

MATHEMATICAL PROGRAMMING

V. A. GORELIK

The modern methods of solutions of nonlinear extremum problems with restrictions imposed by equality and inequality are described.

Рассматриваются современные методы решения нелинейных экстремальных задач с ограничениями типа равенств и неравенств.

МАТЕМАТИЧЕСКОЕ ПРОГРАММИРОВАНИЕ

В. А. ГОРЕЛИК

Московский педагогический государственный университет

ВВЕДЕНИЕ

В последние десятилетия появился ряд классов экстремальных задач, к которым классические методы решения, основанные на принципах Ферма и Лагранжа, оказались непосредственно неприменимыми. Такие задачи возникают в различных прикладных областях техники, экономики, экологии, и для них характерны ограничения не только типа равенств, но и неравенств, наличие большого числа переменных и ограничений, часто недифференцируемость целевых функций и функций ограничений, невыполнение условий регулярности, приводящее к вырожденным случаям и т.д. Теоретически классические подходы могут быть распространены на некоторые классы таких задач, но часто они становятся малоэффективными или вообще практически неприменимыми. Поэтому появилась потребность в разработке новых идей и методов решения экстремальных задач, что привело к формированию нового раздела математики – математического программирования. Рассмотрим некоторые характерные задачи математического программирования, развитие для них принципа Лагранжа, приводящее к важному понятию двойственности, общие схемы численных методов их решения.

ЗАДАЧИ МАТЕМАТИЧЕСКОГО ПРОГРАММИРОВАНИЯ

Общей задачей математического программирования называется задача нахождения глобального максимума функции $f(x)$ при ограничениях

$$x \in R, \quad (1)$$

$$g_i(x) \geq 0, \quad i = 1, 2, \dots, m, \quad (2)$$

где R – некоторое непустое подмножество n -мерного евклидова пространства, $f(x)$, $g_i(x)$ – функции от n переменных.

Если ввести множество

$$X = \{x \in R \mid g_i(x) \geq 0, i = 1, 2, \dots, m\}, \quad (3)$$

то кратко эту задачу можно записать как задачу нахождения x^0 , для которого

$$f(x^0) = \max_{x \in X} f(x). \quad (4)$$

Множество X называется допустимым множеством или множеством допустимых планов, а вектор x^0 – решением или оптимальным планом.

Вообще говоря, в задаче математического программирования могут быть одновременно ограничения типа равенств и неравенств, однако такую задачу нетрудно преобразовать к указанному виду, так как ограничение типа равенства $g(x) = 0$ эквивалентно двум ограничениям типа неравенства $g(x) \geq 0$ и $g(x) \leq 0$, а ограничения типа меньше или равно преобразуются в ограничения больше или равно умножением на -1 . Также нет необходимости рассматривать отдельно задачу на минимум, так как она сводится к задаче на максимум путем умножения целевой функции $f(x)$ на -1 .

Разделение ограничений на (1) и (2) также не носит принципиального характера, так как каждое из них может быть записано в одном из этих двух видов, но иногда оказывается полезным.

Частным случаем задачи (4) является задача линейного программирования, для которой функции $f(x)$, $g_i(x)$ являются линейными, а множество R — неотрицательным ортантом. Таким образом, задача линейного программирования состоит в нахождении максимума линейной функции $f(x)$ на многогранном множестве X . При этом для нахождения максимума достаточно перебрать вершины множества X , число которых конечно. Методы линейной алгебры позволяют достаточно эффективно описывать эти вершины, что и используется в общем методе решения задач линейного программирования — симплекс-методе, реализующем направленный перебор вершин [1].

Другой важный частный случай задачи (4) — задача выпуклого программирования, для которой множество X является выпуклым, а функция $f(x)$ — вогнутой (для задачи на минимум выпуклой). Для выпуклости X достаточно выпуклости множества R и вогнутости функций $g_i(x)$.

Существенным свойством задачи выпуклого программирования является то, что любой локальный максимум вогнутой функции $f(x)$ является и глобальным максимумом (для выпуклой функции это справедливо для минимумов). Это и другие свойства задачи выпуклого программирования существенно облегчают ее решение, так как большинство численных методов обеспечивает нахождение только локальных экстремумов. Заметим, что задача линейного программирования является частным случаем выпуклого программирования, но свойство линейности столь специфично, что позволяет применять специальные методы.

Линейное и выпуклое программирование являются наиболее разработанными разделами математического программирования. Существуют и другие разделы со своими специальными методами (целочисленное, геометрическое, динамическое программирование и т.д.).

Что касается общей задачи математического программирования, то ее решение связано с боль-

шими сложностями, так как универсальные методы, как правило, малоэффективны. Поэтому развитие математического программирования идет в основном по пути выделения специальных классов задач и разработки соответствующих им методов.

ДВОЙСТВЕННОСТЬ

Введем для задачи математического программирования с ограничениями (1), (2) функцию Лагранжа

$$F(x, y) = f(x) + \sum_{i=1}^m y_i g_i(x), \quad (5)$$

где $x \in R$, $y = (y_1, \dots, y_m)$ — вектор множителей Лагранжа, принадлежащий неотрицательному ортанту Y , то есть $y \geq 0$. Нетрудно показать, что задача (4) эквивалентна задаче нахождения $\max_{x \in R} \inf_{y \in Y} F(x, y)$, то есть

$$\max_{x \in X} f(x) = \max_{x \in R} \inf_{y \in Y} F(x, y), \quad (6)$$

и точки реализации максимума по x в этих задачах совпадают. Соотношение (6) ничего не дает для решения задачи (4), но при некоторых условиях возможна перестановка операций максимизации и минимизации в правой части с сохранением равенства, что оказывается весьма полезным для анализа и решения задачи (4).

