A_preuso_inv) %*% t(A)
      [,1]      [,2]      [,3]      [,4]
[1,] 0.04545455 1.821460e-15 0.04545455 9.090909e-02
[2,] 0.09090909 -8.534840e-16 0.09090909 1.818182e-01
[3,] -0.22727273 -2.775558e-16 -0.22727273 5.454545e-01
[4,] -0.50000000 1.000000e+00 -0.50000000 1.110223e-16
[5,] -0.27272727 1.665335e-16 -0.27272727 4.545455e-01

> t(A) %*% t(A_preuso_inv) %*% A_preuso_inv
      [,1]      [,2]      [,3]      [,4]
[1,] 0.04545455 -2.244964e-16 0.04545455 9.090909e-02
[2,] 0.09090909 -4.489928e-16 0.09090909 1.818182e-01
[3,] -0.22727273 -1.094496e-16 -0.22727273 5.454545e-01
```

```
[4,] -0.50000000  1.000000e+00 -0.50000000 -1.094496e-16
[5,] -0.27272727  1.150468e-16 -0.27272727  4.545455e-01
```

```
> pseudoinverse(t(A) %*% A) %*% t(A)
      [,1]      [,2]      [,3]      [,4]
[1,] 0.04545455 -3.469447e-17 0.04545455 9.090909e-02
[2,] 0.09090909 -3.469447e-17 0.09090909 1.818182e-01
[3,] -0.22727273 0.000000e+00 -0.22727273 5.454545e-01
[4,] -0.50000000 1.000000e+00 -0.50000000 -2.775558e-17
[5,] -0.27272727 0.000000e+00 -0.27272727 4.545455e-01
```

```
> t(A) %*% pseudoinverse(A %*% t(A))
      [,1]      [,2]      [,3]      [,4]
[1,] 0.04545455 1.175953e-16 0.04545455 9.090909e-02
[2,] 0.09090909 2.351906e-16 0.09090909 1.818182e-01
[3,] -0.22727273 -2.154716e-16 -0.22727273 5.454545e-01
[4,] -0.50000000 1.000000e+00 -0.50000000 1.214984e-16
[5,] -0.27272727 -3.330669e-16 -0.27272727 4.545455e-01
```

Таким образом, свойство 10 для приведённого численного примера выполняется. Аналогично могут быть проверены и другие свойства.

Системы Maple и Mathematica пока не имеют специальных функций для нахождения псевдообратных матриц. Однако возможно косвенное вычисление псевдообратной матрицы путём многократного обращения к имеющимся функциям в различных режимах. Принципы решения поставленной задачи в Maple описаны, например, в [60].

Нахождение псевдообратной матрицы «ручным» счётом

В следующих ниже примерах проиллюстрируем работу целочисленных алгоритмов с использованием «ручного» счёта.

Псевдообращение матрицы проведём целочисленным алгоритмом Фаддеева.

Шаг 1. $\Phi_1 = I$,

$$\varphi_1 = \text{tr}(A^T A) = \text{tr} \begin{pmatrix} 4 & 8 & 1 & 1 & -3 \\ 8 & 16 & 2 & 2 & -6 \\ 1 & 2 & 1 & 0 & 0 \\ 1 & 2 & 0 & 1 & -1 \\ -3 & -6 & 0 & -1 & 3 \end{pmatrix} = 25.$$

Шаг 2.

$$\Phi_2 = \varphi_1 I - A^T A = \begin{pmatrix} 21 & -8 & -1 & -1 & 3 \\ -8 & 9 & -2 & -2 & 6 \\ -1 & -2 & 24 & 0 & 0 \\ -1 & -2 & 0 & 24 & 1 \\ 3 & 6 & 0 & 1 & 22 \end{pmatrix},$$

$$\begin{aligned} \varphi_2 &= \text{div} \left(\frac{\text{tr}(A^T A \Phi_2)}{2} \right) = \\ &= \text{div} \left(\text{tr} \begin{pmatrix} 9 & 18 & 4 & 1 & -5 \\ 18 & 36 & 8 & 2 & -10 \\ 4 & 8 & 19 & -5 & 15 \\ 1 & 2 & -5 & 18 & -6 \\ -5 & -10 & 15 & -6 & 20 \end{pmatrix} / 2 \right) = 51. \end{aligned}$$

Шаг 3.

$$\Phi_3 = \varphi_2 I - A^T A \Phi_2 = \begin{pmatrix} 42 & -18 & -4 & -1 & 5 \\ -18 & 15 & -8 & -2 & 10 \\ -4 & -8 & 32 & 5 & -15 \\ -1 & -2 & 5 & 33 & 6 \\ 5 & 10 & -15 & 6 & 31 \end{pmatrix},$$

$$\varphi_3 = \text{div} \left(\frac{\text{tr}(A^T A \Phi_3)}{3} \right) = \text{div} \left(\text{tr} \begin{pmatrix} 4 & & & & \\ & 16 & & & \\ & & 12 & & \\ & & & 22 & \\ & & & & 12 \end{pmatrix} / 3 \right) = 22.$$

Шаг 4.

$$\Phi_4 = \varphi_3 I - A^T A \Phi_3 = \begin{pmatrix} 17 & 2 & -2 & 2 & 2 \\ -8 & 5 & -16 & 0 & 4 \\ -5 & -40 & 10 & 0 & -15 \\ 0 & 10 & 0 & 0 & 0 \\ 2 & 4 & -10 & 0 & 10 \end{pmatrix},$$

$$\varphi_4 = \operatorname{div} \left(\frac{\operatorname{tr}(A^T A \Phi_4)}{4} \right) = \operatorname{div} \left(\operatorname{tr} \begin{pmatrix} -7 & & & & \\ & 28 & & & \\ & & -23 & & \\ & & & 2 & \\ & & & & 0 \end{pmatrix} / 4 \right) = 0.$$

Поэтому $\Phi_3 A^T = \Phi = \begin{pmatrix} 1 & 0 & 1 & 2 \\ 2 & 0 & 2 & 4 \\ -5 & 0 & -5 & 12 \\ -11 & 22 & -11 & 0 \\ -6 & 0 & -6 & 10 \end{pmatrix}, \quad \sigma_1 = \sigma = 22,$

$$A^+ = \frac{1}{22} \begin{pmatrix} 1 & 0 & 1 & 2 \\ 2 & 0 & 2 & 4 \\ -5 & 0 & -5 & 12 \\ -11 & 22 & -11 & 0 \\ -6 & 0 & -6 & 10 \end{pmatrix}.$$

Применим к этой матрице целочисленный алгоритм Гревилля.

Шаг 1. $T_1 = a_1^T = \begin{pmatrix} 1 & 1 & 1 & 1 \end{pmatrix}, \quad t_1 = a_1^T a_1 = 4.$

Шаг 2. $C_2 = t_1 a_2 - A_1 T_1 a_2 = 0, \quad t_2 = t_1^2 + |T_1 a_2|^2 = 80,$

$$F_2 = a_2^T T_1 T_1 = \begin{pmatrix} 8 & 8 & 8 & 8 \end{pmatrix},$$

$$t_2 I - a_2 F_2 = \begin{pmatrix} 64 & -16 & -16 & -16 \\ -16 & 64 & -16 & -16 \\ -16 & -16 & 64 & -16 \\ -16 & -16 & -16 & 64 \end{pmatrix},$$

$$\begin{aligned}
T_2 &= \begin{pmatrix} \operatorname{div} \left(\frac{T_1(t_2 I - a_2 F_2)}{t_1} \right) \\ \dots \\ F_2 \end{pmatrix} = \\
&= \begin{pmatrix} \operatorname{div} \left(\frac{16}{4} \right) & \operatorname{div} \left(\frac{16}{4} \right) & \operatorname{div} \left(\frac{16}{4} \right) & \operatorname{div} \left(\frac{16}{4} \right) \\ \dots & \dots & \dots & \dots \\ 8 & 8 & 8 & 8 \end{pmatrix} = \\
&= \begin{pmatrix} 4 & 4 & 4 & 4 \\ 8 & 8 & 8 & 8 \end{pmatrix}.
\end{aligned}$$

Так как t_2 и T_2 имеют общий множитель 4, то сократим его, не меняя обозначения: $t_2 = 20$, $T_2 = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \end{pmatrix}$.

Шаг 3. $C_3 = t_2 a_3 - A_2 T_2 a_3 = \begin{pmatrix} -5 \\ -5 \\ -5 \\ 15 \end{pmatrix} \neq 0$,

$$t_3 = t_2 |a_3|^2 - a_3^T A_2 T_2 a_3 = 15, \quad F_3 = C_3^T = \begin{pmatrix} -5 & -5 & -5 & 15 \end{pmatrix},$$

$$t_3 I - a_3 F_3 = \begin{pmatrix} 15 & 0 & 0 & 0 \\ 0 & 15 & 0 & 0 \\ 0 & 0 & 15 & 0 \\ 5 & 5 & 5 & 0 \end{pmatrix},$$

$$T_2(t_3 I - a_3 F_3) = \begin{pmatrix} 20 & 20 & 20 & 0 \\ 40 & 40 & 40 & 0 \end{pmatrix},$$

$$T_3 = \begin{pmatrix} \operatorname{div} \left(\frac{T_2(t_3 I - a_3 F_3)}{t_2} \right) \\ \dots \\ F_3 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 2 & 2 & 2 & 0 \\ -5 & -5 & -5 & 15 \end{pmatrix}.$$

$$\text{Шаг 4.} \quad C_4 = t_3 a_4 - A_3 T_3 a_4 = \begin{pmatrix} -5 \\ 10 \\ -5 \\ 0 \end{pmatrix} \neq 0,$$

$$t_4 = t_3 |a_4|^2 - a_4^T A_3 T_3 a_4 = 10, \quad F_4 = C_4^T = \begin{pmatrix} -5 & 10 & -5 & 0 \end{pmatrix},$$

$$t_4 I - a_4 F_4 = \begin{pmatrix} 10 & 0 & 0 & 0 \\ 5 & 0 & 5 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 10 \end{pmatrix},$$

$$T_3(t_4 I - a_4 F_4) = \begin{pmatrix} 15 & 0 & 15 & 0 \\ 30 & 0 & 30 & 0 \\ -75 & 0 & -75 & -150 \end{pmatrix},$$

$$T_4 = \begin{pmatrix} \operatorname{div} \left(\frac{T_3(t_4 I - a_4 F_4)}{t_3} \right) \\ \dots \\ F_4 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 2 & 0 & 2 & 0 \\ -5 & 0 & -5 & 10 \\ -5 & 10 & -5 & 0 \end{pmatrix}.$$

$$\text{Шаг 5.} \quad C_5 = t_4 a_5 - A_4 T_4 a_5 = 0, \quad t_5 = t_4^2 + |T_4 a_5|^2 = 220,$$

$$F_4 = a_5^T T_4^T T_4 = \begin{pmatrix} -60 & 0 & -60 & 100 \end{pmatrix},$$

$$t_5 I - a_5 F_5 = \begin{pmatrix} 160 & 0 & -60 & 100 \\ -60 & 220 & -60 & 100 \\ -60 & 0 & 60 & 100 \\ 0 & 0 & 0 & 220 \end{pmatrix},$$

$$T_4(t_5 I - a_5 F_5) = \begin{pmatrix} 100 & 0 & 100 & 200 \\ 200 & 0 & 200 & 400 \\ -500 & 0 & -500 & -1200 \\ -1100 & 2200 & -1100 & 0 \end{pmatrix},$$

$$T_5 = \begin{pmatrix} \operatorname{div} \left(\frac{T_4(t_5 I - a_5 F_5)}{t_4} \right) \\ \dots \\ F_5 \end{pmatrix} = \begin{pmatrix} 10 & 0 & 10 & 20 \\ 20 & 0 & 20 & 40 \\ -50 & 0 & -50 & -120 \\ -110 & 220 & -110 & 0 \\ -60 & 0 & -60 & 100 \end{pmatrix}.$$

Так как t_5 и T_5 имеют общий множитель 10, сократим его, не меняя обозначения:

$$t = t_5 = 22, \quad T = T_5 = \begin{pmatrix} 1 & 0 & 1 & 2 \\ 2 & 0 & 2 & 4 \\ -5 & 0 & -5 & -12 \\ -11 & 22 & -11 & 0 \\ -6 & 0 & -6 & 10 \end{pmatrix}.$$

§ 10. Системы линейных уравнений и метод наименьших квадратов

В данном параграфе рассматривается применение псевдообращения к исследованию и решению систем линейных уравнений.

Псевдообращение при решении совместных систем линейных уравнений

Дана система $Ax = b$, где $x \in \mathbb{R}^{n \times 1}$ – столбец неизвестных, $A \in \mathbb{R}_r^{m \times n}$ – матрица коэффициентов, $b \in \mathbb{R}^{m \times 1}$ – столбец свободных членов.

Система совместна (уравнение разрешимо) тогда и только тогда, когда выполняется критерий совместности:

$$AA^+b = b \text{ или } (I - AA^+)b = 0,$$

общее решение совместной системы записывается в виде

$$x = A^+b + (I - A^+A)y, \quad y - \text{произвольный вектор.}$$

Действительно, подставим выражение для x в уравнение и воспользуемся определяющим соотношением (III) для псевдообратной матрицы; тогда

$$A(A^+b + (I - A^+A)y) = AA^+b + (A - AA^+A)y = AA^+b.$$

Если критерий выполнен, то заключаем, что x удовлетворяет уравнению, так что система совместна: наоборот, если x – решение, то $AA^+b = b$, то есть критерий выполняется.

Подчёркиваем, что в проведённом рассуждении использовано только одно из четырех соотношений Мура-Пенроуза: матрица, удовлетворяющая только этому соотношению, называется *обобщённой обратной* к матрице A и обозначается A^- : $AA^-A = A$. Таким образом, наряду с критерием Кронекера-Капелли, формулируемым в терминах рангов основанной A и расширенной $(A | b)$ матриц системы: $\text{rg}(A | b) = \text{rg}A$ – установлен критерий совместности системы линейных уравнений в терминах обобщённой обратной A^- и, в частности, систематически изучаемой здесь псевдообратной A^+ .

В общем решении неоднородной системы выделяются:

- *общее решение соответствующей однородной системы*

$$x_o = (I - A^+A)y,$$

- *частное решение данной неоднородной системы*

$$x_n = A^+b,$$

так что общее решение имеет структуру $x = x_o + x_n$.

Если выполняется дополнительное условие $I - A^+A = 0$, или $A^+A = I$, то решение системы единственно: $x = x_n$.

В дальнейшем несколько раз используется тот факт, что если два вектора p и q ортогональны: $(p, q) = p^Tq = 0$ – то норма каждого из них не превосходит норму их суммы: $|p| \leq |p + q|$; действительно,

$$|p + q|^2 = (p + q)^T(p + q) = p^Tp + p^Tq + q^Tp + q^Tq = |p|^2 + |q|^2 \geq |p|^2.$$

Решение $x_n = A^+b$ обладает тем свойством, что имеет минимальную норму среди всех решений системы $Ax = b$:

$$|x_n| < |x| \text{ для всех } x \text{ таких, что } Ax = b.$$

Действительно, в силу определяющих соотношений (II), (IV) для псевдообратной матрицы $(A^+)^T = (A^+AA^+)^T = (A^+)^T(A^+A)^T = (A^+)^TA^+A$, так что $(A^+)^T(I - A^+A) = 0$: отсюда для любых векторов b, y получим $C^T(A^+)^T(I - A^+A)y = (A^+b)^T(I - A^+A)y = 0$, так что векторы A^+b и $(I - A^+A)y$ ортогональны; поэтому

$$|x_n| = |A^+b| \leq |A^+b + (I - A^+A)y| = |x|.$$

Такое решение x_n называется *нормальным*. Подчеркнём, что в проведённом рассуждении использовались только два из четырёх определяющих соотношений Мура-Пенроуза.

Если найдена псевдообратная матрица A^+ , то задача исследования и решения совместной системы $Ax = b$ может считаться полностью решённой: знание A^+ позволяет как проверить критерий совместимости, так и, в случае его выполнения, записать общее решение системы. Для других целей, например, выделения базисных и свободных переменных – псевдообращение может быть использовано следующим образом.

Выполнение соотношения $LL^+l = l$ для любых матриц L и столбца l означает, что столбец l является линейной комбинацией столбцов матрицы L ; действительно, в этом случае

$$l = L\lambda = \lambda_1 l_1 + \lambda_2 l_2 + \dots + \lambda_n l_n,$$

где l_k – столбцы матрицы L , λ_k – коэффициенты линейной комбинации – компоненты некоторого решения $\lambda^T = (\lambda_1, \dots, \lambda_m)^T$ совместной системы $L\lambda = l$. Применение этого соображения к транспонированной расширенной матрице $L = (A | b)^T \in \mathbb{R}^{(n+1) \times m}$ исследуемой системы позволяет избавиться от «лишних» уравнений, которые являются линейными комбинациями остальных. Это выполняется путём нижеследующей процедуры.

Для первого столбца $L_1 = l_1 = (a_{11} \dots a_{1n} | b_1)^T$ матрицы L находится строка l_1^+ , матрица $I - l_1 l_1^+$ и проверяется выполнение условия $(I - l_1 l_1^+)l_2 = 0$, где $l_2 = (a_{21} \dots a_{2n} | b_2)^T$ – второй столбец матрицы L . Если условие выполняется, то второе уравнение является следствием первого и удаляется из системы, после чего аналогичным образом исследуется третье уравнение и так далее. Если же условие не выполняется, то второе уравнение линейно не зависит от первого и остается в системе.

В этом случае для матрицы $L_2 = (l_1 | l_2)$ находятся матрицы L_2^+ , $I - L_2 L_2^+$ и проверяется выполнение условия

$$(I - L_2 L_2^+)l_3 = 0,$$

где l_3 – третий столбец матрицы L . Если условие выполняется, то третье уравнение удаляется из системы, в противном случае остается, и так далее. Повторив эти действия $m - 1$ раз, получаем систему из ρ линейно независимых уравнений

$$\bar{A}x = \bar{b}, \quad \bar{A} \in \mathbb{R}^{\rho \times n}, \quad \bar{b} \in \mathbb{R}^{\rho \times 1}, \quad \text{rg}(\bar{A} | \bar{b}) = \rho \leq m -$$

ранг расширенной матрицы системы.

Применяя далее эту же процедуру к матрице \bar{A} и формируя из её линейно независимых столбцов матрицу $\bar{B} \in \mathbb{R}^{\rho \times r}$, а из линейно зависящих от них – матрицу $\bar{C} \in \mathbb{R}^{\rho \times (n-r)^r}$, получаем представление матрицы \bar{A} в виде

$$\bar{A} = (\bar{B} | \bar{C}) P^T,$$

где P – матрица перестановок, $\text{rg} \bar{B} = r$ – ранг основной матрицы.

Если $r < \rho$, то в соответствии с критерием Кронекера-Капелли система несовместна. Если же $r = \rho$, то система совместна и представляется в виде

$$(\bar{B} | \bar{C}) \begin{pmatrix} x_\delta \\ \dots \\ x_c \end{pmatrix} = \bar{b},$$

где $\bar{B} \in \mathbb{R}_r^{r \times r}$ – невырожденная матрица, $\det \bar{B}$ – базисный минор, $x = P^T x$ – столбец переставленных неизвестных, x_δ – базисные, x_c – свободные неизвестные. Решение системы завершается в виде, выражающем базисные неизвестные через свободные:

$$\bar{x}_\delta = \bar{B}^{-1}(\bar{b} - \bar{c}\bar{x}_c),$$

что уточняет и детализирует структуру общего решения.

Псевдообращение при решении несовместных систем линейных уравнений

Пусть теперь система $Ax = b$ несовместна, то есть $AA^+b \neq b$.

В этом случае не существует вектора x , для которого выполнялось бы равенство $Ax = b$, то есть при всех $x \in R^{n \times 1}$ отлична от нуля невязка $Ax - b$, и возникает задача метода наименьших квадратов отыскания псевдообращения x^* – вектора, минимизирующего норму невязки

$$|Ax^* - b| \leq |Ax - b| \text{ для всех } x \in R^{n \times 1}.$$

Использование псевдообращения позволяет решить и эту задачу: все её решения представляются в виде

$$x^* = A^+b + (I - AA^+)y,$$

где $y \in R^{n \times 1}$ – произвольный вектор, то есть в том же виде, что и общее решение совместной системы.

Действительно, учитывая определяющее соотношение (III) для псевдообратной матрицы, запишем

$$|Ax^* - b| = |A(A^+b + (I - A^+A)y) - b| = |(AA^+ - I)b|.$$

Из определяющих соотношений (I), (III) следует, что $A^T = (AA^+A)^T = A^T(AA^+)^T = A^TAA^+$, так что $A^T(AA^+ - I) = 0$; отсюда для любых векторов a, b получим $a^T A^T (AA^+ - I)b = (Aa)^T (AA^+ - I)b = 0$, так что векторы Aa и $(AA^+ - I)b$ ортогональны; поэтому, полагая $a = x - A^+b$, получаем $|(AA^+ - I)b| \leq |(AA^+ - I)b + Aa| = |AA^+b - b + Aa - AA^+b| = |Ax - b| \Rightarrow |Ax^* - b| \leq |Ax - b|$.

Подчеркнём, что в проведённом рассуждении использовались только два из четырёх определяющих соотношений Мура-Пенроуза.

Псевдообращение $x_n^* = A^+b$ называется нормальным, так как имеет минимальную норму среди всех псевдообращений $|x_n^*| < |x^*|$ для всех x^* таких, что $|Ax^* - b| \leq |Ax - b|$.

Это обосновывается так же как свойство нормальности решения $x_n = A^+b$ совместной системы в предыдущем пункте: с использованием двух других определяющих соотношений Мура-Пенроуза. Тем самым, в конечном счёте, использованы все четыре определяющих соотношения Мура-Пенроуза для псевдообратной матрицы.

Если найдена псевдообратная матрицы A^+ , то задача исследования и решения произвольной системы $Ax = b$ полностью решается: знание A^+ позволяет проверить критерий совместности, записать общее решение совместной системы и общее псевдообращение несовместной системы и нормальное

псевдообращение несовместной системы. При этом по форме записи эти решения и псевдорешения не отличаются, что является одним из достоинств применения псевдообращения к исследованию и решению систем линейных уравнений.

$$\text{Рассмотрим пример: } \begin{cases} x_1 + 2x_2 = 3, \\ 3x_2 + x_3 = 2, \\ x_1 + 2x_2 = 1. \end{cases}$$

Для основной матрицы коэффициентов системы

$$A = \begin{pmatrix} 1 & 2 & 0 \\ 0 & 3 & 1 \\ 1 & 2 & 0 \end{pmatrix}$$

вычислена псевдообратная матрица A^+ :

$$A^+ = \frac{1}{14} \begin{pmatrix} 5 & -6 & 5 \\ 1 & 3 & 1 \\ -3 & 5 & -3 \end{pmatrix}.$$

$$\text{Из } (I - AA^+)b = \frac{1}{2} \begin{pmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 3 \\ 2 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix} \neq 0$$

следует, что система несовместна и её псевдообращение имеет вид:

$$x = A^+b + (I - A^+A)y = \frac{1}{7} \begin{pmatrix} 4 \\ 5 \\ 2 \end{pmatrix} + \frac{1}{7} \begin{pmatrix} 4 & -2 & 6 \\ -2 & 1 & -3 \\ 6 & -3 & 9 \end{pmatrix} y.$$

Применение псевдообращения для синтеза алгоритмов управления

Пусть динамическая система с вектором состояний $x(t) \in \mathbb{R}^{\nu \times 1}$ и вектором управления $u(t) \in \mathbb{R}^{\mu \times 1}$ описывается уравнением

$$\mathcal{A}x(t) = Bu(t),$$

где $B \in \mathbb{R}^{\nu \times \mu}$ – некоторая матрица, \mathcal{A} – линейный оператор, $t \in \Gamma$ – аргумент, относительно которого эволюционирует система, причем \mathcal{A} и Γ могут быть конкретизированы в зависимости от типа рассматриваемой системы различными способами.

Так, для линейной стационарной непрерывной системы Γ – непрерывное время, $\Gamma = \{t \geq 0\}$, \mathcal{A} – дифференциальный оператор $\mathcal{A}x(t) = \dot{x}(t) - Ax(t)$, где $A \in \mathbb{R}^{\nu \times \nu}$ – постоянная матрица, так что $B \in \mathbb{R}^{\nu \times \mu}$.

Для линейной стационарной непрерывной вырожденной системы Γ конкретизируется так же, а оператор \mathcal{A} определяется соотношением $\mathcal{A}x(t) = E\dot{x}(t) - Ax(t)$, где E, A – постоянные матрицы.

Для линейной стационарной дискретной системы Γ – дискретное время $\Gamma = \{t = 0, 1, \dots\}$, \mathcal{A} – разностный оператор $\mathcal{A}x(t) = x(t+1) - Ax(t)$, $A \in \mathbb{R}^{\nu \times \nu}$ – постоянная матрица, $B \in \mathbb{R}^{\nu \times \mu}$. Вырожденный аналог такой системы определяется очевидным образом.

Для линейной (пространственно-) однородной двумерной дискретной системы Γ – дискретное двумерное пространство (плоскость), $\Gamma = \{(t, s) : t, s = 0, 1, \dots\}$, \mathcal{A} – частно-разностный оператор $\mathcal{A}x(t, s) = x(t, s) - A_1x(t-1, s) - A_2x(t, s-1)$, $A_1, A_2 \in \mathbb{R}^{\nu \times \nu}$ – постоянные матрицы, $B \in \mathbb{R}^{\nu \times \mu}$. Могут быть рассмотрены вырожденный и многомерный аналоги такой системы и другие классы динамических систем.

Рассмотрим задачу управления движением по назначенной траектории: для всех $t \in \Gamma$ задается траектория (в обычном или обобщённом смысле) желаемого движения данной системы, обозначаемая через $x^*(t), t \in \Gamma$. Требуется построить алгоритм управления $u^*(t), t \in \Gamma$, при котором движение системы осуществляется по заданной траектории $x^*(t)$.

Подставим траекторию в левую часть уравнения, описывающего динамическую систему; тогда для искомого управления получим уравнение

$$Bu_*(t) = \mathcal{A}x^*(t), t \in \Gamma.$$

В соответствии с критерием совместности систем линейных уравнений задача построения алгоритма управления движением заданной динамической

системы по назначенной траектории разрешима тогда и только тогда, когда $(I - BB^+)Ax^*(t) = 0, t \in \Gamma$; её решение (в случае выполнения этого условия) или псевдорешение (в случае его невыполнения) записывается в виде $u^*(t) = B^+Ax^*(t) + (I - B^+B)v(t)$, где $v(t) \in \mathbb{R}^\mu, t \in \Gamma$ – произвольные векторы.

В случае несовместности псевдорешение $u_*(t)$ для любого t минимизирует невязку

$$|Bu_*(t) - Ax^*(t)|, t \in \Gamma$$

среди всех входных воздействий на данную систему, синтезируя алгоритм оптимального (в этом смысле) уравнения. Кроме того, управление

$$u_{*H}(t) = B^+Ax(t)$$

является оптимальным ещё и в том смысле, что для любого t имеет минимальную норму среди всех уравнений $u_*(t)$.

Условие разрешимости поставленной задачи связывает между собой параметры A, B уравнения, описывающего систему, и назначенной траектории $x^*(t)$. Оно допускает различные толкования, наиболее важным из которых является нижеследующее.

Если считать параметры A, B системы заданными, то условие разрешимости накладывает ограничение на выбор задаваемых траекторий и может быть названо в этом случае критерием осуществимости назначаемых траекторий. Оно представляет собой уравнение того же типа, что и исходная система, но отличающееся двумя особенностями:

1) это уравнение однородно (в смысле линейного уравнения с нулевой правой частью), иначе говоря, описываемая ими новая динамическая система (система осуществимых назначаемых траекторий) автономна (отсутствуют внешние воздействия);

2) вообще говоря, оно относится к вырожденному типу систем (даже если исходная система таковой не была). Совокупность решений такого уравнения, будучи полученной, дает полное описание осуществимых назначаемых траекторий для заданной динамической системы. Следует отметить, что во многих случаях это уравнение допускает дальнейший анализ, позволяющий достаточно глубоко вскрыть структуру осуществимых траекторий.

Если считать, что задана траектория, а условие разрешимости накладывает ограничения на параметры системы, то его естественно назвать критерием управляемости движением системы по назначенной траектории. Сопоставляя с традиционными критериями управляемости, известными для различных классов систем, следует отметить, что данный критерий, вообще говоря, является более жёстким, что естественно следует и из самой постановки такой задачи управления; но в ряде случаев данный критерий может совпадать с традиционными, несмотря на отличие в постановках задач управления.

Критерий управляемости движением системы по назначенной траектории также допускает дальнейший анализ, позволяющий достаточно глубоко вскрыть структуру системы, вплоть до выделения в ней неуправляемых (инвариантных к внешним воздействиям) подсистем.

Невыполнение критерия разрешимости поставленной задачи – «по вине» то ли системы, то ли траектории – также может повлечь за собой анализ, результатом которого может оказаться «подправление» параметров системы или траектории, обеспечивающее выполнение критерия, и тем самым точное решение задачи. Наконец, если это оказывается невозможным, приходится довольствоваться алгоритмом оптимального, в смысле минимума нормы невязки по всей траектории управления.

Как намеченные схемы анализа критерия разрешимости, так и фактический синтез алгоритма управления существенно используют псевдообращение, и, в частности, псевдообратную матрицу.

Решение более общей задачи минимизации взвешенной нормы невязки требует отдельного рассмотрения.

В качестве примера рассмотрим линейную однородную двумерную дискретную систему

$$x(t, s) = \begin{pmatrix} 3 & 1 & 2 \\ -0,5 & -1 & 1,3 \\ 1 & 1 & 4 \end{pmatrix} x(t-1, s) +$$

$$\begin{array}{c}
A_1 \\
+ \begin{pmatrix} 0,2 & -0,7 & 0 \\ 4 & 0 & 2,4 \\ 0,2 & -0,7 & 0 \end{pmatrix} x(t, s-1) + \begin{pmatrix} 1 & 2 & 0 \\ 0 & 3 & 1 \\ 1 & 2 & 0 \end{pmatrix} u(t, s), \\
A_2 \qquad \qquad \qquad B
\end{array}$$

где при $t, s = 0, 1, \dots$ $x(t, s) \in \mathbb{R}^{3 \times 1}$, $u(t, s) \in \mathbb{R}^{3 \times 1}$.

