ГЛАВА VII

ПОИСК МИНИМУМА

В главе VII рассмотрены способы нахождения такого значения аргумента, которое минимизирует некоторую зависящую от него скалярную величину. В § 1 изложена задача о минимуме функции одного переменного, лежащая в основе всех более сложных задач. В § 2 рассмотрена задача о минимуме функции многих переменных в неограниченной области. В § 3 область изменения переменных ограничена; наряду с общим случаем рассмотрена частная задача линейного программирования, важная в приложениях к экономике. В § 4 разобрана задача о минимизации функционала, когда аргумент сам является функцией одного или нескольких переменных.

§ 1. Минимум функции одного переменного

1. Постановка задачи. Пусть имеется некоторое множество X, состоящее из элементов x, принадлежащих какому-нибудь метрическому пространству, и на нем определена скалярная функция $\Phi(x)$. Говорят, что $\Phi(x)$ *имеет локальный минимум на элементе* $\bar{x}$, если существует некоторая конечная ε-окрестность этого элемента, в которой выполняется

$$\Phi(\bar{x}) < \Phi(x), \quad \|x - \bar{x}\| \leqslant \varepsilon. \tag{1}$$

У функции может быть много локальных минимумов. Если же выполняется

$$\Phi(\bar{x}) = \inf_X \Phi(x), \tag{2}$$

то говорят о достижении функцией *абсолютного минимума на данном множестве* X.

Естественно требовать, чтобы функция $\Phi(x)$ была непрерывной или, по крайней мере, кусочно-непрерывной, а множество X было компактно*) и замкнуто**) (в частности, если X само является

*) Множество компактно, если из каждого бесконечного и ограниченного его подмножества можно выделить сходящуюся последовательность.

**) Множество замкнуто, если предел любой сходящейся последовательности его элементов принадлежит этому множеству.

пространством, то это пространство должно быть банаховым). Если эти требования не соблюдены, то вряд ли возможно построить разумный алгоритм нахождения решения. Например, если $\Phi(x)$ не является кусочно-непрерывной, то единственным способом решения задачи является перебор всех элементов x, на которых задана функция; этот способ нельзя считать приемлемым. Чем более жестким требованиям удовлетворяет $\Phi(x)$ (таким, как существование непрерывных производных различного порядка), тем легче построить хорошие численные алгоритмы.

Перечислим наиболее важные примеры множеств, на которых приходится решать задачу нахождения минимума. Если множество X является числовой осью, то (1) или (2) есть задача на минимум функции одного вещественного переменного. Если X есть n-мерное векторное пространство, то мы имеем дело с задачей на минимум функции n переменных. Если X есть пространство функций $x(t)$, то (1) называют задачей на минимум функционала.

Для нахождения абсолютного минимума есть только один способ: найти все локальные минимумы, сравнить их и выбрать наименьшее значение. Поэтому задача (2) сводится к задаче (1), и мы будем в основном заниматься задачей поиска локальных минимумов.

Известно, что решение задачи (1) удовлетворяет уравнению

$$\frac{\delta\Phi}{\delta x} = 0. \tag{3}$$

Если множество X есть числовая ось, то написанная здесь производная является обычной производной, и тогда уравнение (3) есть просто одно (нелинейное) уравнение с одним неизвестным. Для n-мерного векторного пространства соотношение (3) оказывается системой нелинейных уравнений $\partial\Phi/\partial x_i = 0$, $1 \leqslant i \leqslant n$. Для пространства функций уравнение (3) является дифференциальным или интегро-дифференциальным. В принципе такие уравнения можно решать численными методами, описанными в главах V и XIV. Однако эти уравнения нередко имеют сложный вид, так что итерационные методы их решения могут очень плохо сходиться или вообще не сходиться. Поэтому в данной главе мы рассмотрим численные методы, применимые непосредственно к задаче (1), без приведения ее к форме (3).

Пусть X является некоторым множеством, принадлежащим какому-то пространству. Тогда (1) называют задачей на минимум в ограниченной области. В частности, если множество X выделено из пространства с помощью ограничивающих условий типа равенств, то задачу (1) называют задачей на условный экстремум; такие задачи методом неопределенных множителей Лагранжа часто можно свести к задачам на безусловный экстремум. Однако при

численном решении обычно удобнее иметь дело непосредственно с исходной задачей (1), хотя при ее решении в ограниченной области возникают свои трудности.

Функция $\Phi(x)$ может иметь на множестве X более одного локального минимума. В конкретных прикладных задачах далеко не всегда удается заранее исследовать свойства функции. Поэтому желательно, чтобы численный алгоритм позволял определить число минимумов и их расположение и аккуратно найти абсолютный минимум.

Задачу называют *детерминированной*, если погрешностью вычисления (или экспериментального определения) функции $\Phi(x)$ можно пренебречь. В противном случае задачу называют *стохастической*. Мы будем рассматривать в основном детерминированные задачи. Для решения стохастических задач есть специальные методы, но они очень медленные, и применять их к детерминированным задачам невыгодно.

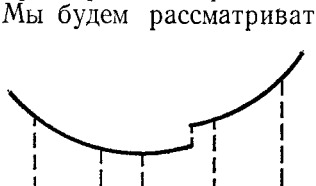

Рис. 36.

2. Золотое сечение. В этом параграфе мы рассмотрим задачу нахождения минимума функции одной действительной переменной. Эта одномерная задача нередко возникает в практических приложениях. Кроме того, большинство методов решения многомерных задач сводится к поиску одномерного минимума.

Сейчас мы рассмотрим метод золотого сечения, применимый к недифференцируемым функциям. Будем считать, что $\Phi(x)$ задана и кусочно-непрерывна на отрезке $a \leqslant x \leqslant b$, и имеет на этом отрезке (включая его концы) только один локальный минимум. Построим итерационный процесс, сходящийся к этому минимуму.

Вычислим функцию на концах отрезка, а также в двух внутренних точках x_1, x_2, сравним все четыре значения функции между собой и выберем среди них наименьшее. Пусть наименьшим оказалось $\Phi(x_1)$. Очевидно, минимум расположен в одном из прилегающих к нему отрезков (рис. 36). Поэтому отрезок $[x_2, b]$ можно отбросить и оставить отрезок $[a, x_2]$. Первый шаг процесса сделан.

На отрезке $[a, x_2]$ снова надо выбрать две внутренние точки, вычислить в них и на концах отрезка значения функции, и сделать следующий шаг процесса. Но на предыдущем шаге вычислений мы уже нашли $\Phi(x)$ на концах нового отрезка a, x_2 и в одной его внутренней точке x_1. Поэтому достаточно выбрать внутри $[a, x_2]$ еще одну точку x_3, определить в ней значение функции и провести необходимые сравнения. Это вчетверо уменьшает объем вычислений на одном шаге процесса.

Как выгодно размещать точки? Всякий раз мы делим оставшийся отрезок на три части (причем одна из точек деления уже определена предыдущими вычислениями) и затем отбрасываем один из крайних отрезков. Очевидно, надо, чтобы следующий отрезок был поделен подобно предыдущему. Для этого должны выполняться соотношения

$$b-x_2=x_1-a, \qquad \frac{x_1-a}{b-a}=\frac{x_2-x_1}{x_2-a}.$$

Решение этих уравнений дает

$$\frac{b-x_2}{b-a}=\frac{x_1-a}{b-a}=\xi, \quad \xi=\frac{2}{3+\sqrt{5}}\approx 0{,}38. \tag{4}$$

После проведения очередного вычисления отрезок сокращается в $1-\xi\approx 0{,}62$ раза; после n вычислений функции он составляет $(1-\xi)^{n-3}$ долю первоначальной величины (три первых вычисления в точках a, b, x_1 еще не сокращают отрезок). Следовательно, при $n\to\infty$ длина оставшегося отрезка стремится к нулю как геометрическая прогрессия со знаменателем $1-\xi\approx 0{,}62$, т. е. метод золотого сечения всегда сходится, причем линейно.

Запишем алгоритм вычисления. Для единообразия записи обозначим

$$a=x_0, \quad b=x_1,$$

а поочередно вводимые внутренние точки будут x_2, x_3, ... На первом шаге полагаем согласно (4)

$$x_2=x_0+\xi(x_1-x_0), \quad x_3=x_1-\xi(x_1-x_0). \tag{5}$$

После сравнения может быть отброшена точка с любым номером, так что на следующих шагах оставшиеся точки будут перенумерованы беспорядочно. Пусть на данном отрезке есть четыре точки x_i, x_j, x_k, x_l, из которых какие-то две являются концами отрезка. Выберем ту точку, в которой функция принимает наименьшее значение; пусть это оказалось x_i:

$$\Phi(x_i)<\Phi(x_j), \quad \Phi(x_k), \quad \Phi(x_l). \tag{6}$$

Затем отбрасываем ту точку, которая более всего удалена*) от x_i; пусть этой точкой оказалась x_l:

$$|x_l-x_i|>|x_j-x_l|, \ |x_k-x_l|. \tag{7}$$

Определим порядок расположения оставшихся трех точек на числовой оси; пусть, для определенности,

$$x_k<x_i<x_j. \tag{8}$$

*) Это верно не при всяких делениях отрезка, но для деления в соответствии (4) это справедливо.

Тогда новую внутреннюю точку введем таким соотношением *):

$$x = x_j + x_k - x_i, \tag{9}$$

и присвоим ей очередной номер. Минимум находится где-то внутри последнего отрезка, $x_k \leqslant \bar{x} \leqslant x_j$. Поэтому итерации прекращаем, когда длина этого отрезка станет меньше заданной погрешности δ:

$$x_j - x_k < \delta. \tag{10}$$

Метод золотого сечения является наиболее экономичным аналогом метода дихотомий применительно к задачам на минимум. Он применим даже к недифференцируемым функциям и всегда сходится; сходимость его линейна. Если на отрезке $[a, b]$ функция имеет несколько локальных минимумов, то процесс сойдется к одному из них (но не обязательно к наименьшему).

Этот метод нередко применяют в технических или экономических задачах оптимизации, когда минимизируемая функция недифференцируема, а каждое вычисление функции — это дорогой эксперимент.

Метод золотого сечения рассчитан на детерминированные задачи. В стохастических задачах из-за ошибок эксперимента можно неправильно определить соотношения между значениями функций в точках; тогда дальнейшие итерации пойдут по ложному пути. Поэтому если различия функций в выбранных точках стали того же порядка, что и ошибки эксперимента, то итерации надо прекращать. Поскольку вблизи минимума чаще всего $\delta\Phi \sim (\delta x)^2$, то небольшая погрешность функции приводит к появлению довольно большой области неопределенности $\delta x \sim \sqrt{\delta\Phi}$.

3. Метод парабол. Метод золотого сечения надежный, но медленный. Если $\Phi(x)$ дифференцируема, то можно построить гораздо более быстрые методы, основанные на решении уравнения $\Phi'(x) = 0$. Напомним, что корень $\bar{x}$ этого уравнения является точкой минимума, если $\Phi''(\bar{x}) > 0$, и точкой максимума при $\Phi''(\bar{x}) < 0$.

На практике часто $\Phi(x)$ имеет и первую производную и вторую. Поэтому для нахождения нулей первой производной применяют метод линеаризации, что приводит к такому итерационному процессу:

$$x_{s+1} = x_s - \frac{\Phi'(x_s)}{\Phi''(x_s)}; \tag{11}$$

в простейших задачах нулевое приближение можно выбрать графически. Формулу (11) можно получить несколько иным способом. Разложим $\Phi(x)$ в точке x_s по формуле Тейлора, ограничившись

*) См. предыдущую сноску.

тремя членами, т. е. аппроксимируем кривую параболой

$$\Phi(x) \approx \Phi(x_s) + (x - x_s)\Phi'(x_s) + \frac{1}{2}(x - x_s)^2 \Phi''(x_s);$$

минимум этой параболы достигается в точке, определяемой формулой (11). Итерационный процесс (11) является ньютоновским; вблизи простого корня уравнения $\Phi'(x) = 0$, т. е. вблизи экстремума с ненулевой второй производной, он сходится квадратично. Если же $\Phi''(\bar{x}) = 0$, то сходимость в достаточно малой окрестности экстремума есть, но она более медленная — линейная.

Обычно для первой и тем более второй производной получаются очень громоздкие выражения. Поэтому выгоднее заменить их конечно-разностными аппроксимациями. Наиболее часто берут симметричные разности (3.6)—(3.7) с постоянным шагом, что приводит к формуле

$$x_{s+1} = x_s - \frac{h}{2} \frac{\Phi(x_s + h) - \Phi(x_s - h)}{\Phi(x_s + h) - 2\Phi(x_s) + \Phi(x_s - h)}. \qquad (12)$$

Это эквивалентно замене кривой на интерполяционную параболу, построенную по трем точкам $x_s - h$, x_s, $x_s + h$. Обычно выбирают вспомогательный шаг $h \approx 0{,}1 - 0{,}01$ при ручных расчетах с небольшим числом знаков и $h \approx 0{,}01 - 0{,}001$ при расчетах на ЭВМ; тогда характер сходимости вблизи экстремума вплоть до расстояний $\sim h^2$ практически не отличается от квадратичного. Формула (12) наиболее часто употребляется в практических расчетах.

Этот способ кажется неэкономным, ибо на каждой итерации надо вычислять три значения функции. Построение параболы по трем последовательным итерациям, как это делалось в методе парабол при нахождении корней многочлена, дает

$$2x_{s+1} = x_s + x_{s-1} - \frac{\Phi(x_s,\ x_{s-1})}{\Phi(x_s,\ x_{s-1},\ x_{s-2})} \qquad (13)$$

и требует только одного вычисления функции за итерацию. Однако ранее уже отмечалось, что такая замена производных разделенными разностями уменьшает скорость сходимости. Можно показать, используя описанную в главе V, § 2, п. 7 технику, что вблизи невырожденного минимума

$$|x_{s+1} - \bar{x}| \approx \left| \frac{\Phi'''(\bar{x})}{6\Phi''(\bar{x})} \right|^{0{,}325} |x_s - \bar{x}|^{1{,}325}. \qquad (14)$$

Во-первых, отсюда видно, что $|x_{s+1} - \bar{x}| < |x_s - \bar{x}|$, только если выполнено условие $|x_s - \bar{x}| < |6\Phi''/\Phi'''|$; это приблизительно показывает размеры окрестности корня, в которой итерации сходятся. Эта окрестность может быть небольшой, если $\Phi'''(\bar{x})$ велика.