Система осуществимых назначаемых траекторий, в соответствии с матрицами $B^+, I - BB^+$, запишется в виде

$$\begin{aligned}
\begin{pmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{pmatrix} x^*(t, s) &= \begin{pmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 3 & 1 & 2 \\ -0,5 & -1 & 1,3 \\ 1 & 1 & 4 \end{pmatrix} x^*(t-1, s) + \\
&+ \begin{pmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0,2 & -0,7 & 0 \\ 4 & 0 & 2,4 \\ 0,2 & -0,7 & 0 \end{pmatrix} x^*(t, s-1) = \\
&= \begin{pmatrix} 2 & 0 & -2 \\ 0 & 0 & 0 \\ -2 & 0 & 2 \end{pmatrix} x^*(t-1, s) + 0 \cdot x^*(t, s-1),
\end{aligned}$$

а алгоритм управления движением по траектории $x^*(t, s)$ – в виде

$$\begin{aligned}
u_*(t, s) &= \frac{1}{14} \begin{pmatrix} 5 & -6 & 5 \\ 1 & 3 & 1 \\ -3 & 5 & -3 \end{pmatrix} \times \\
&\times \left(x^*(t, s) - \begin{pmatrix} 3 & 1 & 2 \\ -0,5 & -1 & 1,3 \\ 1 & 1 & 1 \end{pmatrix} x^*(t-1, s) - \begin{pmatrix} 0,2 & -0,7 & 0 \\ 4 & 0 & 2,4 \\ 0,2 & -0,7 & 0 \end{pmatrix} x^*(t, s-1) \right) +
\end{aligned}$$

$$+\frac{1}{14} \begin{pmatrix} 4 & -2 & 6 \\ -2 & 1 & -3 \\ 6 & -3 & 9 \end{pmatrix} v(t, s).$$

Система осуществимых траекторий может быть записана в виде

$$\begin{aligned} x_1^*(t, s) - x_3^*(t, s) &= 2x_1^*(t-1, s) - 2x_3^*(t-1, s), \\ -x^*(t, s) + x_3^*(t, s) &= -2x_1^*(t-1, s) + 2x_3^*(t-1, s), \end{aligned}$$

и сводится к одному уравнению:

$$x_1^*(t, s) - x_3^*(t, s) = 2(x_1^*(t-1, s) - x_3^*(t-1, s)), \quad t, s = 0, 1, \dots$$

Отсюда следует, что осуществимы траектории $x^*(t, s)$, координаты $x_1^*(t, s)$, $x_2^*(t, s)$ вектора состояния на которых произвольны, а координата $x_3^*(t, s)$ связана с $x_1^*(t, s)$ соотношением

$$x_3^*(t, s) = x_1^*(t, s) - 2^t, \quad t, s = 0, 1, \dots$$

Если назначена такая траектория, то указанное управление точно реализует движение системы по ней. Если же назначена траектория, не обладающая указанными свойствами, то управление оптимально в том смысле, что при любых $t, s = 0, 1, \dots$ минимизирует невязку

$$|Bu_*(t, s) - x^*(t, s) - A_1x^*(t-1, s) - A_2x^*(t, s-1)|.$$

Управление

$$u_{*H}(t, s) = \frac{1}{14} \begin{pmatrix} 5 & -6 & 5 \\ 1 & 3 & 1 \\ -3 & 5 & -3 \end{pmatrix} (x^*(t, s) - A_1x^*(t-1, s) - A_2x^*(t, s-1))$$

оптимально ещё и в том смысле, что при любых $t, s = 0, 1, \dots$ имеет минимальную норму $u_{*H}(t, s)$ среди всех управлений, указанных общей формулой.

Применение псевдообращения в других задачах управления, а также в задачах оценивания состояния и идентификации динамических систем, требует отдельного рассмотрения.

ГЛАВА III

Итерации и рекурсии

Нелинейный метод наименьших квадратов служит для решения нелинейной задачи о наименьших квадратах, которая относится к классу задач нелинейной оптимизации или нелинейного программирования, однако характеризуются в этом классе спецификой [18], оправдывающей её самостоятельное исследование.

Общая задача нелинейной условной минимизации формулируется в виде: найти

$$\min_{x \in X} f(x) = f(x^*) = f^*,$$

где $x = (x_1 \dots x_n)^T \in X \subset \mathbb{R}^n$ – векторный аргумент, $X \subset \mathbb{R}^n$ – некоторое множество, $f(x) = f(x_1, \dots, x_n) \in \mathbb{R}$ – нелинейная скалярная функция векторного аргумента (целевая функция, критерий оптимальности), f^* – её минимум, x^* – значение аргумента, доставляющее этот минимум. Частным случаем общей задачи нелинейной условной минимизации при $X = \mathbb{R}^n$ является общая задача нелинейной безусловной минимизации: найти

$$\min_{x \in \mathbb{R}^n} f(x) = f(x^*) = f^*.$$

Классификации и исследованию этих задач и их разновидностей, разработке методов и алгоритмов их решения посвящено большое число работ (например, [9, 33, 42, 54, 55, 57, 58, 76, 79, 80, 84, 105, 123]), которые по различным причинам не всегда доступны студенту или инженеру и не всегда пригодны для первого знакомства с предметом. В параграфах 11, 12 изложены сведения по этим вопросам в виде не только удобном для последую-

щего использования, но и позволяющем читателю, впервые знакомящемуся с нелинейным программированием, сориентироваться в этой области, чему способствует предложенная здесь и представляющая самостоятельный интерес детальная классификация, опирающаяся на уже известные в литературе (например, [42, 53, 54, 57, 58, 74, 76, 80, 105, 123]), но развивающая и дополняющая их.

Общая многооткликовая взвешенная условная нелинейная задача о наименьших квадратах в классе характеризуется тем, что целевая функция имеет специальный вид:

$$f(x) = \frac{1}{2} |||R(x)|||_w^2 = \frac{1}{2} \sum_{v=1}^l \sum_{t=1}^l \sum_{u=1}^m \sum_{s=1}^m w_{us}^{vt} r_{uv}(x) r_{st}(x).$$

Здесь множитель $\frac{1}{2}$ введён для удобства рассуждений (иногда он опускается). Величина $|||R(x)|||_w$ определена записанным выражением: это некоторая взвешенная норма матричной функции $R : \mathbb{R}^n \rightarrow \mathbb{R}^{m \times l}$ или $R(x) \in \mathbb{R}^{m \times l}$ при $x \in \mathbb{R}^n$, причём $R(x) = [r_{uv}(x)] = [r_{st}(x)]$, где $r_{uv}(x)$, $r_{st}(x)$, $1 \leq u, s \leq m$, $1 \leq v, t \leq l$ – элементы матрицы $R(x)$. Величины w_{us}^{vt} служат весовыми коэффициентами и должны удовлетворять некоторым условиям положительности и симметрии [5].

Матричная функция $R(x)$ обычно называется функцией невязок откликов; каждый её столбец $R_v(x) = (r_{1v}(x), \dots, r_{mv}(x))^T$, $1 \leq v \leq l$, трактуется как столбец невязок v -го отклика. Используемые термины связаны с приложениями, рассмотренными ниже.

Если выполняется условие $w_{us}^{vt} = 0$ при $v \neq t$, $u \neq s$, то целевая функция записывается в виде:

$$f(x) = \frac{1}{2} \sum_{v=1}^l \sum_{u=1}^m w_{uu}^{vv} r_{uv}^2(x),$$

а при $w_{uu}^{vv} = 1$ – в виде:

$$f(x) = \frac{1}{2} \sum_{v=1}^l \sum_{u=1}^m r_{uv}^2(x) = \frac{1}{2} \|R(x)\|_F^2,$$

где $\|R(x)\|$ означает норму Фробениуса, Шура или евклидову матричную форму матричной функции $R(x)$ [73].

Если выполняются условия $w_{us}^{vt} = 0$ при $v \neq t$, то целевая функция записывается в виде:

$$f(x) = \frac{1}{2} \sum_{v=1}^l \sum_{u=1}^m \sum_{s=1}^m w_{us}^{vv} r_{uv}(x) r_{sv}(x) = \frac{1}{2} \sum_{v=1}^l R_v^T(x) W_v R_v(x),$$

где при каждом $1 \leq v \leq l$, $w_v = [w_{us}^{vv}] \in \mathbb{R}^{m \times m}$ – весовая матрица невязок v -го отклика. При $W_v = I$ (единичная матрица):

$$f(x) = \frac{1}{2} \sum_{v=1}^l R_v^T(x) R_v(x) = \frac{1}{2} \sum_{v=1}^l \|R_v(x)\|^2,$$

где $\|R_v(x)\|$ обозначает обычную евклидову векторную норму вектора $R_v(x)$.

Если $l = 1$, то задача называется однооткликковой и индекс v опускается. Если $W = \text{diag}\{w_u\}$ (диагональная матрица с элементами w_u , $1 \leq u \leq m$, на главной диагонали), то

$$f(x) = \frac{1}{2} \sum_{u=1}^m w_u r_u^2(x).$$

Различные из перечисленных (и некоторые из неупомянутых) разновидности общей многооткликковой взвешенной условной нелинейной задачи о наименьших квадратах рассмотрены, например, в работах [1, 2, 5, 31, 49, 50, 73, 83, 101, 103]. Чаще всего рассматривается случай одного отклика при единичных весах и при отсутствии ограничений, как наиболее простой (примеры многооткликковых задач рассмотрены в [1, 5, 83]; взвешенных – в [1, 83]; условных – в [9]. Именно этому случаю и уделено основное внимание в следующих параграфах.

Нелинейная задача о наименьших квадратах рассматривается далее как частная однооткликковая невзвешенная безусловная задача: найти

$$\min_{x \in \mathbb{R}^n} f(x) = \frac{1}{2} \|R(x)\|^2 = \frac{1}{2} \sum_{u=1}^m r_u^2(x) = f(x^*) = f^*,$$

где $R(x) = (r_1(x), \dots, r_m(x))^T \in \mathbb{R}^m$ – нелинейная векторная функция векторного аргумента $x = (x_1, \dots, x_n) \in \mathbb{R}^n$ – функция невязок $R : \mathbb{R}^n \rightarrow \mathbb{R}^m$. Для разъяснения использованных выше терминов остановимся на некоторых приложениях сформулированной задачи.

Пусть $A : \mathbb{R}^n \rightarrow \mathbb{R}^m$ – нелинейный оператор, $b \in \mathbb{R}^m$ – некоторый вектор; рассмотрим уравнение $A(x) = b$, или, что то же, систему m нелинейных уравнений с n неизвестными:

$$a_u(x_1, \dots, x_n) = b_u, \quad 1 \leq u \leq m.$$

Вектор $R(x) = b - A(x)$ называется *невязкой* системы. Вариационный метод решения системы состоит в минимизации евклидовой векторной нормы невязки, что непосредственно приводит к нелинейной задаче о наименьших квадратах, а в случае линейного оператора A – к линейной задаче. Тесная связь решения уравнений и оптимизации подробно рассмотрена, например, в [53, 85].

Пусть $a_u(x)$, $1 \leq u \leq m$ – набор некоторых скалярных критериев оптимизации, то есть векторный критерий. Таким образом рассматривается многокритериальная задача. Пусть b_u , $1 \leq u \leq m$ – так называемые номинальные значения соответствующих показателей качества. Одним из методов многокритериальной оптимизации является свёртка частных критериев. Распространённым способом свёртки служит переход к сумме квадратов отклонений соответствующих критериев от номиналов, что снова приводит к задачам о наименьших квадратах. Возможны и другие способы свёртки; различные подходы подробно рассмотрены, например, в [31, 39].

В рамках этих приложений уже находит разъяснение термин «невязка»; он же и термин «отклик» связаны с, пожалуй, наиболее важными и распространёнными приложениями задач о наименьших квадратах – задачами регрессии (например, [1, 2, 5, 47–50, 52, 74, 83, 86, 87, 101, 103, 111]).

Пусть некоторая величина b , называемая в этих задачах откликом, зависит от вектора факторов $\xi = (\xi_1, \dots, \xi_k) \in \mathbb{R}^k$, причём зависимость пред-

ставлена массивом данных – выборкой объема m :

$$\{\xi_u; b_u\} = \{\xi_{1u}, \dots, \xi_{ku}; b_u\}, \quad 1 \leq u \leq m.$$

Пусть избрана модель для описания этой зависимости, включающая вектор параметров $x = (x_1, \dots, x_n) \in \mathbb{R}^n$: $b = a(\xi; x) = a(\xi_1, \dots, \xi_k; x_1, \dots, x_n)$.

Подстановка данных в модель определяет прогнозируемые значения отклика $\hat{b}_u = a(\xi_u; x) = a_u(x)$, $1 \leq u \leq m$, отличающиеся от данных b_u , $1 \leq u \leq m$, что определяет, в свою очередь, вектор невязки $R(x) \in \mathbb{R}^m$ с координатами $r_u(x) = b_u - \hat{b}_u = b_u - a_u(x) = b_u - a(\xi_u; x)$, $1 \leq u \leq m$.

Регрессионный метод настройки модели на данные состоит в выборе вектора x из условия минимизации некоторой меры отклонения прогноза от данных. Избрание в качестве такой меры евклидовой векторной нормы невязки и приводит к задачам о наименьших квадратах. Принятие именно суммы квадратов в качестве меры для наилучшего приближения данных оправдывается статистическими соображениями (например, [1, 5, 48, 50, 74, 98, 99, 101, 103, 111]). Как и в двух предыдущих задачах, возможны и другие или более общие меры; достаточно широкий их класс образуют, например, декомпозиционные функции [49]. Перечисленные задачи могут рассматриваться также и в комбинациях.

Далее задачи регрессии рассматриваются в качестве основного приложения результатов, формулируемых для нелинейной задачи о наименьших квадратах.

Линейная задача о наименьших квадратах в контексте задач регрессии возникает в том случае, когда модель линейна относительно параметров x (например, [1, 2, 48, 50, 73, 74, 83, 86, 101, 103, 111]):

$$b = a^T(\xi)x = \sum_{i=1}^n a_i(\xi_1, \dots, \xi_k)x_i.$$

В этом случае вектор невязки определяется в виде $R(x) = b - Ax$, $A = [a_{ui}] = [a_i(\xi_u)] \in \mathbb{R}^{m \times n}$.

Функции $a_i(\xi)$, $1 \leq i \leq n$, называемые базисными функциями линейной регрессионной модели, не предполагаются в общем случае линейными как функции векторного аргумента ξ : в простейших случаях они могут быть одночленами различных степеней относительно ξ_1, \dots, ξ_k , что приводит к линейной по параметрам x_1, \dots, x_n полиномиальной по факторам ξ_1, \dots, ξ_k регрессии. Линейной задаче о наименьших квадратах посвящён параграф 13, который служит подготовительным к параграфу 14.

В задачах собственно нелинейной регрессии модель нелинейна как относительно факторов ξ_1, \dots, ξ_k , так и относительно параметров x_1, \dots, x_n . Нелинейной задаче о наименьших квадратах, трактующей модель в общем виде, без её конкретизации, и посвящён параграф 14.

Между тем, многообразие различных нелинейных моделей более или менее общей природы чрезвычайно велико (например, [1, 5, 47–50, 52, 58, 74, 87, 123]). Чтобы дать читателю, впервые знакомящемуся с предметом, это почувствовать, отметим некоторые наиболее характерные классы и конкретные примеры.

Квазилинейные модели считаются простейшими из нелинейных [49]

$$a(\xi; x) = \alpha(\xi^T x) = \alpha \left(\sum_{i=1}^n \xi_i x_i \right),$$

где α – некоторая скалярная функция скалярного аргумента.

Если эта функция тождественна, то получаем наиболее простую линейную и по факторам, и по параметрам модель.

Степенные модели – частный случай квазилинейных [49]:

$$a(\xi; x) = (\xi^T x)^p, \quad p \neq 0.$$

Логлинейные модели – другой частный случай квазилинейных [49]:

$$a(\xi; x) = \exp(\xi^T x) = e^{(\xi_1 x_1 + \dots + \xi_n x_n)};$$

они характеризуются тем, что становятся линейными после логарифмирования. С другой стороны, такую модель можно записать в виде $a(\xi; x) =$

$= e^{\xi_1 x_1} \cdot \dots \cdot e^{\xi_n x_n}$; полагая $e^{\xi_i} = z_i$, $i = 1, \dots, n$, приходим ещё к одному типу нелинейных моделей, а именно: *позиномиальные одночленные модели* [58, 105]: $\xi \rightarrow z$,

$$a(\xi; x) = p(z; x) = z_1^{x_1} \cdot \dots \cdot z_n^{x_n}, \quad z_i > 0, \quad i = 1, \dots, n.$$

Позиномиальные общие модели определяются как линейные комбинации одночленных [58, 105]: $\xi \rightarrow z$, $x \rightarrow (x, c)$,

$$a(\xi; x) = P(z; x; c) = \sum_{j=1}^q c_j z_1^{x_{j1}} \cdot \dots \cdot z_n^{x_{jn}}, \quad c_j, z_i > 0;$$

они, в свою очередь, входят в некоторый более широкий класс нелинейных моделей, определяемых следующим образом.

Смешанные линейно-нелинейные модели [53]; или модели с линейной частью [49] (линейные по части параметров), характеризуются тем, что вектор параметров $x \in \mathbb{R}^{\hat{n}}$ разбивается на два подвектора: $x^{(1)}$, $x^{(2)}$, причём $x^{(2)} \in \mathbb{R}^p$, $x^{(1)} \in \mathbb{R}^{\hat{n}-p}$, $0 < p < \hat{n}$, а сама модель представляется в виде $a(\xi; x) = a(\xi; x^{(1)}, x^{(2)}) = k^T(\xi; x^{(2)})x^{(1)}$, где $k(\xi; x^{(2)}) \in \mathbb{R}^{\hat{n}-p}$, $\xi \in \mathbb{R}^k$, $x^{(2)} \in \mathbb{R}^p$.

Очевидно, именно такую структуру имеют позиномиальные общие модели, причём $\hat{n} = q + qn$, $x^{(1)} = (c_j) \in \mathbb{R}^q$, $x^{(2)} = (x_{ji}) \in \mathbb{R}^{qn}$.

Укажем ещё два класса моделей такого типа.

Полилинейные модели [49] имеют вид:

$$a(\xi; x) = a(\xi; x^{(1)}, x^{(2)}) = k^T(\xi)x^{(1)} + l^T(\xi)x^{(2)} + (x^{(1)})^T M(\xi)x^{(2)},$$

где $k(\xi) \in \mathbb{R}^{\hat{n}-p}$, $l(\xi) \in \mathbb{R}^p$, $M(\xi) \in \mathbb{R}^{(\hat{n}-p) \times p}$, $\xi \in \mathbb{R}^k$. Такие модели становятся линейными относительно одного из подвекторов вектора параметров при фиксировании второго подвектора.

Дробно-линейные модели [49] имеют вид:

$$a(\xi; x) = a(\xi; x^{(1)}, x^{(2)}) = \frac{\xi^{T^{(1)}} x^{(1)}}{\xi^{T^{(2)}} x^{(2)}},$$

где разбиение вектора ξ соответствует разбиению вектора x .

Обобщённые производственные функции представляют класс нелинейных моделей, не попадающих в предыдущий [49]:

$$a(\xi; x) = \frac{1}{x_n} \ln \left(\sum_{i=1}^{n-1} x_i e^{x_n \xi_i} \right);$$

их частными случаями являются или к ним приводятся разнообразные конкретные производственные функции [49].

Связные порядка (n, R) , или (n, R) -связные модели [49], являются линейными относительно факторов ξ_1, \dots, ξ_k и имеют структуру

$$a(\xi; x) = g^T(x)\xi = \sum_{j=1}^k g_j(x)\xi_j,$$

где $g_j(x)$ – скалярные функции векторного аргумента; класс таких моделей весьма широк. Перечень классов нелинейных моделей можно было бы продолжить.

Укажем некоторые конкретные нелинейные модели, часто встречающиеся в литературе; не все из них попадают в перечисленные выше классы:

$$\begin{aligned} a(\xi; x_1, x_2) &= e^{\xi x_1} + e^{\xi x_2}, \\ a(\xi; x_1, x_2, x_3, x_4) &= x_3 e^{\xi x_1} + x_4 e^{\xi x_2}, \\ a(z_1, z_2, z_3; x_1, x_2, x_3) &= z_1^{x_1} z_2^{x_2} z_3^{x_3}, \\ a(\xi_1, \xi_2, \xi_3; x_1, x_2) &= \xi_1 \left(x_1 \xi_2^{-x_2} + (1 - x_1) \xi_3^{-x_2} \right)^{-\frac{1}{x_2}}, \\ a(\xi; x_1, x_2, x_3) &= x_1 + \frac{1}{x_3} \ln \left(x_2 e^{\xi x_3} + 1 - x_2 \right), \\ a(\xi_1, \xi_2; x_1, x_2) &= \exp \left(-\xi_1 x_1 e^{-\frac{x_2}{\xi_2}} \right). \end{aligned}$$

Остановимся теперь подробнее на представленных методах и алгоритмах решения линейной и нелинейной задач о наименьших квадратах.

Линейная задача о наименьших квадратах, состоящая в минимизации $\|b - Ax\|$, очень детально и глубоко исследована в связи с различными её приложениями, первую очередь – с линейной регрессией (например, [1, 2, 48, 50, 73, 83, 86, 101, 103, 111]). Разработано большое число мето-

дов, алгоритмов и пакетов прикладных программ для её решения (например, [1, 2, 50, 87, 103]). Среди разнообразных методов весьма эффективным в формальном, а по мнению ряда специалистов, и в вычислительном отношении является метод псевдообращения, детально рассмотренный во второй главе [2, 16, 17, 73, 101, 103, 111, 114, 127]. Прежде всего отметим (откладывая детали до параграфа 13), что решение линейной задачи о наименьших квадратах можно записать в виде $x = A^+b$ – предельно простом, прозрачном и охватывающем случаи, различающиеся при других подходах. Например, если исходить из стандартной предпосылки регрессионного анализа $\text{rg}A = n$ (матрица данных имеет полный столбцовый ранг), то $A^+ = (A^T A)^{-1} A^T$ [2] и решение запишется в традиционном для таких задач виде: $x = (A^T A)^{-1} A^T b$, то есть как решение системы нормальных уравнений линейного метода наименьших квадратов: $A^T A x = A^T b$. И действительно [1], решение системы нормальных уравнений начали применять раньше, чем непосредственное решение системы: $Ax = b$.

В [1] подробно исследуются вопросы вычислительной реализации процедур решения этих уравнений, в частности, сопоставляется обусловленность матриц A и $A^T A$. Как подчёркивается в [127], обусловленность матрицы $A^T A$ существенно хуже, чем матрицы A . Отмечается, что при решении плохо обусловленных задач линейного метода наименьших квадратов представляет интерес использование методов псевдообращения, которые могут работать в условиях вырожденности матрицы $A^T A$. В [1, 5] указано на возможность использования $(A^T A)^+$ вместо $(A^T A)^{-1}$ в последнем случае; при этом, как известно [2], вторая запись решения равносильна первой. Отмечается, однако, что наличие ошибок округления делает применение точного псевдообращения нежелательным. Это замечание согласуется с широко обсуждаемой численной неустойчивостью псевдообращения (в [127], однако приведены некоторые положительные результаты). В настоящее время хорошо известны подходы к преодолению этого недостатка – это методы регуляризации (например, [82, 124]), тесно связанные с одним из способов определения псевдообратной матрицы. В нелинейном методе наименьших квадратов та-

кой подход находит отражение в одном из основных алгоритмов – алгоритме Левенберга-Марквардта (параграф 14). С другой стороны, матрица $(A^T A)^{-1}$, как информационная, находит применение в регрессионном анализе при статистической оценке результатов. С этой точки зрения представляется целесообразным при наличии пакета прикладных программ, включающего как подпрограмму псевдообращения, так и подпрограмму обращения симметричной положительно определенной матрицы, использовать обе записи решения, что может послужить дополнительным контролем, столь необходимым при таких расчетах.

Метод псевдообращения принят в качестве основного при решении линейной и нелинейной задач о наименьших квадратах далее. Для нелинейной задачи такой подход не является традиционным. Для решения взвешенных задач о наименьших квадратах может использоваться взвешенное псевдообращение. С этих позиций пособие может рассматриваться как продолжение работ [16, 17].

Задачи о наименьших квадратах допускают по меньшей мере две специфических постановки, истоки которых нагляднее всего прослеживаются в задачах линейной регрессии, где может изменяться как количество членов модели (число параметров n), так и количество данных (объем выборки m). В каждом случае возникает необходимость пересчёта уже имеющихся результатов. Эти ситуации для линейной задачи о наименьших квадратах достаточно подробно исследованы в [2]. Ситуация, когда увеличивается на один (или на некоторое число ρ) количество членов модели, условно названа последовательной или пошаговой (поблочной) регрессией. Допускается и такая трактовка, когда число параметров неизменно, но их оценки рассчитываются последовательно (такая трактовка существенна при переходе к нелинейным моделям). Ситуация, когда увеличивается на единицу (или на некоторое число q) объём выборки, условно названа рекуррентным (поблочным) методом наименьших квадратов. Следует отметить, что в статистике может возникать необходимость в пересчёте результатов и в случае уменьшения объёмов выборки (в качестве примера можно привести так называемый метод «скалярно-

го ножа» [83], когда из выборки последовательно исключают по одному или по несколько наблюдений); может представить интерес и исключение некоторых членов из модели (см., например, [86, 111]). Все эти ситуации находят отражение в изменении размеров матрицы A . Пересчёт результатов основан на использовании алгоритма Гревия рекуррентного псевдообращения матрицы при последовательном добавлении столбцов (строк транспонированной матрицы) [2, 16, 127], формулы Клайна блочного псевдообращения [2, 16] и их сочетания – рекуррентного блочного алгоритма. Соответствующие алгоритмы для линейной задачи о наименьших квадратах представлены в параграфе 13 в систематизированном и ориентированном на использование в параграфе 14 виде.

Большинство из методов нелинейного программирования носит итерационный характер: по текущей точке $x^{(k)}$ определяется следующая точка $x^{(k+1)}$ так, что последовательность точек $\{x^{(k)}\}$ сходится к решению x^* . Расчёт направления $S^{(k)}$ перехода из $x^{(k)}$ в $x^{(k+1)}$ является основной частью таких алгоритмов. В одном из распространённых методов второго порядка – методе Ньютона – направление $S^{(k)}$ ищется как решение уравнения

$$G(x^{(k)})S^{(k)} = -g(x^{(k)}) = -\nabla f(x^{(k)}),$$

где G – матрица Гессе (вторых производных) целевой функции, g – градиент (вектор первых производных). Уже при решении этого уравнения может возникнуть необходимость в применении псевдообращения. Развитие метода Ньютона по пути отказа от непосредственного использования матрицы Гессе и определения направления $S^{(k)}$ только с помощью первых производных целевой функции приводит к методам первого порядка, наиболее распространёнными среди которых являются квазиньютоновские методы или методы переменной метрики, например, методы BFGS и DFP, подробно рассмотренные в параграфе 11 (это изложение базируется на [33, 42, 53, 55]).

В [53] обращено внимание на интригующую двойственность, которой пронизан вывод основных формул этих методов, названных положительно определённой и обратной положительно определённой формулами текущих.

Обращает на себя внимание и присутствие в этих итерационных формулах и их выводах элементов, характерных для формул рекуррентного обращения и псевдообращения матриц специальной природы. Достаточно отметить, что существенно используемая здесь лемма Шермана-Моррисона-Вудбери [42, 53, 55, 80] приводится в [2] именно в контексте рекуррентного псевдообращения. Кроме того, строгий вывод указанных формул глубоко уходит своими корнями в задачи псевдорешения матричных уравнений с ограничениями [53]. Однако отмеченные выше специфические постановки, характерные для задач о наименьших квадратах, в контексте общих задач нелинейного программирования не проявляются.

Нелинейная задача о наименьших квадратах в классе задач нелинейного программирования характеризуется с точки зрения указанных методов тем, что вектор g и матрица G имеют специальную структуру [42, 53] (подробности в параграфе 14): $g = J^T R$, $G = J^T J + Q$, где $J \in \mathbb{R}^{m \times n}$ – матрица Якоби векторной функции векторного аргумента R (образованная записанными по строкам градиентами компонент r_u). Q – член в G , несущий информацию о производных второго порядка. Уравнение для определения направления $S^{(k)}$ записывается в виде:

$$[J^T(x^{(k)})J(x^{(k)}) + Q(x^{(k)})]S^{(k)} = -J^T(x^{(k)})R(x^{(k)}).$$

Здесь также могут быть использованы, с одной стороны, псевдообращение, а с другой стороны – квазиньютоновские методы. При этом отмеченные выше специфические постановки задачи проявляются в полной мере и находят отражение в изменении размеров матриц J и Q . Сочетание итерационных процедур нелинейной оптимизации с рекуррентными алгоритмами псевдообращения приводит к представленным в параграфе 14 рекуррентно-итерационным процедурам нелинейного метода наименьших квадратов. Наиболее законченный вид им удастся придать в ряде случаев, характерных для решения нелинейных задач о наименьших квадратах.

Метод Гаусса-Ньютона относится к методам первого порядка, так как основан на пренебрежении членом Q , включающим вторые производные. В

этом случае

$$S^{(k)} = -[J^T(x^{(k)})J(x^{(k)})]^{-1}J^T(x^{(k)})R(x^{(k)}) = -J^+(x^{(k)})R(x^{(k)}),$$

после чего может применяться подход, изложенный в параграфе 13.

Метод Левенберга-Марквардта основан на замене члена Q выражением μI . В этом случае

$$S^{(k)} = -[J^T(x^{(k)})J(x^{(k)}) + \mu^{(k)}I]^{-1}J^T(x^{(k)})R(x^{(k)}) = -J_\mu^+(x^{(k)})R(x^{(k)}),$$

где матрица J_μ^+ представляет собой регуляризатор матрицы J^+ , для которого также разработаны рекуррентные процедуры пересчёта при изменении размеров матрицы J [2].

Методы квазиньютоновского типа для нелинейной задачи о наименьших квадратах строятся на основе замены члена Q его квазиньютоновскими приближениями $M^{(k)}$. В этом случае

$$S^{(k)} = -[J^T(x^{(k)})J(x^{(k)}) + M^{(k)}]^{-1}J^T(x^{(k)})R(x^{(k)}) = -J_M^+(x^{(k)})R(x^{(k)}),$$

а для определения $M^{(k)}$ служит итерационная процедура, формулируемая на основе квазиньютоновских алгоритмов. Рекуррентный пересчёт при изменении размеров матрицы J здесь организовать уже сложнее. То же можно отнести и к методам полностью ньютоновского типа, скорректированному методу Гаусса-Ньютона и некоторым другим [42, 53, 55, 80].

На возможность использования метода псевдообращения в задачах нелинейного программирования и нелинейного метода наименьших квадратов имеются отдельные указания, например, в [1, 5, 48, 50, 53]. Так, в [53] указан метод решения смешанной линейно-нелинейной задачи о наименьших квадратах, использующий выражение линейно входящих параметров через остальные с помощью псевдообращения. В [5] отмечено, что в методе Гаусса-Ньютона, в случае вырождения матрицы $J^T J$ в текущей точке, то есть в случае неполного ранга матрицы J , вместо $(J^T J)^{-1}$ можно использовать $(J^T J)^+$. Как уже отмечалось, это равносильно использованию J^+ в приведённой вы-

ше формуле метода. С другой стороны, разрабатываются и иные подходы к анализу «вырожденных» случаев нелинейной задачи о наименьших квадратах. Они представлены, например, в [1, 47, 48, 113], где рассматриваются явление мультиколлинеарности и нелинейные модели неполного ранга.

Закljučая краткий обзор методов, рассматриваемых далее, отметим, что вопросы сходимости итерационных процедур уже достаточно глубоко исследованы и продолжают интенсивно исследоваться в отечественной и зарубежной литературе. Читателю, заинтересовавшемуся этими вопросами, можно порекомендовать, например, [42, 53–55, 57, 58, 76, 79, 80, 84, 85, 105] и цитированную там литературу. В [50] указано, что большое число публикаций посвящено алгоритмическому и программному обеспечению задач нелинейного оценивания, например, [1, 5, 47–49, 74, 87, 123].

§ 11. Общая классификация алгоритмов оптимизации

Известно большое число экономических, технических, управленческих и других задач, постановки которых укладываются в общую схему математического программирования. Для решения таких задач эффективно используются различные методы оптимизации, направленные на отыскание минимума (максимума) широких классов функций цели при наличии (или отсутствии) разнообразных ограничений, от структуры и вида которых зависит не только название методов, но и их математическое и логическое построение.

Существует большое количество численных методов оптимизации, которые по виду решаемой задачи можно сгруппировать в следующие разделы математического программирования:

- линейное программирование – раздел математического программирования, изучающий задачу отыскания минимума (максимума) линейной функции многих переменных при линейных ограничениях в виде равенств или неравенств;
- нелинейное программирование – раздел математического программирования, изучающий методы решения и характер экстремумов в задачах

оптимизации с нелинейной целевой функцией и (или) допустимым множеством, определяемым нелинейными ограничениями;

- стохастическое программирование – раздел математического программирования, изучающий модели выбора оптимальных решений в ситуациях, характеризующихся случайными величинами.

Существуют также методы, которые при решении задач оптимизации учитывают специфику этих задач. Такие методы, как правило, превосходят по эффективности общие алгоритмы, и их можно выделить в отдельный класс методов для решения задач специальной структуры:

- а) целочисленное программирование – задачи оптимизации, в которых на значения всех или части переменных целевой функции и (или) ограничений наложено требование целочисленности;
- б) квадратичное программирование – задачи оптимизации с квадратичной целевой функцией и линейными ограничениями;
- в) геометрическое программирование – задачи оптимизации, в которых целевая функция и ограничения представляют собой обобщенные многочлены с положительными переменными (позиномы).

Далее рассматриваются задачи нелинейного программирования, поэтому подробнее остановимся лишь на этом классе методов оптимизации.

Хотя в настоящее время известно довольно много методов нелинейного программирования, вместе с тем ведётся постоянная работа в направлении совершенствования и разработки алгоритмов решения задач этого класса. Это обусловлено тем, что не существует универсальных методов оптимизации, пригодных для решения задач произвольного вида. Для ориентирования во всех этих методах и правильного выбора алгоритма решения конкретной задачи необходимо их систематизировать. Авторы ряда работ (например, [42,54,57]) уже рассматривали численные методы с этих позиций, однако приведённые ими классификации не являются исключительными, и в этом вопросе можно использовать различные подходы. Здесь же авторы предлагают свой вариант систематизации методов нелинейного программирования.

Алгоритмы нелинейного программирования можно классифицировать:

- а) по наличию ограничений:
 - методы безусловной оптимизации,
 - методы условной оптимизации;
- б) по количеству независимых переменных целевой функции и ограничений:
 - методы решения одномерных задач,
 - методы решения многомерных задач;
- в) по способу поиска экстремальной точки:
 - методы прямого поиска,
 - методы косвенного поиска;
- г) по способу определения направлений поиска:
 - методы первого порядка,
 - методы второго порядка;
- д) по способу вычисления производных:
 - градиентные методы,
 - квазиградиентные методы.

Алгоритмы прямого и косвенного поиска для многих задач малоэффективны, поэтому они чаще используются либо как составные части более сложных алгоритмов, либо для выбора начальной точки поиска экстремума другими методами.

1. Безусловная минимизация. В задаче безусловной минимизации требуется найти

$$\min f(x), \quad x \in \mathbb{R}^n \quad (11.1)$$

при отсутствии ограничений на вектор управляемых переменных размерности n ; f – скалярная целевая функция. Вектор x^* называется точкой локального минимума функции f , если существует $\varepsilon > 0$ такое, что

$$f(x^*) \leq f(x), \quad x \in S(x^*, \varepsilon).$$

Часто удобно предполагать, что функция f и её производные существуют и непрерывны. Можно привести известные условия оптимальности, доказательства которых имеются, например, в [53].

Теорема (необходимые условия локального минимума). Пусть x^* – точка локального минимума f , непрерывно дифференцируемой на выпуклом множестве $S \subset \mathbb{R}^n$. Тогда градиент $g(x^*) = \nabla f(x^*) = 0$, а матрица Гессе $G(x^*)$ положительно полуопределена.

Теорема (достаточные условия локального минимума). Пусть f дважды непрерывно дифференцируема на выпуклом множестве $S \subset \mathbb{R}^n$ и $g(x^*) = 0$, $G(x^*) > 0$. Тогда x^* есть точка строго локального минимума.

Вектор x^* называется точкой глобального минимума функции f , если $f(x^*) \leq f(x)$, $x \in \mathbb{R}^n$.

Теорема. Пусть f выпукла на всем \mathbb{R}^n . Тогда при $g(x^*) = 0$ x^* есть точка глобального минимума.

2. Условная минимизация. В задаче условной минимизации требуется найти

$$\min f(x), \quad x \in X, \quad (11.2)$$

где X – заданное множество в \mathbb{R}^n .

Назовём вектор $x^* \in X$ точкой локального условного минимума функции f , если существует $\varepsilon > 0$ такое, что

$$f(x^*) \leq f(x), \quad x \in P(x^*, \varepsilon) = S(x^*, \varepsilon) \cap X.$$

Теорема. Предположим, что множество X выпукло и функция f непрерывно дифференцируема на множестве $P(x^*, \varepsilon)$, где $x^* \in X$, $\varepsilon > 0$. Тогда справедливы следующие утверждения:

а) если x^* – точка локального условного минимума, то

$$g(x^*)(x - x^*) \geq 0, \quad x \in X; \quad (11.3)$$

б) если в дополнение к сделанным предположениям f выпукла на X и имеет место (11.3), то x^* – точка глобального условного минимума.

Обычно в задачах условной оптимизации ограничения учитывают либо с помощью множителей Лагранжа, либо с помощью различных штрафных и барьерных функций. В последнем случае эти задачи преобразуют к такому

виду, что их можно решать с помощью методов безусловной оптимизации. Подробнее с условной оптимизацией можно ознакомиться в работах [9, 42, 105].

3. Функции одной переменной. Поиск экстремального значения функции одной переменной относится к наиболее простому типу задач оптимизации. Однако важность изучения таких задач обусловлена реализацией итерационных процедур, ориентированных на многомерные задачи. Важным свойством функций одной переменной является унимодальность.

Функция $f(x)$ является унимодальной на отрезке $a \leq x \leq b$ в том случае, если она монотонна по обе стороны от единственной на интервале (a, b) оптимальной точки x^* . Существует множество алгоритмов минимизации, использующих это свойство.

Наибольшую известность получили следующие методы:

- исключения интервалов (деления интервала пополам, золотого сечения, дихотомии, Фибоначчи), которые позволяют искать оптимум путем последовательного уменьшения интервалов поиска;
- полиномиальной аппроксимации (квадратичная аппроксимация, кубическая аппроксимация), основная идея которых связана с возможностью аппроксимации гладкой функции полиномом, используемым затем для оценивания координаты точки минимума;
- аппроксимации «секущей прямой» (Ньютона-Рафсона, секущих), основанные на условии существования локального минимума: $f'(x) = 0$.

4. Функции многих переменных. При решении многомерных задач оптимизации трудность заключается не только в увеличении объёма вычислений, хотя это тоже важная характеристика алгоритмов, но и в том, что с увеличением числа измерений пространства уменьшается вероятность унимодальности целевой функции. При этом используют итерационные процедуры, основанные на исследованиях функций одной переменной, а также положения линейной алгебры и дифференциального исчисления. Очень часто при решении многомерных задач эти понятия совмещают. В дальнейшем мы подробнее рассмотрим эти алгоритмы.

5. Методы прямого поиска. Основным достоинством методов прямого поиска является то, что с их помощью можно решать задачи оптимизации с разрывными целевой функцией $f(x)$ и её производными (последние вообще могут не существовать). Однако при этом существенно увеличивается время вычислений. Эти методы основаны на постепенном приближении к оптимуму и сравнении вычисляемых значений целевой функции в различных точках. К алгоритмам этого класса можно отнести: методы исключения областей; методы случайного поиска; методы поиска по симплексу; методы конфигураций (Розенброка, Хука-Дживса, Пауэлла). Хотя эти методы не позволяют найти оптимальное решение с необходимой точностью, они создают подходящие предпосылки для применения в дальнейшем других методов поиска [42, 53, 57, 85].

6. Методы косвенного поиска. Алгоритмы методов косвенного поиска основаны на стремлении найти решение задачи оптимизации, не исследуя неоптимальные точки. Обычно эти методы эффективно используются в составе более сложных алгоритмов для поиска линейного минимума в заданном направлении. В этом случае итерационный процесс строится на определении новых точек $x^{(k+1)} = x^{(k)} + \lambda^{(k)} s^{(k)}$, где $\lambda^{(k)}$ – неотрицательное число, минимизирующее функцию $f(x^{(k)} + \lambda^{(k)} s^{(k)})$ вдоль направления $s^{(k)}$ на k -ой итерации. С помощью методов линейного поиска необходимо определить $\lambda^{(k)}$, обеспечивающее неравенство $f(x^{(k)} + \lambda^{(k)} s^{(k)}) < f(x^{(k)})$. Для этой цели широкое применение получили методы полиномиальной интерполяции.

Метод квадратичной интерполяции. Идея метода заключается в том, что вблизи минимума функции $f(x + \lambda s)$ определяем полином второй степени, совпадающий с $f(x + \lambda s)$ при $\lambda = 0$ и $\lambda = \lambda^{(0)}$, наклон которого (при $\lambda = 0$) совпадает с $\frac{\partial f(x)}{\partial s}$. Обозначим $\hat{f}(\lambda) = f(x + \lambda s)$, тогда имеем $\hat{f}(0) = f(x)$, $\hat{f}'(0) = \nabla^T f(x) s$, $\hat{f}(\lambda^{(0)}) = f(x + \lambda^{(0)} s)$. Первоначальный шаг $\lambda^{(0)}$ обычно определяется как

$$\lambda^{(0)} = \min \left\{ 1, \frac{-2(f(x) - f_0)}{\nabla^T f(x) s} \right\}, \quad (11.4)$$

где f_0 – предполагаемое экстремальное значение целевой функции.

Таким образом, функция $\hat{f}(\lambda)$ моделируется одномерным квадратичным полиномом $P_q(\lambda) = [\hat{f}(\lambda^{(0)}) - \hat{f}(0) - \hat{f}'(0)]\lambda^2 + \hat{f}'(0)\lambda + \hat{f}(0)$.

Вычислим $\lambda^{(1)}$ — точку экстремума этого полинома:

$$\lambda^{(1)} = \frac{1}{2} \frac{-\hat{f}'(0)}{\hat{f}(\lambda^{(0)}) - \hat{f}(0) - \hat{f}'(0)}. \quad (11.5)$$

Метод кубической интерполяции. Отличается от предыдущего алгоритма тем, что строится вблизи точки минимума полином третьей степени. Кубическая интерполяция выполняется на отрезке (a, b) , причём a — точка, в которой производная $\frac{\partial f(x)}{\partial s} < 0$, а b — точка, в которой производная положительна. Тогда значение точки минимума на этом отрезке вычисляется по формуле:

$$\lambda^{(1)} = b - \frac{(\partial f(x + bs)/\partial s - z + w)(b - a)}{\partial f(x - bs)/\partial s - \partial f(x + as)/\partial s + 2w}, \quad (11.6)$$

где

$$z = 3(b - a)^{-1}[f(x + as) - f(x + bs)] + \frac{\partial f(x + as)}{\partial s} + \frac{\partial f(x + bs)}{\partial s},$$

$$w = \left[z^2 - \frac{\partial f(x + as)}{\partial s} \cdot \frac{\partial f(x + bs)}{\partial s} \right]^{\frac{1}{2}}.$$

Метод кубической интерполяции считается наиболее эффективным среди методов линейного поиска. Однако он требует дополнительных вычислений производных. Поэтому в некоторых задачах рациональнее использовать квадратичную интерполяцию или даже методы исключения интервалов. Во всех случаях минимизируется функция $f(x + \lambda s)$ от одной переменной λ .

7. Градиентные методы. Кроме процедур вычисления шага, частично описанных выше, надо определить направление поиска $s^{(k)}$. Расчёт $s^{(k)}$ является главной частью алгоритма оптимизации и определяет его «лицо» и название.

Линейная аппроксимация функции f в окрестности текущей точки $x^{(k)}$ при единичном шаге $\lambda = 1$ вдоль направления $s^{(k)}$, основанная на тейлоров-

ском разложении, имеет вид:

$$f(x^{(k)} + s^{(k)}) \approx f(x^{(k)}) + (g^{(k)})^T s^{(k)}, \quad (11.7)$$

где $g^{(k)}$ – значение градиента минимизируемой функции в текущей точке. Из (11.7) видно, что при поиске минимума f необходимо стремиться к выполнению условия $(g^{(k)})^T s^{(k)} < 0$, для которого в данном случае можно варьировать только вектором $s^{(k)}$, который выбирается на множестве всех нормированных соответствующим образом векторов s , то есть

$$s^{(k)} = \arg \min_{s \in \mathbb{R}^n} = \frac{(g^{(k)})^T s}{\|s\|}. \quad (11.8)$$

Согласно (11.8) $s^{(k)}$ полностью зависит от нормы $\|s\|$, которая может быть задана симметричной, положительно определённой матрицей D . Тогда $\|s\| = \sqrt{s^T D s}$, а

$$s^{(k)} = -D^{-1} g^{(k)}. \quad (11.9)$$

В более простом варианте, если в качестве такой нормы принять евклидову норму $\|s\| = \sqrt{s^T s}$, то (11.9) примет вид:

$$s^{(k)} = -g^{(k)}. \quad (11.10)$$

Тогда вектор $s^{(k)}$ оказывается антиградиентом и является направлением наискорейшего спуска, а алгоритм соответственно называют методом наискорейшего спуска [42].

В ряде случаев, когда матрицы D в (11.9) оказываются слишком большими, а (11.10) дают неудовлетворительные результаты, используют методы, у которых направление поиска $s^{(k)}$ выбирается с учётом информации об изменении градиента при переходе от точки $x^{(k-1)}$ к точке $x^{(k)}$ итерационного процесса:

$$s^{(k)} = -g^{(k)} + \varphi(g^{(k)}, g^{(k-1)}, s^{(k-1)}). \quad (11.11)$$

Существует несколько способов расчёта φ (см., например, [42, 105]). Среди них наиболее популярны методы сопряженных градиентов, Флетчера-Ривса.

Алгоритмы, основанные на формулах (11.9)–(11.11), в полной мере можно отнести к градиентным методам, так как в их основе заложено использование компонентов градиента, которые записаны в аналитическом виде.

8. Квазиградиентные методы. На практике часто приходится сталкиваться с задачами оптимизации, функции цели у которых бывают такие, что очень сложно, а порой даже и невозможно аналитически вычислить значения градиентов (например, с использованием некоторых имитационных моделей). В таком случае необходимо воспользоваться конечно-разностной аппроксимацией истинного градиента.

Заменой аналитического дифференцирования на численное любой градиентный метод можно перевести в класс квазиградиентных, однако и без детального анализа ясно, что эффективность получающегося алгоритма будет пропорциональна точности вычисления производных. Основные формулы конечно-разностной аппроксимации:

$$\frac{\partial f(x)}{\partial x_i} \approx \hat{g}_i(x) = \frac{f(x_i + h_i e_i) - f(x_i)}{h_i}, \quad (11.12)$$

$$\frac{\partial f(x)}{\partial x_i} \approx \hat{g}_i(x) = \frac{f(x_i + h_i e_i) - f(x_i - h_i e_i)}{2h_i}. \quad (11.13)$$

Аппроксимация значений производных по формулам (11.12), (11.13) связана с ошибками следующих типов: ошибка отбрасывания остаточного члена тейлоровского разложения; абсолютные ошибки вычисления функции в точках x_i и $x_i + h_i e_i$; ошибки округления машинных арифметических операций, которыми, как правило, пренебрегают из-за их малости. Существует множество способов численного дифференцирования, однако практически все они основаны на минимизации суммарной ошибки некоторым подбором конечно-разностного интервала, то есть h_i [42, 53]. Некоторые из них будут рассмотрены в параграфе 12.

9. Методы второго порядка. Среди методов второго порядка наиболее известны алгоритмы, основанные на квадратичной аппроксимации минимизируемой функции (например, метод Ньютона). В этих случаях направление $s^{(k)}$ определяется с использованием первых и вторых производных целевой функции уравнением

$$G^{(k)} s^{(k)} = -g^{(k)}, \quad (11.14)$$

где G – матрица Гессе (вторых производных), g – градиент.

Метод Ньютона обладает рядом достоинств:

- при положительно определенной матрице G , независимо от выбора начальной точки, решение выдается за одну итерацию (при $\lambda = 1$);
- использование вторых производных позволяет осуществлять проверку достаточных условий оптимальности.

В связи с этим алгоритмы, использующие (11.9), стараются максимально приблизить к методу Ньютона. Однако у методов, основанных на (11.14), существуют и недостатки:

- при неточном описании целевой функции f её квадратичными приближениями вне окрестностей точек $x^{(k)}$ результат решения задачи оптимизации ньютоновскими методами может оказаться неверным;
- если матрица Гессе не является положительно определённой, то процедура выбора направления поиска $s^{(k)}$ существенно усложняется;
- необходимо использовать точные значения вычисленных аналитически вторых производных;
- необходимо использовать алгоритм обращения матрицы G на каждом шаге итерационного процесса.

10. Методы первого порядка. Идея методов первого порядка основана на использовании при определении направления поиска $s^{(k)}$ лишь первых производных целевой функции. Так, в квазиньютоновских методах матрицу Гессе G в (11.14) заменяют на её аппроксимацию, которая вычисляется на каждом шаге итерационного процесса.

Авторы наиболее распространенных квазиньютоновских алгоритмов (например, DFP, BFGS) использовали процедуры аппроксимации не самой мат-

рицы Гессе, а обратной к ней матрицы. Тогда, с учетом (11.9), направление поиска $s^{(k)}$ вычисляется по формуле

$$s^{(k)} = -H^{(k)}g^{(k)}, \quad (11.15)$$

где H – приближения обращённой матрицы Гессе.

Основными требованиями к матрице H являются её симметричность и положительная определённость.

Таковы основные положения предлагаемой классификации методов нелинейного программирования. Как и всякая вводная классификация, она не является исчерпывающей. В параграфе 12 будут более подробно рассмотрены итерационные методы и, в частности, их модельная схема, квазиньютоновские алгоритмы, методы сопряжённых градиентов и возможности их перевода в класс квазиградиентных процедур оптимизации. Далее в параграфе 14 итерационные методы, специально предназначенные для решения нелинейной задачи о наименьших квадратах, модифицируются в направлении последовательного пересчёта координат текущей точки и учёта изменения числа слагаемых суммы выражения, минимизируемого в нелинейной задаче о наименьших квадратах.

§ 12. Нелинейная оптимизация: итерационные процедуры

Принято считать, что наиболее эффективными и распространенными из перечисленных в параграфе 11 методов являются методы первого и второго порядка. Как правило, эти методы носят итерационный характер и имеют единую модельную схему [42]. На k -й итерации выполняется такая последовательность шагов.

Шаг 1. Проверка соблюдения условий останова.

Если эти условия выполнены, вычисления прекратить и взять найденную точку $x^{(k)}$ в качестве искомого решения x^* , в противном случае перейти к следующему шагу.

Шаг 2. Расчёт направления поиска.

Вычислить ненулевой вектор $s^{(k)}$ – направление поиска.

Шаг 3. Расчёт длины шага итерационного процесса.

Вычислить положительное число $\lambda^{(k)}$, обеспечивающее неравенство

$$f(x^{(k)} + \lambda^{(k)} s^{(k)}) < f(x^{(k)}).$$

Шаг 4. Пересчёт оценки решения.

Положить $x^{(k+1)} \leftarrow x^{(k)} + \lambda^{(k)} s^{(k)}$, $k \leftarrow k + 1$, и вернуться к шагу 1 (уже на $(k + 1)$ -й итерации).

На каждой итерации необходимо контролировать выполнение условия $(g^{(k)})^T s^{(k)} < 0$, где $g^{(k)} = g(x^{(k)})$ – значение градиента минимизируемой функции в текущей точке итерационного процесса. Это условие означает, что вектор $s^{(k)}$ является направлением спуска из $x^{(k)}$.

Эффективность алгоритма, выполненного по модельной схеме, зависит от двух параметров – вектора направления поиска $s^{(k)}$ и шага $\lambda^{(k)}$. Так, при удачном выборе $s^{(k)}$ всё можно испортить неправильным построением $\lambda^{(k)}$ и наоборот. Основные идеи различных способов определения $\lambda^{(k)}$, гарантирующих существенное убывание функции на текущей итерации, изложены практически во всех работах по нелинейному программированию. Главной же частью алгоритма оптимизации, определяющей его «лицо» и название, является расчёт $s^{(k)}$. Рассмотрим его подробнее для двух классов методов первого порядка: квазиньютоновских и градиентных.

Как уже говорилось в параграфе 11, минимизация целевой функции на k -ой итерации в квазиньютоновских методах осуществляется в направлении $s^{(k)} = -H^{(k)} g^{(k)}$.

Очевидно, что матрица $H^{(k)}$ на каждой итерации корректируется таким образом, чтобы для квадратичной функции вида $q(x) = \frac{1}{2}x^T Gx + g^T x$ матрицы $H^{(k)}$ сходились к обращённому гессиану G^{-1} функции q . Существует несколько различных способов выбора нормы представления $H^{(k)}$. В общем виде коррекцию $H^{(k)}$ можно записать в виде [42]

$$H^{(k+1)} = H^{(k)} + \Delta^{(k)}, \quad (12.1)$$

где Δ – корректирующая матрица.

Требуется, чтобы выполнялось соотношение

$$H^{(k+1)} \left[g(x^{(k+1)}) - g(x^{(k)}) \right] = x^{(k+1)} - x^{(k)}. \quad (12.2)$$

Симметричную одноранговую матрицу можно представить в виде:

$$\Delta^{(k)} = \alpha^{(k)} u^{(k)} u^{T(k)},$$

где $u^{(k)}$, $\alpha^{(k)}$ – вектор и скаляр, выбранные так, чтобы выполнилось условие (12.1).

Положим $\delta^{(k)} = x^{(k+1)} - x^{(k)}$, $y^{(k)} = g^{(k+1)} - g^{(k)}$. Определим $\alpha^{(k)}$ и $u^{(k)}$, чтобы

$$H^{(k+1)} y^{(k)} = \delta^{(k)}. \quad (12.3)$$

Представим (12.1), (12.2) в виде $[H^{(k)} + \alpha^{(k)} u^{(k)} u^{T(k)}] y^{(k)} = \delta^{(k)}$, тогда $y^{T(k)} H^{(k)} y^{(k)} + \alpha^{(k)} (y^{T(k)} u^{(k)}) (u^{T(k)} y^{(k)}) = y^{T(k)} \delta^{(k)}$.

Пусть

$$\begin{aligned} \alpha^{(k)} (u^{T(k)} y^{(k)})^2 &= y^{T(k)} (\delta^{(k)} - H^{(k)} y^{(k)}), \\ \alpha^{(k)} u^{(k)} u^{T(k)} &= \frac{(\alpha^{(k)} u^{(k)} u^{T(k)} y^{(k)}) (\alpha^{(k)} u^{(k)} u^{T(k)} y^{(k)})^T}{\alpha^{(k)} (u^{T(k)} y^{(k)})^2}. \end{aligned} \quad (12.4)$$

Заменим $\alpha^{(k)} u^{(k)} u^{T(k)} y^{(k)}$ на $\delta^{(k)} - H^{(k)} y^{(k)}$, $\alpha^{(k)} (u^{(k)} u^{T(k)})^2$ на $y^{T(k)} (\delta^{(k)} - H^{(k)} y^{(k)})$, тогда с учётом (12.3) получим

$$H^{(k+1)} - H^{(k)} = \alpha^{(k)} u^{(k)} u^{T(k)} = \frac{(\delta^{(k)} - H^{(k)} y^{(k)}) (\delta^{(k)} - H^{(k)} y^{(k)})^T}{y^{T(k)} (\delta^{(k)} - H^{(k)} y^{(k)})}.$$

Следовательно,

$$H^{(k+1)} = H^{(k)} + \frac{(\delta^{(k)} - H^{(k)} y^{(k)}) (\delta^{(k)} - H^{(k)} y^{(k)})^T}{y^{T(k)} (\delta^{(k)} - H^{(k)} y^{(k)})}. \quad (12.5)$$

С другой стороны, подставив (12.1) в (12.3), получим

$$\Delta^{(k)} y^{(k)} = \delta^{(k)} - H^{(k)} y^{(k)}. \quad (12.6)$$

Решением этого уравнения является матрица

$$\Delta^{(k)} = \frac{\delta^{(k)} \delta^{T(k)}}{\delta^{T(k)} y^{(k)}} - \frac{H^{(k)} y^{(k)} y^{T(k)} H^{(k)}}{y^{T(k)} H^{(k)} y^{(k)}}. \quad (12.7)$$

Тогда из (12.1) получим формулу, реализующую известный метод Дэвидона-Флетчера-Пауэла (DFP):

$$H^{(k+1)} = H^{(k)} + \frac{\delta^{(k)} \delta^{T(k)}}{\delta^{T(k)} y^{(k)}} - \frac{H^{(k)} y^{(k)} y^{T(k)} H^{(k)}}{y^{T(k)} H^{(k)} y^{(k)}}. \quad (12.8)$$

Если в (12.8) переставить местами $\delta^{(k)}$ и $y^{(k)}$, а матрицу H заменить на W (объяснение матрицы W дано ниже), то можно получить последовательность матриц следующего типа:

$$W^{(k+1)} = W^{(k)} + \frac{y^{(k)} y^{T(k)}}{y^{T(k)} \delta^{(k)}} - \frac{W^{(k)} \delta^{(k)} \delta^{T(k)} W^{(k)}}{\delta^{T(k)} W^{(k)} \delta^{(k)}}. \quad (12.9)$$

Матрицы W будут удовлетворять условию, обратному (12.3):

$$W^{(k+1)} \delta^{(k)} = y^{(k)}. \quad (12.10)$$

Формула (12.9), полученная независимо Бройденом, Флетчером, Голдфарбом, Шанно, позволяет строить оценки прямой матрицы Гессе. Если в (12.9) взять обращение обеих её частей, то получим формулу аппроксимации обратного гессиана H , которая реализуется в алгоритме BFGS:

$$H^{(k+1)} = H^{(k)} + \frac{(\delta^{(k)} - H^{(k)} y^{(k)}) \delta^{T(k)} + \delta^{(k)} (\delta^{(k)} - H^{(k)} y^{(k)})^T}{y^{T(k)} \delta^{(k)}} - \frac{\langle \delta^{(k)} - H^{(k)} y^{(k)}, y^{(k)} \rangle \delta^{(k)} \delta^{T(k)}}{(y^{T(k)} \delta^{(k)})^2}. \quad (12.11)$$

Аналогично, при необходимости, для формулы (12.8) можно построить оценку W , которая в конечном счёте имеет вид:

$$W^{(k+1)} = W^{(k)} + \frac{(y^{(k)} - W^{(k)}\delta^{(k)})y^{T(k)} + y^{(k)}(y^{(k)} - W^{(k)}\delta^{(k)})^T}{\delta^{T(k)}y^{(k)}} - \frac{\langle y^{(k)} - W^{(k)}\delta^{(k)}, \delta^{(k)} \rangle y^{(k)}y^{T(k)}}{(y^{T(k)}\delta^{(k)})^2}. \quad (12.12)$$

Таким образом, формулы DFP и BFGS для $H^{(k)}$ продолжают формулы для пересчёта $W^{(k)}$ путём перехода к обратным матрицам, при этом по структуре формулы меняются местами. В [53, 80] обращено особое внимание на имеющую место двойственность формул и подробно обоснована сходимость указанных алгоритмов, хотя причины, по которым применение этих формул оказывается успешным, всё ещё до конца не поняты.

Квазиньютоновские методы (или, как их ещё называют, методы переменной метрики) оказываются наиболее приспособленными для перевода в класс квазигradientных методов, необходимость в которых на практике обоснована в параграфе 11 тем, что они несут наибольшую информацию для корректного выбора шага численного дифференцирования. В этом случае производные заменяются их конечно-разностными приближениями и эффективность получаемых алгоритмов существенно зависит от точности этой замены.

Подробно ошибки численного дифференцирования рассматриваются в работах [42, 53].