Во-вторых, асимптотическая скорость сходимости определяется показателем степени при $|x_s - \bar{x}|$ в правой части соотношения (14). Этот показатель невелик; поэтому сходимость настолько медленна, что три итерации по этой формуле только немного сильней уменьшают погрешность, чем одна итерация по формуле (12). А поскольку формула (13) недостаточно испытана на практике, то нет уверенности, что она окажется лучше.

Заметим, что во всех вариантах метода парабол для успешной работы необходимы «кухонные» поправки к алгоритму. В ходе вычислений надо проверять, движемся ли мы к минимуму: вторая разность, стоящая в знаменателе формулы (12), или вторая производная в знаменателе формулы (11) должна быть положительной. Если она отрицательна, то итерации сходятся к максимуму, и надо сделать какой-то шаг в обратном направлении, причем достаточно большой.

Вычислив новое приближение, надо обязательно проверить, уменьшилась ли функция. Если оказалось, что

$$\Phi(x_{s+1}) > \Phi(x_s),$$

то значение x_{s+1} нельзя использовать и надо просто сделать от точки x_s какой-то шаг в сторону убывания функции. Обычно делают шаг величиной $\tau(x_{s+1} - x_s)$ с $\tau = 1/2$ и проверяют условие убывания функции; если оно снова не выполнено, то уменьшают τ вдвое и делают шаг опять из точки x_s, и так до тех пор, пока не добьются убывания функции.

Фактическая скорость работы программы очень сильно зависит от того, насколько тщательно обдуманы эти поправки к алгоритму.

Если функция имеет несколько локальных минимумов, то итерационный метод может сойтись к любому из них. Удалять найденные минимумы можно только в том случае, когда мы располагаем явным выражением для $\Phi'(x)$ и решаем не исходную задачу (1), а уравнение $\Phi'(x) = 0$; тогда удаляют уже найденные корни этого уравнения при помощи техники, описанной в главе V.

Если так сделать не удается, то выбирают несколько начальных приближений в разных участках отрезка $[a, b]$, и из каждого начального приближения проводят какой-нибудь итерационный процесс поиска минимума. Некоторые из этих итерационных процессов могут сходиться к одному и тому же локальному минимуму, а некоторые — к другим. Остается сравнить найденные локальные минимумы между собой и выбрать наименьший (если это требуется по условиям задачи).

Описанный способ не дает гарантии того, что будут найдены все минимумы (и тем самым, что будет найден абсолютный минимум). Но для недостаточно изученной функции такой гарантии не дают никакие способы.

4. Стохастические задачи. Опишем один алгоритм, рассчитанный на стохастические задачи. Он основан на предположении, что ошибки определения функции $\Phi(x)$ имеют статистическую природу, т. е. они целиком случайны, а систематической погрешности нет. Тогда можно определить минимум со сколь угодно высокой точностью (фактически игнорируя область неопределенности $\delta x \sim \sqrt{\delta\Phi}$), если воспользоваться таким итерационным

процессом:

$$x_{n+1} = x_n - \frac{a_n}{b_n}[\Phi(x_n + b_n) - \Phi(x_n - b_n)], \tag{15}$$

где a_n, b_n — последовательности положительных чисел, удовлетворяющие следующим условиям:

$$a_n,\ b_n \underset{n\to\infty}{\to} 0, \quad \sum_{n=1}^{\infty} a_n = \infty, \quad \sum_{n=1}^{\infty} \left(\frac{a_n}{b_n}\right)^2 < \infty. \tag{16}$$

При выполнении этих условий $x_n \to \bar{x}$ с вероятностью единица при $n \to \infty$ (напомним, что стремление с вероятностью единица означает «почти всегда стремится», а не «обязательно стремится»). Условиям (16) удовлетворяют, например, $a_n = 1/n$ и $b_n = n^{-1/3}$.

Этот алгоритм является обобщением алгоритма Роббинса — Монро, описанного в главе V, на задачи поиска минимума. Он сходится весьма медленно, ибо изменение аргумента за шаг равно $|x_{n+1} - x_n| \approx 2a_n |\Phi'(x_n)|$, а величины a_n убывают очень медленно, как видно из второго условия (16). Поэтому применять этот алгоритм к детерминированным задачам невыгодно.

§ 2. Минимум функции многих переменных

1. Рельеф функции. Основные трудности многомерного случая удобно рассмотреть на примере функции двух переменных $\Phi(x, y)$. Она описывает некоторую поверхность в трехмерном пространстве с координатами x, y, Φ. Задача $\Phi(x, y) = \min$ означает поиск низшей точки этой поверхности.

Как в топографии, изобразим рельеф этой поверхности линиями уровня. Проведем равноотстоящие плоскости $\Phi = \text{const}$ и найдем линии их пересечения с поверхностью $\Phi(x, y)$; проекции этих линий на плоскость x, y называют линиями уровня. Направление убывания функции будем указывать штрихами, рисуемыми около линий уровня. Полученная картина напоминает топографическое изображение рельефа горизонталями. По виду линий уровня условно выделим три типа рельефа: котловинный, овражный и неупорядоченный.

При *котловинном* рельефе линии уровня похожи на эллипсы (рис. 37, *а*). В малой окрестности невырожденного минимума рельеф функции котловинный. В самом деле, точка минимума гладкой функции определяется необходимыми условиями

$$\frac{\partial \Phi}{\partial x} = \frac{\partial \Phi}{\partial y} = 0, \tag{17}$$

и разложение функции по формуле Тейлора вблизи минимума

имеет вид

$$\Phi(x, y) = \Phi(\bar{x}, \bar{y}) + \frac{1}{2}(\Delta x)^2 \Phi_{xx} + \\ + \Delta x \Delta y \Phi_{xy} + \frac{1}{2}(\Delta y)^2 \Phi_{yy} + \dots, \qquad (18)$$

причем квадратичная форма (18) — положительно определенная *), иначе эта точка не была бы невырожденным минимумом. А линии уровня знакоопределенной квадратичной формы — это эллипсы.

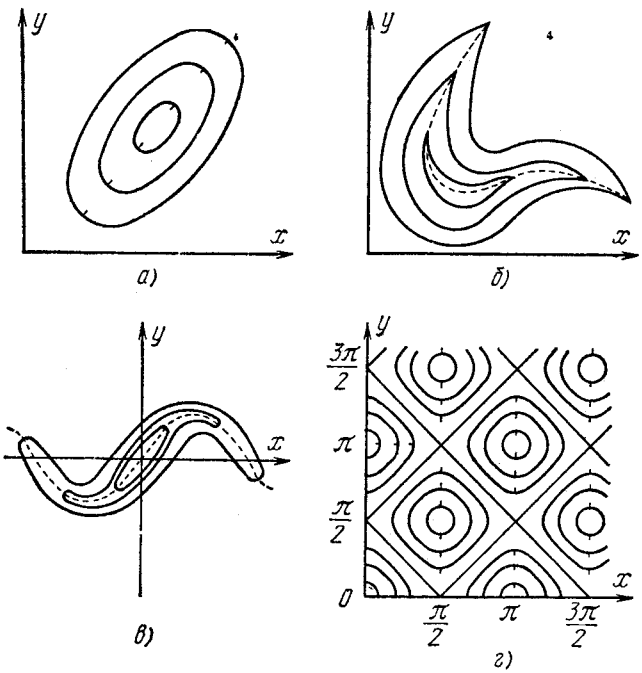

Рис. 37.

Случай, когда все вторые производные равны в этой точке нулю и минимум определяется более высокими производными, по существу ничего нового не дает, и мы не будем его специально рассматривать (линии уровня вместо эллипсов будут похожими на них кривыми четвертого порядка).

Отметим, что условию (17) удовлетворяют также точки максимумов и седловые точки. Но в точках максимумов квадратичная

*) Квадратичная форма $\sum_{i,\ k} a_{ik} z_i z_k$ называется положительно определенной, если при любых z_i (за исключением обращающихся одновременно в нуль) она положительна.

форма (18) отрицательно определенная, а в седловинах она знакопеременна.

Вблизи минимума функция мало меняется при заметных изменениях переменных. Поэтому даже если мы не очень точно определим те значения переменных, которые должны минимизировать функцию, то само значение функции при этом обычно будет мало отличаться от минимального.

Рассмотрим *овражный* тип рельефа. Если линии уровня кусочно-гладкие, то выделим на каждой из них точку излома. Геометрическое место точек излома назовем *истинным оврагом*, если угол направлен в сторону возрастания функции, и *гребнем* — если в сторону убывания (рис. 37, *б*). Чаще линии уровня всюду гладкие, но на них имеются участки с большой кривизной; геометрические места точек с наибольшей кривизной назовем *разрешимыми* оврагами или гребнями (рис. 37, *в*). Например, рельеф функции

$$\Phi(x,\ y) = 10\,(y - \sin x)^2 + 0{,}1\,x^2, \tag{19}$$

изображенный на этом рисунке, имеет ярко выраженный извилистый разрешимый овраг, «дно» которого — синусоида, а низшая точка — начало координат.

В физических задачах овражный рельеф указывает на то, что вычислитель не учел какую-то закономерность, имеющую вид связи между переменными. Обнаружение и явный учет этой закономерности облегчают решение математической задачи. Так, если в примере (19) ввести новые переменные $\xi = x$, $\eta = y - \sin x$, то рельеф становится котловинным.

Неупорядоченный тип рельефа (рис. 37, *г*) характеризуется наличием многих максимумов, минимумов и седловин. Примером может служить функция

$$\Phi(x,\ y) = (1 + \sin^2 x)\,(1 + \sin^2 y), \tag{20}$$

рельеф которой изображен на этом рисунке; она имеет минимумы в точках с координатами $\bar{x}_k = \pi k$, $\bar{y}_l = \pi l$ и максимумы в точках, сдвинутых относительно минимумов на $\pi/2$ по каждой координате.

Все эффективные методы поиска минимума сводятся к построению траекторий, вдоль которых функция убывает; разные методы отличаются способами построения таких траекторий. Метод, приспособленный к одному типу рельефа, может оказаться плохим на рельефе другого типа.

2. Спуск по координатам. Казалось бы, для нахождения минимума достаточно решить систему уравнений типа (17) методом линеаризации или простых итераций и отбросить те решения, которые являются седловинами или максимумами. Однако в реальных задачах минимизации эти методы обычно сходятся в настолько малой окрестности минимума, что выбрать подходящее

нулевое приближение далеко не всегда удается. Проще и эффективнее провести спуск по координатам. Изложим этот метод на примере функции трех переменных $\Phi(x, y, z)$.

Выберем нулевое приближение x_0, y_0, z_0. Фиксируем значения двух координат $y=y_0$, $z=z_0$. Тогда функция будет зависеть только от одной переменной x; обозначим ее через $f_1(x) = \Phi(x, y_0, z_0)$. Используя описанные в § 1 методы, найдем минимум функции одной переменной $f_1(x)$ и обозначим его через x_1. Мы сделали шаг из точки (x_0, y_0, z_0) в точку (x_1, y_0, z_0) по направлению, параллельному оси x; на этом шаге значение функции уменьшилось.

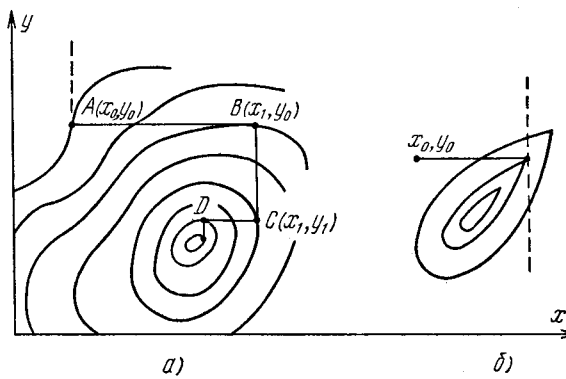

Рис. 38.

Затем из новой точки сделаем спуск по направлению, параллельному оси y, т. е. рассмотрим $f_2(y) = \Phi(x_1, y, z_0)$, найдем ее минимум и обозначим его через y_1. Второй шаг приводит нас в точку (x_1, y_1, z_0). Из этой точки делаем третий шаг — спуск параллельно оси z и находим минимум функции $f_3(z) = \Phi(x_1, y_1, z)$. Приход в точку (x_1, y_1, z_1) завершает *цикл спусков*.

Будем повторять циклы. На каждом спуске функция не возрастает, и при этом значения функции ограничены снизу ее значением в минимуме $\overline{\Phi} = \Phi(\bar{x}, \bar{y}, \bar{z})$. Следовательно, итерации сходятся к некоторому пределу $\tilde{\Phi} \geqslant \overline{\Phi}$. Будет ли здесь иметь место равенство, т. е. сойдутся ли спуски к минимуму и как быстро?

Это зависит от функции и выбора нулевого приближения. На примере функции двух переменных легко убедиться, что существуют случаи сходимости спуска по координатам к искомому минимуму и случаи, когда этот спуск к минимуму не сходится.

В самом деле, рассмотрим геометрическую трактовку спуска по координатам (рис. 38). Будем двигаться по выбранному направлению, т. е. по некоторой прямой в плоскости x, y.

В тех участках, где прямая пересекает линии уровня, мы при движении переходим от одной линии уровня к другой, так что при этом движении функция меняется (возрастает или убывает, в зависимости от направления движения). Только в той точке, где данная прямая касается линии уровня (рис. 38, *а*), функция имеет экстремум вдоль этого направления. Найдя такую точку, мы завершаем в ней спуск по первому направлению, и должны начать спуск по второму направлению (поскольку направления мы сейчас выбираем параллельно координатным осям, то второе направление перпендикулярно первому).

Пусть линии уровня образуют истинный овраг. Тогда возможен случай (рис. 38, *б*), когда спуск по одной координате приводит нас на «дно» оврага, а любое движение по следующей координате (пунктирная линия) ведет нас на подъем. Никакой дальнейший спуск по координатам невозможен, хотя минимум еще не достигнут; процесс спуска по координатам в данном случае не сходится к минимуму.

Наоборот, если функция достаточно гладкая, то в некоторой окрестности минимума процесс спуска по координатам сходится к этому минимуму. Пусть функция имеет непрерывные вторые производные, а ее минимум не вырожден. Для простоты опять рассмотрим функцию двух переменных $\Phi(x, y)$. Выберем некоторое нулевое приближение x_0, y_0 и проведем линию уровня через эту точку. Пусть в области G, ограниченной этой линией уровня, выполняются неравенства, означающие положительную определенность квадратичной формы (18):

$$\Phi_{xx} \geqslant a > 0,\ \Phi_{yy} \geqslant b > 0,\ |\Phi_{xy}| \leqslant c,\ ab > c^2. \qquad (21)$$

Докажем, что тогда спуск по координатам из данного нулевого приближения сходится к минимуму, причем линейно.

Значения функции вдоль траектории спуска не возрастают; поэтому траектория не может выйти из области G, и неравенства (21) будут выполняться на всех шагах. Рассмотрим один из циклов, начинающийся в точке A (рис. 38, *а*). Предыдущий цикл окончился поиском минимума по направлению y, следовательно, $(\Phi_y)_A = 0$ и $|\Phi_x|_A = \xi_1 \neq 0$. Первый шаг нового цикла спускает нас по направлению x в точку B, в которой $\Phi_x = 0$ и $|\Phi_y| = \eta \neq 0$. Поскольку вторые производные непрерывны, можно применить теорему о среднем; получим

$$\xi_1 = |(\Phi_x)_A - (\Phi_x)_B| = |\Phi_{xx}|\,\rho_{AB} \geqslant a\,\rho_{AB},$$
$$\eta = |(\Phi_y)_A - (\Phi_y)_B| = |\Phi_{xy}|\,\rho_{AB} \leqslant c\,\rho_{AB},$$

где через ρ обозначены расстояния между точками. Отсюда получаем $c\xi_1 \geqslant a\eta$. Выполним второй шаг цикла — спуск по направлению y в точку C, после которого $(\Phi_y)_C = 0$ и $|\Phi_x|_C = \xi_2$.

Аналогичные рассуждения дают соотношение $c\eta \geqslant b\xi_2$. Объединяя эти неравенства, найдем

$$\xi_2 \leqslant q\xi_1, \quad q = \frac{c^2}{ab}, \quad 0 < q < 1.$$

Следовательно, за один цикл Φ_x уменьшается в q раз: то же справедливо для Φ_y, если рассмотреть цикл, сдвинутый на один шаг, т. е. начинающийся в точке B и кончающийся в точке D.

Значит, когда число циклов $n \to \infty$, то все первые производные линейно стремятся к нулю:

$$|\Phi_x|_n \leqslant q^n |\Phi_x|_0 \to 0 \text{ и } |\Phi_y|_n \sim q^n \to 0.$$

Первые производные одновременно обращаются в нуль в точке минимума и вблизи него являются линейными однородными функциями приращений координат. Поэтому координаты точек спуска линейно стремятся к координатам точки минимума, т. е. в данном случае спуск по координатам сходится, причем линейно.

Случай (21) заведомо реализуется в достаточно малой окрестности невырожденного минимума, ибо эти условия эквивалентны требованию положительной определенности квадратичной формы (18). Таким образом, вблизи невырожденного минимума достаточно гладкой функции спуск по координатам линейно сходится к минимуму. В частности, для квадратичной функции этот метод сходится при любом нулевом приближении.

Фактическая скорость сходимости будет неплохой при малых q, когда линии уровня близки к эллипсам, оси которых параллельны осям координат. Для эллипсов, сильно вытянутых под значительным углом к осям координат, величина $q \approx 1$ и сходимость очень медленная.

Если сходимость медленная, но траектория уже попала в близкую окрестность минимума, то итерации можно уточнять процессом Эйткена; разумеется, при этом надо брать в качестве исходных значения не на трех последних спусках, а на трех *циклах* спусков (т. е. не точки A, B, C, а точки B, D и третья точка, которой нет на рис. 38, *а*).

Разрешимый овраг напоминает сильно вытянутую котловину (см. рис. 38, *б*). При попадании траектории спуска в такой овраг сходимость становится настолько медленной, что расчет практически невозможно вести. Отметим, что в стохастических задачах наличие ошибок эквивалентно превращению истинных оврагов и гребней в разрешимые; расчет при этом можно продолжать, хотя практическая ценность такого расчета невелика: сходимость очень медленная.

Метод спуска по координатам несложен и легко программируется на ЭВМ. Но сходится он медленно, а при наличии оврагов — очень плохо. Поэтому его используют в качестве первой попытки при нахождении минимума.

Пример. Рассмотрим квадратичную функцию $\Phi(x, y) = x^2+y^2+xy$ и выберем нулевое приближение $x_0=1$, $y_0=2$. Выполняя вычисления, получим

$$x_1=-1,\ y_1={}^1/_2;\ x_2=-{}^1/_4,\ y_2={}^1/_8;\ x_3=-{}^1/_{16},\ y_3={}^1/_{32}.$$

Уточнение по Эйткену дает $\tilde{x}=\tilde{y}=0$, т. е. точное положение минимума (заметим, что делать уточнение с использованием нулевого приближения нельзя; читателям предлагается объяснить, почему).

3. **Наискорейший спуск.** Спускаться можно не только параллельно осям координат. Вдоль любой прямой $\boldsymbol{r}=\boldsymbol{r}_0+\boldsymbol{a}t$ функция зависит только от одной переменной, $\Phi(\boldsymbol{r}_0+\boldsymbol{a}t)=\varphi(t)$, и минимум на этой прямой находится описанными в § 1 методами.

Наиболее известным является метод наискорейшего спуска, когда выбирается $\boldsymbol{a}=-(\operatorname{grad}\Phi)_{\boldsymbol{r}=\boldsymbol{r}_0}$, т. е. направление, в котором функция быстрее всего убывает при бесконечно малом движении из данной точки. Спуск по этому направлению до минимума определяет новое приближение $\boldsymbol{r}_1$. В этой точке снова определяется градиент и делается следующий спуск.

Однако этот метод значительно сложнее спуска по координатам, ибо требуется вычислять производные и градиент (это нередко делают конечно-разностными методами) и переходить к другим переменным. К тому же, по сходимости наискорейший спуск не лучше спуска по координатам. При попадании траектории в истинный овраг спуск прекращается, а в разрешимом овраге сильно замедляется.

Если функция является положительно определенной квадратичной функцией

$$\Phi(\boldsymbol{r})=(\boldsymbol{r},\ A\boldsymbol{r})+(\boldsymbol{b},\ \boldsymbol{r})+c, \tag{22}$$

то формулы наискорейшего спуска приобретают несложный вид. Вдоль прямой $\boldsymbol{r}=\boldsymbol{r}_n+\boldsymbol{a}t$ функция (22) квадратично зависит от параметра t:

$$\varphi(t)\equiv\Phi(\boldsymbol{r}_n+\boldsymbol{a}t)=\Phi(\boldsymbol{r}_n)+(2A\boldsymbol{r}_n+\boldsymbol{b},\ \boldsymbol{a})\,t+(\boldsymbol{a},\ A\boldsymbol{a})\,t^2. \tag{23}$$

Из уравнения $(d\varphi/dt)=0$ легко находим ее минимум

$$t=-(2A\boldsymbol{r}_n+\boldsymbol{b},\ \boldsymbol{a})/2\,(\boldsymbol{a},\ A\boldsymbol{a}), \tag{24}$$

дающий нам следующую точку спуска:

$$\begin{gathered}\boldsymbol{r}_{n+1}=\boldsymbol{r}_n+\boldsymbol{a}t,\\ \Phi(\boldsymbol{r}_{n+1})=\Phi(\boldsymbol{r}_n)-\frac{(2A\boldsymbol{r}_n+\boldsymbol{b},\ \boldsymbol{a})^2}{4\,(\boldsymbol{a},\ A\boldsymbol{a})}.\end{gathered} \tag{25}$$

Направление наискорейшего спуска определяется градиентом квадратичной функции (22):

$$\boldsymbol{a}=-(\operatorname{grad}\Phi)_{\boldsymbol{r}_n}=-(2A\boldsymbol{r}_n+\boldsymbol{b}). \tag{26}$$

Подставляя это значение в формулы (24) — (25), получим окончательные выражения для вычисления последовательных спусков.

Если воспользоваться разложением всех движений по базису, состоящему из собственных векторов матрицы A, то можно доказать, что для квадратичной функции метод наискорейшего спуска линейно сходится, причем

$$|r_{n+1}-\bar{r}|\leqslant q|r_n-\bar{r}|, \text{ где } q=\frac{\lambda_{\max}-\lambda_{\min}}{\sqrt{\lambda_{\max}^2+\lambda_{\min}^2}}<1; \qquad (27)$$

здесь λ — собственные значения положительно определенной матрицы A (они вещественны и положительны). Если $\lambda_{\min}\ll\lambda_{\max}$, что соответствует сильно вытянутым эллипсам — линиям уровня, то $q\approx 1$ и сходимость может быть очень медленной. Есть такие начальные приближения (рис. 39), когда точно реализуется наихудшая возможная оценка, т. е. в (27) имеет место равенство.

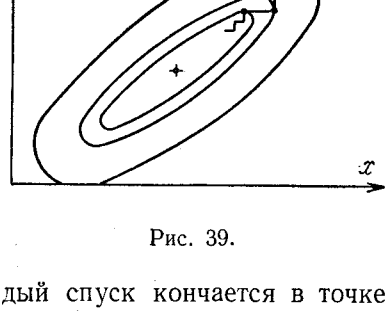

Рис. 39.

Причины нетрудно понять. Во-первых, в данной точке любую прямую, в том числе невыгодную для спуска, можно сделать направлением градиента, если специально подобрать изменение масштабов по осям. Во-вторых, каждый спуск кончается в точке, где его направление касается линии (поверхности) уровня. Градиент перпендикулярен поверхности уровня. Следовательно, в методе наискорейшего спуска каждый спуск перпендикулярен предыдущему. В двумерном случае это означает, что мы совершаем спуск по координатам, повернутым так, что одна ось параллельна градиенту в начальной точке.

Для улучшения метода наискорейшего спуска предлагают «кухонные» поправки к алгоритму — например, совершают по каждому направлению спуск не точно до минимума. Наиболее любопытным представляется такое видоизменение алгоритма. Будем делать по направлению, противоположному градиенту, только бесконечно малый шаг и после него вновь уточнять направление спуска. Это приводит к движению по кривой $r(t)$, являющейся решением системы обыкновенных дифференциальных уравнений:

$$\frac{dr}{dt}=-\operatorname{grad}\Phi(r(t)). \qquad (28)$$

Вдоль этой кривой $d\Phi/dt=(d\Phi/dr)(dr/dt)=-(\operatorname{grad}\Phi)^2<0$, т. е. функция убывает, и мы движемся к минимуму при $t\to+\infty$.

Уравнение (28) моделирует безынерционное движение материальной точки вниз по линии градиента. Можно построить и другие уравнения — например, дифференциальное уравнение второго порядка, моделирующее движение точки при наличии вязкого трения.

Однако от идеи метода еще далеко до надежного алгоритма. Фактически систему дифференциальных уравнений (28) надо численно интегрировать (см. главу VIII). Если интегрировать с большим шагом, то численное решение будет заметно отклоняться от линии градиента. А при интегрировании малым шагом сильно возрастает объем расчетов. Кроме того, если рельеф имеет извилистые овраги, то трудно ожидать хорошей сходимости этого метода.

Алгоритмы наискорейшего спуска и всех его видоизменений сейчас недостаточно отработаны. Поэтому метод наискорейшего спуска для сложных нелинейных задач с большим числом переменных ($m \gtrsim 5$) редко применяется, но в частных случаях он может оказаться полезным.

4. Метод оврагов. Рассмотрим задачу $\Phi(r) = \min$. Выберем произвольно точку ρ_0 и спустимся из нее (например, по координатам), делая не очень много шагов, т. е. не требуя высокой точности сходимости. Конечную точку спуска обозначим r_0. Если рельеф овражный, эта точка окажется вблизи дна оврага (рис. 40).

Теперь выберем другую точку ρ_1 не слишком далеко от первой. Из нее также сделаем спуск и попадем в некоторую точку r_1. Эта точка тоже лежит вблизи дна оврага. Проведем через точки r_0 и r_1 на дне оврага прямую — приблизительную линию дна оврага, передвинемся по этой линии в сторону убывания функции и выберем новую точку

$$\rho_2 = r_1 \pm (r_1 - r_0)\, h, \qquad h = \text{const} > 0. \tag{29}$$

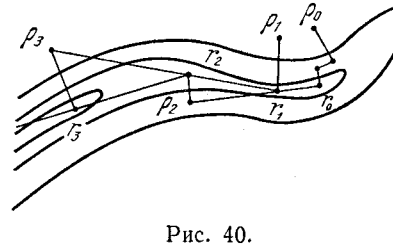

Рис. 40.

В формуле (29) выбирается плюс, если $\Phi(r_1) < \Phi(r_0)$, и минус в обратном случае, так что движение направлено в сторону понижения дна оврага. Величина h называется *овражным шагом* и для каждой функции подбирается в ходе расчета.

Дно оврага не является отрезком прямой, поэтому точка ρ_2 на самом деле лежит не на дне оврага, а на склоне. Из этой точки снова спустимся на дно и попадем в некоторую точку r_2. Затем соединим точки r_1 и r_2 прямой, наметим новую линию дна оврага и сделаем новый шаг по оврагу. Продолжим процесс до тех пор, пока значения функции на дне оврага, т. е. в точках r_n, убывают. В случае, когда

$$\Phi(r_{n+1}) > \Phi(r_n),$$

процесс надо прекратить и значение r_{n+1} не использовать.

Метод оврагов рассчитан на то, чтобы пройти вдоль оврага и выйти в котловину около минимума. В этой котловине значения минимума лучше уточнять другими методами.

Методом оврагов удается находить минимумы достаточно сложных функций от 5—10 переменных. Но этот метод довольно капризен. Для каждой функции приходится подбирать свой овражный шаг, визуально наблюдать за ходом расчета и вносить коррективы. Программирование этого метода на ЭВМ несложно.

5. Сопряженные направления. Методы наискорейшего спуска или спуска по координатам даже для квадратичной функции требуют бесконечного числа итераций. Однако можно построить такие направления спуска, что для квадратичной функции

$$\Phi(\boldsymbol{r}) = (\boldsymbol{r},\ A\boldsymbol{r}) + (\boldsymbol{b},\ \boldsymbol{r}) + c \tag{30}$$

(где $\boldsymbol{r}$ есть n-мерный вектор) с симметричной положительно определенной матрицей A процесс спуска сойдется точно к минимуму за конечное число шагов.