Ошибка численного оценивания производной $f'(x)$, как уже было сказано в параграфе 11, складывается из ошибок отбрасывания, абсолютных ошибок вычисления функции и ошибок округления. Рассмотрим для простоты одномерный случай, когда конечно-разностная формула аппроксимации производной имеет вид:

$$\varphi(f, h) = \frac{f(x + h) - f(x)}{h}. \quad (12.13)$$

Ошибка отбрасывания равна остаточному члену тейлоровского разложения, то есть

$$\varphi(f, h) - f'(x) = O(h). \quad (12.14)$$

Абсолютная ошибка равна ошибке вычисления функции в точках x и $x + h$. Это значит, что вместо $f(x)$ и $f(x + h)$ в правую часть формулы (12.13) войдут величины $\bar{f}(x)$ и $\bar{f}(x + h)$. Тогда $\bar{f}(x) = f(x) + \gamma$, $\bar{f}(x + h) = f(x + h) + \gamma_h$, где $|\gamma|$, $|\gamma_h|$ – абсолютные ошибки вычисления функции в точках x и $x + h$. Численной оценкой производной при отсутствии прочих ошибок будет величина

$$\varphi(\bar{f}, h) = \frac{\bar{f}(x + h) - \bar{f}(x)}{h}$$

или

$$\varphi(\bar{f}, h) = \frac{f(x + h) - f(x)}{h} + \frac{\gamma_h - \gamma}{h} = \varphi(f, h) + O\left(\frac{\Delta\gamma}{h}\right). \quad (12.15)$$

Ошибки округления возникают при машинных арифметических расчётах. Погрешность при этом довольно мала, поэтому в ряде случаев ошибкой машинной арифметики пренебрегают.

Таким образом, первая из ошибок прямо пропорциональна конечно-разностному интервалу h , а вторая – обратно пропорциональна его величине. Следовательно, суммарная ошибка, допускаемая при численном дифференцировании, зависит от того, насколько удачно выбран шаг h .

Существует ряд методик выбора конечно-разностных интервалов с использованием матриц Гессе или их оценок, основанных на минимизации вычисляемой оценки суммарной ошибки.

При использовании формулы (11.12), согласно работе [79], шаг h_i в многомерных задачах определяется из решения уравнения, полученного путём разложения в ряд Тейлора $f(x_i + h_i e_i)$ и пренебрежения членами высоких порядков:

$$O(h) + O\left(\frac{\Delta\gamma}{h}\right) \leq \tilde{O} = 2|f(x)| \frac{\eta_f}{|h_i|} + \eta_f \bar{g}_i + \frac{1}{2}(1 + \eta_f)|h_i A_{ii}|, \quad (12.16)$$

где η_f – верхний предел относительной погрешности вычисления функции для всех точек x ; \bar{g}_i – оценка первой производной; A_{ii} – оценка второй производной по i -ой координате.

Из требования минимизации суммарной ошибки получим:

$$h_i = 2\sqrt{\frac{\eta_f |f(x)|}{|A_{ii}|(1 + \eta_f)}}. \quad (12.17)$$

Аналогично рассуждая, для формулы (11.13) конечно-разностный интервал определяем в виде:

$$h_i = |A_{ii}|^{-1} \left[-|\bar{g}_i| + \sqrt{\bar{g}_i^2 + 2|A_{ii}f(x)|\frac{\varepsilon}{\varepsilon_1}} \right], \quad (12.18)$$

где ε – относительная погрешность вычисления значений функции в точке x , $\varepsilon \leq \eta_f$; ε_1 – верхний предел допустимой относительной ошибки в вычислении производной.

Рекомендуется выбирать ε_1 по крайней мере на порядок выше ε , а в качестве \bar{g}_i – оценку $\frac{\partial f(x)}{\partial x_i}$, вычисленную на предыдущей итерации процесса.

Предлагается матрицу для метода DFP задавать в виде:

$$A_{ii}^{(k+1)} = A_{ii}^{(k)} + \left[(\beta_i \lambda_i^{(k)})^{-1} - \rho_i \beta_i^{-1} \right] y^{(k)} y^{T(k)} + \beta_i^{-1} (g^{(k)} y^{T(k)} + y^{(k)} g^{T(k)}), \quad (12.19)$$

где $\beta = \langle y^{(k)}, s^{(k)} \rangle$, $\rho = \langle g^{(k)}, s^{(k)} \rangle$.

Однако практически реализация алгоритмов на ЭВМ показала преимущество использования ii -ых элементов матриц $W^{(k)}$ формул (12.9), (12.12) в методах DFP и BFCS соответственно вместо A_{ii} . Перейдём к рассмотрению градиентных методов.

Среди наиболее популярных градиентных методов, использующих условия (11.8)-(11.11), является метод сопряжённых градиентов Флетчера-Ривса с формулой пересчёта направлений на k -ой итерации:

$$s^{(k)} = -g^{(k)} + \sum_{i=0}^{k-1} \nu^{(i)} s^{(i)}, \quad s^{(0)} = -g^{(0)}. \quad (12.20)$$

Значения $\nu^{(i)}$, $i = 1, 2, \dots, k - 1$, выбираются из условия сопряжения построенных ранее (до k -ой итерации) направлений к матрице D в (11.9):

$$\nu^{(k-1)} = \frac{\|g^{(k)}\|^2}{\|g^{(k-1)}\|^2}. \quad (12.21)$$

При переводе таких методов в класс квазиградиентных одним из главных недостатков является невозможность использования эффективных процедур определения оценок производных целевой функции на k -ой итерации с учётом информации о её кривизне на предыдущих итерациях, как это было в квазиньютоновских методах. Тем не менее, авторы работ [42, 53] предлагают алгоритмы определения конечно-разностных интервалов h_i , реализация которых на ЭВМ даёт приемлемые результаты. Так, в [42] в качестве оценки минимизации суммарной ошибки рассматривается величина

$$\frac{h}{2}|f''(\mu)| + \frac{2}{h}\delta_f, \quad (12.22)$$

где μ – некоторое число из отрезка $(x, x+h)$; δ_f – оценка абсолютной ошибки вычисления f в x и в $(x+h)$, ограниченная сверху.

Из (12.6) получается, что минимум ошибки достигается при

$$\hat{h} = 2\sqrt{\frac{\delta_f}{|f''(\mu)|}}. \quad (12.23)$$

Но при этом в (12.23) неизвестны ни производная f'' точка μ . Следовательно, для нахождения оценки \hat{h} необходимы дополнительные предположения.

Удобнее всего говорить о точности вычислений в относительных единицах, то есть о значащих цифрах в машинной арифметике. Верхней границей относительной погрешности вычисления f является число $\eta_f = 10^{-t} \geq \geq macheps$, где t – число достоверных значащих цифр машинного значения f . Величины δ_f и η_f связаны между собой выражением

$$\delta_f = |f|\eta_f. \quad (12.24)$$

Для аппроксимации f'' предлагается использовать формулу

$$\Phi(f, h_\Phi) = \frac{f(x + h_\Phi) - 2f(x) + f(x - h_\Phi)}{h_\Phi^2}. \quad (12.25)$$

Здесь погрешность также имеет порядок $O(h)$ для достаточно малых h_Φ , но необходимо чаще вычислять значение функции. Кроме того, определение h_Φ существенно увеличивает вычислительные затраты.

При решении задач с использованием формулы (11.12) выражение (12.23)-(12.25) переносятся без изменений на многомерный случай. При этом конечно-разностные интервалы h_i строятся независимо друг от друга по (12.23). Однако построение этих интервалов на каждом шаге основного процесса по (12.23) авторы [42] считают нецелесообразным. Предлагается использовать эту процедуру один раз – в начальной точке $x^{(0)}$. Обозначим через \hat{h}_i оценку оптимального абсолютно конечно-разностного интервала, вычисленную в $x^{(0)}$, а через δ_i – набор относительных интервалов, которые определяют как

$$\delta_i = \frac{\hat{h}_i}{1 + \tau_i |x^{(0)}|}, \quad (12.26)$$

где τ_i – параметр, удовлетворяющий условию $0 \leq \tau_i \leq 1$; обычно τ_i полагают равным единице.

Тогда на k -ой итерации конечно-разностные интервалы вычисляются по формуле

$$h_i = \delta_i (1 + \tau_i |x_i^{(k)}|). \quad (12.27)$$

Обычно оптимальные значения относительных интервалов устойчивее абсолютных, но в тех редких случаях, когда из-за h_i возникают трудности, предусматривается возможность повторных обращений к формуле (12.23).

В работе [53] предложен алгоритм минимизации суммарного влияния ошибок аппроксимация производных за счёт балансировки $O(h)$ и $O\left(\frac{\Delta\gamma}{h}\right)$ и с учётом достоверных разрядов прикомпьютерных вычислений. Если $f(x)$ имеет t достоверных десятичных разрядов, то целесообразно стремиться к тому, чтобы $f(x + h_i e_i)$ отличалось от $f(x)$ правой половиной этих разрядов.

Для правой конечно-разностной формулы (11.12) рекомендуемая длина шага

$$h_i = \sqrt{\eta_f} \max \{|x_i|, bx_i\} \operatorname{sign}(x_i), \quad (12.28)$$

где bx_i – характерный размер x_i , задаваемой пользователем.

Соответствующие элементы $g(x)$ и $\hat{g}(x)$ будут приблизительно совпадать в их первых $\frac{t}{2}$ десятичных разрядах. Величина bx_i характеризует параметры масштабирования и задаётся, когда ожидается, что величины x будут сильно различаться. Предлагаемое значение по умолчанию $bx_i = 1$, $i = 1, \dots, n$.

При использовании формулы центральных разностей (11.13) длина шага вычисляется как

$$h_i = \eta_f^{\frac{1}{3}} \max \{|x_i|, bx_i\} \operatorname{sign}(x_i). \quad (12.29)$$

Обычно аппроксимация градиента по правым разностям оказывается вполне достаточной и в два раза дешевле, чем аппроксимация по центральным разностям, так как последняя требует больших вычислений значений функции. Однако точность аппроксимации по формуле (11.12) ограничена и в некоторых редких случаях целесообразнее использовать формулу (11.13). При этом желательно, чтобы переключение в программах с (11.12) на (11.13) и наоборот происходило автоматически. Этот переход производится в том случае, когда на какой-либо итерации использование аппроксимации градиента по правой конечно-разностной формуле не приводит к достаточному убыванию функции.

§ 13. Линейный метод наименьших квадратов: рекуррентные процедуры

Как указано в предисловии к главе, линейная задача о наименьших квадратах состоит в отыскании набора параметров x_1, \dots, x_n из условия: найти

$$\min_{x \in \mathbb{R}^n} f(x) = \frac{1}{2} \sum_{u=1}^m \left[b_u - \sum_{i=1}^n x_i a_i(\xi_u) \right]^2. \quad (13.1)$$

Введём обозначения:

$$A = A^{\langle m, n \rangle} = \begin{pmatrix} a_1(\xi_1) & \dots & a_n(\xi_1) \\ \vdots & a_i(\xi_u) & \vdots \\ a_1(\xi_m) & \dots & a_n(\xi_m) \end{pmatrix} \in \mathbb{R}^{m \times n},$$

$$x = x^{\langle m, n \rangle} = \begin{pmatrix} x_1 \\ \vdots \\ x_i \\ \vdots \\ x_n \end{pmatrix} \in \mathbb{R}^n, \quad b = b^{\langle m \rangle} = \begin{pmatrix} b_1 \\ \vdots \\ b_u \\ \vdots \\ b_m \end{pmatrix} \in \mathbb{R}^m.$$

В этих векторно-матричных обозначениях соотношение (13.1) запишется в следующем виде: найти

$$\min_{x \in \mathbb{R}^n} f(x) = \frac{1}{2} \|b - Ax\|^2 = \frac{1}{2} \varepsilon, \quad (13.2)$$

где $\varepsilon = \|b - Ax\|^2$ – невязка уравнения

$$Ax = b. \quad (13.3)$$

В редких случаях матрица A уравнения (13.3) оказывается квадратной и невырожденной; в этом случае его единственное решение

$$x = A^{-1}b \quad (13.4)$$

является и решением задачи (13.2) (или, что то же, (13.1)) с нулевой невязкой.

Чаще встречаются случаи, когда матрица $A \in \mathbb{R}^{m \times n}$ является матрицей общего вида, но оказывается выполненным условие совместности уравнения (13.3), имеющее вид:

$$AA^+b = b, \quad (13.5)$$

где $A^+ \in \mathbb{R}^{n \times m}$ – псевдообратная к матрице A ; в этом случае общее решение уравнения записывается в виде:

$$x = A^+b + (I - A^+A)y, \quad (13.6)$$

где $I \in \mathbb{R}^{n \times n}$ – единичная матрица, $y \in \mathbb{R}^n$ – произвольный вектор; любой вектор вида (13.6) является и решением задачи (13.2) с нулевой невязкой. Это решение единственно, если выполняется наряду с (13.5) условие

$$A^+A = I, \quad (13.7)$$

и записывается в виде:

$$x = A^+b. \quad (13.8)$$

В других случаях выбор единственного решения из (13.6) подчиняется тем или иным дополнительным требованиям; таково, например, требование: найти

$$\min_{x=A^+b+(I-A^+A)y} \|x\|. \quad (13.9)$$

Решение задачи (13.3), (13.9) называется нормальным; оно также записывается в виде (13.8).

Наиболее реальный случай, когда матрица имеет общий вид и условие (13.5) не выполняется, так что решение уравнения (13.3) не существует. Замечателен тот факт, что и в этих случаях соотношение (13.6) даёт решение задачи (13.2), а соотношение (13.3) – решение задачи (13.2), (13.9); для уравнения (13.3) они называются соответственно псевдорешением и нормальным псевдорешением.

В предисловии к главе указано, что необходимость в рекуррентных процедурах решения задачи о наименьших квадратах возникает, по крайней мере, в связи со следующими ситуациями: увеличивается число слагаемых во внутренней (стоящей в квадратных скобках) сумме соотношения (13.1); увеличивается число слагаемых во внешней сумме соотношения (13.1). Рассмотрим подробнее эти ситуации.

Предварительно, для удобства дальнейших формулировок, уточним обозначения векторов и матрицы в (13.2): найти

$$\min_{x^{\langle m,n \rangle} \in \mathbb{R}^n} \frac{1}{2} \|b^{\langle m \rangle} - A^{\langle m,n \rangle} x^{\langle m,n \rangle}\|^2 = \frac{1}{2} \varepsilon^{\langle m,n \rangle}. \quad (13.10)$$

Пусть число слагаемых во внутренней сумме увеличилось на единицу так, что она приняла вид

$$\sum_{i=1}^{n+1} x_i^{\langle m,n+1 \rangle} a_i(\xi_u) = \sum_{i=1}^{n+1} x_i^{\langle m,n+1 \rangle} a_i(\xi_u) + x_{n+1}^{\langle m,n+1 \rangle} a_{n+1}(\xi_u),$$

и решается задача: найти

$$\min_{x^{\langle m,n+1 \rangle} \in \mathbb{R}^{n+1}} \frac{1}{2} \|b^{\langle m \rangle} - A^{\langle m,n+1 \rangle} x^{\langle m,n+1 \rangle}\|^2 = \frac{1}{2} \varepsilon^{\langle m,n+1 \rangle}, \quad (13.11)$$

причём матрица $A^{\langle m,n+1 \rangle}$ связана с матрицей $A^{\langle m,n \rangle}$ из (13.10) соотношением $A^{\langle m,n+1 \rangle} = (A^{\langle m,n \rangle} | a^{\langle m,n+1 \rangle})$, где дополнительный столбец имеет вид

$$a^{\langle m,n+1 \rangle} = (a_{n+1}(\xi_1), \dots, a_{n+1}(\xi_m))^T,$$

а вектор $x^{\langle m,n+1 \rangle}$ имеет структуру $x^{\langle m,n+1 \rangle} = \left(x_{(n)}^{\langle m,n+1 \rangle T} | x_{n+1}^{\langle m,n+1 \rangle} \right)^T$, где

$$x_{(n)}^{\langle m,n+1 \rangle} = \left(x_1^{\langle m,n+1 \rangle}, \dots, x_n^{\langle m,n+1 \rangle} \right)^T.$$

В соответствии с (13.8) решение задачи (13.11), (13.9) запишется в виде:

$$x^{\langle m,n+1 \rangle} = A^{\langle m,n+1 \rangle +} b^{\langle m \rangle}$$

или

$$\begin{pmatrix} x_{(n)}^{\langle m,n+1 \rangle} \\ \dots \\ x_{n+1}^{\langle m,n+1 \rangle} \end{pmatrix} = \left(A^{\langle m,n \rangle} | a^{\langle m,n+1 \rangle} \right)^+ b^{\langle m \rangle}. \quad (13.12)$$

Известный алгоритм Гревия рекуррентного псевдообращения матриц, записываемый «по наращиванию столбцов», определяет следующее выраже-

ние этого решения через решение $x^{\langle m,n \rangle}$ задачи (13.10), (13.9):

$$x_{n+1}^{\langle m,n+1 \rangle} = l^{\langle m,n+1 \rangle T} b^{\langle m \rangle}, \quad (13.13)$$

$$x_{(n)}^{\langle m,n+1 \rangle} = x^{\langle m,n \rangle} - A^{\langle m,n \rangle +} a^{\langle m,n+1 \rangle} x_{n+1}^{\langle m,n+1 \rangle}, \quad (13.14)$$

где вектор $l^{\langle m,n+1 \rangle T}$ определяется в виде

$$l^{\langle m,n+1 \rangle T} = \begin{cases} c^{\langle m,n+1 \rangle +} = \frac{c^{\langle m,n+1 \rangle T}}{\|c^{\langle m,n+1 \rangle}\|^2}, \\ \text{если } c^{\langle m,n+1 \rangle} = \left(I - A^{\langle m,n \rangle} A^{\langle m,n \rangle +}\right) a^{\langle m,n+1 \rangle} \neq 0; \\ \text{в остальных случаях} \\ \frac{a^{\langle m,n+1 \rangle T} \left(A^{\langle m,n \rangle +}\right)^T A^{\langle m,n \rangle +}}{k^{\langle m,n+1 \rangle}}, \\ k^{\langle m,n+1 \rangle} = 1 + \|A^{\langle m,n \rangle +} a^{\langle m,n+1 \rangle}\|^2. \end{cases} \quad (13.15)$$

Первому варианту этой формулы соответствует, например, случай, когда матрица $A^{\langle m,n \rangle} \in \mathbb{R}_n^{m \times n}$, то есть $\text{rg} A^{\langle m,n \rangle} = n$, а второму варианту – случай, когда матрица $A^{\langle m,n \rangle} \in \mathbb{R}_m^{m \times n}$, то есть $\text{rg} A^{\langle m,n \rangle} = m$; в последнем случае выражение для $l^{\langle m,n+1 \rangle T}$ упрощается:

$$l^{\langle m,n+1 \rangle T} = \frac{a^{\langle m,n+1 \rangle T} \left(A^{\langle m,n \rangle} A^{\langle m,n \rangle T}\right)^{-1}}{1 + a^{\langle m,n+1 \rangle T} \left(A^{\langle m,n \rangle} A^{\langle m,n \rangle T}\right)^{-1} a^{\langle m,n+1 \rangle}}. \quad (13.16)$$

Пусть теперь число слагаемых во внутренней сумме увеличилось на p так, что она приняла вид

$$\sum_{i=1}^{n+p} x_i^{\langle m,n+p \rangle} a_i(\xi_u) = \sum_{i=1}^n x_i^{\langle m,n+p \rangle} a_i(\xi_u) + \sum_{i=n+1}^{n+p} x_i^{\langle m,n+p \rangle} a_i(\xi_u),$$

и решается задача: найти

$$\min_{x^{\langle m,n+p \rangle} \in \mathbb{R}^{n+p}} \frac{1}{2} \|b^{\langle m \rangle} - A^{\langle m,n+p \rangle} x^{\langle m,n+p \rangle}\|^2 = \frac{1}{2} \varepsilon^{\langle m,n+p \rangle}, \quad (13.17)$$

причём матрица $A^{\langle m, n+p \rangle}$ связана с матрицей $A^{\langle m, n \rangle}$ из (13.10) соотношением

$$A^{\langle m, n+p \rangle} = \begin{pmatrix} A^{\langle m, n \rangle} & \vdots & A^{\langle m, (p) \rangle} \end{pmatrix},$$

где дополнительный блок имеет вид:

$$A^{\langle m, (p) \rangle} = \begin{pmatrix} a_{n+1}(\xi_1) & \dots & a_{n+p}(\xi_1) \\ \vdots & & \vdots \\ a_{n+1}(\xi_m) & \dots & a_{n+p}(\xi_m) \end{pmatrix},$$

а вектор $x^{\langle m, n+p \rangle}$ имеет структуру

$$x^{\langle m, n+p \rangle} = \begin{pmatrix} x_{(n)}^{\langle m, n+p \rangle T} & \vdots & x_{(p)}^{\langle m, n+p \rangle T} \end{pmatrix},$$

где

$$x_{(n)}^{\langle m, n+p \rangle} = \begin{pmatrix} x_1^{\langle m, n+p \rangle} \\ \vdots \\ x_n^{\langle m, n+p \rangle} \end{pmatrix}, \quad x_{(p)}^{\langle m, n+p \rangle} = \begin{pmatrix} x_{n+1}^{\langle m, n+p \rangle} \\ \vdots \\ x_{n+p}^{\langle m, n+p \rangle} \end{pmatrix}.$$

В соответствии с (13.8) решение задачи (13.17), (13.9) запишется в виде:

$$x^{\langle m, n+p \rangle} = A^{\langle m, n+p \rangle} b^{\langle m \rangle}$$

или

$$\begin{pmatrix} x_{(n)}^{\langle m, n+p \rangle} \\ \dots \\ x_{(p)}^{\langle m, n+p \rangle} \end{pmatrix} = \left(A^{\langle m, n \rangle} | A^{\langle m, (p) \rangle} \right)^+ b^{\langle m \rangle}. \quad (13.18)$$

Известная формула Клайна псевдообращения блочных матриц определяет следующее выражение этого решения через решение задачи (13.10), (13.9):

$$x_{(p)}^{\langle m, n+p \rangle} = L^{\langle m, n+p \rangle} b^{\langle m \rangle}, \quad (13.19)$$

$$x_{(n)}^{\langle m, n+p \rangle} = x^{\langle m, n \rangle} - A^{\langle m, n \rangle +} A^{\langle m, (p) \rangle} x_{(p)}^{\langle m, n+p \rangle}, \quad (13.20)$$

где матрица $L^{\langle m, n+p \rangle}$ определяется в виде:

$$L^{\langle m, n+p \rangle} = C^{\langle m, n+p \rangle +} + \left(I - C^{\langle m, n+p \rangle +} C^{\langle m, n+p \rangle} \right) \times$$

$$\times K^{\langle m, n+p \rangle} \left(A^{\langle m, n \rangle +} A^{\langle m, (p) \rangle} \right)^T A^{\langle m, n \rangle +} \left(I - A^{\langle m, (p) \rangle} C^{\langle m, n+p \rangle +} \right),$$

причём

$$\begin{aligned} C^{\langle m, n+p \rangle} &= \left(I - A^{\langle m, n \rangle} A^{\langle m, n \rangle +} \right) A^{\langle m, (p) \rangle}, \\ K^{\langle m, n+p \rangle} &= \left(I + \left[A^{\langle m, n \rangle +} A^{\langle m, (p) \rangle} \left(I - C^{\langle m, n+p \rangle +} C^{\langle m, n+p \rangle} \right) \right]^T \times \right. \\ &\quad \left. \times \left[A^{\langle m, n \rangle +} A^{\langle m, (p) \rangle} \left(I - C^{\langle m, n+p \rangle +} C^{\langle m, n+p \rangle} \right) \right] \right)^{-1}. \end{aligned} \quad (13.21)$$

Выражение для $L^{\langle m, n+p \rangle}$ в ряде случаев упрощается. Так, если матрица $A^{\langle m, n \rangle} \in \mathbb{R}_m^{m \times n}$, то есть $\text{rg} A^{\langle m, n \rangle} = m$, то $C^{\langle m, n+p \rangle} = 0$ и

$$\begin{aligned} L^{\langle m, n+p \rangle} &= \left(I + A^{\langle m, (p) \rangle T} \left[A^{\langle m, n \rangle} A^{\langle m, n \rangle T} \right]^{-1} A^{\langle m, (p) \rangle} \right)^{-1} \times \\ &\quad \times A^{\langle m, (p) \rangle T} \left[A^{\langle m, n \rangle} A^{\langle m, n \rangle T} \right]^{-1} \end{aligned} \quad (13.22)$$

(это выражение непосредственно обобщает (13.16)).

Если матрица $A^{\langle m, n+p \rangle} \in \mathbb{R}_{n+p}^{m \times (n+p)}$, то есть $\text{rg} A^{\langle m, n+p \rangle} = n + p$, то

$$L^{\langle m, n+p \rangle} = C^{\langle m, n+p \rangle +} \quad (13.23)$$

(это выражение непосредственно обобщает первый вариант формулы (13.15)).

Если матрица $A^{\langle m, n+p \rangle} \in \mathbb{R}_m^{m \times (n+p)}$, то есть $\text{rg} A^{\langle m, n+p \rangle} = m$, то

$$L^{\langle m, n+p \rangle} = \left(A^{\langle m, (p) \rangle T} C^{\langle m, n+p \rangle} \right)^{-1} C^{\langle m, n+p \rangle T} \quad (13.24)$$

(аналогов этой формулы ранее не было).

Следует отметить, что при использовании в (13.19), (13.20) формулы (13.23) для $L^{\langle m, n+p \rangle}$ справедливо соотношение

$$\varepsilon^{\langle m, n+p \rangle} = \varepsilon^{\langle m, n \rangle} - \|C^{\langle m, n+p \rangle} x_{(p)}^{\langle m, n+p \rangle}\|^2, \quad (13.25)$$

а при использовании формулы (13.24) – соотношение

$$\varepsilon^{\langle m, n+p \rangle} = \varepsilon^{\langle m, n \rangle} - \left(A^{\langle m, (p) \rangle} x_{(p)}^{\langle m, n+p \rangle} \right)^T \left(I - A^{\langle m, n \rangle} A^{\langle m, n \rangle +} \right) b^{\langle m \rangle}. \quad (13.26)$$

Вычитаемые в этих соотношениях, как и в (13.14), (13.20), трактуются как «корректирующие члены» при увеличении числа слагаемых во внутренней сумме выражения (13.1).

Перейдём к рассмотрению другой из указанных выше ситуаций, обуславливающих необходимость в рекуррентных процедурах. Пусть увеличилось на единицу число слагаемых во внешней сумме соотношения (13.1), так что она приняла вид:

$$\sum_{u=1}^{m+1} \left(b_u - \sum_{i=1}^n x_i^{\langle m+1, n \rangle} a_i(\xi_u) \right)^2,$$

и решается задача: найти

$$\min_{x^{\langle m+1, n \rangle} \in \mathbb{R}^n} \frac{1}{2} \|b^{\langle m+1 \rangle} - A^{\langle m+1, n \rangle} x^{\langle m+1, n \rangle}\|^2 = \frac{1}{2} \varepsilon^{\langle m+1, n \rangle}, \quad (13.27)$$

причём матрица $A^{\langle m+1, n \rangle}$ связана с матрицей $A^{\langle m, n \rangle}$ из (13.10) соотношением

$$A^{\langle m+1, n \rangle} = \begin{pmatrix} A^{\langle m, n \rangle} \\ \dots \\ a^{\langle m+1, n \rangle} \end{pmatrix},$$

где дополнительная строка имеет вид:

$$a^{\langle m+1, n \rangle} = (a_1(\xi_{m+1}) \dots a_n(\xi_{m+1})),$$

а вектор $b^{\langle m+1 \rangle}$ имеет структуру:

$$b^{\langle m+1 \rangle} = \left(b^{\langle m \rangle T} \quad \vdots \quad b_{m+1} \right)^T.$$

В соответствии с (13.8) решение задачи (13.27), (13.9) запишется в виде:

$$x^{\langle m+1, n \rangle} = A^{\langle m+1, n \rangle +} b^{\langle m+1 \rangle}$$

или

$$x^{\langle m+1, n \rangle} = \begin{pmatrix} A^{\langle m, n \rangle} \\ \dots \\ a^{\langle m+1, n \rangle} \end{pmatrix}^+ \begin{pmatrix} b^{\langle m \rangle} \\ \dots \\ b_{m+1} \end{pmatrix}. \quad (13.28)$$

Алгоритм Гревия рекуррентного псевдообращения матриц, записанный «по наращиванию строк», определяет следующее выражение этого решения через решение $x^{\langle m, n \rangle}$ задачи (13.10), (13.9):

$$x^{\langle m+1, n \rangle} = x^{\langle m, n \rangle} + l^{\langle m+1, n \rangle} \left(b_{m+1} - a^{\langle m+1, n \rangle} x^{\langle m, n \rangle} \right), \quad (13.29)$$

где вектор $l^{\langle m+1, n \rangle}$ определяется в виде:

$$l^{\langle m+1, n \rangle} = \begin{cases} c^{\langle m+1, n \rangle} = \frac{c^{\langle m+1, n \rangle T}}{\|c^{\langle m+1, n \rangle}\|^2}, \\ \text{если } c^{\langle m+1, n \rangle} = a^{\langle m+1, n \rangle} \left(I - A^{\langle m, n \rangle} A^{\langle m, n \rangle} \right) \neq 0; \\ \text{в остальных случаях} \\ \frac{A^{\langle m, n \rangle} A^{\langle m, n \rangle} a^{\langle m+1, n \rangle T}}{k^{\langle m+1, n \rangle}}, \\ k^{\langle m+1, n \rangle} = 1 + \|a^{\langle m+1, n \rangle} A^{\langle m, n \rangle}\|^2. \end{cases} \quad (13.30)$$

Первому варианту этой формулы соответствует, например, случай, когда матрица $A^{\langle m, n \rangle} \in \mathbb{R}_m^{m \times n}$, то есть $\text{rg} A^{\langle m, n \rangle} = m$, а второму варианту – случай, когда матрица $A^{\langle m, n \rangle} \in \mathbb{R}_n^{m \times n}$, то есть $\text{rg} A^{\langle m, n \rangle} = n$; в последнем случае выражение для $l^{\langle m+1, n \rangle}$ упрощается:

$$l^{\langle m+1, n \rangle} = \frac{\left(A^{\langle m, n \rangle T} A^{\langle m, n \rangle} \right)^{-1} a^{\langle m+1, n \rangle T}}{1 + a^{\langle m+1, n \rangle} \left(A^{\langle m, n \rangle T} A^{\langle m, n \rangle} \right)^{-1} a^{\langle m+1, n \rangle T}} \quad (13.31)$$

(это выражение является аналогом (13.16)).

Следует отметить определенное отличие структуры рекуррентного выражения (13.29) от рекуррентных выражений (13.13), (13.14). А именно, в (13.29) «корректирующий член» пропорционален разности между добав-

ленным значением b_{m+1} и его «предсказанной» по предыдущим значениям b_1, \dots, b_m величиной $a^{\langle m+1, n \rangle} x^{\langle m, n \rangle}$, так что новое значение $x^{\langle m, n \rangle}$ и члена, пропорционального ошибке предсказания по нему добавленного значения b_{m+1} . Вектор пропорциональности $l^{\langle m+1, n \rangle}$, часто называемый «вектором сглаживания», не зависит от значений b_u , $u = 1, \dots, m+1$.