Положительно определенная матрица позволяет ввести норму вектора следующим образом:

$$\|\boldsymbol{x}\|^2 = (\boldsymbol{x},\ A\boldsymbol{x}) > 0 \quad \text{при } \boldsymbol{x} \neq 0. \tag{31}$$

Нетрудно проверить, что все аксиомы нормы при этом выполнены. Определение (31) означает, что под скалярным произведением двух векторов $\boldsymbol{x}$ и $\boldsymbol{y}$ теперь подразумевается величина $(\boldsymbol{x},\ A\boldsymbol{y})$. Векторы, ортогональные в смысле этого скалярного произведения

$$(\boldsymbol{x},\ A\boldsymbol{y}) = 0, \tag{32}$$

называют *сопряженными* (по отношению к данной матрице A). Ниже мы увидим, что поочередный спуск по сопряженным направлениям особенно выгоден при поиске минимума.

На этом основана большая группа методов: сопряженных градиентов, сопряженных направлений, параллельных касательных и другие. Для квадратичной функции они применяются с одинаковым успехом. На произвольные функции наиболее хорошо обобщается метод *сопряженных направлений*, у которого детали алгоритма тщательно отработаны; этот метод излагается в данном пункте.

а) Сначала рассмотрим, как применяется этот метод к квадратичной форме (30). Для этого нам потребуются некоторые свойства сопряженных векторов. Пусть имеется некоторая система попарно сопряженных векторов $\boldsymbol{x}_i$. Нормируем каждый из этих векторов в смысле нормы (31); тогда соотношения между ними примут вид

$$(\boldsymbol{x}_i,\ A\boldsymbol{x}_j) = \delta_{ij}. \tag{33}$$

Докажем, что взаимно сопряженные векторы линейно-независимы. Из равенства $\boldsymbol{x}_1 = \sum_{i=2} \alpha_i \boldsymbol{x}_i$ следует $(\boldsymbol{x}_1,\ A\boldsymbol{x}_1) = \sum_{i=2} \alpha_i (\boldsymbol{x}_1,\ A\boldsymbol{x}_i) = 0,$

что противоречит положительной определенности матрицы. Это противоречие доказывает наше утверждение. Значит, система n сопряженных векторов является базисом в n-мерном пространстве. Для данной матрицы имеется бесчисленное множество базисов, состоящих из взаимно сопряженных векторов.

Пусть мы нашли некоторый сопряженный базис $\boldsymbol{x}_i$, $1 \leqslant i \leqslant n$. Выберем произвольную точку $\boldsymbol{r}_0$. Любое движение из этой точки можно разложить по сопряженному базису

$$\boldsymbol{r} = \boldsymbol{r}_0 + \sum_{i=1}^{n} \alpha_i \boldsymbol{x}_i. \tag{34}$$

Подставляя это выражение в правую часть формулы (30), преобразуем ее с учетом сопряженности базиса (33) к следующему виду:

$$\Phi(\boldsymbol{r}) = \Phi(\boldsymbol{r}_0) + \sum_{i=1}^{n} [\alpha_i^2 + 2\alpha_i (\boldsymbol{x}_i,\ A\boldsymbol{r}_0) + \alpha_i (\boldsymbol{x}_i,\ \boldsymbol{b})]. \tag{35}$$

Последняя сумма состоит из членов, каждый из которых соответствует только одной компоненте суммы (34). Это означает, что движение по одному из сопряженных направлений $\boldsymbol{x}_i$ меняет только один член суммы (35), не затрагивая остальных.

Совершим из точки $\boldsymbol{r}_0$ поочередные спуски до минимума по каждому из сопряженных направлений $\boldsymbol{x}_i$. Каждый спуск минимизирует свой член суммы (35), так что *минимум квадратичной функции точно достигается после выполнения одного цикла спусков*, то есть за конечное число действий.

Поясним геометрический смысл сопряженного базиса. Если осями координат сделать главные оси эллипсоидов уровня квадратичной функции, то один цикл спусков по этим координатам приводит точно в минимум. Если перейти к некоторым аффинным координатам, то функция останется квадратичной, но коэффициенты квадратичной формы изменятся. Можно формально рассмотреть нашу квадратичную функцию с измененными коэффициентами как некоторую новую квадратичную форму в декартовых координатах и найти главные оси ее эллипсоидов. Положение этих главных осей в исходных аффинных координатах будет некоторой системой сопряженных направлений. Разный выбор аффинных координат естественно приводит к разным сопряженным базисам.

б) Сопряженный базис можно построить способом *параллельных касательных плоскостей*.

Пусть некоторая прямая параллельна вектору $\boldsymbol{x}$, а квадратичная функция достигает на этой прямой минимального значения в точке $\boldsymbol{r}_0$. Подставим уравнение этой прямой $\boldsymbol{r} = \boldsymbol{r}_0 + \alpha \boldsymbol{x}$ в выражение (30) и потребуем выполнения условия минимума

функции $\varphi(\alpha) \equiv \Phi(\boldsymbol{r}_0 + \alpha \boldsymbol{x})$ в точке $\boldsymbol{r} = \boldsymbol{r}_0$, т. е. при $\alpha = 0$. Для этого воспользуемся выражением (35), где в сумме оставим только один член:

$$\varphi(\alpha) = \Phi(\boldsymbol{r}_0) + \alpha^2 + \alpha(\boldsymbol{x},\ 2A\boldsymbol{r}_0 + \boldsymbol{b}),$$

и положим $(d\varphi/d\alpha)_{\alpha=0} = 0$. Отсюда следует уравнение, которому удовлетворяет точка минимума:

$$(\boldsymbol{x},\ 2A\boldsymbol{r}_0 + \boldsymbol{b}) = 0. \tag{36}$$

Пусть на какой-нибудь другой прямой, параллельной первой, функция принимает минимальное значение в точке $\boldsymbol{r}_1$; тогда аналогично найдем $(\boldsymbol{x},\ 2A\boldsymbol{r}_1 + \boldsymbol{b}) = 0$. Вычитая это равенство из (36), получим

$$(\boldsymbol{x},\ A(\boldsymbol{r}_1 - \boldsymbol{r}_0)) = 0. \tag{37}$$

Следовательно, *направление, соединяющее точки минимума на двух параллельных прямых, сопряжено направлению этих прямых.*

Таким образом, всегда можно построить вектор, сопряженный произвольному заданному вектору $\boldsymbol{x}$. Для этого достаточно провести две прямые, параллельные $\boldsymbol{x}$, и найти на каждой прямой минимум квадратичной формы (30). Вектор $\boldsymbol{r}_1 - \boldsymbol{r}_0$, соединяющий эти минимумы, сопряжен $\boldsymbol{x}$. Заметим, что прямая касается линии уровня в той точке, где функция на данной прямой принимает минимальное значение; с этим связано название способа.

Пусть имеются две параллельные m-мерные плоскости, порожденные системой сопряженных векторов $\boldsymbol{x}_i$, $1 \leqslant i \leqslant m < n$. Пусть квадратичная функция достигает своего минимального значения на этих плоскостях соответственно в точках $\boldsymbol{r}_0$ и $\boldsymbol{r}_1$. Аналогичными рассуждениями можно доказать, что вектор $\boldsymbol{r}_1 - \boldsymbol{r}_0$, соединяющий точки минимума, сопряжен всем векторам $\boldsymbol{x}_i$. Следовательно, если задана неполная система сопряженных векторов $\boldsymbol{x}_i$, то этим способом всегда можно построить вектор $\boldsymbol{r}_1 - \boldsymbol{r}_0$, сопряженный всем векторам этой системы.

Рассмотрим один цикл процесса построения сопряженного базиса. Пусть уже построен базис, в котором последние m векторов взаимно сопряжены, а первые $n - m$ векторов не сопряжены последним. Найдем минимум квадратичной функции (30) в какой-нибудь m-мерной плоскости, порожденной последними m векторами базиса. Поскольку эти векторы взаимно сопряжены, то для этого достаточно произвольно выбрать точку $\boldsymbol{r}_0$ и сделать из нее спуск поочередно по каждому из этих направлений (до минимума!). Точку минимума в этой плоскости обозначим через $\boldsymbol{r}_1$.

Теперь из точки $\boldsymbol{r}_1$ сделаем поочередный спуск по первым $n - m$ векторам базиса. Этот спуск выведет траекторию из первой плоскости и приведет ее в некоторую точку $\boldsymbol{r}_2$. Из точки $\boldsymbol{r}_2$

снова совершим по последним m направлениям спуск, который приведет в точку r_3. Этот спуск означает точное нахождение минимума во второй плоскости, параллельной первой плоскости. Следовательно, направление $r_3 - r_1$ сопряжено последним m векторам базиса.

Если одно из несопряженных направлений в базисе заменить направлением $r_3 - r_1$, то в новом базисе уже $m+1$ направление будет взаимно сопряжено.

Начнем расчет циклов с произвольного базиса; для него можно считать, что $m=1$. Описанный процесс за один цикл увеличивает на единицу число сопряженных векторов в базисе. Значит, за $n-1$ цикл все векторы базиса станут сопряженными, и следующий цикл приведет траекторию в точку минимума квадратичной функции (30).

в) Хотя понятие сопряженного базиса определено только для квадратичной функции, описанный выше процесс построен так, что его можно формально применять для произвольной функции. Разумеется, что при этом находить минимум вдоль направления надо методом парабол, не используя нигде формул, связанных с конкретным видом квадратичной функции (30).

В малой окрестности минимума приращение достаточно гладкой функции обычно представимо в виде симметричной положительно определенной квадратичной формы типа (18). Если бы это представление было точным, то метод сопряженных направлений сходился бы за конечное число шагов. Но представление приближенно, поэтому число шагов будет бесконечным; зато сходимость этого метода вблизи минимума будет квадратичной.

Благодаря квадратичной сходимости метод сопряженных направлений позволяет находить минимум с высокой точностью. Методы с линейной сходимостью обычно определяют экстремальные значения координат менее точно.

Замечание 1. Реально даже для квадратичной функции процесс не всегда укладывается в n циклов. Построение сопряженного базиса означает ортогонализацию в метрике, порожденной матрицей A. Ранее отмечалось, что в процессе ортогонализации теряется точность; при большом числе переменных погрешность настолько возрастает, что процесс приходится повторять.

Замечание 2. Теоретически безразлично, какое из несопряженных направлений выкинуть из базиса в конце цикла. Обычно выкидывают то направление, при спуске по которому на данном цикле функция изменилась менее всего. Поскольку для произвольной функции понятие сопряженности ввести нельзя, то направление наиболее слабого убывания выкидывают независимо от того, под каким номером оно стоит в базисе. Любопытно, что это оказывается выгодным даже для квадратичной функции, хотя

на основании этого критерия иногда можно выкинуть сопряженное направление, оставив несопряженные; зато уменьшается потеря точности при ортогонализации.

Замечание 3. Описанный выше цикл метода включает два спуска по сопряженным направлениям и один — по несопряженным. Более выгоден цикл, при котором сразу после нахождения нового сопряженного направления по нему делают спуск из точки r_3, приходя в некоторую точку r_4. Тогда спуск из r_2 в r_4 будет спуском в плоскости всех новых сопряженных направлений, т. е. его можно считать первой группой нового цикла спусков. Поэтому из точки r_4 сразу можно спускаться по несопряженным направлениям.

При этом новое направление ставят в базис на последнее место и выкидывают то направление, на котором функция слабее всего уменьшилась при спусках от точки r_1 до точки r_4. Наименее выгодным может оказаться и новое направление; тогда следующий цикл спусков будет сделан со старым базисом.

Метод сопряженных направлений является, по-видимому, наиболее эффективным методом спуска. Он неплохо работает и при вырожденном минимуме, и при разрешимых оврагах, и при наличии слабо наклонных участков рельефа — «плато», и при большом числе переменных — до двух десятков.

6. Случайный поиск. Методы спуска неполноценны на неупорядоченном рельефе. Если локальных экстремумов много, то спуск из одного нулевого приближения может сойтись только к одному из локальных минимумов, не обязательно абсолютному. Тогда для исследования задачи применяют случайный поиск.

Предполагают, что интересующий нас минимум (или все минимумы) лежит в некоторой замкнутой области; линейным преобразованием координат помещают ее внутрь единичного n-мерного куба. Выбирают в этом кубе N случайных точек способами, описанными в § 4 главы IV; если о расположении экстремумов заранее ничего не известно, то наилучшие результаты дают $ЛП_\tau$-последовательности точек.

Даже при миллионе пробных точек вероятность того, что хотя бы одна точка попадет в небольшую окрестность локального минимума, ничтожно мала. В самом деле, пусть диаметр котловины около минимума составляет 10% от пределов изменения каждой координаты. Тогда объем этой котловины составляет $0{,}1^n$ часть объема n-мерного куба. Уже при $n > 6$ ни одна точка в котловину не попадет.

Поэтому берут небольшое число точек $N \approx (5-20)\,n$ и каждую точку рассматривают как нулевое приближение. Из каждой точки совершают спуск, быстро попадая в ближайший овраг или котловину; когда шаги спуска сильно укорачиваются, его прекращают, не добиваясь высокой точности. Этого уже достаточно,

чтобы судить о величине функции в ближайшем локальном минимуме с удовлетворительной точностью.

Сравнивая (визуально или при помощи программы) окончательные значения функции на всех спусках между собой, можно изучить расположение локальных минимумов функции и сопоставить их величины. После этого можно отобрать нужные по смыслу задачи минимумы и провести в них дополнительные спуски для получения координат точек минимума с высокой точностью.

Обычно в прикладных задачах нужно в первую очередь добиться того, чтобы исследуемая функция приняла минимальное или почти минимальное значение. Но вблизи минимума значение функции слабо зависит от изменения координат. Зачем тогда нужно находить координаты точки минимума с высокой точностью? Оказывается, что это имеет не только теоретический, но и практический смысл.

Пусть, например, координаты — это размеры деталей механической конструкции, а минимизируемая функция есть мера качества конструкции. Если мы нашли минимум точно, то мы находимся в самом центре котловины около минимума. В этом случае вариации координат влияют на функцию слабее, чем в точках, расположенных ближе к краям котловины. А безопасные вариации координат имеют в данном примере смысл допусков на точность обработки деталей. Значит, при аккуратном вычислении координат минимума мы можем разрешить бо́льшие допуски, т. е. удешевить обработку деталей.

Метод случайного поиска зачастую позволяет найти все локальные минимумы функции от $10-20$ переменных со сложным рельефом. Он полезен и при исследовании функции с единственным минимумом; в этом случае можно обойтись заметно меньшим числом случайных точек. Недостаток метода в том, что надо заранее задать область, в которой выбираются случайные точки. Если мы зададим слишком широкую область, то ее труднее детально исследовать, а если выберем слишком узкую область, то многие локальные минимумы могут оказаться вне ее. Правда, положение несколько облегчается тем, что при спусках траектории могут выйти за пределы заданной области и сойтись к лежащим вне этой области минимумам.