Пусть теперь число слагаемых во внешней сумме соотношения (13.1) увеличилось на q , так что она приняла вид

$$\sum_{u=1}^{m+q} \left(b_u - \sum_{i=1}^n x_i^{\langle m+q, n \rangle} a_i(\xi_u) \right)^2,$$

и решается задача: найти

$$\min_{x^{\langle m+q, n \rangle} \in \mathbb{R}^n} \frac{1}{2} \|b^{\langle m+q \rangle} - A^{\langle m+q, n \rangle} x^{\langle m+q, n \rangle}\|^2 = \frac{1}{2} \varepsilon^{\langle m+q, n \rangle}, \quad (13.32)$$

причём матрица $A^{\langle m+q, n \rangle}$ связана с матрицей $A^{\langle m, n \rangle}$ из (13.10) соотношением

$$A^{\langle m+q, n \rangle} = \begin{pmatrix} A^{\langle m, n \rangle} \\ \dots \\ A^{\langle (q), n \rangle} \end{pmatrix},$$

где дополнительный блок имеет вид:

$$A^{\langle (q), n \rangle} = \begin{pmatrix} a_1(\xi_{m+1}) & \dots & a_n(\xi_{m+1}) \\ \vdots & & \vdots \\ a_1(\xi_{m+q}) & \dots & a_n(\xi_{m+q}) \end{pmatrix},$$

а вектор $b^{\langle m+q \rangle}$ имеет структуру:

$$b^{\langle m+q \rangle} = \begin{pmatrix} b^{\langle m \rangle T} & \vdots & b^{\langle (q) \rangle T} \end{pmatrix}^T,$$

где $b^{\langle m \rangle}$ определён, как и выше, а

$$b^{\langle (q) \rangle} = (b_{m+1} \dots b_{m+q})^T.$$

В соответствии с (13.8) решение задачи (13.32), (13.9) запишется в виде:

$$x^{\langle m+q, n \rangle} = A^{\langle m+q, n \rangle +} b^{\langle m+q \rangle}$$

или

$$x^{\langle m+q, n \rangle} = \begin{pmatrix} A^{\langle m, n \rangle} \\ \dots \\ A^{\langle q, n \rangle} \end{pmatrix}^+ \begin{pmatrix} b^{\langle m \rangle} \\ \dots \\ b^{\langle q \rangle} \end{pmatrix}. \quad (13.33)$$

Формула Клайна псевдообращения блочных матриц, записанная в транспонированном виде, определяет следующее выражение этого решения через решение $x^{\langle m, n \rangle}$ задачи (13.10), (13.9):

$$x^{\langle m+q, n \rangle} = x^{\langle m, n \rangle} + L^{\langle m+q, n \rangle} \left(b^{\langle q \rangle} - A^{\langle q, n \rangle} x^{\langle m, n \rangle} \right), \quad (13.34)$$

где матрица $L^{\langle m+q, n \rangle}$ определяется в виде:

$$L^{\langle m+q, n \rangle} = C^{\langle m+q, n \rangle +} + \left(I - C^{\langle m+q, n \rangle +} A^{\langle q, n \rangle} \right) A^{\langle m, n \rangle +} \times \\ \times \left(A^{\langle q, n \rangle} A^{\langle m, n \rangle +} \right)^T K^{\langle m+q, n \rangle} \left(I - C^{\langle m+q, n \rangle} C^{\langle m+q, n \rangle +} \right),$$

причём

$$C^{\langle m+q, n \rangle} = A^{\langle q, n \rangle} \left(I - A^{\langle m, n \rangle +} A^{\langle m, n \rangle} \right), \\ K^{\langle m+q, n \rangle} = \left(I + \left[\left(I - C^{\langle m+q, n \rangle} C^{\langle m+q, n \rangle +} \right) A^{\langle q, n \rangle} A^{\langle m, n \rangle +} \right] \times \right. \\ \left. \times \left[\left(I - C^{\langle m+q, n \rangle} C^{\langle m+q, n \rangle +} \right) A^{\langle q, n \rangle} A^{\langle m, n \rangle +} \right]^T \right)^{-1}. \quad (13.35)$$

Выражение для $L^{\langle m+q, n \rangle}$ в ряде случаев упрощается. Так, если матрица $A^{\langle m, n \rangle} \in \mathbb{R}_n^{m \times n}$, то есть $\text{rg} A^{\langle m, n \rangle} = n$, то $C^{\langle m+q, n \rangle} = 0$ и

$$L^{\langle m+q, n \rangle} = \left(A^{\langle m, n \rangle T} A^{\langle m, n \rangle} \right)^{-1} A^{\langle q, n \rangle T} \times \\ \times \left(I + A^{\langle q, n \rangle} \left(A^{\langle m, n \rangle T} A^{\langle m, n \rangle} \right)^{-1} A^{\langle q, n \rangle T} \right)^{-1} \quad (13.36)$$

(это выражение непосредственно обобщает (13.31) и является аналогом (13.32)).
Подобным же образом можно рассмотреть и другие случаи.

До сих пор рассматривались ситуации, когда число слагаемых увеличивалось либо только во внутренней, либо только во внешней суммах соотношения (13.1). Может представить интерес и вопрос об одновременном увеличении числа слагаемых в обеих суммах. Из предыдущих рекуррентных процедур следуют по крайней мере два альтернативных варианта двойной или двухиндексной рекуррентной процедуры:

$$\begin{aligned}
 x^{\langle m+1, n+1 \rangle} &= \begin{pmatrix} x^{\langle m, n \rangle} + l^{\langle m+1, n \rangle} (b_{m+1} - a^{\langle m+1, n \rangle} x^{\langle m, n \rangle}) - \\ - A^{\langle m+1, n \rangle +} a^{\langle m+1, n+1 \rangle} (l^{\langle m+1, n+1 \rangle})^T b_{m+1} \\ \dots\dots\dots \\ l^{\langle m+1, n+1 \rangle T} b_{m+1} \end{pmatrix}, \\
 x^{\langle m+1, n+1 \rangle} &= \begin{pmatrix} x^{\langle m, n \rangle} - A^{\langle m, n+1 \rangle +} a^{\langle m, n+1 \rangle} (l^{\langle m, n+1 \rangle})^T b^{\langle m \rangle} \\ \dots\dots\dots \\ l^{\langle m, n+1 \rangle T} b^{\langle m \rangle} \end{pmatrix} + \\
 &+ l^{\langle m+1, n+1 \rangle} \left(b_{m+1} - a^{\langle m+1, n+1 \rangle} \begin{pmatrix} x^{\langle m, n \rangle} - A^{\langle m, n+1 \rangle +} a^{\langle m, n+1 \rangle} (l^{\langle m, n+1 \rangle})^T b^{\langle m \rangle} \\ \dots\dots\dots \\ l^{\langle m, n+1 \rangle T} b^{\langle m \rangle} \end{pmatrix} \right).
 \end{aligned}$$

§ 14. Нелинейный метод наименьших квадратов: рекуррентно-итерационные процедуры

Как указано в предисловии к главе, нелинейная задача о наименьших квадратах рассматривается здесь в следующей постановке: найти

$$\min_{x \in \mathbb{R}^n} f(x) = \frac{1}{2} \|R(x)\|^2 = \frac{1}{2} \sum_{u=1}^m r_u^2(x) = \frac{1}{2} \varepsilon. \quad (14.1)$$

Отметим сразу же, что если в параграфе 13 линейная задача о наименьших квадратах формулировалась в контексте задач регрессии, то в данном параграфе нелинейная задача о наименьших квадратах формулируется в кон-

тексте задач нелинейного программирования без конкретизации структуры невязок $r_u(x)$, $u = 1, \dots, m$.

В предисловии к главе, в параграфах 11, 12 указано также, что большинство методов нелинейного программирования носит итерационный характер: по текущей точке x_c определяется следующая точка x_+ так, что последовательность точек $\{x^{(k)}\}$ сходится к решению x^* , причём $f(x^{(k+1)}) < f(x^{(k)})$ на каждом шаге итерации. В предисловии к главе охарактеризованы основные методы решения нелинейной задачи о наименьших квадратах: метод Гаусса-Ньютона, Левенберга-Марквардта и др. Рассмотрим их подробнее, следуя при этом схеме параграфа 13.

Метод Гаусса-Ньютона основан на аппроксимации функции $R(x)$ в окрестности текущей точки x_c её аффинной моделью

$$M_c(x) = R(x_c) + J(x_c)(x - x_c) = R(x_c) + J(x_c)\Delta x$$

и соответствующей замене нелинейной задачи о наименьших квадратах (14.1) на линейную вида: найти

$$\min_{x \in \mathbb{R}^n} f_c(x) = \frac{1}{2} \|M_c(x)\|^2 = \frac{1}{2} \|R(x_c) + J(x_c)\Delta x\|^2 = \frac{1}{2} \varepsilon_c, \quad (14.2)$$

где $\varepsilon_c = \|R(x_c) + J(x_c)\Delta x\|^2$ – невязка уравнения

$$J(x_c)\Delta x = -R(x_c). \quad (14.3)$$

Здесь J – матрица Якоби вектор-функции $R = (r_1, \dots, r_m)$,

$$J = \begin{pmatrix} \nabla^T r_1 \\ \vdots \\ \nabla^T r_m \end{pmatrix}, \quad \nabla r_n = \begin{pmatrix} \frac{\partial r_u}{\partial x_1} \\ \vdots \\ \frac{\partial r_u}{\partial x_n} \end{pmatrix}, \quad u = 1, \dots, m.$$

В редких случаях, матрица $J(x_c)$ уравнения (14.3) оказывается квадратной и невырожденной; в этом случае его единственное решение

$$\Delta x = -J^{-1}(x_c)R(x_c), \quad x_+ = x_c + \Delta x \quad (14.4)$$

является решением задачи (14.2) с нулевой невязкой.

Чаще встречаются случаи, когда матрица $J(x_c) \in \mathbb{R}^{m \times n}$ является матрицей общего вида, но оказывается выполненным условие совместности уравнения (14.3), имеющее вид:

$$J(x_c)J^+(x_c)R(x_c) = R(x_c), \quad (14.5)$$

где $J^+(x_c) \in \mathbb{R}^{m \times n}$ – псевдообратная к матрице $J(x_c)$. Следует отметить, что если уравнение (14.3) переписать в эквивалентном виде

$$J(x_c)x = J(x_c)x_c - R(x_c),$$

то условие его совместности:

$$J(x_c)J^+(x_c)(J(x_c)x_c - R(x_c)) = J(x_c)x_c - R(x_c),$$

в силу одного из определяющих соотношений для псевдообратных, а именно $JJ^+J = J$, оказывается эквивалентным условию (14.5).

При выполнении последнего условия общее решение уравнения (14.3)

$$\Delta x = -J^+(x_c)R(x_c) + (I - J^+(x_c)J(x_c))y, \quad x_+ = x_c + \Delta x \quad (14.6)$$

является решением задачи (14.2) с нулевой связкой. Это решение единственно, если выполняется наряду с (14.5) условие

$$J^+(x_c)J(x_c) = I, \quad (14.7)$$

и записывается в виде:

$$\Delta x = -J^+(x_c)R(x_c), \quad x_+ = x_c + \Delta x. \quad (14.8)$$

В других случаях выбор единственного решения из (14.6) подчиняется тем или иным дополнительным требованиям; таково, например, требование: найти

$$\min_{\Delta x = -J^+(x_c)R(x_c) + (I - J^+(x_c)J(x_c))y} \|\Delta x\|. \quad (14.9)$$

Решение задачи (14.3), (14.9) называется нормальным; оно также записывается в виде (14.8).

Наиболее реальный случай, когда матрица $J(x_c)$ имеет общий вид и условие (14.5) не выполняется, так что решение уравнения (14.3) не существует. Замечателен тот факт, что и в этих случаях соотношение (14.6) даёт решение задачи (14.2), а соотношение (14.8) – решение задачи (14.2), (14.9); для уравнения (14.3) они служат соответственно общим и нормальным псевдорешениями. Они также в методе Гаусса-Ньютона определяют основной шаг итерационной процедуры, записываемой обычно в виде (при традиционном предположении $J(x_c) \in \mathbb{R}_n^{m \times n}$, то есть $\text{rg} J(x_c) = n$, когда $J^+ = (J^T J)^{-1} J^T$):

$$\Delta x = - (J^T(x_c) J(x_c))^{-1} J^T(x_c) R(x_c), \quad x_+ = x_c + \Delta x.$$

Далее за основу принимается запись (14.3) как более общая и краткая, а также лучше приспособленная для формулировки рекуррентных процедур.

В предисловии к главе указано, что необходимость в рекуррентных процедурах решения нелинейной задачи о наименьших квадратах возникает наряду с итерационными, по крайней мере, в связи с такими ситуациями: последовательным пересчетом координат следующей точки x_+ по текущей точке x_c ; увеличением числа координат вектор-функции R . Рассмотрим подробнее эти ситуации применительно к методу Гаусса-Ньютона.

Предварительно, для удобства дальнейших формулировок, уточним обозначения векторов и матриц в (14.2): найти

$$\min_{\Delta x^{(m,n)} \in \mathbb{R}^n} \frac{1}{2} \|R^{(m)}(x_c^{(m,n)}) + J^{(m,n)}(x_c^{(m,n)}) \Delta x^{(m,n)}\|^2 = \frac{1}{2} \varepsilon^{(m,n)}. \quad (14.10)$$

При последовательном пересчёте координат точки x_c в координаты точки x_+ на отдельно взятой итерации, рассматриваемом как вариант методов покоординатной оптимизации, обозначим через $x^{(m,\nu)}$, $0 \leq \nu \leq n$, промежуточные точки такие, что $x^{(m,0)} = x_c^{(m,n)}$, $x^{(m,n)} = x_+^{(m,n)}$, $\Delta x^{(m,\nu)} = x^{(m,\nu)} - x^{(m,\nu-1)}$, причём у точки $x^{(m,\nu)}$ первые ν координат уже пересчитаны, а остальные совпадают с координатами точки $x_c^{(m,n)}$, так что $(n - \nu)$ последних координат вектора $\Delta x^{(m,\nu)}$ с $(\nu + 1)$ -й по n -ю равны нулю, в силу

чего в аффинной модели участвует при этом только часть $J^{\langle m, \nu \rangle} \in \mathbb{R}^{m \times \nu}$ матрицы Якоби. Именно по индексу ν формируется рекуррентная процедура на каждом шаге итерации, что приводит в конечном счёте к одной из разновидностей рекуррентно-итерационных процедур метода Гаусса-Ньютона.

В принятых обозначениях вектор $\Delta x^{\langle m, \nu \rangle}$ ищется как решение задачи: найти

$$\min_{\Delta x^{\langle m, \nu \rangle} \in \mathbb{R}^n} \frac{1}{2} \|R^{\langle m \rangle}(x^{\langle m, \nu-1 \rangle}) + J^{\langle m, \nu \rangle}(x^{\langle m, \nu-1 \rangle}) \Delta x^{\langle m, \nu \rangle}\|^2 = \frac{1}{2} \varepsilon^{\langle m, \nu \rangle}. \quad (14.11)$$

Переходя к пересчёту $(\nu + 1)$ -й координаты, то есть к определению $x^{\langle m, \nu+1 \rangle}$, а значит – к определению $\Delta x^{\langle m, \nu+1 \rangle}$, отметим, что при этом, вообще говоря, пересчитываются и все ранее найденные координаты, что отражено в последующих обозначениях. Вектор $\Delta x^{\langle m, \nu+1 \rangle}$ ищется как решение задачи: найти

$$\min_{\Delta x^{\langle m, \nu+1 \rangle} \in \mathbb{R}^n} \frac{1}{2} \|R^{\langle m \rangle}(x^{\langle m, \nu \rangle}) + J^{\langle m, \nu+1 \rangle}(x^{\langle m, \nu \rangle}) \Delta x^{\langle m, \nu+1 \rangle}\|^2 = \frac{1}{2} \varepsilon^{\langle m, \nu+1 \rangle}, \quad (14.12)$$

где матрица $J^{\langle m, \nu+1 \rangle}$ структурно связана с матрицей $J^{\langle m, \nu \rangle}$ из (14.11) соотношением

$$J^{\langle m, \nu+1 \rangle} = \begin{pmatrix} J^{\langle m, \nu \rangle} & \vdots & j^{\langle m, \nu+1 \rangle} \end{pmatrix}.$$

Отметим, что матрица $J^{\langle m, \nu \rangle}$ в последней формуле вычисляется в точке $x^{\langle m, \nu \rangle}$, тогда как в (14.11) эта же матрица вычисляется в точке $x^{\langle m, \nu-1 \rangle}$. Тем самым, в отличие от линейной задачи о наименьших квадратах, рассмотренной в параграфе 13, в данном случае на следующем шаге рекурсии не удаётся непосредственно использовать результат предыдущего шага, а именно псевдообратную матрицу $(J^{\langle m, \nu \rangle}(x^{\langle m, \nu-1 \rangle}))^+$ и приходится определять вспомогательную матрицу $(J^{\langle m, \nu \rangle}(x^{\langle m, \nu \rangle}))^+$, то есть дополнительно псевдообращать ту же матрицу $J^{\langle m, \nu \rangle}$, но вычисленную в уже найденной точке $x^{\langle m, \nu \rangle}$. В этом состоит особенность формируемой рекуррентной процедуры применительно к нелинейной задаче о наименьших квадратах в отличие от линейной задачи. Тем не менее сохраняется основной смысл рекуррентного псевдообращения: сведение псевдообращения матрицы определённого разме-

ра к псевдообращению матрицы меньшего размера. Последствия отмеченной особенности встречаются и в дальнейших рассмотрениях.

Дополнительный столбец матрицы $J^{\langle m, \nu+1 \rangle}$ имеет вид:

$$j^{\langle m, \nu+1 \rangle} = \begin{pmatrix} \frac{\partial r_1}{\partial x_{\nu+1}} & \dots & \frac{\partial r_m}{\partial x_{\nu+1}} \end{pmatrix}^T.$$

Вектор $\Delta x^{\langle m, \nu+1 \rangle}$ имеет структуру:

$$\Delta x^{\langle m, \nu+1 \rangle} = \begin{pmatrix} \Delta x_{(\nu)}^{\langle m, \nu+1 \rangle T} & \vdots & \Delta x_{\nu+1}^{\langle m, \nu+1 \rangle} \end{pmatrix}^T,$$

причём, как отмечалось, $x_{(\nu)}^{\langle m, \nu+1 \rangle} \neq x^{\langle m, \nu \rangle}$, так что

$$\Delta x_{(\nu)i}^{\langle m, \nu+1 \rangle} = x_i^{\langle m, \nu+1 \rangle} - x_i^{\langle m, \nu \rangle}, \quad i = 1, \dots, \nu,$$

$$\Delta x_{\nu+1}^{\langle m, \nu+1 \rangle} = x_{\nu+1}^{\langle m, \nu+1 \rangle} - x_{\nu+1}^{\langle m, \nu \rangle},$$

так как пересчёт каждой очередной координаты, вообще говоря, влечёт пересчёт предыдущих координат.

В соответствии с (14.8) решение задачи (14.9), (14.12) запишется в виде: $\Delta x^{\langle m, \nu+1 \rangle} = - \left(J^{\langle m, \nu+1 \rangle} (x^{\langle m, \nu \rangle}) \right)^+ R^{\langle m \rangle} (x^{\langle m, \nu \rangle})$ или

$$\begin{pmatrix} \Delta x_{(\nu)}^{\langle m, \nu+1 \rangle} \\ \vdots \\ \Delta x_{\nu+1}^{\langle m, \nu+1 \rangle} \end{pmatrix} = - \begin{pmatrix} J^{\langle m, \nu \rangle} (x^{\langle m, \nu \rangle}) & j^{\langle m, \nu+1 \rangle} (x^{\langle m, \nu \rangle}) \end{pmatrix}^+ R^{\langle m \rangle} (x^{\langle m, \nu \rangle}). \quad (14.13)$$

Известный алгоритм Гревия рекуррентного псевдообращения матриц, записываемый «по наращиванию столбцов» определяет следующее выражение этого решения через решение $\Delta \tilde{x}^{\langle m, \nu \rangle}$ вспомогательной задачи типа (14.11), (14.9), в которой $x^{\langle m, \nu-1 \rangle}$ заменено на $x^{\langle m, \nu \rangle}$:

$$\Delta x_{\nu+1}^{\langle m, \nu+1 \rangle} = - \left(l^{\langle m, \nu+1 \rangle} \right)^T R^{\langle m \rangle} (x^{\langle m, \nu \rangle}), \quad (14.14)$$

$$\Delta x_{(\nu)}^{\langle m, \nu+1 \rangle} = \Delta \tilde{x}^{\langle m, \nu \rangle} + \left(J^{\langle m, \nu \rangle} (x^{\langle m, \nu \rangle}) \right)^+ j^{\langle m, \nu+1 \rangle} (x^{\langle m, \nu \rangle}) \Delta x_{\nu+1}^{\langle m, \nu+1 \rangle}, \quad (14.15)$$

где вектор $(l^{\langle m, \nu+1 \rangle})^T$ определяется в виде:

$$(l^{\langle m, \nu+1 \rangle})^T = \begin{cases} c^{\langle m, \nu+1 \rangle+} = \frac{c^{\langle m, \nu+1 \rangle T}}{\|c^{\langle m, \nu+1 \rangle}\|^2}, \\ \text{если } c^{\langle m, \nu+1 \rangle} = \left(I - J^{\langle m, \nu \rangle}(x^{\langle m, \nu \rangle}) J^{\langle m, \nu \rangle+}(x^{\langle m, \nu \rangle}) \right) \times \\ \times j^{\langle m, \nu+1 \rangle}(x^{\langle m, \nu \rangle}) \neq 0; \\ \text{в остальных случаях} \\ \frac{j^{\langle m, \nu+1 \rangle T}(x^{\langle m, \nu \rangle}) \left(J^{\langle m, \nu \rangle+}(x^{\langle m, \nu \rangle}) \right)^T J^{\langle m, \nu \rangle+}(x^{\langle m, \nu \rangle})}{k^{\langle m, \nu+1 \rangle}}, \\ k^{\langle m, \nu+1 \rangle} = 1 + \|J^{\langle m, \nu \rangle+}(x^{\langle m, \nu \rangle}) j^{\langle m, \nu+1 \rangle}(x^{\langle m, \nu \rangle})\|^2. \end{cases} \quad (14.16)$$

Первому варианту этой формулы соответствует, например, случай (упомянутый ранее), когда матрица $J^{\langle m, \nu \rangle} \in \mathbb{R}_\nu^{m \times \nu}$, то есть $\text{rg} J^{\langle m, \nu \rangle}(x^{\langle m, \nu \rangle}) = \nu$, а второму варианту – случай, когда матрица $J^{\langle m, \nu \rangle} \in \mathbb{R}_m^{m \times \nu}$, то есть $\text{rg} J^{\langle m, \nu \rangle}(x^{\langle m, \nu \rangle}) = m$; в последнем случае выражение для $l^{\langle m, \nu+1 \rangle T}$ упрощается:

$$(l^{\langle m, \nu+1 \rangle})^T = \quad (14.17)$$

$$= \frac{j^{\langle m, \nu+1 \rangle T}(x^{\langle m, \nu \rangle}) \left(J^{\langle m, \nu \rangle}(x^{\langle m, \nu \rangle}) J^{\langle m, \nu \rangle T}(x^{\langle m, \nu \rangle}) \right)^{-1}}{1 + j^{\langle m, \nu+1 \rangle T}(x^{\langle m, \nu \rangle}) \left(J^{\langle m, \nu \rangle}(x^{\langle m, \nu \rangle}) J^{\langle m, \nu \rangle T}(x^{\langle m, \nu \rangle}) \right)^{-1} j^{\langle m, \nu+1 \rangle}(x^{\langle m, \nu \rangle})}.$$

Пусть теперь пересчёт координат точки на шаге итерационного процесса осуществляется «блоками» (поблочная координатная оптимизация), то есть после точки $x^{\langle m, \nu \rangle}$ определяется точка $x^{\langle m, \nu+p \rangle}$, $\nu+p \leq n$, у которой $n - (\nu+p)$ последних координат совпадают с координатами точки $x_c^{\langle m, n \rangle}$; в этом случае вектор $\Delta x^{\langle m, \nu+p \rangle}$ отыскивается как решение задачи: найти

$$\min_{\Delta x^{\langle m, \nu+p \rangle} \in \mathbb{R}^n} \frac{1}{2} \|R^{\langle m \rangle}(x^{\langle m, \nu \rangle}) + J^{\langle m, \nu+p \rangle}(x^{\langle m, \nu \rangle}) \Delta x^{\langle m, \nu+p \rangle}\|^2 = \frac{1}{2} \varepsilon^{\langle m, \nu+p \rangle}, \quad (14.18)$$

причём матрица $J^{\langle m, \nu+p \rangle}$ связана с матрицей $J^{\langle m, \nu \rangle}$ из (14.11) соотношением (вычисляются они, как уже отмечалось, в разных точках)

$$J^{\langle m, \nu+p \rangle} = \begin{pmatrix} J^{\langle m, \nu \rangle} & \vdots & J^{\langle m, (p) \rangle} \end{pmatrix},$$

где дополнительный блок имеет вид:

$$J^{\langle m, (p) \rangle} = \begin{pmatrix} \frac{\partial r_1}{\partial x_{\nu+1}} & \cdots & \frac{\partial r_1}{\partial x_{\nu+p}} \\ \vdots & & \vdots \\ \frac{\partial r_m}{\partial x_{\nu+1}} & \cdots & \frac{\partial r_m}{\partial x_{\nu+p}} \end{pmatrix},$$

а вектор $\Delta x^{\langle m, \nu+p \rangle}$ имеет структуру $\Delta x^{\langle m, \nu+p \rangle} = \left(\Delta x_{(\nu)}^{\langle m, \nu+p \rangle T} \quad \vdots \quad \Delta x_{(p)}^{\langle m, \nu+p \rangle T} \right)^T$, причём

$$\begin{aligned} \Delta x_{(\nu)i}^{\langle m, \nu+p \rangle} &= x_i^{\langle m, \nu+p \rangle} - x_i^{\langle m, \nu \rangle}, \quad i = 1, \dots, \nu, \\ \Delta x_{(p)i}^{\langle m, \nu+p \rangle} &= x_i^{\langle m, \nu+p \rangle} - x_c^{\langle m, n \rangle}, \quad i = \nu + 1, \dots, \nu + p, \end{aligned}$$

так как пересчёт каждого очередного блока координат, вообще говоря, влечёт пересчёт предшествующих координат.

В соответствии с (14.8) решение задачи (14.18), (14.9) запишется в виде:

$$\Delta x^{\langle m, \nu+p \rangle} = - \left(J^{\langle m, \nu+p \rangle} (x^{\langle m, \nu \rangle}) \right)^+ R^{\langle m \rangle} (x^{\langle m, \nu \rangle})$$

или

$$\begin{pmatrix} \Delta x_{(\nu)}^{\langle m, \nu+p \rangle} \\ \vdots \\ \Delta x_{(p)}^{\langle m, \nu+p \rangle} \end{pmatrix} = - \left(J^{\langle m, \nu \rangle} (x^{\langle m, \nu \rangle}) \quad \vdots \quad J^{\langle m, (p) \rangle} (x^{\langle m, \nu \rangle}) \right)^+ R^{\langle m \rangle} (x^{\langle m, \nu \rangle}). \quad (14.19)$$

Известная формула Клайна псевдообращения блочных матриц определяет следующее выражение этого решения через решение $\Delta \tilde{x}^{\langle m, \nu \rangle}$ задачи (14.11), (14.9), в которой $x^{\langle m, \nu-1 \rangle}$ заменено на $x^{\langle m, \nu \rangle}$:

$$\Delta x_{(p)}^{\langle m, \nu+p \rangle} = -L^{\langle m, \nu+p \rangle} R^{\langle m \rangle} (x^{\langle m, \nu \rangle}), \quad (14.20)$$

$$\Delta x_{(\nu)}^{\langle m, \nu+p \rangle} = \Delta \tilde{x}^{\langle m, \nu \rangle} + \left(J^{\langle m, \nu \rangle} (x^{\langle m, \nu \rangle}) \right)^+ J^{\langle m, (p) \rangle} (x^{\langle m, \nu \rangle}) \Delta x_{(p)}^{\langle m, \nu+p \rangle}, \quad (14.21)$$

где матрица $L^{\langle m, \nu+p \rangle}$ определяется в виде:

$$L^{\langle m, \nu+p \rangle} = C^{\langle m, \nu+p \rangle +} + \left(I - C^{\langle m, \nu+p \rangle +} C^{\langle m, \nu+p \rangle} \right) \times$$

$$\begin{aligned} & \times K^{\langle m, \nu+p \rangle} \left(J^{\langle m, n \rangle +} (x^{\langle m, \nu \rangle}) J^{\langle m, (p) \rangle} (x^{\langle m, \nu \rangle}) \right)^T \times \\ & \times J^{\langle m, \nu \rangle +} (x^{\langle m, \nu \rangle}) \left(I - J^{\langle m, (p) \rangle} (x^{\langle m, \nu \rangle}) C^{\langle m, \nu+p \rangle +} \right), \end{aligned}$$

причём

$$\begin{aligned} C^{\langle m, \nu+p \rangle} &= \left(I - J^{\langle m, n \rangle} (x^{\langle m, \nu \rangle}) J^{\langle m, n \rangle +} (x^{\langle m, \nu \rangle}) \right) J^{\langle m, (p) \rangle} (x^{\langle m, \nu \rangle}), \\ K^{\langle m, \nu+p \rangle} &= \left(I + \left[J^{\langle m, \nu \rangle +} (x^{\langle m, \nu \rangle}) J^{\langle m, (p) \rangle} (x^{\langle m, \nu \rangle}) \left(I - C^{\langle m, \nu+p \rangle +} C^{\langle m, \nu+p \rangle} \right) \right]^T \times \right. \\ & \times \left. \left[J^{\langle m, \nu \rangle +} (x^{\langle m, \nu \rangle}) J^{\langle m, (p) \rangle} (x^{\langle m, \nu \rangle}) \left(I - C^{\langle m, \nu+p \rangle +} C^{\langle m, \nu+p \rangle} \right) \right] \right)^{-1}. \quad (14.22) \end{aligned}$$

Выражение для $L^{\langle m, \nu+p \rangle}$ в ряде случаев упрощается. Так, если матрица Якоби в точке $x^{\langle m, \nu \rangle}$ удовлетворяет условию $J^{\langle m, \nu \rangle} (x^{\langle m, \nu \rangle}) \in \mathbb{R}_m^{m \times \nu}$, то есть $\text{rg} J^{\langle m, \nu \rangle} = m$, то $C^{\langle m, \nu+p \rangle} = 0$ и

$$\begin{aligned} L^{\langle m, \nu+p \rangle} &= \{ I + J^{\langle m, (p) \rangle T} (x^{\langle m, \nu \rangle}) \left[J^{\langle m, \nu \rangle} (x^{\langle m, \nu \rangle}) J^{\langle m, \nu \rangle T} (x^{\langle m, \nu \rangle}) \right]^{-1} \times \\ & \times J^{\langle m, (p) \rangle} (x^{\langle m, \nu \rangle}) \}^{-1} J^{\langle m, (p) \rangle T} (x^{\langle m, \nu \rangle}) \left[J^{\langle m, \nu \rangle} (x^{\langle m, \nu \rangle}) J^{\langle m, \nu \rangle T} (x^{\langle m, \nu \rangle}) \right]^{-1} \quad (14.23) \end{aligned}$$

(это выражение непосредственно обобщает (14.17)).

Если матрица $J^{\langle m, \nu+p \rangle} (x^{\langle m, \nu \rangle}) \in \mathbb{R}_{\nu+p}^{m \times (\nu+p)}$, то есть $\text{rg} J^{\langle m, \nu+p \rangle} (x^{\langle m, \nu \rangle}) = \nu + p$, то

$$L^{\langle m, \nu+p \rangle} = C^{\langle m, \nu+p \rangle +} \quad (14.24)$$

(это выражение непосредственно обобщает первый вариант формулы (14.16)).

Если матрица $J^{\langle m, \nu+p \rangle} (x^{\langle m, \nu \rangle}) \in \mathbb{R}_m^{m \times (\nu+p)}$, то есть $\text{rg} J^{\langle m, \nu+p \rangle} (x^{\langle m, \nu \rangle}) = m$, то

$$L^{\langle m, \nu+p \rangle} = \left(J^{\langle m, (p) \rangle T} (x^{\langle m, \nu \rangle}) C^{\langle m, \nu+p \rangle} \right)^{-1} C^{\langle m, \nu+p \rangle T} \quad (14.25)$$

(аналогов этой формулы в данном параграфе не было).

Следует отметить, что при использовании в (14.20), (14.21) формулы (14.24) для $L^{\langle m, \nu+p \rangle}$ справедливо соотношение

$$\varepsilon^{\langle m, \nu+p \rangle} = \varepsilon^{\langle m, \nu \rangle} - \|C^{\langle m, \nu+p \rangle} \Delta x_{(p)}^{\langle m, \nu+p \rangle}\|^2, \quad (14.26)$$

а при использовании формулы (14.25) – соотношение

$$\varepsilon^{\langle m, \nu+p \rangle} = \varepsilon^{\langle m, \nu \rangle} - \left(J^{\langle m, (p) \rangle} (x^{\langle m, \nu \rangle}) \Delta x_{(p)}^{\langle m, \nu+p \rangle} \right)^T \times \\ \times \left(I - J^{\langle m, \nu \rangle} (x^{\langle m, \nu \rangle}) J^{\langle m, \nu \rangle+} (x^{\langle m, \nu \rangle}) \right) R^{\langle m \rangle} (x^{\langle m, \nu \rangle}). \quad (14.27)$$

Вычитаемые в этих соотношениях, как и добавки к $\Delta \tilde{x}^{\langle m, \nu \rangle}$ в (14.15), (14.21), трактуются как «корректирующие члены» при рекуррентной покоординатной или поблочно-координатной оптимизации на каждом шаге итерационного процесса Гаусса-Ньютона.

Перейдём к рассмотрению, применительно к тому же методу, другой из указанных выше ситуаций, обуславливающих необходимость в рекуррентных процедурах. Пусть увеличилось на единицу число координат вектор функции R так, что

$$R^{\langle m+1 \rangle} = \begin{pmatrix} r_1 \\ \vdots \\ r_m \\ \dots\dots \\ r_{m+1} \end{pmatrix} = \begin{pmatrix} R^{\langle m \rangle} \\ \dots\dots \\ r_{m+1} \end{pmatrix}, \quad J^{\langle m+1, n \rangle} = \begin{pmatrix} \nabla^T r_1 \\ \vdots \\ \nabla^T r_m \\ \dots\dots \\ \nabla^T r_{m+1} \end{pmatrix} = \begin{pmatrix} J^{\langle m, n \rangle} \\ \dots\dots \\ \nabla^T r_{m+1} \end{pmatrix}.$$

Ограничимся рассмотрением случая, когда дополнительная координата вектор-функции R появляется в процессе итерационного счёта, после определения текущей точки $x_c = x_c^{\langle m, n \rangle}$. Если эта координата не появилась, то следующая точка $x_+ = x_+^{\langle m, n \rangle}$ определяется в виде неоднократно использованном выше: $x_+^{\langle m, n \rangle} = x_c^{\langle m, n \rangle} + \Delta x^{\langle m, n \rangle}$, где $\Delta x^{\langle m, n \rangle}$ – решение задачи (14.10). Если же дополнительная координата появилась, то следующая точка $x_+ = x_+^{\langle m+1, n \rangle}$ определяется в виде: $x_+^{\langle m+1, n \rangle} = x_c^{\langle m, n \rangle} + \Delta x^{\langle m+1, n \rangle}$, где $\Delta x^{\langle m+1, n \rangle}$ является решением задачи: найти

$$\min_{\Delta x^{\langle m+1, n \rangle} \in \mathbb{R}^n} \frac{1}{2} \| R^{\langle m+1 \rangle} (x_c^{\langle m, n \rangle}) + J^{\langle m+1, n \rangle} (x_c^{\langle m, n \rangle}) \Delta x^{\langle m+1, n \rangle} \|^2 = \frac{1}{2} \varepsilon^{\langle m+1, n \rangle}. \quad (14.28)$$

Здесь матрица $J^{\langle m+1, n \rangle} (x_c^{\langle m, n \rangle})$ непосредственно связана с матрицей $J^{\langle m, n \rangle} (x_c^{\langle m, n \rangle})$ из (14.10) соотношением

$$J^{\langle m+1, n \rangle}(x_c^{\langle m, n \rangle}) = \begin{pmatrix} J^{\langle m, n \rangle}(x_c^{\langle m, n \rangle}) \\ \dots\dots\dots \\ \nabla^T r_{m+1}(x_c^{\langle m, n \rangle}) \end{pmatrix}.$$

Таким образом, в данной ситуации, в отличие от предыдущей, матрица $J^{\langle m, n \rangle}(x_c^{\langle m, n \rangle})$ из (14.10) и результаты манипулирования с ней (в частности, её псевдообратная) могут быть непосредственно использованы при переходе от m к $(m + 1)$ координат вектор-функции R .

В соответствии с (14.8) решение задачи (14.28), (14.9) запишется в виде:

$$\begin{aligned} \Delta x^{\langle m+1, n \rangle} &= - \left(J^{\langle m+1, n \rangle}(x_c^{\langle m, n \rangle}) \right)^+ R^{\langle m+1 \rangle}(x_c^{\langle m, n \rangle}) \quad \text{или} \\ \Delta x^{\langle m+1, n \rangle} &= - \begin{pmatrix} J^{\langle m, n \rangle}(x_c^{\langle m, n \rangle}) \\ \dots\dots\dots \\ \nabla^T r_{m+1}(x_c^{\langle m, n \rangle}) \end{pmatrix}^+ \begin{pmatrix} R^{\langle m \rangle}(x_c^{\langle m, n \rangle}) \\ \dots\dots\dots \\ r_{m+1}(x_c^{\langle m, n \rangle}) \end{pmatrix}. \end{aligned} \quad (14.29)$$

Алгоритм Гревия рекуррентного псевдообращения матриц, записанный «по наращиванию строк», определяет следующее выражение этого решения $\Delta x^{\langle m, n \rangle}$ через решение задачи (14.10), (14.9):

$$\begin{aligned} \Delta x^{\langle m+1, n \rangle} &= \\ &= \Delta x^{\langle m, n \rangle} - l^{\langle m+1, n \rangle} \left(r_{m+1}(x_c^{\langle m, n \rangle}) + \nabla^T r_{m+1}(x_c^{\langle m, n \rangle}) \Delta x^{\langle m, n \rangle} \right), \end{aligned} \quad (14.30)$$

где вектор $l^{\langle m+1, n \rangle}$ определяется в виде:

$$l^{\langle m+1, n \rangle} = \begin{cases} c^{\langle m+1, n \rangle+} = \frac{c^{\langle m+1, n \rangle T}}{\|c^{\langle m+1, n \rangle}\|^2}, \\ \text{если } c^{\langle m+1, n \rangle} = \nabla^T r_{m+1}(x_c^{\langle m, n \rangle}) \times \\ \times \left(I - J^{\langle m, n \rangle+}(x_c^{\langle m, n \rangle}) J^{\langle m, n \rangle}(x_c^{\langle m, n \rangle}) \right) \neq 0; \\ \text{в остальных случаях} \\ \frac{J^{\langle m, n \rangle+}(x_c^{\langle m, n \rangle}) \left(J^{\langle m, n \rangle+}(x_c^{\langle m, n \rangle}) \right)^T \nabla r_{m+1}(x_c^{\langle m, n \rangle})}{k^{\langle m+1, n \rangle}}, \\ k^{\langle m+1, n \rangle} = 1 + \|\nabla^T r_{m+1}(x_c^{\langle m, n \rangle}) J^{\langle m, n \rangle+}(x_c^{\langle m, n \rangle})\|^2. \end{cases} \quad (14.31)$$

Первому варианту этой формулы соответствует, например, случай, когда матрица $J^{\langle m,n \rangle}(x_c^{\langle m,n \rangle}) \in \mathbb{R}_m^{m \times n}$, то есть $\text{rg} J^{\langle m,n \rangle}(x_c^{\langle m,n \rangle}) = m$, а второму варианту – случай, когда матрица $J^{\langle m,n \rangle}(x_c^{\langle m,n \rangle}) \in \mathbb{R}_n^{m \times n}$, то есть $\text{rg} J^{\langle m,n \rangle}(x_c^{\langle m,n \rangle}) = n$; в последнем случае выражение для $l^{\langle m,n \rangle}(x_c^{\langle m+1,n \rangle})$ упрощается:

$$l^{\langle m+1,n \rangle} = \frac{\left(J^{\langle m,n \rangle T}(x_c^{\langle m,n \rangle}) J^{\langle m,n \rangle}(x_c^{\langle m,n \rangle}) \right)^{-1} \nabla r_{m+1}(x_c^{\langle m,n \rangle})}{1 + \nabla^T r_{m+1}(x_c^{\langle m,n \rangle}) \left(J^{\langle m,n \rangle T}(x_c^{\langle m,n \rangle}) J^{\langle m,n \rangle}(x_c^{\langle m,n \rangle}) \right)^{-1} \nabla r_{m+1}(x_c^{\langle m,n \rangle})} \quad (14.32)$$

(это выражение является аналогом (14.17)).

Следует отметить определённое отличие структуры рекуррентного выражения (14.30) от рекуррентных выражений (14.14), (14.15). А именно, в (14.30) «корректирующий член» пропорционален «предсказанному» по $\Delta x^{\langle m,n \rangle}$ значению аффинной модели добавленной координаты r_{m+1} вектор-функции R , причём это значение совпадает со своим отклонением от нуля, определяя ошибку данного предсказания. Тем самым «новое» значение $\Delta x^{\langle m+1,n \rangle}$ получается в виде разности его «старого» значения $\Delta x^{\langle m,n \rangle}$ и члена, пропорционального ошибке предсказания. Вектор пропорциональности $l^{\langle m+1,n \rangle}$, как и в линейной задаче о наименьших квадратах, может быть назван «вектором сглаживания».

Пусть теперь число координат вектор-функции R увеличилось на q , так что

$$R^{\langle m+q \rangle} = \begin{pmatrix} r_1 \\ \vdots \\ r_m \\ \dots\dots\dots \\ r_{m+1} \\ \vdots \\ r_{m+q} \end{pmatrix} = \begin{pmatrix} R^{\langle m \rangle} \\ \dots\dots\dots \\ R^{\langle (q) \rangle} \end{pmatrix}, \quad J^{\langle m+q,n \rangle} = \begin{pmatrix} \nabla^T r_1 \\ \vdots \\ \nabla^T r_m \\ \dots\dots\dots \\ \nabla^T r_{m+1} \\ \vdots \\ \nabla^T r_{m+q} \end{pmatrix} = \begin{pmatrix} J^{\langle m,n \rangle} \\ \dots\dots\dots \\ J^{\langle (q),n \rangle} \end{pmatrix}.$$

Следующая точка $x_+ = x_+^{\langle m+q, n \rangle}$ определяется в этом случае в виде $x_+^{\langle m+q, n \rangle} = x_c^{\langle m, n \rangle} + \Delta x^{\langle m+q, n \rangle}$, где $\Delta x^{\langle m+q, n \rangle}$ является решением задачи: найти

$$\min_{\Delta x^{\langle m+q, n \rangle} \in \mathbb{R}^n} \frac{1}{2} \|R^{\langle m+q \rangle}(x_c^{\langle m, n \rangle}) + J^{\langle m+q, n \rangle}(x_c^{\langle m, n \rangle}) \Delta x^{\langle m+q, n \rangle}\|^2 = \frac{1}{2} \varepsilon^{\langle m+q, n \rangle}. \quad (14.33)$$

Здесь матрица $J^{\langle m+q, n \rangle}(x_c^{\langle m, n \rangle})$ непосредственно связана с матрицей $J^{\langle m, n \rangle}(x_c^{\langle m, n \rangle})$ из (14.10) соотношением

$$J^{\langle m+q, n \rangle}(x_c^{\langle m, n \rangle}) = \begin{pmatrix} J^{\langle m, n \rangle}(x_c^{\langle m, n \rangle}) \\ \dots\dots\dots \\ J^{\langle (q), n \rangle}(x_c^{\langle m, n \rangle}) \end{pmatrix}.$$

В соответствии с (14.8) решение задачи (14.33), (14.9) запишется в виде:

$$\Delta x^{\langle m+q, n \rangle} = - \left(J^{\langle m+q, n \rangle}(x_c^{\langle m, n \rangle}) \right)^+ R^{\langle m+q \rangle}(x_c^{\langle m, n \rangle})$$

или

$$\Delta x^{\langle m+q, n \rangle} = - \begin{pmatrix} J^{\langle m, n \rangle}(x_c^{\langle m, n \rangle}) \\ \dots\dots\dots \\ J^{\langle (q), n \rangle}(x_c^{\langle m, n \rangle}) \end{pmatrix}^+ \begin{pmatrix} R^{\langle m \rangle}(x_c^{\langle m, n \rangle}) \\ \dots\dots\dots \\ R^{\langle (q) \rangle}(x_c^{\langle m, n \rangle}) \end{pmatrix}. \quad (14.34)$$

Формула Клайна псевдообращения блочных матриц, записанная в транспонированном виде, определяет следующее выражение этого решения через решение $\Delta x^{\langle m, n \rangle}$ задачи (14.10), (14.9):

$$\begin{aligned} \Delta x^{\langle m+q, n \rangle} = \\ = \Delta x^{\langle m, n \rangle} - L^{\langle m+q, n \rangle} \left(R^{\langle (q) \rangle}(x_c^{\langle m, n \rangle}) + J^{\langle (q), n \rangle}(x_c^{\langle m, n \rangle}) \Delta x^{\langle m, n \rangle} \right), \end{aligned} \quad (14.35)$$

где матрица $L^{\langle m+q, n \rangle}$ определяется в виде:

$$\begin{aligned} L^{\langle m+q, n \rangle} = C^{\langle m+q, n \rangle}{}^+ + \left(I - C^{\langle m+q, n \rangle}{}^+ J^{\langle (q), n \rangle}(x_c^{\langle m, n \rangle}) \right) J^{\langle m, n \rangle}{}^+(x_c^{\langle m, n \rangle}) \times \\ \times \left(J^{\langle (q), n \rangle}(x_c^{\langle m, n \rangle}) J^{\langle m, n \rangle}{}^+(x_c^{\langle m, n \rangle}) \right)^T K^{\langle m+q, n \rangle} \left(I - C^{\langle m+q, n \rangle} C^{\langle m+q, n \rangle}{}^+ \right), \end{aligned}$$

причём

$$\begin{aligned}
C^{\langle m+q,n \rangle} &= J^{\langle (q),n \rangle}(x_c^{\langle m,n \rangle}) \left(I - J^{\langle m,n \rangle+}(x_c^{\langle m,n \rangle}) J^{\langle m,n \rangle}(x_c^{\langle m,n \rangle}) \right), \\
K^{\langle m+q,n \rangle} &= (I + \left[\left(I - C^{\langle m+q,n \rangle} C^{\langle m+q,n \rangle+} \right) J^{\langle (q),n \rangle}(x_c^{\langle m,n \rangle}) J^{\langle m,n \rangle+}(x_c^{\langle m,n \rangle}) \right] \times \\
&\times \left[\left(I - C^{\langle m+q,n \rangle} C^{\langle m+q,n \rangle+} \right) J^{\langle (q),n \rangle}(x_c^{\langle m,n \rangle}) J^{\langle m,n \rangle+}(x_c^{\langle m,n \rangle}) \right]^T)^{-1}. \quad (14.36)
\end{aligned}$$

Выражение для $L^{\langle m+q,n \rangle}$ в ряде случаев упрощается.

Так, если матрица $J^{\langle m,n \rangle}(x_c^{\langle m,n \rangle}) \in \mathbb{R}_n^{m \times n}$, то есть $\text{rg} J^{\langle m,n \rangle}(x_c^{\langle m,n \rangle}) = n$, то $C^{\langle m+q,n \rangle} = 0$ и

$$\begin{aligned}
L^{\langle m+q,n \rangle} &= \left(J^{\langle m,n \rangle T}(x_c^{\langle m,n \rangle}) J^{\langle m,n \rangle}(x_c^{\langle m,n \rangle}) \right)^{-1} J^{\langle (q),n \rangle T}(x_c^{\langle m,n \rangle}) \times \\
&\times (I + J^{\langle (q),n \rangle}(x_c^{\langle m,n \rangle}) (J^{\langle m,n \rangle T}(x_c^{\langle m,n \rangle}) \times \\
&\times J^{\langle m,n \rangle}(x_c^{\langle m,n \rangle}))^{-1} J^{\langle (q),n \rangle T}(x_c^{\langle m,n \rangle}))^{-1} \quad (14.37)
\end{aligned}$$

(это выражение непосредственно обобщает (14.32) и является аналогом (14.23)).

Подобным же образом можно рассмотреть и другие случаи.

Применительно к методу Левенберга-Марквардта рассмотрим ситуацию увеличения числа координат вектор функции R . В соответствии с указанным в предисловии к главе основным соотношением этого метода запишем:

$$\begin{aligned}
\Delta x^{\langle m,n \rangle} &= - \left(J^{\langle m,n \rangle T}(x_c^{\langle m,n \rangle}) J^{\langle m,n \rangle}(x_c^{\langle m,n \rangle}) + \mu_c I \right)^{-1} \times \\
&\times J^{\langle m,n \rangle T}(x_c^{\langle m,n \rangle}) R^{\langle m \rangle}(x_c^{\langle m,n \rangle}), \\
\Delta x^{\langle m+1,n \rangle} &= - \left(J^{\langle m+1,n \rangle T}(x_c^{\langle m,n \rangle}) J^{\langle m+1,n \rangle}(x_c^{\langle m,n \rangle}) + \mu_c I \right)^{-1} \times \\
&\times J^{\langle m+1,n \rangle T}(x_c^{\langle m,n \rangle}) R^{\langle m+1 \rangle}(x_c^{\langle m,n \rangle}).
\end{aligned}$$

Соотношение Жуковского-Липцера позволяет записать аналогичное (14.30) соотношение:

$$\begin{aligned}
\Delta x^{\langle m+1,n \rangle} &= \Delta x^{\langle m,n \rangle} - v^{\langle m+1,n \rangle} \times \\
&\times \left(r_{m+1}(x_c^{\langle m,n \rangle}) + \nabla^T r_{m+1}(x_c^{\langle m,n \rangle}) \Delta x_c^{\langle m,n \rangle} \right), \quad (14.38)
\end{aligned}$$

где вектор $v^{\langle m+1, n \rangle}$ определяется в виде:

$$v^{\langle m+1, n \rangle} = \frac{\Gamma^{\langle m, n \rangle} \nabla r_{m+1}(x_c^{\langle m, n \rangle})}{\gamma^{\langle m, n \rangle}}, \quad (14.39)$$

а матрица $\Gamma^{\langle m, n \rangle}$, в свою очередь, определяется в виде:

$$\Gamma^{\langle m+1, n \rangle} = \Gamma^{\langle m, n \rangle} - \frac{\Gamma^{\langle m, n \rangle} \nabla r_{m+1}(x_c^{\langle m, n \rangle}) \nabla^T r_{m+1}(x_c^{\langle m, n \rangle}) \Gamma^{\langle m, n \rangle}}{\gamma^{\langle m, n \rangle}}, \quad (14.40)$$

при этом скаляр $\gamma^{\langle m, n \rangle}$ определяется в виде:

$$\gamma^{\langle m, n \rangle} = \mu_c + \nabla^T r_{m+1}(x_c^{\langle m, n \rangle}) \Gamma^{\langle m, n \rangle} \nabla r_{m+1}(x_c^{\langle m, n \rangle}). \quad (14.41)$$

В этих соотношениях в качестве начальных условий естественным образом полагается $\Delta x^{\langle 0, n \rangle} = 0$, $\Gamma^{\langle 0, n \rangle} = I$ (первое из них относится к (14.30)).

В случае методов полностью ньютоновского типа возникает необходимость связать указанное в предисловии к главе основное соотношение:

$$\begin{aligned} \Delta x^{\langle m, n \rangle} = & - \left(J^{\langle m, n \rangle T}(x_c^{\langle m, n \rangle}) J^{\langle m, n \rangle}(x_c^{\langle m, n \rangle}) + Q^{\langle m, n \rangle}(x_c^{\langle m, n \rangle}) \right)^{-1} \times \\ & \times J^{\langle m, n \rangle T}(x_c^{\langle m, n \rangle}) R^{\langle m \rangle}(x_c^{\langle m, n \rangle}), \end{aligned}$$

либо с

$$\begin{aligned} \Delta x^{\langle m, \nu+1 \rangle} = & - \left(J^{\langle m, \nu+1 \rangle T}(x^{\langle m, \nu \rangle}) J^{\langle m, \nu+1 \rangle}(x^{\langle m, \nu \rangle}) + Q^{\langle m, \nu+1 \rangle}(x^{\langle m, \nu \rangle}) \right)^{-1} \times \\ & \times J^{\langle m, \nu+1 \rangle T}(x^{\langle m, \nu \rangle}) R^{\langle m \rangle}(x^{\langle m, \nu \rangle}), \end{aligned}$$

либо с

$$\begin{aligned} \Delta x^{\langle m+1, n \rangle} = & - \left(J^{\langle m+1, n \rangle T}(x_c^{\langle m, n \rangle}) J^{\langle m+1, n \rangle}(x_c^{\langle m, n \rangle}) + Q^{\langle m+1, n \rangle}(x_c^{\langle m, n \rangle}) \right)^{-1} \times \\ & \times J^{\langle m+1, n \rangle T}(x_c^{\langle m, n \rangle}) R^{\langle m+1 \rangle}(x_c^{\langle m, n \rangle}). \end{aligned}$$

По сравнению с рассмотренными выше методами основная возникающая здесь проблема связана с появлением матриц типа Q , рекуррентный пересчет которых при изменении m и n теперь организовать уже сложнее. Это обусловлено тем, что матрица Q , несущая информацию о производных второго порядка, имеет структуру $Q = \sum_{u=1}^m r_u \cdot G_u$, где G_u – матрицы Гессе функций r_u , $u = 1, \dots, m$, образованные их производными второго порядка по переменным x_ν , $\nu = 1, \dots, n$.

Основы изложенных в данном параграфе рекуррентно-итерационных процедур представлены в работе [18], а их развёрнутое детализированное изложение – в учебном пособии [19]. Следует отметить, что ещё в работе [108] присутствовали элементы рекуррентно-итерационного подхода, выраженные в схеме процесса подстройки, при появлении новых наблюдений, идентифицируемых параметров модели – см. также рис. 4.2.1 в [109]. В монографии [110] представлено развитие уже нового подхода к глобальной поисковой недифференцируемой оптимизации, основанного на усреднении координат и адаптации области поиска.

Задания и упражнения

В этом разделе приведены задания и упражнения, охватывающие основные разделы оптимизации и псевдообращения. Первая часть заданий посвящена одномерной оптимизации, а также многомерной оптимизации без ограничений и при наличии ограничений-равенств и смешанных ограничений. Во второй части заданий представлены упражнения на нахождение псевдообратных матриц, а также на доказательства некоторых свойств псевдообратных матриц. В третьей части заданий приведены содержательные задачи, требующие нахождения оптимальных параметров некоторых систем. Четвёртая часть включает набор тестовых функций для апробации алгоритмов и методов оптимизации.

Часть 1. Оптимизационная математика

Задача 1. Одномерная оптимизация. Найдите оптимумы функций.

Вариант	Функция	Вариант	Функция
1	$f(x) = 2\sqrt[3]{x^5} - 5\sqrt[3]{x^2} + 1$	11	$f(x) = e^{-x} + e^{2x}$
2	$f(x) = 3x - \operatorname{tg} x$	12	$f(x) = (2x + 1)^2(x - 4)$
3	$f(x) = 2x^2 - e^{0,5x^2}$	13	$f(x) = (1 - 4x)^2 + e^{0,3x^2}$
4	$f(x) = x + 2 - \sqrt[3]{x^5}$	14	$f(x) = -e^{-x} \ln(x)$
5	$f(x) = e^{-x} \cos(x)$	15	$f(x) = (x - 1)^6 + x$
6	$f(x) = 4x \sin x$	16	$f(x) = (x - 3)\sqrt{x}$
7	$f(x) = -2e^{-2x} \ln(2x)$	17	$f(x) = \sin x + \cos x$
8	$f(x) = x - 2e^{2x}$	18	$f(x) = x^5 + x^4 - (x^3/3) + 2$
9	$f(x) = 4x(x^2 + 4)$	19	$f(x) = 16x(x - 1)^3$
10	$f(x) = x/\ln(x)$	20	$f(x) = x^2/2 + 8/x^2$

Задача 2. Многомерная оптимизация, задачи без ограничений.

2.1. Найдите оптимумы функции $f(x, y) = e^{-(x^2+y^2)}(ax^2 + by^2)$.

Вариант	a	b	Вариант	a	b
1	1	2	11	9	1
2	7	3	12	3	5
3	1	4	13	3	1
4	8	5	14	7	2
5	1	6	15	3	6
6	8	3	16	4	1
7	2	4	17	6	7
8	2	5	18	4	3
9	7	6	19	1	5
10	2	7	20	5	6

2.2. Найдите оптимумы функции $f(x, y) = \frac{ax + by + c}{\sqrt{x^2 + y^2 + 1}}$.

Вариант	a	b	c	Вариант	a	b	c
1	4	-1	2	11	-2	4	1
2	4	2	-1	12	4	-2	-1
3	1	2	4	13	6	1	6
4	3	1	2	14	6	1	-6
5	3	-1	-2	15	4	6	-1
6	1	3	-4	16	2	3	3
7	-1	3	4	17	3	1	4
8	2	5	2	18	6	-1	-2
9	1	3	-3	19	1	5	4
10	2	1	3	20	1	-1	-4

2.3. Найдите оптимумы функции $f(x, y) = ax^3 + ax^2y + bx + \frac{1}{3}y^3 + cy$.

Вариант	a	b	c	Вариант	a	b	c
1	-5/4	-5/2	-11/2	11	-2	-2	-17/8
2	-5/4	5/2	1/2	12	-2	2	7/8
3	-5/4	-10	-22	13	-2	-8	-17/2
4	-5/4	10	2	14	-2	8	7/2
5	-5/4	-5	-11	15	-2	-4	-17/4
6	-5/4	5	1	16	-2	4	7/4
7	-5/4	-5/2	-19/3	17	-2	-2	-7/3
8	-5/4	5/2	-1/3	18	-2	2	2/3
9	-5/4	-15/2	-19	19	-2	-6	-7
10	-5/4	15/2	-1	20	-2	6	2

Задача 3. Многомерная оптимизация, задачи с ограничениями-равенствами.

3.1. Найдите оптимумы функции $f(x, y) = ax^2 + 2xy + by^2$
при условии $4x^2 + cy^2 = 9$.

Вариант	a	b	c	Вариант	a	b	c
1	1	1	3	11	5	8	6
2	3	4	5	12	5	3	2
3	1	1	2	13	9	3	1
4	1	2	6	14	9	7	3
5	1	1	1	15	3	6	7
6	3	2	2	16	7	4	2
7	5	4	3	17	13	4	1
8	3	5	6	18	7	11	6
9	7	9	5	19	9	5	2
10	5	2	1	20	7	13	7

3.2. Найдите оптимумы функции $f(x, y) = ax^3 + by^3$
при условии $x^2 + y^2 = c^2$.

Вариант	a	b	c	Вариант	a	b	c
1	3	2	2	11	1	8	1
2	7	3	3	12	4	5	1
3	1	2	5	13	5	3	4
4	7	2	1	14	2	5	3
5	3	4	2	15	2	1	2
6	3	1	3	16	2	3	4
7	4	3	4	17	2	7	1
8	1	4	5	18	8	1	2
9	5	2	2	19	1	3	5
10	3	5	1	20	3	7	3

3.3. Найдите оптимумы функции $f(x, y) = y^2$
при условии $ax^3 + by^3 - 3cxy = d$.

Вариант	a	b	c	d	Вариант	a	b	c	d
1	9	2	9	-16	11	-1	2	1	-4
2	-36	1	9	8	12	-4	1	2	-16
3	-9	2	9	16	13	-18	4	9	4
4	7	7	7	16	14	9	16	9	-2
5	3	2	3	-4	15	14	4	7	4
6	12	1	6	-16	16	-7	16	7	-2
7	1	2	1	4	17	28	1	7	8
8	4	1	2	16	18	1	2	1	12
9	-7	2	7	-16	19	1	16	2	12
10	36	1	9	-8	20	1	1	1	3

Задача 4. Многомерная оптимизация, задачи со смешанными ограничениями.