§ 3. Минимум в ограниченной области

1. Формулировка задачи. Пусть в n-мерном векторном пространстве задана скалярная функция $\Phi(\boldsymbol{x})$. Рассмотрим задачу на минимум с дополнительными условиями двух типов:

$$\Phi(\boldsymbol{x}) = \min, \quad \varphi_i(\boldsymbol{x}) = 0, \quad 1 \leqslant i \leqslant m,$$
$$\psi_j(\boldsymbol{x}) \geqslant 0, \quad 1 \leqslant j \leqslant p. \tag{38}$$

Условия типа равенств выделяют в пространстве некоторую $(n-m)$-мерную поверхность; поэтому должно выполняться неравенство $m < n$. Условия типа неравенств выделяют n-мерную область, ограниченную гиперповерхностями $\psi_j(\boldsymbol{x}) = 0$; число таких условий

может быть произвольным. Следовательно, задача (38) есть поиск минимума функции n переменных в $(n-m)$-мерной области G.

Функция может достигать минимального значения как внутри области, так и на ее границе. Эта задача и особенно последний случай трудны для расчета. Вид дополнительных условий в любой реальной задаче не слишком прост, так что явно ввести в области G собственную $(n-m)$-мерную систему координат практически никогда не удается. Значит, при численном расчете мы вынуждены вести спуск не на $(n-m)$-мерной поверхности, а во всем n-мерном пространстве. Тогда даже если нулевое приближение лежит в области G, естественная траектория спуска сразу выходит из этой области; особенно сложно «заставить» траекторию идти вдоль границы области.

В математических задачах экономики поиск минимума при дополнительных условиях называют (в зависимости от типа функций) линейным, нелинейным и т. д. программированием.

2. Метод штрафных функций. Рассмотрим задачу на абсолютный минимум во всем n-мерном пространстве для такой вспомогательной функции:

$$F(\boldsymbol{x}) \equiv \Phi(\boldsymbol{x}) + \\ + \mu\left\{\sum_{i=1}^{m} \varphi_i^2(\boldsymbol{x}) + \sum_{j=1}^{p} \psi_j^2(\boldsymbol{x})\,[1 - \operatorname{sign} \psi_j(\boldsymbol{x})]\right\} = \min, \quad \mu > 0. \qquad (39)$$

Прибавляемые к $\Phi(\boldsymbol{x})$ члены взяты таким образом, что они обращаются в нуль, если дополнительные условия в (38) выполнены. Если же условия нарушены, то эти члены положительны, т. е. они увеличивают $F(\boldsymbol{x})$, причем тем больше, чем сильнее нарушены дополнительные условия. Это своеобразный штраф за нарушение условий.

Если коэффициент штрафа μ достаточно велик, то за границами области G функция $F(\boldsymbol{x})$ быстро возрастает. Значит, минимум $F(\boldsymbol{x})$ расположен или внутри области G, или снаружи вблизи ее границы. Если он лежит в области G, то он совпадает с минимумом $\Phi(\boldsymbol{x})$, ибо там дополнительные члены в условии (39) обращаются в нуль. Если же минимум $F(\boldsymbol{x})$ лежит снаружи, то минимум $\bar{\boldsymbol{x}}$ исходной функции лежит на границе; при разумных предположениях о свойствах функций $\Phi(\boldsymbol{x})$, $\varphi_i(\boldsymbol{x})$ и $\psi_j(\boldsymbol{x})$ доказано, что его отличие от минимума $\bar{\boldsymbol{x}}_\mu$ вспомогательной функции не превышает

$$|\bar{\boldsymbol{x}} - \bar{\boldsymbol{x}}_\mu| \leqslant \frac{\text{const}}{\mu}, \qquad (40)$$

где величина константы зависит от конкретных свойств функций (38). Поэтому если взять последовательность $\mu_k \to \infty$ и найти для нее минимумы $\bar{\boldsymbol{x}}_k$ вспомогательной функции $F(\boldsymbol{x};\ \mu_k)$, то $\bar{\boldsymbol{x}}_k \to \bar{\boldsymbol{x}}$.

Задачу (39) на безусловный экстремум удобнее всего решать методом случайного поиска со спуском по сопряженным направлениям: здесь естественно задана область, где надо выбирать случайные точки.

При малых значениях μ согласно оценке (40) точность может быть плохой. Но при большом μ благодаря дополнительным членам в (39) вблизи границы области появляются глубокие овраги и крутые откосы, так что методы спуска сходятся медленно. Полезен следующий прием, заметно ускоряющий сходимость.

Сначала берут небольшое μ_1 и легко находят соответствующий минимум $\bar{x}_1$. Затем берут большее значение μ_2, а значение $\bar{x}_1$ используют в качестве начального приближения для спуска; поэтому спуск будет не длинный, и новый минимум $\bar{x}_2$ определится быстро. Эту процедуру повторяют до тех пор, пока «штраф» — фигурная скобка в (39) — не станет достаточно малым. Тогда можно считать, что точка $\bar{x}_k$ близка к границе области G и хорошо аппроксимирует минимум $\bar{x}$.

Метод штрафных функций медленный и не слишком надежный. Он применим только при небольшом числе переменных $n \lesssim 10$. Но существенно более хороших методов для общей нелинейной задачи (38) пока нет. Перспективным кажется метод штрафных оценок, являющийся комбинацией описанного метода и метода неопределенных множителей Лагранжа; однако он еще мало изучен.

3. Линейное программирование. При оптимизации экономических планов возникают задачи на минимум линейной функции n переменных при наличии линейных дополнительных условий трех типов:

$$L(x) \equiv \sum_{i=1}^{n} c_i x_i = \min, \tag{41а}$$

$$x_i \geqslant 0, \quad 1 \leqslant i \leqslant n, \tag{41б}$$

$$\sum_{i=1}^{n} a_{ji} x_i = b_j, \quad 1 \leqslant j \leqslant m, \tag{41в}$$

$$\sum_{i=1}^{n} a_{ji} x_i \leqslant b_j, \quad m < j \leqslant M. \tag{41г}$$

Каждое из условий типа неравенств (41б) или (41г) определяет полупространство, ограниченное гиперплоскостью; все эти условия вместе определяют выпуклый n-мерный многогранник J, являющийся пересечением соответствующих полупространств. С математической точки зрения условия (41б) и (41г) однотипны; но по традиции их записывают указанным образом. Условия типа равенств (41в) выделяют из n-мерного пространства $(n-m)$-мерную

плоскость. Ее пересечение с областью J дает *выпуклый* $(n-m)$-*мерный многогранник* G; наша задача состоит в том, чтобы найти минимум линейной функции (41а) в этом многограннике G.

Примером такой задачи является распределение производства однотипной продукции по разным заводам. Пусть x_i — выпускаемое i-м заводом количество продукции (оно должно быть неотрицательным), c_i — себестоимостью одного изделия на этом заводе, a_{ji} при $j>m$ — расход сырья j-го вида и a_{ji} при $2\leqslant j\leqslant m$ — расход заработной платы и других аналогичных показателей j-го вида при выпуске единицы продукции на данном заводе. Положим $a_{1i}=1$; тогда b_1 будет суммарным выпуском продукции по всем заводам, b_j, $2\leqslant j\leqslant m$, — полной заработной платой и аналогичными данными по всей отрасли, суммы (41г) — расходом сырья по всем заводам, а L — себестоимостью общей продукции. Требуется, чтобы себестоимость продукции была минимальной, выпуск продукции, расход заработной платы и т. д. — заданными, а фонды сырья b_j, $m<j$, не перерасходовались. Нас интересует, как распределить неотрицательные плановые задания x_i по заводам так, чтобы удовлетворить всем этим требованиям.

Отметим терминологию, установившуюся в экономике. Вектор $\boldsymbol{x}$, удовлетворяющий всем дополнительным условиям, называют *планом*; если он, к тому же, соответствует вершине многогранника G, то *опорным планом*. Решение экстремальной задачи (41) называют *оптимальным планом*, столбцы прямоугольной матрицы A — *векторами условий*, а столбец $\boldsymbol{b}$ — *вектором ограничений*. В задачах экономики обычно все коэффициенты a, b, $c\geqslant 0$, хотя для последующего изложения это несущественно.

Многогранник условий G — выпуклый (он может быть и неограниченным). Поэтому внутри него линейная функция $L(\boldsymbol{x})$ не может достигать минимума. Ее минимум (если он существует) достигается обязательно в какой-нибудь вершине многогранника. При вырождении он может достигаться во всех точках ребра или даже p-мерной ограничивающей плоскости ($p<n-m$). Поэтому теоретически задача линейного программирования проста. Достаточно вычислить значения функции в конечном числе точек — в вершинах многогранника и найти среди этих значений наименьшее.

Сложность заключается в другом. Типичное в экономике число переменных — это сотни и даже тысячи. При этом число вершин многогранника G становится астрономическим. Для того чтобы оценить это число, рассмотрим способ нахождения вершин.

Находить вершины самого многогранника G неудобно. Лучше преобразовать задачу к канонической форме, не содержащей условий третьего типа. Для этого введем в качестве новых переменных невязки условий третьего типа:

$$x_i=b_{i+m-n}-\sum_{q=1}^{n}a_{i+m-n,\,q}\,x_q\geqslant 0,\quad n<i\leqslant N,\quad N=n+M-m. \tag{42}$$

Доопределим коэффициенты экстремальной задачи (41) следующим образом:

$$c_i=0,\quad a_{ji}=\delta_{j,\,i+M-N}\quad\text{при } n<i\leqslant N,\ 1\leqslant j\leqslant M. \tag{43}$$

Тогда задача линейного программирования примет *каноническую форму*:

$$L(x) \equiv \sum_{i=1}^{n} c_i x_i = \min, \tag{44а}$$

$$x_i \geqslant 0, \quad 1 \leqslant i \leqslant N, \tag{44б}$$

$$\sum_{i=1}^{N} a_{ji} x_i = b_j, \quad 1 \leqslant j \leqslant M \quad (M < N). \tag{44в}$$

Многогранник новых канонических условий образован пересечением новой $(N-M)$-мерной плоскости условий с первым координатным углом. Значит, все его вершины лежат на координатных гиперплоскостях, т. е. у каждой вершины часть координат — нули, а остальные координаты положительны.

Будем считать, что строки новой матрицы A линейно-независимы: в противном случае или одно условие лишнее, или система условий несовместна. Тогда ранг этой прямоугольной матрицы равен M, и среди ее столбцов найдется по крайней мере один набор из M линейно-независимых столбцов. Все линейно-независимые наборы столбцов матрицы A соответствуют точкам пересечения плоскости условий с координатными гиперплоскостями.

Чтобы найти вершину, возьмем один такой набор столбцов. Для удобства записи перенумеруем переменные так, чтобы первыми стояли столбцы, соответствующие этому набору (базису). Перепишем условия второго типа (44в) в следующем виде:

$$\sum_{i=1}^{M} a_{ji} x_i = b_j - \sum_{i=M+1}^{N} a_{ji} x_i, \quad 1 \leqslant j \leqslant M. \tag{45}$$

Обозначим через α_{ji}, $1 \leqslant j$, $i \leqslant M$, элементы матрицы, обратной к базисной квадратной матрице, стоящей в левой части системы (45). Приравнивая внебазисные координаты нулю и решая эту систему, получим координаты точки пересечения плоскости условий с координатной гиперплоскостью

$$\begin{aligned} x_i &= \sum_{j=1}^{M} \alpha_{ij} b_j, \quad 1 \leqslant i \leqslant M, \\ x_i &= 0, \qquad\qquad M < i \leqslant N. \end{aligned} \tag{46}$$

Если найденные координаты неотрицательны, точка пересечения принадлежит первому координатному углу, т. е. является вершиной многогранника канонических условий. Если хотя бы одно $x_i < 0$, эту точку надо отбросить и исследовать другой набор столбцов матрицы A. Если мы забракуем все точки, это

означает, что условия первого и второго рода образуют несовместную систему.

Различные столбцы матрицы A могут образовать C_N^M наборов. Поэтому в самом неблагоприятном случае ($M \approx {}^1/_2 N$) многогранник условий может иметь до $C_N^{N/2} \approx 2^N$ вершин. Если $N \sim 100$, то это число настолько велико, что простой перебор вершин невозможен. Нетрудно подсчитать, что для ЭВМ типа БЭСМ-6 простой перебор посилен только при $N \leqslant 15$.

4. Симплекс-метод позволяет найти решение задачи линейного программирования за гораздо меньшее число действий. Изложим идею метода.

Найдем какую-нибудь вершину многогранника и все ребра, выходящие из этой вершины. Пойдем вдоль того из ребер, по которому функция убывает. Придем в следующую вершину, найдем выходящие из нее ребра и повторим процесс. Когда мы придем в такую вершину, что вдоль всех выходящих из нее ребер функция возрастает, то минимум достигнут. Поскольку $L(\boldsymbol{x})$ — линейная функция, а многогранник условий выпуклый, то этот процесс всегда сходится к решению задачи, причем за конечное число шагов.

При канонической форме записи многогранника условий из каждой его вершины исходит $N - M$ ребер. Выбирая одно ребро, мы выбрасываем из рассмотрения вершины, лежащие на остальных траекториях. Следовательно, за k шагов мы рассматриваем $(N-M)^{-k}$-ю часть вершин, проходя мимо остальных. Нам надо найти искомую вершину среди C_N^M вершин многогранника. Приравнивая число вершин C_N^M величине $(N-M)^k$, получим, что минимум достигается примерно за $k \sim N$ шагов, т. е. достаточно быстро.

Выведем формулы шага. Первую вершину находим по формуле (46). Чтобы найти ребро, надо одну из внебазисных переменных x_l сделать положительной; тогда координаты точек ребра можно выразить через нее из (45) при помощи обратной матрицы

$$\tilde{x}_i = x_i - x_l \sum_{j=1}^{M} \alpha_{ij} a_{jl}, \qquad 1 \leqslant i \leqslant M, \qquad \tilde{x}_l = x_l; \tag{47}$$

остальные внебазисные координаты остаются равными нулю. Будем увеличивать x_l до тех пор, пока одна из базисных координат не обратится в нуль. Это будет при

$$\bar{x}_l = \min_{1 \leqslant i \leqslant M} \left(\frac{x_i}{S_{il}} \right), \qquad S_{il} = \sum_{j=1}^{M} \alpha_{ij} a_{jl} > 0; \tag{48}$$

минимум ищется только среди тех индексов i, для которых $S_{il} > 0$, ибо только эти координаты вдоль данного ребра уменьшаются и,

следовательно, могут обратиться в нуль. Если все суммы S_{il} при данном l отрицательны, то это ребро неограниченное и весь многогранник условий — тоже.

Подставляя найденное $\bar{x}_l$ в формулы (47), получим координаты новой вершины и вычислим в ней значение функции $L(\boldsymbol{x}) = L_l$. Поочередно меняя каждую внебазисную переменную, найдем все $N - M$ ребер, выходящих из исходной вершины и проводящих в смежные вершины. Сравним все значения функции в смежных вершинах L_l и выберем из них наименьшее. Если оно меньше, чем значение функции в исходной вершине L_0, то переместимся в наинизшую из новых вершин и повторим процесс. Если же $\min L_l \geqslant L_0$, то минимум уже достигнут в исходной вершине.

Для всех неограниченных ребер, исходящих из вершины, надо проверять знак производной функции

$$\Delta_l = \frac{dL}{dx_l} = c_l - \sum_{i=1}^{M} c_i \sum_{j=1}^{M} \alpha_{ij} a_{jl} = c_l - \sum_{i=1}^{M} c_i S_{il}. \qquad (49)$$

Если эта величина отрицательна, то задача линейного программирования вообще не имеет решения ($\min L(\boldsymbol{x}) = -\infty$). Если же она неотрицательна, то это ребро не ведет к минимуму, и оно нас не интересует.

Нетрудно оценить, что для выполнения всех шагов и получения минимума требуется примерно до $10N^2M^2$ арифметических действий. Это уже приемлемо для крупных современных ЭВМ.

Симплекс-метод является примером высоко специализированного метода. Он пригоден только для нахождения минимума линейной функции в многомерном выпуклом многограннике определенного вида — симплексе. Зато он позволяет решать задачи с огромным числом переменных.

5. Регуляризация линейного программирования. Задача линейного программирования часто оказывается плохо обусловленной. Так, себестоимость единицы продукции или норм расхода сырья на разных заводах не должна сильно отличаться. Поэтому даже заметное перераспределение заказов между заводами слабо влияет на суммарную стоимость продукции. Соответственно малая вариация суммарной стоимости приводит к большой вариации распределения заказов.

По тем же причинам небольшое изменение себестоимости или других показателей на отдельных заводах сильно меняет оптимальный план, так что решение очень чувствительно к вариациям коэффициентов. А сами эти коэффициенты не вполне точно известны. Поэтому на практике задача (41) нередко оказывается настолько плохо обусловленной, что не удается даже проверить, совместна ли система дополнительных условий, т. е. может ли существовать решение поставленной задачи.

Для регуляризации задачи линейного программирования воспользуемся тем же способом, что и для решения плохо обусловленных линейных систем (см. главу V, § 1). Будем искать *нормальное* решение $\boldsymbol{x}$, т. е. наименее уклоняющееся от некоторого заданного вектора $\boldsymbol{x}_0$. Обычно в качестве $\boldsymbol{x}_0$ берут ранее составленный план. Тогда регуляризованное решение будет почти не уступать оптимальному по величине $L(\boldsymbol{x})$ и в то же время мало отличаться от старого плана, так что перестройка планов будет небольшой.

Возьмем исходную задачу в канонической форме (44) и рассмотрим формулы регуляризации. Надо минимизировать положительную функцию $L(\boldsymbol{x})$ или, что то же самое, функцию $L^2(\boldsymbol{x})$. Дополнительным условием служит система уравнений $A\boldsymbol{x}=\boldsymbol{b}$ с прямоугольной матрицей. Поскольку коэффициенты системы известны не точно, то достаточно найти приближенное решение. Тогда требование приближенного соблюдения этих условий эквивалентно введению штрафной функции $\mu\|A\boldsymbol{x}-\boldsymbol{b}\|^2$, т. е. постановке следующей задачи:

$$L^2(\boldsymbol{x})+\mu\|A\boldsymbol{x}-\boldsymbol{b}\|^2=\min, \quad \mu>0. \tag{50}$$

Здесь норму будем определять, как $\|\boldsymbol{y}\|^2=(\boldsymbol{y},\boldsymbol{y})$; тогда все минимизируемые выражения будут квадратичными функциями $\boldsymbol{x}$, что облегчит вычисления. Заметим, что пока мы не учитывали требования неотрицательности компонент решения.

Условием близости решения к заданному вектору можно считать малость величины $\|\boldsymbol{x}-\boldsymbol{x}_0\|$ или, в более общем виде, малость величины

$$\Omega[\boldsymbol{x}]=\sum_{i=1}^{N} p_i(x_i-x_{i0})^2, \quad p_i>0. \tag{51}$$

Эту величину также можно считать штрафом и прибавлять в качестве дополнительного слагаемого в левую часть (50); тогда получаем регуляризованную задачу

$$M[\boldsymbol{x}]=L^2(\boldsymbol{x})+\mu\|A\boldsymbol{x}-\boldsymbol{b}\|^2+\lambda\Omega[\boldsymbol{x}]=\min, \quad \mu,\lambda>0. \tag{52}$$

Отклонение регуляризованного решения от $\boldsymbol{x}_0$ не должно быть большим. Но $\boldsymbol{x}_0$ есть некоторый план; следовательно, его компоненты неотрицательны. Значит, если у решения, найденного из условия (52), и будут отрицательные компоненты, то небольшие по абсолютной величине, что в итоге несущественно. Поэтому при решении регуляризованной задачи (52) условия неотрицательности (44б) обычно можно не принимать во внимание.

Величина $M[\boldsymbol{x}]$ является квадратичной формой, так что нахождение ее минимума (путем обычного дифференцирования по координатам) сводится к решению системы линейных уравнений. Поскольку задача регуляризована, то полученная линейная система

будет хорошо обусловлена; тогда ее решение даже при большом числе неизвестных $N \sim 200$ легко вычислить методом исключения Гаусса.

Более сложен вопрос о выборе параметров регуляризации μ и λ. Величину μ подбирают так, чтобы для найденного регуляризованного решения выполнялось условие $\|A\boldsymbol{x} - \boldsymbol{b}\| \approx \|\delta \boldsymbol{b}\|$, где $\delta \boldsymbol{b}$ — допустимая погрешность вектора $\boldsymbol{b}$, связанная с тем, что его компоненты и коэффициенты матрицы A известны неточно. Аналогичным образом величину λ связывают с погрешностями коэффициентов c_i и с допустимыми отклонениями функции $L(\boldsymbol{x})$ от своего минимального значения.

При численном решении задачи (52) приходится находить серию регуляризованных решений, соответствующих разным значениям параметров μ и λ, и выбирать оптимальные параметры. Несмотря на это, общий объем вычислений в описанном методе, по-видимому, не больше, чем в симплекс-методе для нерегуляризованной задачи (44).

§ 4. Минимизация функционала

1. Задачи на минимум функционала. Если каждой функции $y(x)$ из некоторого множества функций Y сопоставлено число $\Phi[y(x)]$, то говорят, что на множестве Y задан функционал. Задача минимизации функционала формулируется так: *найти функцию $\bar{y}(x) \in Y$, на которой функционал достигает своей точной нижней грани на этом множестве:*

$$\Phi[\bar{y}(x)] = \inf \Phi[y(x)], \qquad y(x),\ \bar{y}(x) \in Y. \tag{53}$$

Иногда эту задачу называют минимизацией функционала *по аргументу*, а просто минимизацией называют нахождение числа $\bar{\Phi} = \inf \Phi[y(x)]$, когда не требуется определять функцию, минимизирующую этот функционал.

Не всякий функционал и не на всяком множестве имеет минимум. Например, решения задачи (53) не существует, если функционал не ограничен снизу на заданном множестве: решения может также не существовать, если множество не компактно в себе, или функционал разрывен и т. д. (хотя условия непрерывности или компактности не являются, вообще говоря, необходимыми). Но мы не исследуем постановки задач и дальше будем предполагать, что конкретные решаемые нами задачи типа (53) корректно поставлены.

Дадим несколько примеров задач на минимум функционала. Пусть требуется решить операторное уравнение

$$Ay(x) = f(x), \qquad a \leqslant x \leqslant b. \tag{54}$$

Составим функционал

$$\Phi[y(x)]=\int_a^b\{Ay(x)-f(x)\}^2\rho(x)\,dx,\quad \rho(x)>0. \tag{55}$$

Очевидно, он равен нулю при $Ay(x)=f(x)$ и положителен, если $Ay(x)\neq f(x)$ на сколь угодно малом, но конечном интервале Δx. Таким образом, найдя функцию $\bar{y}(x)$, на которой функционал (55) достигает своего абсолютного минимума, мы получим решение уравнения (54). Заметим, что этот функционал ограничен снизу на любом множестве функций и непрерывно зависит от $Ay(x)$. Описанный способ решения операторных уравнений называется *методом наименьших квадратов*.

Если задача (54) некорректно поставлена (например, неустойчива по правой части), то наиболее употребительным общим методом регуляризации является замена исходной задачи на задачу минимизации функционала А. Н. Тихонова:

$$M[y(x),\alpha]=\int_a^b\{Ay(x)-f(x)\}^2\rho(x)\,dx+\alpha\Omega[y(x)]=\min,\quad \alpha>0, \tag{56}$$

где так называемый *стабилизатор* $\Omega[y(x)]$ — специально подобранный положительный функционал, обладающий свойствами нормы; он несколько напоминает штрафную функцию. В главе XIV будет показано, что для стабилизаторов типа

$$\Omega[y(x)]=\int_a^b\{p(x)\,y^2(x)+q(x)\,y'^2(x)\}\,dx,\qquad p(x),q(x)>0, \tag{57}$$

решение задачи (56) непрерывно зависит от $f(x)$, причем при правильном подборе α оно одновременно достаточно близко в чебышевской норме к решению $\bar{y}(x)$ уравнения (54).

Уравнение (54) может привести и к другим функционалам. Пусть оператор A аддитивен, положителен и симметричен, так что $(y,\ Ay)>0$ при $y\neq 0$ и $(z,\ Ay)=(Az,\ y)$, где под скалярным произведением подразумевается интеграл от произведения функций. Рассмотрим функционал

$$\Phi[y(x)]=(y,\ Ay)-2(y,f), \tag{58}$$

где

$$(y,\ z)=\int_a^b y(x)\,z(x)\,dx.$$

Покажем, что задача на минимум этого функционала эквивалентна задаче решения операторного уравнения (54).

В самом деле, запишем произвольную функцию $y(x)$ в следующем виде:

$$y(x) = \bar{y}(x) + \lambda z(x). \tag{59}$$

Подставляя это выражение в правую часть формулы (58), получим

$$\Phi[y(x)] = \Phi[\bar{y}(x)] + 2\lambda(z,\ A\bar{y} - f) + \lambda^2(z,\ Az). \tag{60}$$

Если $\bar{y}(x)$ есть решение уравнения (54), то второе слагаемое в правой части (60) обращается в нуль; последний же член в правой части неотрицателен благодаря положительности оператора A. Значит, $\Phi[\bar{y}] = \inf \Phi[y]$, т. е. функционал (58) достигает минимума на решении операторного уравнения (54).

Наоборот, если $\bar{y}(x)$ в представлении (59) есть функция, на которой функционал (58) достигает минимума, то первая вариация функционала на этой функции равна нулю. Следовательно, $(d\Phi/d\lambda)_{\lambda=0} = 0$, каково бы ни было $z(x)$. Применяя это условие к (60) и одновременно полагая $z(x) = A\bar{y}(x) - f(x)$, получим

$$(A\bar{y} - f,\ A\bar{y} - f) = 0,$$

что выполняется только при $A\bar{y}(x) = f(x)$. Это означает, что функция, на которой функционал (58) достигает минимума, является решением операторного уравнения (54). Утверждение доказано.

Классическим примером применения описанного приема является краевая задача

$$\begin{gathered} -\frac{d}{dx}\left[p(x)\frac{dy}{dx}\right] + q(x)\,y(x) = f(x), \\ p(x) > 0, \quad q(x) > 0, \quad y(-\infty) = y(+\infty) = 0. \end{gathered} \tag{61}$$

Интегрированием по частям легко убедиться в симметричности и положительности дифференциального оператора и получить следующее выражение для функционала (58):

$$\Phi[y(x)] = \int_{-\infty}^{+\infty} \left\{p(x)\left(\frac{dy}{dx}\right)^2 + q(x)\,y^2(x) - 2f(x)\,y(x)\right\} dx. \tag{62}$$

Отметим, что оператор A включает в себя не только дифференциальное (или интегральное) уравнение, но также краевые условия, если последние имеются. Краевые условия должны некоторым образом войти в функционал, соответственно изменив его вид. Например, для задачи на ограниченном отрезке с краевыми условиями третьего рода

$$-\frac{d}{dx}\left[p(x)\frac{dy}{dx}\right] + q(x)\,y(x) = f(x), \quad p(x),\ q(x) > 0, \tag{63а}$$

$$\alpha_0 y(a) + \alpha_1 y'(a) = \alpha, \quad \beta_0 y(b) + \beta_1 y'(b) = \beta, \tag{63б}$$

надо минимизировать в классе достаточно гладких функций функционал

$$\Phi[y(x)] = \int_a^b \left\{p(x)\left(\frac{dy}{dx}\right)^2 + q(x)\,y^2(x) - 2f(x)\,y(x)\right\} dx + \\ + \frac{p(a)}{\alpha_1}[2\alpha y(a) - \alpha_0 y^2(a)] + \frac{p(b)}{\beta_1}[\beta_0 y^2(b) - 2\beta y(b)]. \quad (64)$$

От функций, минимизирующих этот функционал, уже не надо требовать удовлетворения краевым условиям — они автоматически будут им удовлетворять.

В теоретической физике встречаются функционалы более сложные, чем квадратичные. Например, в статистической модели атома Томаса — Ферми при температуре абсолютного нуля энергия выражается через электронную плотность следующим образом:

$$E[\rho(\boldsymbol{r})] = \int_V dv \left\{\frac{3(3\pi^2)^{2/3}\hbar^2}{10m}\rho^{5/3}(\boldsymbol{r}) - \frac{Ze^2}{r}\rho(\boldsymbol{r}) + \frac{e^2}{2}\rho(\boldsymbol{r})\int_V \frac{\rho(\boldsymbol{r}')\,dv'}{|\boldsymbol{r}-\boldsymbol{r}'|}\right\}. \quad (65)$$

Поскольку при нулевой температуре и заданном объеме энергия минимальна, то нахождение электронной плотности сводится к задаче на условный экстремум для этого функционала (дополнительное условие заключается в том, что полное число электронов равно заряду ядра).