Вариант 1. Найдите $\min f(x) = x_1^2 + x_2$ при условии

$$\begin{cases} q_1(x) = -x_1^2 - x_2^2 + 9 \geq 0, \\ q_2(x) = -x_1 - x_2 + 1 \geq 0. \end{cases}$$

Вариант 2. Найдите $\min f(x) = 4x_1^2 + 4x_2^2$ при условии

$$\begin{cases} q_1(x) = -4x_1^2 - 4x_2^2 + 8 \geq 0, \\ q_2(x) = \ln x_2 \geq 0, \\ q_3(x) = -x_1 \geq 0, \\ q_4(x) = x_2 \geq 0. \end{cases}$$

Вариант 3. Найдите $\min f(x) = 3x_1^2 + 4x_2^2 - x_1$ при условии

$$\begin{cases} q_1(x) = 7 - x_1^2 - x_2^2 \geq 0, \\ q_2(x) = 3 - x_1 \geq 0, \\ q_3(x) = x_1 \geq 0, \\ q_4(x) = x_2 \geq 0. \end{cases}$$

Вариант 4. Найдите $\min f(x) = -\ln x_1 + x_2$ при условии

$$\begin{cases} q_1(x) = -x_1 + x_2 + 2 \geq 0, \\ q_2(x) = -x_1 - x_2^2 + 4 \geq 0, \\ q_3(x) = x_1 \geq 0. \end{cases}$$

Вариант 5. Найдите $\min f(x) = 2x_1^2 + 2x_2^2 - 2x_1$ при условии

$$\begin{cases} q_1(x) = -x_1^2 - 2x_2^2 + 7 \geq 0, \\ q_2(x) = -4x_1 + 3x_2 + 8 \geq 0, \\ q_3(x) = \ln x_1 \geq 0. \end{cases}$$

Вариант 6. Найдите $\min f(x) = x_1^2 + 2x_2^2 - x_2$ при условии

$$\begin{cases} q_1(x) = 4x_1 + 3x_2 \leq 10, \\ q_2(x) = -x_1 - x_2^2 \geq -2, \\ q_3(x) = x_1 \geq 0, \\ q_4(x) = x_2 \geq 0. \end{cases}$$

Вариант 7. Найдите $\min f(x) = 3x_1^2 + x_2^2 + 6x_3^2$ при условии

$$\begin{cases} q_1(x) = x_1 - 2x_2^2 + 4x_3 \geq 0, \\ q_2(x) = -x_2^2 - x_3^2 + 5x_1 \geq 0, \\ h_1(x) = x_1 + 2x_2 - 3x_3 - 4 = 0, \\ h_2(x) = x_1 + 4x_2 - x_3 + 3 = 0. \end{cases}$$

Вариант 8. Найдите $\min f(x) = x_1^2 + x_2^2 - 4x_1 + 4$ при условии

$$\begin{cases} q_1(x) = x_1 - x_2 + 2 \geq 0, \\ q_2(x) = -x_1^2 + x_2 - 1 \geq 0, \\ q_3(x) = x_1 \geq 0, \\ q_4(x) = x_2 \geq 0. \end{cases}$$

Вариант 9. Найдите $\min f(x) = x_1^2 - 4x_1 + x_2^2 - 3x_2$ при условии

$$\begin{cases} q_1(x) = x_1^2 + x_2^2 \leq 7, \\ q_2(x) = x_1 \leq 2, \\ q_3(x) = x_1 \geq 0, \\ q_4(x) = x_2 \geq 0. \end{cases}$$

Вариант 10. Найдите $\min f(x) = 2x_1^2 + 2x_2^2 - x_1$ при условии

$$\begin{cases} q_1(x) = 2x_1 + x_2 \leq 3, \\ q_2(x) = -x_1 - 2x_2^2 \geq -1, \\ q_3(x) = x_1 \geq 0, \\ q_4(x) = x_2 \geq 0. \end{cases}$$

Вариант 11. Найдите $\min f(x) = x_1^2 - 4x_1 + 2x_2^2 - 3x_2$ при условии

$$\begin{cases} q_1(x) = 5 - x_2 \leq 0, \\ q_2(x) = 7 - x_1^2 - 3x_2^2 \geq 0, \\ q_3(x) = x_1 \geq 0, \\ q_4(x) = x_2 \geq 0. \end{cases}$$

Вариант 12. Найдите $\min f(x) = x_1^2 - 2x_1 + 2x_2^2 - x_2$ при условии

$$\begin{cases} q_1(x) = 2x_1^2 + 3x_2^2 \leq 8, \\ q_2(x) = x_2 \leq 5, \\ q_3(x) = x_1 \geq 0, \\ q_4(x) = x_2 \geq 0. \end{cases}$$

Вариант 13. Найдите $\min f(x) = 2x_1 + 3x_2^2 + 4x_3^2$ при условии

$$\begin{cases} h_1(x) = x_1 + x_2 + x_3 = 0, \\ h_2(x) = 2x_1 - 3x_2 - x_3 = 0, \\ q_1(x) = x_1 - 2x_2^2 + x_3 \geq 0, \\ q_2(x) = -3x_1^2 + x_2 - 5x_3^2 \geq 0. \end{cases}$$

Вариант 14. Найдите $\min f(x) = x_1^2 + x_2^2 + x_3 + 9$ при условии

$$\begin{cases} h_1(x) = x_1 - x_2 - 2x_3 + 4 = 0, \\ h_2(x) = x_1 + x_2 + x_3 + 1 = 0, \\ q_1(x) = 2x_1 - x_2^2 - x_3^2 + 3 \geq 0, \\ q_2(x) = -3x_1^2 - x_2 - x_3^2 + 20 \geq 0. \end{cases}$$

Вариант 15. Найдите $\min f(x) = x_1^2 + 2x_2^2 + 3x_3$ при условии

$$\begin{cases} q_1(x) = 20 - 3x_2^2 - x_3^2 \geq 0, \\ q_2(x) = 22 - x_3^2 - x_2^2 - x_1 \geq 0, \\ h_1(x) = x_1 - 2x_2 - 3x_3 - 4 = 0, \\ h_2(x) = 2x_1 - x_2 - x_3 - 4 = 0. \end{cases}$$

Вариант 16. Найдите $\min f(x) = x_1^2 + x_2^2 + 2x_3$ при условии

$$\begin{cases} q_1(x) = 20 - 3x_2^2 - x_3^2 \geq 0, \\ q_2(x) = 22 - x_3^2 - x_2^2 - x_1 \geq 0, \\ h_1(x) = x_1 - 2x_2 - 3x_3 - 4 = 0, \\ h_2(x) = 2x_1 - x_2 - x_3 - 4 = 0. \end{cases}$$

Вариант 17. Найдите $\min f(x) = x_1^2 + x_2^2 + x_3 + 4$ при условии

$$\begin{cases} h_1(x) = x_1 + x_2 + x_3 + 1 = 0, \\ h_2(x) = x_1 - x_2 - 2x_3 + 3 = 0, \\ q_1(x) = -3x_1^2 - x_2 - x_3^2 + 20 \geq 0, \\ q_2(x) = -2x_1^2 - x_2^2 - x_3^2 + 30 \geq 0. \end{cases}$$

Вариант 18. Найдите $\min f(x) = 2x_1^2 + x_2^2 + x_3^2 + x_4^2$ при условии

$$\begin{cases} q_1(x) = -x_1 - x_2 - x_3 - x_4 + 4 \leq 0, \\ q_2(x) = x_1 - 2x_2 - x_3 - 2x_4 \geq 0, \\ h_1(x) = 5x_1 - 5x_2 - 9x_3 - x_4 = 0, \\ h_2(x) = x_1 + x_2 - x_3 - x_4 = 0. \end{cases}$$

Вариант 19. Найдите $\min f(x) = x_1^2 + x_2^2 + x_3 + 10$ при условии

$$\begin{cases} h_1(x) = x_1 - x_2 - 2x_3 + 3 = 0, \\ h_2(x) = x_1 + x_2 + x_3 + 1 = 0, \\ q_1(x) = -2x_1 - x_2^2 - x_3^2 + 30 \geq 0, \\ q_2(x) = -3x_1^2 - x_2 - x_3^2 + 20 \geq 0. \end{cases}$$

Вариант 20. Найдите $\min f(x) = x_1^2 + x_2^2 + x_3 + 1$ при условии

$$\begin{cases} h_1(x) = 3 - x_1^2 - x_2^2 - x_3^2 \geq 0, \\ h_2(x) = 1 - x_1^2 \geq 0, \\ q_1(x) = x_1 - x_2 = 0, \\ q_2(x) = x_2 - x_3 - 2 = 0. \end{cases}$$

Часть 2. Псевдообращение

1. Докажите формулы Фробениуса блочных определителей

$$\det A \neq 0 \Rightarrow \det \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \det A \cdot \det H, H = D - CA^{-1}B,$$

$$\det D \neq 0 \Rightarrow \det \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \det K \cdot \det D, K = A - BD^{-1}C.$$

2. Докажите формулы Фробениуса блочных матриц

$$\det A \neq 0 \Rightarrow \det \begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} =$$

$$= \begin{bmatrix} A^{-1} + A^{-1}BH^{-1}CA^{-1} & \vdots & -A^{-1}BH^{-1} \\ \dots & \dots & \dots \\ -H^{-1}CA^{-1} & \vdots & H^{-1} \end{bmatrix},$$

$$\det D \neq 0 \Rightarrow \det \begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} =$$

$$= \begin{bmatrix} K^{-1} & \vdots & -K^{-1}BD^{-1} \\ \dots & \dots & \dots \\ -D^{-1}CK^{-1} & \vdots & D^{-1} + D^{-1}CK^{-1}BD^{-1} \end{bmatrix},$$

$$\det A \neq 0, \det D \neq 0 \Rightarrow \det \begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} =$$

$$= \begin{bmatrix} K^{-1} & \vdots & -K^{-1}BD^{-1} \\ \dots & \dots & \dots \\ -H^{-1}CA^{-1} & \vdots & H^{-1} \end{bmatrix} =$$

$$= \begin{bmatrix} K^{-1} & \vdots & -A^{-1}BH^{-1} \\ \dots & \dots & \dots \\ -D^{-1}CK^{-1} & \vdots & H^{-1} \end{bmatrix},$$

$$\det A \neq 0, \quad \det B \neq 0, \quad \det c \neq 0, \quad \det D \neq 0 \Rightarrow$$

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} = \begin{bmatrix} K^{-1} & L^{-1} \\ C^{-1} & H^{-1} \end{bmatrix}, \quad C = B - AC^{-1}D, L = C - DB^{-1}A.$$

3. Докажите формулы обращения некоторых матричных сумм

$$\begin{aligned} (A + UMV)^{-1} &= A^{-1} - A^{-1}(I + UMV A^{-1})^{-1}UMV A^{-1}, \\ &= A^{-1} - A^{-1}U(I + MVA^{-1}U)^{-1}MVA^{-1}, \\ &= A^{-1} - A^{-1}UM(I + VA^{-1}UM)^{-1}VA^{-1}, \\ &= A^{-1} - A^{-1}UMV(A^{-1}MV)^{-1}A^{-1}, \\ &= A^{-1} - A^{-1}UMVA^{-1}(I + UMV A^{-1})^{-1}. \end{aligned}$$

4. Докажите

$$(A^T A)^+ = A^+(AA^T)^+ A = A^T(AA^T)^+(A^T)^+.$$

5. Докажите

$$\begin{aligned} AA^+ &= (AA^T)(AA^T)^+ = (AA^T)^+(AA^T), \\ A^+A &= (A^T A)(A^T A)^+ = (A^T A)^+(A^T A). \end{aligned}$$

6. Докажите

$$AA^+A^k = A^k, \quad k > 0.$$

7. Докажите

$$\begin{aligned} A^T = A, k > 0 &\Rightarrow (A^k)^+ = (A^+)^k, \\ A^k A^+ &= A^+ A^k, A^k (A^k)^+ = (A^k)^+ A^k = AA^+ = A^+ A. \end{aligned}$$

8. Докажите

$$P^T = P, P^2 = P \Rightarrow (PA)^+ = (PA)^T P, (AP)^+ = P(AP)^+.$$

9. Докажите

$$\begin{aligned} \operatorname{rg} A &= \operatorname{rg} AA^+ A \leq \operatorname{rg} A^+, \operatorname{rg} A^+ = \operatorname{rg} A^+ AA^+ \leq \operatorname{rg} A \Rightarrow \operatorname{rg} A = \operatorname{rg} A^+, \\ \operatorname{rg} A &= \operatorname{rg} A^T = \operatorname{rg} AA^T = \operatorname{rg} A^T A = \operatorname{rg} A^+ = \operatorname{rg} AA^+ = \operatorname{rg} A^+ A = \\ &= \operatorname{tr} A^+ A = \operatorname{tr} AA^+ = \operatorname{tr} A^+ A, \\ A^T B &= 0, BA^T = 0 \Rightarrow (A + B)^+ = A^+ + B^+. \end{aligned}$$

10. Докажите $(AB)^+ = B^+A^+$, если:

или $B = A^T$, или $B = A^+$,

или $A^T A = I$ или $BB^T = I$,

или $\text{rg} A = \text{rg} B = r, A \in R_r^{m \times r}, B \in R_r^{r \times n}$.

11. Докажите $(A \otimes B)^+ = A^+ \otimes B^+$ (тензорное произведение).

12. Докажите

$$A = \begin{bmatrix} B & \vdots & C \end{bmatrix}, \quad B^T C = 0 \Rightarrow A^+ = \begin{bmatrix} B^+ \\ \dots \\ C^+ \end{bmatrix},$$

$$A = \begin{bmatrix} B \\ \dots \\ C \end{bmatrix}, \quad BC^T = 0 \Rightarrow A^+ = \begin{bmatrix} B^+ & \vdots & C^+ \end{bmatrix}.$$

13. Докажите формулу Клайна псевдообращения блочных матриц

$$\begin{bmatrix} U & \vdots & V \end{bmatrix} = \begin{bmatrix} U^+ - U^+ V J \\ \dots \\ J \end{bmatrix},$$

где $C = (I - UU^+)V$,

$$K = I + [U^+ V (I - C^+ C)]^T [U^+ V (I - C^+ C)]^{-1},$$

$$J = C^+ (I - C^+ C) K V^T (U^+)^T U^+ (I - V C^+).$$

14. Докажите формулу псевдообращения некоторых матричных сумм

$$\begin{aligned} (UU^T + VV^T)^+ &= (CC^T)^+ + (I - (VC^+)^T) \times \\ &\times ((UU^T)^+ - (UU^T)^+ V (I - C^+ C) K V^T (UU^T)^+) \cdot (I - VC^+), \end{aligned}$$

где $C = (I - (UU^T)(UU^T)^+)V$,

$$K = I + (I - C^+ C) V (UU^T)^+ V (I - C^+ C)^{-1}.$$

15. Докажите $J_{m \times n} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & 1 & \dots & 1 \\ \vdots & \vdots & & \vdots \\ 1 & 1 & \dots & 1 \end{bmatrix}$, $J_{m \times n}^+ = \frac{1}{m-n} J_{n \times m}$.

16. Докажите $K_2 = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$, $K_2^+ = \frac{1}{4} K_2$.

17. Докажите

$$\begin{bmatrix} I_m & \vdots & I_m \vdots & \dots & \vdots I_m \end{bmatrix}^+ = \frac{1}{n} \begin{bmatrix} I_m & \vdots & I_m \vdots & \dots & \vdots I_m \end{bmatrix}^T.$$

18. Докажите

$$\begin{bmatrix} & & I_m & & \\ \dots & \dots & \dots & \dots & \dots \\ 1 & 1 & \dots & 1 & 1 \end{bmatrix}^+ = \frac{1}{m+1} \begin{bmatrix} m & -1 & \dots & -1 & \vdots & 1 \\ -1 & m & \dots & -1 & \vdots & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ -1 & -1 & \dots & m & \vdots & 1 \end{bmatrix}.$$

19. Вычислите

$$\begin{pmatrix} -1 & 0 & 1 & 2 \\ -1 & 1 & 0 & -1 \\ 0 & -1 & 1 & 3 \\ 0 & 1 & -1 & -3 \\ 1 & -1 & 0 & 1 \\ 1 & 0 & -1 & 2 \end{pmatrix}^+ = \frac{1}{102} \begin{pmatrix} -15 & 18 & 3 & -3 & 18 & 15 \\ 8 & 13 & -5 & 5 & -13 & -8 \\ 7 & 5 & 2 & -2 & -5 & -7 \\ 6 & -3 & 9 & 9 & 3 & -6 \end{pmatrix}.$$

20. Вычислите

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 83 & -47 & 1 & 0 & 0 \\ -55 & 94 & 0 & 1 & 0 \\ 62 & -71 & 0 & 0 & 1 \end{pmatrix}^+ = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ -83 & 47 & 1 & 0 & 0 \\ 55 & -94 & 0 & 1 & 0 \\ -62 & 71 & 0 & 0 & 1 \end{pmatrix}.$$

Часть 3. Содержательные задачи

Задача 1. Водный инспектор получил задание поставить опознавательный знак на самом глубоком месте некоего водоёма. Площадь водоёма представляет собой систему координат. Известно, что дно водоёма на всей его площади может быть описано функцией

$$f(x_1, x_2) = ax_1^2 + 2x_1x_2 + bx_2^2 - 2x_1 - 3x_2,$$

указывающей глубину в метрах над уровнем моря. Найдите координаты места, в котором инспектору необходимо поставить этот опознавательный знак.

Вариант	a	b	Вариант	a	b	Вариант	a	b	Вариант	a	b
1	1	2	6	2	1	11	3	1	16	4	1
2	1	3	7	2	2	12	3	2	17	4	2
3	1	4	8	2	3	13	3	3	18	4	3
4	1	5	9	2	4	14	3	4	19	4	4
5	1	6	10	2	5	15	3	5	20	5	1

Задача 2. Перед государственной инспекцией безопасности дорожного движения была поставлена задача соединить две автомагистрали. Местность представляет собой систему координат. Известно, что одна автомобильная магистраль приближённо представляет собой параболу $ax_1^2 - x_2 = 0$, а другая прямую $x_1 - x_2 + a = 0$. Определите такое соединение автомагистралей, при котором государство понесёт наименьшие затраты.

Вариант	a	Вариант	a	Вариант	a	Вариант	a
1	0	6	-3	11	-4	16	4,5
2	1	7	-2	12	2,5	17	-4,5
3	-1	8	5	13	-0,5	18	2,25
4	2	9	5,5	14	0,5	19	-2,25
5	3	10	-1,5	15	5	20	-0,75

Задача 3. При изучении объёмных фигур по геометрии преподаватель в качестве домашнего задания задал своим ученикам собрать усечённую пирамиду с квадратным основанием (x_1 и x_2 – рёбра квадратов оснований) так, чтобы её объём был максимальный, а высота пирамиды составила c см. Для этого нужно найти максимум функции

$$f(x_1, x_2) = \frac{1}{3}c(x_1 + x_2 + \sqrt{x_1x_2}).$$

Найдите соответствующие длины рёбер квадратов оснований.

Вариант	a	Вариант	a	Вариант	a	Вариант	a
1	15	6	38	11	56	16	17
2	18	7	90	12	72	17	27
3	25	8	45	13	64	18	31
4	21	9	10	14	30	19	53
5	54	10	74	15	40	20	89

Задача 4. Эллиптический параболоид имеет структуру

$$f(x) = (x_1 - p)^2 + (x_2 - q)^2.$$

Определите точку минимума параболоида численно и сравните достоверность полученного результата с аналитически вычисленным оптимумом.

Вариант	p	q	Вариант	p	q
1	2	3	11	11	5
2	-1	4	12	9	2
3	6	7	13	1	5
4	3	2	14	-6	0
5	-5	3	15	4	4
6	4	5	16	9	3
7	1	3	17	-2	4
8	7	2	18	-1	3
9	2	2	19	-3	5
10	-7	4	20	6	-2

Задача 5. На околоземной орбите находится спутник.

Уравнение орбиты $(x - a)^2 + (y - b)^2 + (z - c)^2 - 100d^2 = 0$. Со спутником должна состыковаться ракета. Найдите точку стыковки, если спутник может свободно перемещаться по орбите, а траектория ракеты описывается системой

$$\begin{cases} abx + by + cz + d = 0, \\ cx + bdy + (a + b)z + a = 0. \end{cases}$$

Вариант	a	b	c	d	$x_0^{(0)}$	$x_1^{(0)}$	$x_2^{(0)}$
1	0,5	1	10	4	0,5	1	0,5
2	2	1	4,5	3,5	2	5	3
3	3	4	0,5	2	0,5	1,5	2,5
4	2	1	3	5	2	-0,5	2
5	7	-5	5	2,5	11	5	3
6	5	2,5	3,5	3	2	4	4,5
7	0,5	2	0,5	2,5	5	5,5	2
8	-0,5	1,5	3	2	2,5	4	5,5
9	2,5	2	2	3,5	26	5	3
10	2,5	2	1	1,5	4	-0,5	1,5
11	0,5	1	-0,5	0,5	2	1,5	5
12	1,5	2	3	2	2	3	2,5
13	3	5,5	2	2,5	-5	0,5	1
14	10	5	1	6	1	0,5	1,5
15	2	1	-0,5	-5	2,5	2	5
16	1	3	4	4	2	11	2,5
17	3	7	11	3	2	5	1
18	2	9	10	1	1	5,5	1,5
19	6	4	4	1	10	1	-0,5
20	7	6	11	2	1	1,5	3,5

Задача 6. Кривая спроса описывается функцией $-\exp(x_0) + abx_1 = 0$.
Кривая предложения $-cx_0 + dx_1 + cd = 0$. Найдите оптимальную цену.

Вар.	a	b	c	d	$x_0^{(0)}$	$x_1^{(0)}$	Вар.	a	b	c	d	$x_0^{(0)}$	$x_1^{(0)}$
1	1,5	2	3	2	2	3	11	2,5	2	2	3,5	26	5
2	3	5,5	2	2,5	-5	0,5	12	2,5	10	3,5	4	1	8
3	2	3	1	2,5	-6	11	13	2	11	1,5	1	5	3
4	2	1	-0,5	-5	16	15	14	4	2	-1	2	7,5	1
5	1	3	4	4	1	0,5	15	2	1	2	3,5	4	1,5
6	3	7	11	3	5	3	16	2	1,5	-5	0,5	4,5	4
7	2	2	3	4	16	15	17	-5	5	10	2	1	7
8	-9	2,5	7	3	3	15	18	1	6	-0,5	-1	2	4,5
9	0,5	2	0,5	2,5	5	5,5	19	7	11	3	2	5	1
10	-0,5	1,5	3	2	2,5	4	20	9	10	1	1	5,5	1,5

Задача 7. Для построения графика корневого годографа необходимо найти точки пересечения с мнимой осью. Решите систему уравнений

$$\begin{cases} U(w) = aw^2 + bk = 0, \\ V(w) = cw - w^3 = 0. \end{cases}$$

Вариант	a	b	c	$x_0^{(0)}$	$x_1^{(0)}$	Вариант	a	b	c	$x_0^{(0)}$	$x_1^{(0)}$
1	2	3	4	2	10	11	2	10	6	4,5	1,5
2	5,5	2	0,5	3	4,5	12	1,5	1,5	5	16	5
3	3	3	-0,5	1,5	0,5	13	4	3	2	1	2,5
4	4,5	1	2,5	16	3	14	-0,5	15	3,4	5	2
5	7,5	2	2,5	2	5	15	3	4	4	1	0,5
6	3,5	4	2	7	3,5	16	7	11	3	5	3
7	1	2,5	4	5	0,5	17	2	3	4	16	15
8	15	4,5	4,5	4	3	18	2	4,5	2	-1	2
9	3	7	7	3,5	5,5	19	1	1,3	1	2	3,5
10	16	9,5	8	16	3	20	4	7,8	1,5	-5	0,5

Задача 8. Река Ока впадает в реку Волга. Русло Волги описывается функцией $a(|x_0 + b|)^{\frac{1}{2}} + cx_1 = 0$. Русло Оки описывается функцией $a \exp(x_0) + cx_1 = 0$. Найдите координаты точки впадения Оки в Волгу.

Вариант	a	b	c	$x_0^{(0)}$	$x_1^{(0)}$	Вариант	a	b	c	$x_0^{(0)}$	$x_1^{(0)}$
1	2	3	4	2	10	11	4,4	10	3,4	10	1,5
2	5,5	2	0,5	3	4,5	12	5	2,5	3,5	3	2
3	2	11	1,5	1	5	13	0,5	2	0,5	2,5	5
4	4,5	2	-1	2	7,5	14	-0,5	1,5	3	2	2,5
5	2	1	2	3,5	4	15	2,5	2	2	3,5	26
6	3,5	4	2	4	3,5	16	5	3	4,8	3	6
7	1	2,5	8,5	2,5	0,5	17	2,4	7	2,9	5,1	7,5
8	2,5	4,5	6	4,5	3	18	3,4	4,5	2	-1	2
9	2,5	7	5,5	7	5,5	19	4,9	1,3	1	2	3,5
10	2	9,5	2	9,5	3	20	4,8	7,8	1,5	-5	0,5

Часть 4. Тестовые функции

Далее приведены тестовые задачи, каждая из которых подходит для безусловной минимизации, решения систем нелинейных уравнений и нелинейных задач о наименьших квадратах. Решения многих из этих задач могут быть найдены при различных значениях n . Для каждой задачи указана точка оптимума x^* .

Расширенная функция Розенброка

n – любое положительное число, кратное 2, для $i = 1, \dots, \frac{n}{2}$:

$$f_{2i-1}(x) = 10(x_{2i} - x_{2i-1}^2), \quad f_{2i}(x) = 1 - x_{2i-1};$$

$$x^* = (1, 1, \dots, 1, 1).$$

Расширенная обобщённая функция Пауэлла

n – любое положительное число, кратное 4, для $i = 1, \dots, \frac{n}{4}$:

$$f_{4i-3}(x) = x_{4i-3} + 10x_{4i-2}, \quad f_{4i-2}(x) = \sqrt{5}(x_{4i-1} - x_{4i}),$$

$$f_{4i-1}(x) = (x_{4i-2} - 2x_{4i})^2, \quad f_{4i}(x) = \sqrt{10}(x_{4i-3} - x_{4i})^2;$$

$$x^* = (0, 0, 0, 0, \dots, 0, 0, 0, 0).$$

Функция Вуда

$$n = 4,$$

$$f(x) = 100(x_1^2 - x_2)^2 + (1 - x_1)^2 + 90(x_3^2 - x_4)^2 + (1 - x_3^2) + \\ + 10,1((1 - x_2)^2 + (1 - x_4)^2) + 19,8(1 - x_2)(1 - x_4);$$

$$x^* = (1, 1, 1, 1).$$

Функция Растригина

n – любое положительное число,

$$f(x) = An + \sum_{i=1}^n [x_i^2 - A \cos(2\pi x_i)],$$

где $A = 10$ и $x_i \in (-5.12, 5.12)$;

$$x^* = (0, \dots, 0).$$

Функция Де Йонга 1 (сфера)

n – любое положительное число,

$$f(x) = \sum_{i=1}^n x_i^2;$$

$$x^* = (0, \dots, 0).$$

Функция Грайвенка

n – любое положительное число,

$$f(x) = 1 + \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right), \quad x_i \in (-600, 600);$$

$$x^* = (0, \dots, 0).$$

Функция Изома

$$n = 2,$$

$$f(x_1, x_2) = -\cos(x_1) \cos(x_2) \exp(-(x_1 - \pi)^2 + (x_2 - \pi)^2),$$

$$x_i \in (-100, 100); x^* = (-\pi, \pi).$$

Заключение

Для решения большого числа задач, описываемых общим понятием «математическое программирование» эффективно используются различные методы оптимизации, направленные на отыскание минимума (максимума) широких классов целевых функций при наличии (или отсутствии) разнообразных ограничений, от структуры и вида которых зависит не только название методов, но и их математическое и логическое построение.

Общая задача нелинейной условной минимизации, сформулированная в виде: найти $\min_{x \in X} f(x) = f(x^*) = f^*$, где $x = (x_1 \dots x_n)^T \in X \subset \mathbb{R}^n$ – векторный аргумент, $X \subset \mathbb{R}^n$ – некоторое множество, $f(x) = f(x_1, \dots, x_n) \in \mathbb{R}$ – нелинейная скалярная функция векторного аргумента (целевая функция, критерий оптимальности), f^* – её минимум, x^* – значение аргумента, доставляющее этот минимум, представлена в большом количестве работ. Классификации и исследованию частных задач и их разновидностей, разработке методов и алгоритмов их решения посвящено также большое число работ, которые по различным причинам не всегда доступны студенту или инженеру и не всегда пригодны для первого знакомства с предметом. Поэтому в пособии изложены сведения по данным вопросам в виде не только удобном для последующего использования, но и позволяющем читателю, впервые обратившемуся к задачам нелинейного программирования, сориентироваться в этой области. В пособии приведён пример содержательной задачи из области регулирования дорожного движения – нахождение оптимального времени цикла светофорного регулирования и оптимального распределения найденного времени по фазам регулирования. Для решения поставленной задачи предложен численный алгоритм многомерной нелинейной оптимизации.

Обобщением понятия обратной, однозначно определённой для квадратной невырожденной матрицы является понятие псевдообратной матрицы, которая существует и единственна для любой матрицы с произвольным числом строк, столбцов и произвольного ранга. Псевдообратная совпадает с обратной для квадратной невырожденной матрицы.

Аппарат псевдообращения широко применяется при точном или приближённом исследовании и решении систем линейных алгебраических уравнений и приводящихся к ним разнообразных прикладных задач из таких областей как математическое программирование, математическая статистика, математическое моделирование систем и управление ими и многих других. Применения к нелинейным задачам опираются на различные схемы линеаризации.

Нелинейный метод наименьших квадратов служит для решения нелинейной задачи о наименьших квадратах, которая относится к классу задач нелинейной оптимизации или нелинейного программирования, однако характеризуется в этом классе спецификой, оправдывающей её самостоятельное исследование.

Библиографический список

1. Айвазян, С.А. Прикладная статистика. Исследование зависимостей [Текст] / С.А. Айвазян, И.С. Енюков, Л.Д. Мешалкин. – Москва: Финансы и статистика, 1985. – 487 с.
2. Алберт, А. Регрессия, псевдоинверсия и рекуррентное оценивание [Текст] / А. Алберт. – Москва: Наука, 1977. – 224 с.
3. Алгоритмы: построение и анализ [Текст] / Т. Кормен [и др.]. – Москва: Изд. дом «Вильямс», 2005. — 1296 с.