К еще более сложным функционалам приводят задачи *оптимального управления*, в которых ищется минимум функционала $\Phi[y(x)]$, причем функция $y(x)$ является решением задачи Коши для дифференциального уравнения $\frac{dy}{dx} = F(x,\ y(x),\ u(x)),\ y(0) = y_0$. Требуется найти такую *управляющую* функцию $u(x)$, при которой заданный функционал минимален. К задачам оптимального управления относится, например, определение оптимального режима расхода горючего $u(t)$ при запуске ракеты, приводящего к максимальной высоте подъема Φ при заданном начальном количестве горючего.

2. Метод пробных функций. Общая схема численного решения заключается в сведении задачи (53) к поиску минимума функции многих переменных. Рассмотрим класс V_n *пробных функций* заданного вида $v_n(x;\ \boldsymbol{a}) = v_n(x;\ a_1,\ a_2,\ \ldots,\ a_n)$, содержащих n свободных параметров и принадлежащих множеству V_n. На этом классе функций рассматриваемый функционал будет функцией n переменных — свободных параметров:

$$\Phi[v_n(x;\ \boldsymbol{a})] = F_n(\boldsymbol{a}) \equiv F_n(a_1,\ a_2,\ \ldots,\ a_n); \quad (66)$$

численное нахождение минимума функции многих переменных было подробно рассмотрено в предыдущих параграфах. Найдя

минимум функции $F_n(\boldsymbol{a})$ и соответствующие ему значения параметров $\bar{\boldsymbol{a}}$, мы определим функцию $v_n(x; \bar{\boldsymbol{a}})$, на которой функционал достигает своего минимума в классе V_n.

Можно ли считать найденную функцию $v_n(x; \bar{\boldsymbol{a}})$ приближенным значением искомого решения $\bar{y}(x)$? Чтобы выяснить это, рассмотрим предельный переход $n \to \infty$.

Построим бесконечную последовательность классов функций V_n (принадлежащих заданному множеству Y) с увеличивающимся числом параметров так, чтобы каждая функция предыдущего класса получалась из функции последующего класса фиксированием некоторого значения последнего параметра:

$$v_{n-1}(x; a_1, a_2, \ldots, a_{n-1}) = v_n(x; a_1, a_2, \ldots, a_{n-1}, \bar{a}_n). \quad (67)$$

Тогда каждый класс V_n вложен в классы с бóльшим индексом. Если обозначить через Φ_n минимум функционала на этом классе

$$\Phi_n = \Phi[v_n(x; \bar{\boldsymbol{a}})] = \inf_{V_n} \Phi[v_n(x; \boldsymbol{a})], \quad (68)$$

то

$$\Phi_1 \geqslant \Phi_2 \geqslant \Phi_3 \geqslant \ldots \geqslant \bar{\Phi} = \inf_{Y} \Phi[y(x)].$$

Последовательность Φ_n не возрастает и ограничена снизу; значит, она сходится к пределу, который больше или равен $\bar{\Phi}$. Если $\lim_{n\to\infty} \Phi_n = \bar{\Phi}$, то последовательность функций $v_n(x; \bar{\boldsymbol{a}})$, на которых достигается минимум функционала в классах V_n, называют *минимизирующей* (или минимизирующей функционал).

Рассмотрим два понятия, нужных для дальнейшего изложения.

Будем называть функционал $\Phi[y(x)]$ *непрерывным*, если он непрерывно зависит от $y(x)$, т. е. если фиксировать $y(x)$, то для любого $\varepsilon > 0$ найдется такое $\delta(\varepsilon)$, что при $\|y(x) - \tilde{y}(x)\| < \delta(\varepsilon)$ будет выполняться неравенство $|\Phi[y] - \Phi[\tilde{y}]| < \varepsilon$. Очевидно, наличие или отсутствие этого свойства зависит как от вида функционала, так и от выбора нормы функции. Например, наиболее распространенные функционалы имеют вид

$$\Phi[y(x)] = \int_a^b f(x, y(x), y'(x), \ldots, y^{(p)}(x))\, dx, \quad (69)$$

где f — непрерывная функция всех своих аргументов. Их можно рассматривать в пространстве $C^{(p)}$ с нормой $\|y\| = \max\{|y(x)|, |y'(x)|, \ldots, |y^{(p)}(x)|\}$; тогда непрерывность функционала очевидна. А в чебышевском пространстве $C^{(0)}$ такой функционал уже не будет, вообще говоря, непрерывно зависеть от $y(x)$.

Бесконечная система функций заданного вида $\{v_n\}$ называется *полной*, если при $n \to \infty$ она может аппроксимировать в данной норме со сколь угодно высокой точностью любую функцию множества Y. Это значит, что для любой заданной функции $y(x) \in Y$ и любого $\delta > 0$ существует такое N, что при $n > N$ в классах V_n найдутся функции $\tilde{v}_n(x)$, удовлетворяющие условию $\| y(x) - \tilde{v}_n(x) \| < \delta$. Понятие полноты также существенно связано не только с выбором системы $v_n(x; \boldsymbol{a})$, но также с выбором нормы и множества Y.

Достаточные условия сходимости $v_n(x; \bar{\boldsymbol{a}})$ искомому решению дает следующая

Теорема. а) *Если система функций* $v_n(x; \boldsymbol{a})$ *полная, а функционал* $\Phi[y(x)]$ *непрерывен, то последовательность* $v_n(x; \bar{\boldsymbol{a}})$ *является минимизирующей,*

б) *если требования пункта* (а) *выполнены и функционал удовлетворяет дополнительному условию*

$$\Phi[y(x)] - \Phi[\bar{y}(x)] \geqslant \alpha \| y(x) - \bar{y}(x) \|^{\beta}, \quad \alpha, \beta > 0, \tag{70}$$

то последовательность $v_n(x, \bar{\boldsymbol{a}})$ *сходится к решению* $\bar{y}(x)$ *задачи* (53).

Доказательство. Поскольку функционал непрерывен, то для искомого решения $\bar{y}(x)$ задачи (53) и для заданного ε найдется такое δ, что если $\|y - \bar{y}\| < \delta$, то $\varepsilon > \Phi[y] - \Phi[\bar{y}] \geqslant 0$ (в последнем неравенстве не надо ставить знак модуля, ибо $\Phi[\bar{y}]$ есть минимальное значение функционала). Но система $\{v_n\}$ полная; следовательно, для функции $\bar{y}(x)$ и данного δ существует такое N, что во всех классах V_n при $n > N$ найдутся функции $v_n(x; \tilde{\boldsymbol{a}})$, удовлетворяющие условию $\| v_n(x; \tilde{\boldsymbol{a}}) - \bar{y}(x) \| < \delta$. Тогда выполняется неравенство $\varepsilon > \Phi[v_n(x; \tilde{\boldsymbol{a}})] - \bar{\Phi} \geqslant 0$. Поскольку $\Phi_n = \inf \Phi[v_n(x; \boldsymbol{a})]$, то отсюда следует неравенство $\varepsilon > \Phi_n - \bar{\Phi} \geqslant 0$. Оно означает, что

$$\lim_{n \to \infty} \Phi_n = \bar{\Phi},$$

так что первое утверждение теоремы доказано.

Применяя к последнему неравенству условие (70), получим $\| v_n(x, \bar{\boldsymbol{a}}) - \bar{y}(x) \| \leqslant (\varepsilon/\alpha)^{1/\beta}$, так что второе утверждение теоремы также доказано.

Замечание 1. Сходимость $v_n(x; \bar{\boldsymbol{a}}) \to \bar{y}(x)$ доказана в смысле той нормы, которая входила в определения полноты системы функций, непрерывности функционала и условие (70). Пусть в исходных определениях подразумевались разные нормы; в условиях полноты — аппроксимация в $\| \cdot \|_1$, в условии непрерывности функционала — малость $\| \delta y \|_2$ и в условии (70) — неравенство при $\| \cdot \|_3$. Если существует такая норма $\| \cdot \|_4$, которая не сильнее $\| \cdot \|_1$ и $\| \cdot \|_3$, но не слабее $\| \cdot \|_2$, то при переходе к этой норме все не-

равенства сохранятся *). Тогда из теоремы следует сходимость минимизирующей последовательности в $\|\cdot\|_1$.

Замечание 2. Пусть функционал $\Phi[y]$ определен на множестве Y, но при этом известно, что искомое решение принадлежит некоторому подмножеству Y_0. Например, функционал (64) определен на множестве кусочно-гладких функций, а решение является кусочно-гладкой функцией, удовлетворяющей краевым условиям (63б). В этом случае достаточно искать решение только среди функций подмножества Y_0 и проверять полноту системы пробных функций $\{v_n\}$ и непрерывность функционала лишь по отношению к этому подмножеству. Это может существенно облегчить решение поставленной задачи.

Замечание 3. Нетрудно доказать, что если функционал непрерывен, то для сходимости последовательности $v_n(x; \bar{a})$ к $\bar{y}(x)$ необходимо, чтобы эта последовательность была минимизирующей.

Замечание 4. Существуют функционалы, для которых последовательности $v_n(x; \bar{a})$ являются минимизирующими, но при этом ни к какой предельной функции не сходятся. Это нередко встречается в задачах оптимального управления. Такие задачи относятся к некорректно поставленным и требуют регуляризации.

В задачах для конкретных функционалов исследование сходимости сводится к выбору подходящей полной системы функций $\{v_n\}$ и нормы и проверке условий теоремы. Норму обычно выбирают из соображений простоты доказательства, но эта норма не должна быть слишком слабой, иначе результат не будет представлять практической ценности.

Метод пробных функций в своей наиболее общей постановке применяется не часто. Если функционал имеет достаточно сложный вид, как в примере (65), или если выбрана система функций $v_n(x; \boldsymbol{a})$, нелинейно зависящих от свободных параметров, то получающаяся при этом функция $F(\boldsymbol{a})$ имеет достаточно общий вид. Обычно ее минимум удается найти численными методами, только если число переменных (свободных параметров) не превышает $n \sim 10-20$. Такого числа параметров не всегда достаточно, чтобы уверенно констатировать сходимость.

Поэтому для конкретных функционалов сложного вида обычно стараются исследовать качественный характер решения и выбирают пробные функции с небольшим ($n \sim 3-10$) числом параметров так, чтобы по своему качественному поведению — асимптотике, полюсам и т. д. — они были бы близки к искомому решению. Проводят исследование непрерывности функционала и полноты системы. Затем выполняют расчеты с различным числом

*) Напомним, что норма $\|\cdot\|_1$ называется более сильной, чем $\|\cdot\|_2$, если для любой допустимой функции $y(x)$ выполняется неравенство $\|y\|_1 \geqslant C\|y\|_2$, где $C = \text{const}$.

параметров и смотрят, сходятся ли полученные значения Φ_n и функции $v_n(x; \bar{a})$ к какому-то пределу.

Если последовательность $\{v_n\}$ выбрана удачно, то величина Φ_n будет близка к своему пределу Φ уже при небольшом n. Например, для функционала энергии атома (65) пробная функция всего с четырьмя параметрами $\rho(x) \approx \left(\sum_{k=1}^{4} a_k x^k\right)^{-3/2}$ обеспечивает точность расчета полной энергии существенно лучше 1%. Само искомое решение (в данном примере — распределение электронов в атоме) находится при этом с меньшей, но удовлетворительной точностью.

Однако оценить фактическую точность найденного приближения на основании таких расчетов не удается. Далее мы рассмотрим два частных случая метода пробных функций, когда можно получить и более высокую точность, и неплохую оценку погрешности.

3. Метод Ритца. Ряд важных математических задач сводится к минимизации квадратичного функционала. Примером является решение корректно или некорректно поставленных задач для линейного операторного уравнения (54), приводящее к одному из функционалов (55), (56) или (58). Если в качестве пробных функций взять обобщенные многочлены

$$v_n(x; \boldsymbol{a}) = \varphi_0(x) + \sum_{k=1}^{n} a_k \varphi_k(x), \tag{71}$$

то на них квадратичный функционал будет квадратичной функцией параметров a_k. Задача на нахождение минимума квадратичной функции $F(\boldsymbol{a})$ посредством дифференцирования по переменным a_k сводится к системе алгебраических линейных уравнений; ее нетрудно численно решить даже при числе параметров $n \sim 100 - 200$ *). Этот частный случай метода пробных функций называют методом Ритца.

Обсудим выбор функций $\varphi_k(x)$. Его целесообразно связать с краевыми условиями для задач типа (54), которые обычно линейны. Пусть, для определенности, это условия первого рода

$$y(a) = \alpha, \quad y(b) = \beta. \tag{72}$$

Выберем какую-нибудь гладкую функцию $\varphi_0(x)$ так, чтобы она

*) В отдельных случаях число параметров бывает еще больше. Например, в квантовой химии при решении уравнения Шредингера для несферического многоцентрового поля молекулы берут $n \sim 1000$.

удовлетворяла этим краевым условиям, например,

$$\varphi_0(x) = \alpha + \frac{\beta - \alpha}{b - a}(x - a), \tag{73а}$$

или

$$\varphi_0(x) = \alpha + (\beta - \alpha)\sin\frac{\pi(x-a)}{2(b-a)}. \tag{73б}$$

Остальные функции выберем так, чтобы они удовлетворяли однородным краевым условиям типа (72) и при этом образовывали бы полную систему. Например, согласно теореме Вейерштрасса любую непрерывную функцию можно аппроксимировать со сколь угодно высокой точностью алгебраическими или тригонометрическими многочленами. Поэтому можно положить

$$\varphi_k(x) = (x-a)^k(b-x), \quad k = 1, 2, \ldots, \tag{73в}$$

или

$$\varphi_k(x) = \sin\frac{\pi k(x-a)}{b-a}, \quad k = 1, 2, \ldots \tag{73г}$$

В этом случае пробные функции (71) при любых коэффициентах a_k удовлетворяют неоднородным краевым условиям (72) и являются полными на множестве непрерывных функций, удовлетворяющих этим краевым условиям. Согласно замечанию 3 к теореме п. 2 такой выбор пробных функций допустим.

Пример. Рассмотрим задачу на минимум квадратичного функционала (58) с вещественным симметричным положительным оператором A:

$$\Phi[y(x)] = (y, Ay) - 2(f, y) = \min. \tag{74}$$

Подставляя в этот функционал пробные функции Ритца (71), получим квадратичную функцию свободных параметров

$$\Phi[v_n(x; a)] = F(a) = \sum_{k=1}^{n}\sum_{m=1}^{n} a_k a_m(\varphi_k, A\varphi_m) + 2\sum_{k=1}^{n} a_k[(\varphi_k, A\varphi_0) - (\varphi_k, f)] + (\varphi_0, A\varphi_0 - 2f) = \min.$$

Приравнивая нулю производные этой квадратичной функции по параметрам, получим для определения параметров линейную систему уравнений

$$\sum_{m=1}^{n} a_m(\varphi_k, A\varphi_m) = -(\varphi_k, A\varphi_0 - f), \quad 1 \leqslant k \leqslant n. \tag{75}$$

Дадим схему исследования сходимости, не останавливаясь на деталях. В этом примере удобно ввести норму, связанную с данным положительным оператором A:

$$\|y\|_A^2 = (y, Ay). \tag{76}$$

Сделаем естественное предположение, что эта норма не слабее $\|\cdot\|_C$. В самом деле, для операторов A типа (61) такая норма содержит интеграл от квадрата функции и ее производной, а среднеквадратичная близость и функций, и их производных есть более сильное требование, чем равномерная близость функций. Для таких операторов система тригонометрических функций (73г) будет *полной* по норме (76). Действительно, для любой функции $y(x)$, непрерывно дифференцируемой r раз, ее тригонометрический ряд Фурье среднеквадратично сходится к ней вместе со своими r-ми производными. А сходимость по норме (76) отличается от среднеквадратичной только наличием весовых множителей $p(x)$, $q(x)$ под интегралом (62), что несущественно.

Найдем вариацию функционала (74) на произвольной функции

$$\delta\Phi[y]=\Phi[y+\delta y]-\Phi[y]=(\delta y,\ A\delta y)+2(\delta y,\ Ay-f). \tag{77}$$

Первое слагаемое этой вариации равно $\|\delta y\|_A^2$, т. е. является бесконечно малой второго порядка; второе слагаемое, по предположению о силе нормы (76), является бесконечно малой не ниже первого порядка относительно $\|\delta y\|_A$. Отсюда следует *непрерывность* функционала. Наконец, заметим, что решение $\bar{y}$ искомой задачи (74) удовлетворяет уравнению $Ay=f$. Подставляя это решение в (77), получим

$$\delta\Phi[\bar{y}]=\|\delta y\|_A^2.$$

Таким образом, последнее условие (70) теоремы о сходимости выполнено и метод Ритца в данном примере сходится.

Заметим, что для не квадратичных функционалов $\Phi[y]$ линейные по параметрам пробные функции (71) не дают никаких преимуществ, ибо получающиеся функции параметров $F(\boldsymbol{a})=\Phi[v_n(x;\boldsymbol{a})]$ все равно оказываются не квадратичными. Поэтому метод Ритца фактически применяют только для квадратичных функционалов.

4. Сеточный метод. Введем сетку по аргументу x и заменим все производные и интегралы, входящие в функционал, некоторыми разностями и суммами узловых значений функции $y_k=y(x_k)$. Тогда функционал аппроксимируется некоторой вспомогательной функцией многих переменных — значений решения в узлах:

$$\Phi[y(x)]\approx F(y_0,\ y_1,\ y_2,\ \ldots,\ y_n)=\min. \tag{78}$$

Решая задачу $F(y_0,\ \ldots,\ y_n)=\min$ численными методами, мы непосредственно получим приближенные значения решения в узлах сетки. Зная их, решение при остальных значениях аргумента (не совпадающих с узлами сетки) можно найти интерполяцией.

Например, рассмотрим сферически-симметричный сжатый атом в модели Томаса — Ферми; его энергия задается функционалом (65), где интегралы берутся по сферической атомной ячейке радиуса R. Вводя равномерную сетку $0=r_0<r_1<\ldots<r_n=R$ и вычисляя интегралы по формуле прямоугольников, получим

$$E\approx\frac{R}{n}\sum_{i=1}^{n}(\alpha r_i^2\rho_i^{5/3}-\beta r_i\rho_i+\gamma r_i^2\rho_i\varphi_i), \tag{79а}$$

где атомный потенциал $\varphi(r)$ сам зависит от неизвестной электронной плотности $\rho(r)$:

$$\varphi_i \approx \frac{R}{nr_i}\sum_{j=1}^{i} r_j^2\rho_j + \frac{R}{n}\sum_{j=i+1}^{n} r_j\rho_j, \qquad (79б)$$

а коэффициенты α, β, γ выражаются через физические константы. Надо найти минимум энергии при дополнительном условии нормировки

$$\int \rho(r)\, dv = Z,$$

причем это условие также надо приближенно записать в сеточной форме.

Выражения (79а), (79б) достаточно сложные, и при большом числе узлов сетки найти минимум численными методами трудно. Очевидно, что для произвольных функционалов число узлов сетки, которое практически возможно использовать в расчетах, очень невелико: оно не превышает $n \sim 10-20$. Однако даже при таком числе узлов нередко удается получить неплохую точность при умеренном объеме расчетов, используя прием сгущения сеток.

Для этого выполняют серию расчетов на сгущающихся вдвое сетках с числами интервалов $n=1$, 2, 4, 8 и 16. Поскольку порядок точности выбранных разностных формул дифференцирования и интегрирования обычно известен, то проводят уточнение результатов, полученных на разных сетках, рекуррентным методом Рунге. При этом непосредственно наблюдают, сходится ли численный расчет к пределу при увеличении n, и производят апостериорную оценку погрешности.

На каждой сетке минимум функции $F(y_0, \ldots, y_n)$ находят обычно каким-либо итерационным методом спуска. Для уменьшения числа итераций (а тем самым, объема вычислений) организуют расчет следующим образом. Сначала выполняют расчет на самой редкой сетке, где неизвестных мало (при $n=1$ всего два — y_0 и y_1) и объем вычислений заведомо невелик даже при плохом нулевом приближении. Найденный на этой сетке профиль $y(x)$ интерполируют на следующей, более подробной сетке, и используют на ней в качестве нулевого приближения. Вновь найденный профиль снова интерполируют и т. д.

Для квадратичных функционалов при использовании линейных формул численного дифференцирования и интегрирования задача (78), как и в методе Ритца, сводится к нахождению минимума квадратичной функции. Например, возьмем функционал (62), но на ограниченном отрезке $a \leqslant x \leqslant b$; введем на этом отрезке равномерную сетку с шагом h и аппроксимируем интеграл при

помощи обобщенных формул трапеции и средних:

$$\Phi[y] \approx h \sum_{i=1}^{n} \left\{ p_{i-1/2} \left(\frac{y_i - y_{i-1}}{h} \right)^2 + \right.$$
$$\left. + \frac{1}{4} q_{i-1/2} (y_i + y_{i-1})^2 - f_{i-1/2} (y_i + y_{i-1}) \right\}. \qquad (80)$$

Отыскание минимума опять сводится к решению линейной системы уравнений с неизвестными y_i, $0 \leqslant i \leqslant n$, легко выполняемому численными методами. Это позволяет брать очень большое число узлов. Тогда имеет смысл ставить вопрос о теоретическом исследовании сходимости приближенного решения к искомому при $n \to \infty$. Обоснования сходимости мы не будем давать. Укажем только один случай, когда применима сформулированная в п. 2 теорема.

Пусть функционал имеет вид (69), т. е. явно содержит функцию и ее производные вплоть до p-й. Построим последовательность сеток так, чтобы предыдущая сетка содержалась в последующей; это можно делать сгущением сеток вдвое, причем сетки могут быть даже неравномерными.

В качестве пробных функций возьмем сплайны $S(x)$ порядка не ниже p (см. главу II, § 1). Эти сплайны являются многочленами степени p, коэффициенты которых линейно выражаются через узловые значения искомой функции y_i; их производные порядка ниже p непрерывны, а p-я производная всюду существует и кусочно-непрерывна. Нетрудно убедиться в том, что условие вложения классов пробных функций V_n во все последующие классы при этом выполнено, и что такими пробными функциями при $n \to \infty$ можно со сколь угодно высокой точностью аппроксимировать любую p раз непрерывно дифференцируемую функцию вместе с ее производными вплоть до $y^{(p)}(x)$. Следовательно, система сплайн-функций обладает свойствами, нужными для применения теоремы о сходимости.

В качестве примера рассмотрим квадратичный функционал типа (62), содержащий первую производную:

$$\Phi[y] = \int_a^b \left\{ p(x) \left(\frac{dy}{dx} \right)^2 + q(x) y^2(x) - 2f(x) y(x) \right\} dx. \qquad (81)$$

Сплайн должен иметь порядок тоже не ниже первого. Ограничимся простейшим сплайном первого порядка — ломаной линией, проведенной через точки (x_i, y_i):

$$S(x) = y_{i-1} + \frac{y_i - y_{i-1}}{x_i - x_{i-1}} (x - x_{i-1}), \qquad x_{i-1} \leqslant x \leqslant x_i. \qquad (82)$$

Надо разбить интеграл (81) на сумму интегралов по отдельным интервалам сетки и каждый из этих интегралов вычислить, используя заданный закон интерполяции (82). Например, поскольку $S'(x) = (y_i - y_{i-1})/(x_i - x_{i-1})$ при $x_{i-1} < x < x_i$, то

$$\int_a^b p(x)\,[y'(x)]^2\,dx \approx \sum_{i=1}^{n} \left(\frac{y_i - y_{i-1}}{x_i - x_{i-1}}\right)^2 \int_{x_{i-1}}^{x_i} p(x)\,dx.$$

Аналогично вычисляются остальные слагаемые в (81).

Получающиеся выражения имеют более сложный вид, чем при не сплайновой аппроксимации (80); использование сплайнов высших порядков привело бы к еще более сложным выражениям (зато получающиеся при этом сеточные схемы имели бы более высокий порядок точности). Тем не менее, поскольку сами сплайны линейно зависят от узловых значений функции, то подстановка их в квадратичный функционал приводит к задаче на минимум квадратичной формы. Поэтому такой подстановкой пользуются даже для многомерных функционалов, к которым сводятся краевые задачи для эллиптических уравнений в частных производных.

Коснемся построения сплайнов в многомерных задачах. Если область G двумерна, то ее можно разбить на треугольные ячейки (у граничных ячеек одна сторона может быть не прямой). В каждой ячейке g_i по узловым значениям функции двух переменных $z(x, y)$ в трех вершинах ячейки однозначно строится простейший линейный сплайн $S(x, y) = a_i + b_i x + c_i y$, где $(x, y) \in g_i$; он соответствует аппроксимации поверхности $z(x, y)$ плоскостью. Сплайновые плоскости соседних ячеек пересекаются по прямым, проходящим через выбранные узлы поверхности $z(x, y)$; эти прямые проектируются точно на границы ячеек. Следовательно, двумерный линейный сплайн, построенный указанным образом, является непрерывным и кусочно-гладким в области G.

Описанный способ построения линейного сплайна естественно обобщается на случай любого числа измерений. При этом область G следует разбить на многомерные симплексы.

Но для построения сплайнов более высокого порядка этот несложный алгоритм не годится: в этом случае он не гарантирует непрерывности и требуемой гладкости сплайновой поверхности на границах ячеек. Требования непрерывности функции и некоторого числа ее производных на границах ячеек надо формулировать в виде дополнительных уравнений, которым должны удовлетворять коэффициенты сплайнов. Надо, чтобы полное число уравнений равнялось полному числу коэффициентов; это будет не при любой форме ячейки.

Например, рассмотрим двумерный кубический сплайн $S(x, y) = \sum_{k+m=0}^{3} a_{km}x^k y^m$ в прямоугольных ячейках со сторонами, параллельными осям координат. Потребуем непрерывности на границах сплайна вместе со вторыми производными. Этот сплайн имеет 10 коэффициентов в расчете на одну ячейку. Совпадение сплайна с функцией $z(x, y)$ в вершинах ячейки дает 4 уравнения. Потребуем на обеих сторонах ячейки, параллельных оси x, непрерывности $S(x, y)$, S_y и S_{yy}; дифференцируя их по x, нетрудно убедиться, что тогда на этих сторонах величины S_x, S_{xx} и S_{xy} тоже будут непрерывны. Аналогично потребуем непрерывности величин $S(x, y)$, S_x и S_{xx} на сторонах, параллельных оси y. Это дает по 3 уравнения на каждой стороне ячейки, но эти уравнения связывают коэффициенты двух ячеек. Следовательно, всего непрерывность дает 6 уравнений в расчете на одну ячейку. Таким образом, полное число уравнений равно полному числу коэффициентов, и сплайн определяется однозначно (с точностью до условий на границе области G).

ЗАДАЧИ

1. Вывести итерационную формулу (12) поиска минимума функции одной переменной $\Phi(x)$, заменяя истинную кривую интерполяционной параболой, проведенной через три точки $x_s - h$, x_s, $x_s + h$.

2. Дать аналогичный вывод формулы (13), строя интерполяционную параболу по точкам x_s, x_{s-1}, x_{s-2}.

3. Доказать оценку (14) для скорости сходимости процесса (13); для этого можно воспользоваться схемой доказательства, данного в главе V, § 2, п. 7.

4. Написать уравнение для линий уровня квадратичной формы (18); найти главные оси полученных эллипсов и определить отношение длин главных осей.

5. Написать линейную систему уравнений, решение которой минимизирует регуляризованную задачу линейного программирования (52).

6. Построить какую-нибудь полную систему функций в методе Ритца, если вместо краевого условия первого рода (72) задано условие второго рода $y'(a) = \alpha$, $y'(b) = \beta$.

7. Провести аккуратное доказательство сходимости метода Ритца для функционала (62), используя схему, данную в § 4, п. 3.

8. Написать систему уравнений для определения сеточных значений функции y_i, к которой приводится задача минимума функционала (62) после разностной замены (80). Убедиться, что эта линейная система имеет трехдиагональную матрицу и решение ее методом прогонки устойчиво.

9. Показать, что двумерный квадратичный сплайн

$$S(x, y) = \sum_{k+m=0}^{2} a_{km}x^m y^k$$

на треугольной сетке и трехмерный квадратичный сплайн на сетке из тетраэдров определяются однозначно (с точностью до условий на границе области G).