4. Балонин, Н.А. Методы ортогонализации и псевдоинверсии в алгебраических задачах теории автоматического управления [Текст] / Н.А. Балонин // Проектирование авиационных адаптивных систем и их элементов. – Ленинград: Изд-во ЛИАП, 1985. – С. 35-40.
5. Бард, Й. Нелинейное оценивание параметров [Текст] / Й. Бард. – Москва: Статистика, 1979. – 349 с.
6. Барышев, В.Г. К управлению системами с многомерным параметром [Текст] / В.Г. Барышев, С.Л. Блюмин, Л.А. Кузнецов // Автоматика и телемеханика. – 1977. – № 4. – С. 37-42.
7. Бейко, И.В. Методы и алгоритмы решения задач оптимизации [Текст] / И.В. Бейко, Б.Н. Бублик, П.Н. Зинько. – Киев: Головное издательство издательского объединения «Вища школа», 1983. — 512 с.
8. Беклемишев, Д.В. Дополнительные главы линейной алгебры [Текст] / Д.В. Беклемишев. – Москва: Наука, 1983. – 336 с.

9. Бертсекас, Д. Условная оптимизация и метод множителей Лагранжа [Текст] / Д. Бертсекас. – Москва: Радио и связь, 1987. – 400 с.
10. Блюмин, С.Л. Обратные задачи динамики дискретно-аргументных систем [Текст] / С.Л. Блюмин // Изв. АН. СССР. Техн. кибернет. – 1980. – № 6. – С. 198.
11. Блюмин, С.Л. Соотношение типа Кэли-Гамильтона в теории дискретно-аргументных систем [Текст] / С.Л. Блюмин // Автоматика и телемеханика. – 1981. – № 9. – С. 133-142.
12. Блюмин, С.Л. К построению алгоритмов управления движением линейных систем на основе решения обратных задач динамики [Текст] / С.Л. Блюмин, В.В. Дариенко // Известия академии наук СССР. Техническая кибернетика. – 1982. – № 1. – С. 189-194.
13. Блюмин, С.Л. Обратные задачи динамики и ортогональные канонические формы линейных систем [Текст] / С.Л. Блюмин // Известия академии наук СССР. Техническая кибернетика. – 1984. – № 3. – С. 191.
14. Блюмин, С.Л. Обратные задачи динамики и динамические транспортные задачи [Текст] / С.Л. Блюмин, С.П. Миловидов // Известия академии наук СССР. Техническая кибернетика. – 1987. – № 5. – С. 209.
15. Блюмин, С.Л. К исследованию и решению целочисленных систем линейных уравнений [Текст] / С.Л. Блюмин, Ю.И. Денисенко, С.П. Миловидов // Журнал вычислительной математики и математической физики. – 1988. – Т. 28, № 6. – С. 787-792.
16. Блюмин, С.Л. Псевдообращение: учеб. пос. [Текст] / С.Л. Блюмин, С.П. Миловидов. – Воронеж: ВорПИ-ЛипПИ, 1990. – 72 с.
17. Блюмин, С.Л. Взвешенное псевдообращение: учеб. пос. [Текст] / С.Л. Блюмин, С.П. Миловидов. – Воронеж: ВорПИ-ЛипПИ, 1991. – 64 с.

18. Блюмин, С.Л. Блочные рекуррентно-итерационные процедуры решения нелинейной задачи о наименьших квадратах [Текст] / С.Л. Блюмин, А.К. Погодаев // Журнал вычислительной математики и математической физики. – 1992. – Т. 32, № 8. – С. 1180-1186.
19. Блюмин, С.Л. Нелинейный метод наименьших квадратов и псевдообращение [Текст]: учеб. пос. / С.Л. Блюмин, С.П. Миловидов, А.К. Погодаев. – Липецк: ЛПИ, 1992. – 80 с.
20. Блюмин, С.Л. Алгоритмы блочной адаптации линейных и нелинейных моделей технологических зависимостей [Текст] / С.Л. Блюмин, А.К. Погодаев // Известия вузов. Чёрная металлургия. – 1992. – № 9. – С. 67-68.
21. Блюмин, С.Л. Оптимальное моделирование технологических связей [Текст]: учеб. пос. / С.Л. Блюмин, А.К. Погодаев, В.В. Барышев. – Липецк: ЛПИ, 1993. – 68 с.
22. Блюмин, С.Л. Исследование и решение матричных уравнений над ассоциативными кольцами [Текст] / С.Л. Блюмин, С.П. Миловидов // Журнал вычислительной математики и математической физики. – 1994. – Т. 34, № 2. – С. 163-174.
23. Блюмин, С.Л. Суперпозиционная регрессия [Текст] / С.Л. Блюмин, А.К. Погодаев // Журнал вычислительной математики и математической физики. – 1995. – Т. 35, № 10. – С. 1576-1581.
24. Блюмин, С.Л. Пошаговая нелинейная регрессия по последовательно поступающим данным [Текст] / С.Л. Блюмин, А.К. Погодаев // Заводская лаборатория. – 1995. – № 10. – С. 51-57.
25. Блюмин, С.Л. Основы прикладной математики. Оптимизационная математика [Текст]: учеб. пос. / С.Л. Блюмин, С.П. Миловидов, А.К. Погодаев. – Липецк: НОУ ЛЭГИ, 1999. – 81 с.
26. Блюмин, С.Л. Рекуррентно-итерационные алгоритмы адаптивной идентификации нелинейных динамических сосредоточенных систем

[Текст] / С.Л. Блюмин, А.К. Погодаев // Автоматика и телемеханика. – 2003. – № 10. – С. 80-86.

27. Блюмин, С.Л. Адаптивные процедуры в решении задач оптимизации качества металлопродукции [Текст] / С.Л. Блюмин, А.К. Погодаев // Известия вузов. Чёрная металлургия. – 2003. – № 3. – С. 60-62.
28. Блюмин, С.Л. Явное выражение псевдообратной лапласиана двудольного графа [Текст] / С.Л. Блюмин, С.П. Миловидов // Управление большими системами: сб. тр. – 2007. – № 18. – С. 24-29.
29. Блюмин, С.Л. Применение теоремы Лагранжа о конечных приращениях для решения проблем управления транспортными системами / С.Л. Блюмин, А.С. Сысоев // Проблемы управления. – 2014. – № 1. – С. 82-87.
30. Бояринцев, Ю.Е. Регулярные и сингулярные системы линейных обыкновенных дифференциальных уравнений [Текст] / Ю.Е. Бояринцев. – Новосибирск: Наука, 1980. – 222 с.
31. Брахман, Т.Р. Многокритериальность и выбор альтернативы в технике [Текст] / Т.Р. Брахман. – Москва: Радио и связь, 1984. – 283 с.
32. Брегеда, С.Ю. Управление безопасностью дорожного движения на основе моделей регулирования транспортными потоками: автореф. дис. ... канд. техн. наук: 05.13.10 [Текст] / Брегеда Сергей Юрьевич. – Воронеж: 2010. – 18 с.
33. Васильев, Ф.П. Численные методы решения экстремальных задач [Текст] / Ф.П. Васильев. – Москва: Наука, 1980. – 253 с.
34. Вен, В.Л. Решение системы линейных неравенств по методу наименьших квадратов [Текст] / В.Л. Вен, П.И. Литвинцев, К.К. Фролов // Журнал вычислительной математики и математической физики. – 1984. – Т. 24, № 1. – С. 144-148.

35. Воеводин, В.В. Вычислительные основы линейной алгебры [Текст] / В.В. Воеводин. – Москва: Наука, 1977. – 304 с.
36. Владимиров, В.А. Инженерные основы организации дорожного движения [Текст] / В.А. Владимиров. – Москва: Стройиздат, 1975. – 454 с.
37. Воеводин, В.В. Линейная алгебра [Текст] / В.В. Воеводин. – Москва: Наука, 1980. – 452 с.
38. Воеводин, В.В. Матрицы и вычисления [Текст] / В.В. Воеводин, Ю.А. Кузнецов. – Москва: Наука, 1984. – 287 с.
39. Вошинин, А.П. Решение задачи векторной оптимизации на основе регрессионных моделей [Текст] / А.П. Вошинин, Нгуен-Ань-Чинь // Заводская лаборатория. – 1985. – № 7. – С. 26-57.
40. Гантмахер, Ф.Р. Теория матриц [Текст] / Ф.Р. Гантмахер. – Москва: Наука, 1967. – 576 с.
41. Ганжа, О.А. Особенности формирования шумового режима городских транспортных пересечений [Текст] / О.А. Ганжа, Э.С. Косицына // Интернет-вестник ВолгГАСУ. – Политематическая сер. – 2007. – Вып. 2(3).
42. Гилл, Ф. Практическая оптимизация [Текст] / Ф. Гилл, У. Мюррей, М. Райт. – Москва: Мир, 1985. – 509 с.
43. Глазман, И.М. Конечномерный линейный анализ в задачах [Текст] / И.М. Глазман, Ю.И. Любич. – Москва: Наука, 1969. – 476 с.
44. Голуб, Д.И. Метод статистической оптимизации циклов светофорного регулирования [Текст] / Д.И. Голуб. – Ставрополь: СевКавГТУ, 2007.
45. Грегори, Р. Безошибочные вычисления [Текст] / Р. Грегори, Е. Кришнамурти. – Москва: Мир, 1988. – 208 с.
46. Гуревич, Л.В. Управление движением на улицах и дорогах [Текст] / Л.В. Гуревич, П.В. Рушевский. – Москва: Транспорт, 1975. – 198 с.

47. Демиденко, Е.З. Вычислительные вопросы нелинейной регрессии [Текст] / Е.З. Демиденко // Заводская лаборатория. – 1986. – № 3. – С. 51-54.
48. Демиденко, Е.З. Линейная и нелинейная регрессии [Текст] / Е.З. Демиденко. – Москва: Финансы и статистика, 1981. – 302 с.
49. Демиденко, Е.З. Оптимизация и регрессия [Текст] / Е.З. Демиденко. – Москва: Наука, 1989. – 296 с.
50. Дрейпер, Н. Прикладной регрессионный анализ [Текст]: в 2-х кн. / Н. Дрейпер, Г. Смит. – Москва: Финансы и статистика, 1986.
51. Дрю, Д. Теория транспортных потоков и управление ими [Текст] / Д. Дрю. – Москва: Транспорт, 1972. – 424 с.
52. Дубровский, С.А. Прикладной многомерный статистический анализ [Текст] / с.А. Дубровский. – Москва: Финансы и статистика, 1982. – 215 с.
53. Дэннис, Дж. Численные методы безусловной оптимизации и решения нелинейных уравнений [Текст] / Дж. Дэннис, Р. Шнабель. – Москва: Мир, 1988. – 440 с.
54. Евтушенко, Ю.Г. К вопросу о систематизации численных методов нелинейного программирования [Текст] / Ю.Г. Евтушенко, В.Г. Жадан // Сообщения по прикладной математике. – Москва: ВЦ АН СССР, 1988. – 66 с.
55. Евтушенко, Ю.Г. Методы решения экстремальных задач и их применение в системах оптимизации [Текст] / Ю.Г. Евтушенко. – Москва: Наука, 1982. – 432 с.
56. Живоглядов, В.Г. Теория движения транспортных и пешеходных потоков [Текст] / В.Г. Живоглядов. – Ростов-на-Дону: Изд-во журнала «Известия вузов. Северо-Кавказский регион», 2005. – 1082 с.

57. Жилинскас, А.Г. Поиск оптимума [Текст] / А. Жилинскас, В.Р. Шалтянис. – Москва: Наука, 1989. – 128 с.
58. Зангвилл, У. Нелинейное программирование. Единый подход [Текст] / У. Зангвилл. – Москва: Советское радио, 1973. – 312 с.
59. Икрамов, Х.Д. Задачник по линейной алгебре [Текст] / Х.Д. Икрамов. – Москва: Наука, 1975. – 347 с.
60. Икрамов, Х.Д. О компьютерно-алгебраических процедурах для вдообращения матриц / Х.Д. Икрамов, М. Матин фар // Журнал вычислительной математики и математической физики. – 2003. – Т. 43, № 2. – С. 163-168.
61. Иносэ, Х. Управление дорожным движением [Текст] / Х. Иносэ, Т. Хамада. – Москва: Транспорт, 1983. – 248 с.
62. Интрилигатор, М. Математические методы оптимизации и экономическая теория [Текст] / М. Интрилигатор. – Москва: Прогресс, 1975. – 606 с.
63. Кадасев, Д.А. Повышение системной безопасности транспортных потоков оптимизацией светофорного регулирования их движения: автореф. дис. ... канд. техн. наук : 05.22.10 [Текст] / Кадасев Дмитрий Анатольевич. – Москва, 2008. – 18 с.
64. Калман, Р. Очерки по математической теории систем [Текст] / Р. Калман, П. Фалб, М. Арбиб. – Москва: Мир, 1971. – 400 с.
65. Клиновштейн, Г.И. Организация дорожного движения: учебник для вузов [Текст] / Г.И. Клиновштейн. – Москва: Транспорт, 2001. – 247 с.
66. Колдаев, В.Д. Численные методы и программирование: учебное пособие [Текст] / В.Д. Колдаев. – Москва: ИД «ФОРУМ», ИНФРА-М, 2008. – 336 с.

67. Корчагин, В.А. Повышение системной безопасности транспортных потоков: учеб. пос. [Текст] / В.А. Корчагин, Д.А. Кадасев. – Липецк: Изд-во ЛГТУ, 2010. – 159 с.
68. Кременец, Ю.А. Технические средства организации дорожного движения [Текст]: учеб. для вузов / Ю.А. Кременец, М.П. Печерский, М.Б. Афанасьев. – Москва: ИКЦ «Академкнига», 2005. – 279 с.
69. Крутько, П.Д. Обратные задачи динамики управляемых систем. Линейные модели [Текст] / П.Д. Крутько. – Москва: Наука, 1987. – 304 с.
70. Ланкастер, П. Теория матриц [Текст] / П. Ланкастер. – Москва: Наука, 1978. – 280 с.
71. Левашев, А.Г. Проектирование регулируемых пересечений: учеб. пособие [Текст] / А.Г. Левашев, А.Ю. Михайлов, И.М. Головных. – Иркутск: Изд-во ИрГТУ, 2007. – 208 с.
72. Липцер, Р.Ш. Статистика случайных процессов [Текст] / Р.Ш. Липцер, А.Н. Ширяев. – Москва: Наука, 1974. – 696 с.
73. Лоусон, Ч. Численное решение задач метода наименьших квадратов [Текст] / Ч. Лоунсон, Р. Хенсон. – Москва: Наука, 1986. – 232 с.
74. Львовский, Е.Н. Статистические методы построения эмпирических формул / Е.Н. Львовский. – Москва: Высшая школа, 1988. – 239 с.
75. Малашонок, Г.И. Решение системы линейных уравнений в целостном кольце [Текст] / Г.И. Малашонок // Журнал вычислительной математики и математической физики. – 1983. – Т. 23, № 6. – С. 1497-1500.
76. Математические методы исследования операций [Текст] / Под ред. Н.Н. Моисеева, П.С. Краснощёкова. – Москва: МГУ, 1981. – 192 с.
77. Метсон, Т.М. Организация движения (сокращённый перевод с англ.) [Текст] / Т.М. Метсон, У.С. Смит, Ф.В. Хард. – Москва: Автотранзисдат, 1960. – 464 с.

78. Миловидов, С.П. Псевдообращение матрицы условий транспортной задачи [Текст] / с.П. Миловидов. – Деп. в ВИЦИТИ 21.07.82. Ф 6027-82 Деп.
79. Минимизация в инженерных расчётах на ЭВМ [Текст] / С.Ю. Гуснин [и др.]. – Москва: Машиностроение, 1981. – 120 с.
80. Мину, М. Математическое программирование. Теория и алгоритмы [Текст] / М. Мину. – Москва: Наука, 1990. – 488 с.
81. Михалевич, В.С. Методы невыпуклой оптимизации [Текст] / В.С. Михалевич, Ф.М. Гупал, В.Н. Норкин. — Москва: Наука, 1987. – 280 с.
82. Морозов, В.А. Регулярные методы решения некорректно поставленных задач [Текст] / В.А. Морозов. – Москва: Наука, 1987. – 240 с.
83. Мостеллер, Ф. Анализ данных и регрессия [Текст]: в 2-х вып. / Ф. Мостеллер, Дж. Тьюки. – Москва: Финансы и статистика, 1982. – 317, 239 с.
84. Нестеров, Ю.Е. Эффективные методы в нелинейном программировании [Текст] / Ю.Е. Нестеров. – Москва: Радио и связь, 1989. – 304 с.
85. Ортега, Д. Итерационные методы решения нелинейных систем уравнений со многими неизвестными [Текст] / Д. Ортега, В. Рейнболт. – Москва: Мир, 1975. – 392 с.
86. Петунина, М.Ю. Исследование точности модели линейной регрессии при изменении числа параметров [Текст] / М.Ю. Петунина // Украинский математический журнал. – 1990. – № 2. – С. 266-271.
87. Планирование эксперимента в задачах нелинейного оценивания и распознавания образов [Текст] / Г.К. Круг [и др.]. – Москва: Наука, 1981. – 172 с.
88. Погодаев, А.К. Адаптивные методы в многокритериальных задачах оптимизации качества [Текст] / А.К. Погодаев // Современные слож-

ные системы управления CCCU/HTCS'2002: Сб. тр. Международной научно-технической конференции. – Липецк: ЛГТУ, 2002. – С. 199-201.

89. Погодаев, А.К. Адаптивные методы определения приоритетов показателей качества металлопродукции [Текст] / А.К. Погодаев // Известия вузов. Чёрная металлургия. – 2002. – № 7. – С. 51-53.
90. Погодаев, А.К. Адаптация и оптимизация в системах автоматизации и управления [Текст]: монография / А.К. Погодаев, С.Л. Блюмин. – Липецк: ЛЭГИ, 2003. – 128 с.
91. Погодаев, А.К. Развитие современных информационных технологий для повышения эффективности автоматизированных систем управления качеством: дис. ... д-ра. техн. наук: 05.13.06 [Текст] / Погодаев Анатолий Кириянович. Липецк: 2003. – 380 с.
92. Погодаев, А.К. Структурное моделирование производственных связей [Текст] / А.К. Погодаев, В.Д. Бурцев, А.В. Анненков // Теория и практика производства листового проката: Сб. научн. тр. Часть 2. – Липецк: ЛГТУ, 2003. – С. 76-81.
93. Погодаев, А.К. Адаптивная оптимизация качества [Текст] / А.К. Погодаев, С.Л. Блюмин, Д.А. Погодаев // Вести вузов Черноземья. – 2005. – № 1. – С. 40-44.
94. Погодаев, А.К. Система оптимизации распределения потребления тепловой энергии [Текст] / А.К. Погодаев, Е.Н. Кириллов, Д.А. Погодаев // Вести вузов Черноземья. – 2007. – № 2(8). – С. 67-70.
95. Погодаев, А.К. Псевдообращение и численный метод дифференцирования псевдообратных матриц в обучении нейросетевых моделей [Текст] / А.К. Погодаев, П.В. Сараев, М.Н. Тарнакина // Системы управления и информационные технологии. – 2011. – № 4.1(46). – С. 166-171.
96. Погодаев, А.К. Идентификация нейро-нечётких моделей для данных больших объемов [Текст] / А.К. Погодаев, П.В. Сараев // Вестник

Воронежского государственного технического университета. – 2015. – Т. 11, № 4. – С. 8-11.

97. Потёмкин, В.Г. Система инженерных и научных расчётов Matlab5.x [Текст]: В 2-х т. Т.1 / В.Г. Потёмкин. – Москва: Диалог-МИФИ, 1999. – 366 с.
98. Прикладная статистика. Классификация и снижение размерности [Текст] / С.А. Айвазян [и др.]. – Москва: Финансы и статистика, 1989. – 607 с.
99. Прикладная статистика. Основы моделирования и обработки данных [Текст] / С.А. Айвазян [и др.]. – Москва: Финансы и статистика, 1983. – 471 с.
100. Пшеничный, Б.Н. Численные методы в экстремальных задачах [Текст] / Б.Н. Пшеничный, Ю.М. Данилин. – Москва: Наука, 1975. – 320 с.
101. Пытьев, Ю.П. Математические методы интерпретации эксперимента [Текст] / Ю.П. Пытьев. – Москва: Высшая школа, 1989. – 351 с.
102. Пытьев, Ю.П. Псевдообратный оператор. Свойства и применения [Текст] / Ю.П. Пытьев // Математическое образование. – 1982. – Т. 118(160), № 1(5). – С. 19-49.
103. Рао, С.Р. Линейные статистические методы и их применения [Текст] / С.Р. Рао. – Москва: Наука, 1968. – 548 с.
104. Рейзлин, В.И. Численные методы оптимизации [Текст]: учеб. пос. / В.И. Рейзлин. – Томск: Изд-во Томского политехнического университета, 2011. – 105 с.
105. Реклейтис, Г. Оптимизация в технике [Текст]: т. 1-2 / Г. Реклейтис, А. Рейвиндран, К. Рэгсдейл. – Москва: Мир, 1986. – 295, 320 с.
106. Рейцен, Е.А. Оптимизация цикла работы светофора с использованием методов линейного программирования / Е.А. Рейцен, А.Г. Богданов //

Містобудування та територ. планування: наук.-тех. збірн. – Вип. 14. – Київ: КНУБА. – 2003. – С. 143-151.

107. Рубан, А.И. Синтез алгоритмов управления для дискретно-аргументных систем [Текст] / А.И. Рубан // Известия академии наук СССР. Техническая кибернетика. – 1987. – № 5. – С. 204.
108. Рубан, А.И. Идентификация одного класса стохастических нелинейных дискретных объектов [Текст] / А.И. Рубан // Автоматика и вычислительная техника. – 1973. – № 3. – С. 64-70.
109. Рубан, А.И. Идентификация нелинейных динамических объектов на основе алгоритма чувствительности: монография [Текст] / А.И. Рубан. – Томск: ТГУ, 1975. – 271 с.
110. Рубан, А.И. Глобальная оптимизация методом усреднения координат [Текст]: монография / А.И. Рубан. – Красноярск: ИПЦ КГТУ, 2004. – 302 с.
111. Себер, Д. Линейный регрессионный анализ [Текст] / Д. Себер. – Москва: Мир, 1980. – 456 с.
112. Сирл, С. Матричная алгебра в экономике [Текст] / С. Сирл, У. Госман. – Москва: Статистика, 1974. – 374 с.
113. Спивак, С.И. Нелинейные модели неполного ранга и нелинейные параметрические функции в обратных задачах химической кинетики [Текст] / С.И. Спивак, В.Г. Горский // Заводская лаборатория. – 1981. – № 10. – С. 39-47.
114. Судаков, Р.С. Теория псевдополубратных матриц и её применение к задачам оценки надёжности [Текст] / Р.С. Судаков. – Москва: Знание, 1981. – 106 с.
115. Сысоев, А.С. Численный алгоритм оптимизации функции транспортной задержки на регулируемом перекрестке [Текст] / А.С. Сысоев // Научно-технический вестник Поволжья. – 2012. – № 5. – С. 324-330.

116. Сысоев, А.С. Моделирование и оптимизация систем с переменной структурой методами идемпотентной математики и анализа конечных изменений: дис. ... канд. техн. наук: 05.13.18 [Текст] / Сысоев Антон Сергеевич. – Воронеж: 2013. – 129 с.
117. Сысоев, А.С. Идемпотентный подход к графоструктурному моделированию городских транспортных перекрёстков [Текст] / А.С. Сысоев // Научные технологии и инновации: сб. докл. Юбилею Международной науч.-практ. конф., посвященной 60-летию БГТУ им. В.Г. Шухова. – Белгород: Издательство БГТУ, 2014. – С. 67-73.
118. Сысоев, А.С. Моделирование и оптимизация класса систем массового обслуживания с переменной структурой [Текст] / А.С. Сысоев // Управление большими системами УБС-2014: мат. XI Всерос. школы-конф. молодых учёных. – Москва: Издательство института проблем управления им. В.А. Трапезникова РАН, 2014. – С. 1197-1211.
119. Тихонов, А.Н. Методы решения некорректных задач [Текст] / А.Н. Тихонов, В.Я. Арсенин. – Москва: Наука, 1987. – 288 с.
120. Турчак, Л.И. Основы численных методов [Текст]: учеб. пос. / Л.И. Турчак. – Москва: ФИЗМАТЛИТ, 2005. – 304 с.
121. Хачиян, Л.Г. О точном решении систем линейных неравенств и задач линейного программирования [Текст] / Л.Г. Хачиян // Журнал вычислительной математики и математической физики. – 1982. – Т. 22, № 4. – С. 999-1002.
122. Хейт, Ф. Математическая теория транспортных потоков [Текст] / Ф. Хейт. – Москва: Мир, 1966. – 286 с.
123. Химмельблау, Д. Прикладное нелинейное программирование [Текст] / Д. Химмельблау. – Москва: Мир, 1975. – 493 с.
124. Численные методы решения некорректных задач [Текст] / А.Н. Тихонов [и др.]. – Москва: Наука, 1990. – 232 с.

125. Численные методы решения сингулярных систем [Текст] / Ю.Е. Бояринцев [и др.]. – Новосибирск: Наука, 1989. – 223 с.
126. Швецов, В.И. Математическое моделирование транспортных потоков [Текст] / В.И. Швецов // Автоматика и телемеханика. – 2003. – № 11. – С. 3–46.
127. Шумский, В.М. О применении методов псевдообращения для решения плохообусловленных задач МНК [Текст] / В.М. Шумский, Т.Н. Шумская // Заводская лаборатория. – 1989. – № 1. – С. 81-86.
128. Юдин, Д.Б. Математические методы управления в условиях неполной информации [Текст] / Д.Б. Юдин. – Москва: Советское радио, 1974. – 387 с.
129. Ben-Israel, A. Generalized Inverses of Matrices: a Perspective of the Work Penrose [Text] / A. Ben-Israel // Mathematical Proceedings of the Cambridge Philosophical Society. – V. 100, № 3. – 1986. – P. 407-425.
130. Blyumin, S.L. Blockwise Recursive-Iterative Procedures for the Least-squares Solution of Non-linear Problems [Text] / S.L. Blyumin, A.K. Pogodayev // Computational Mathematics and Mathematical Physics. – 1992. – V. 32, № 8. – P. 1059-1064.
131. Blyumin, S.L. Superpositional Regression [Text] / S.L. Blyumin, A.K. Pogodayev // Computational Mathematics and Mathematical Physics. – 1995. – V. 35, № 10. – P. 1269-1273.
132. Blyumin, S.L. Investigation and solution of a pair of linear matrix equations [Text] / S.L. Blyumin, S.P. Milovidov // Mathematical Notes. – 1995. – V. 57, № 2. P. 211-213.
133. Blyumin, S.L. Recursive iterative algorithms for adaptive identification of nonlinear concentrated dynamic systems [Text] / S.L. Blyumin, A.K. Pogodaev // Automation and Remote Control. – 2003. – № 10. – P. 1583-1588.

134. Blyumin, S.L. Multicriteria optimization, regularization, pseudoinversion and multiresponse regression [Text] / S.L. Blyumin, P.V. Saraev // Modern informatization problems: proceedings of the XIX-th International Open Science Conference. – 2014. – P. 32-34.
135. Brent, R.P. Algorithms for Minimization without Derivatives [Text] / R.P. Brent. – Prentice-Hall, Englewood Cliffs, New Jersey, 1973. — 195 p.
136. Campbell, S. Singular System of Differential Equations [Text] / S. Campbell. – London: Pitman, 1980. – 245 p.
137. Canadian capacity guide for signalized intersections. – District 7, Canada: Institute of Transportation Engineers, 2008.
138. El Tohami, M. On the Desing of observers for Generalized State Space Systems Using Singular value Decomposition [Text] / M. El Tohami, V. Lorass-Nagy, R. Mukundan // International Journal of Control. – 1983. – V. 38, № 3. – P. 673-683.
139. Hallum, C. Computational Aspect of Matrix Generalized Inversion for the Computer with Applications [Текст] / C. Hallum, H. Pore // Computers and Mathematics with Applications. – 1975. – V. 1. – P. 145-150.
140. Highway Capacity Manual. Volume 3: Interrupted Flow. – Transportation Research Board of the National Academies. Washington, DC. – 2010.
141. Koecher, M. The Generalized Inverses of Integral Matrices [Text] / M. Koecher // Linear Algebra and Applications. – 1985. – V. 71. – P. 187-198.
142. Moore, E. On the Reciprocal of the General Algebraic Matrix [Text] / E. Moore // Bulletin of the American Mathematical Society. – 1920. – V. 26. – P. 394-395.
143. Penrose, R. A Generalized Inverse of Matrices [Text] / R. Penrose // Proc. Camb. Phil. Soc. – 1955. – V. 51. – PP. 406-413.

144. Pogodaev, A. Neurostructural modelling and prediction of hot-rolled production defects by casting parameters [Text] / A. Pogodaev, P. Saraev // Journal of Chemical Technology and Metallurgy. – 2015. – V. 50, № 6. – P. 595-599.
145. R package «corpcor» [Электронный ресурс]. – Режим доступа: <https://cran.r-project.org/web/packages/corpcor/corpcor.pdf> (дата обращения 01.09.2015).
146. Rao, C.R. Generalized Inverses of Matrices and its Applications [Text] / C.R. Rao, S.K. Mitra. – New York: Wiley, 1971. – 499 p.
147. Springer, J. Verallgemeinerte Inversen ganzzahliger Matrizen // Zeitschrift für angewandte Mathematik und Mechanik. – 1987. – V. 67, № 10. – P. 503-506.
148. Sysoev, A.S. Synthesis of parallel algorithm for multidimensional optimization for traffic delay function [Text] / A.S. Sysoev // Computer Science and Information Technologies (CSIT2013): Proceedings of the Workshop. – V. 2. – Vienna-Budapest-Bratislava: Ufa State Aviation Technical University. – 2013. – P. 88-91.
149. Torn, A. A sampling-search-clustering approach to global optimization [Text] / A.A. Torn // Towards Global Optimization 2. – Amsterdam: North Holland, 1978. – P. 49-62.
150. Zielke, G.A. A survey of Generalized Matrix Inverses [Text] / G.A. Zielke // Computational Mathematics Banach Center Publications. – 1984. – V. 13. – P. 499-526.

Погодаев Анатолий Кириянович

Блюмин Семён Львович

Миловидов Сергей Петрович

Сысоев Антон Сергеевич

**ОПТИМИЗАЦИЯ
ПСЕВДООБРАЩЕНИЕ
ИТЕРАЦИИ И РЕКУРСИИ**

Учебное пособие

Редактор Черникова Е.Н.

Подписано в печать . Формат 60×84 1/16. Бумага офсетная.

Ризография. Объём 12,25 п.л. Тираж 100 экз. Заказ № .

Издательство Липецкого государственного технического университета.

Полиграфическое подразделение Издательства ЛГТУ.

398600 Липецк, ул. Московская, 30.