Задача нахождения $\min_{y \in Y} \sup_{x \in R} F(x, y)$ называется двойственной к задаче (4), а соотношение

$$\max_{x \in X} f(x) = \min_{y \in Y} \sup_{x \in R} F(x, y) \quad (7)$$

называется соотношением двойственности. Теоремы, устанавливающие справедливость (7) (при тех или иных условиях), носят название теорем двойственности (в общем случае правая часть в (7) больше или равна левой части, причем это неравенство может быть строгим).

Справедливость равенства (7) связана с существованием седловой точки у функции $F(x, y)$, а именно такой точки (x^0, y^0) , что

$$F(x, y^0) \leq F(x^0, y^0) \leq F(x^0, y) \quad \forall x \in R, \quad y \in Y. \quad (8)$$

Теорема двойственности. Если функция Лагранжа $F(x, y)$ для задачи (4) имеет седловую точку (x^0, y^0) на прямом произведении множеств $R \times Y$, то справедливо соотношение двойственности (7), причем x^0 является решением задачи (4), а y^0 — решением двойственной задачи.

Соотношение двойственности позволяет свести решение задачи (4) к решению двойственной задачи, которая иногда оказывается проще. Например, двойственная задача к задаче линейного программирования тоже является линейной, причем она может иметь меньшую размерность, что позволяет строить более эффективные методы решения (на

этой идее основан двойственный симплекс-метод, см. [1]). Кроме того, соотношение двойственности позволяет установить сам факт существования решения (например, наличие хотя бы по одному допустимому плану у двойственных задач линейного программирования гарантирует существование оптимальных планов), проанализировать чувствительность решения к малым изменениям параметров задачи.

К сожалению, соотношение двойственности справедливо далеко не всегда. Среди широких классов задач математического программирования в этом плане можно выделить линейное программирование (при условии непустоты множеств допустимых планов) и выпуклое программирование (также при некоторых условиях регулярности, среди которых наиболее известным является условие Слейтера, требующее существования внутренней точки для множества, задаваемого ограничениями (2)).

Теорема Куна–Таккера. Если задача (4) является задачей выпуклого программирования, удовлетворяющей условию Слейтера, то необходимым и достаточным условием оптимальности плана $x^0 \in X$ является существование такого $y^0 \in Y$, что пара (x^0, y^0) является седловой точкой функции Лагранжа, то есть удовлетворяет (8).

В соответствии с теоремами двойственности решение задач математического программирования можно сводить к нахождению седловых точек, причем на множествах более простой структуры (без ограничений вида неравенств (2)).

ЧИСЛЕННЫЕ МЕТОДЫ НЕЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

Градиентные методы

Рассмотрим сначала задачу максимизации функции $f(x)$ без ограничений, то есть в случае, когда X совпадает со всем пространством R^n . Градиент функции $f(x)$ обозначим $f^1(x)$. Условие оптимальности в этом случае имеет вид

$$f^1(x) = 0, \quad (9)$$

однако непосредственное решение системы уравнений (9) может оказаться чересчур сложным. Поэтому на практике поступают следующим образом. Выбирая произвольную начальную точку x_1 , строят итеративный процесс

$$x_{k+1} = x_k + \alpha_k f^1(x_k), \quad k = 1, 2, \dots \quad (10)$$

Число α_k называют длиной шага или просто шагом. Если все α_k равны между собой, то имеем процесс с постоянным шагом.

Процесс (10), лежащий в основе градиентных методов, представляет собой движение в сторону возрастания функции $f(x)$, так как если $f^1(x_k) \neq 0$, то всегда можно выбрать α_k так, что $f(x_{k+1}) > f(x_k)$. Су-

ществуют разные способы выбора α_k . Вообще говоря, наилучшим является выбор такого α_k , при котором обеспечивается максимальный рост функции $f(x)$. Такое α_k находится из условия

$$f(x_k + \alpha_k f^1(x_k)) = \max_{\alpha \geq 0} f(x_k + \alpha f^1(x_k)). \quad (11)$$

Градиентный метод поиска экстремума (10) с выбором шага по способу (11) называется методом скорейшего подъема (или спуска для задачи на минимум). Такой метод требует наименьшего числа итераций, но зато на каждом шаге приходится решать дополнительную задачу поиска экстремума (11) (правда, в одномерном случае). На практике часто довольствуются нахождением любого α_k , обеспечивающего рост функции. Для этого берут произвольное α_k и проверяют условие роста. Если оно не выполняется, то дробят α_k до тех пор, пока это условие не будет выполнено (такое достаточно малое α_k при $f^1(x_k) \neq 0$ существует всегда).

Процесс (10), очевидно, останавливается, когда выполнено условие (9). При этом если функция $f(x)$ вогнута, то найденная стационарная точка будет решением задачи максимизации. В противном случае необходимо провести дополнительно исследование функции $f(x)$ в окрестности найденной точки. Однако, даже если она будет точкой максимума, в невыпуклом случае трудно определить, локальный это максимум или глобальный. Поэтому градиентные методы обеспечивают нахождение глобального экстремума только для вогнутых (выпуклых) функций, а в общем случае дают лишь локальные экстремумы (при этом можно попытаться найти глобальный экстремум, применяя итеративный процесс многократно с разными начальными точками).

Если рассматривается задача максимизации $f(x)$ при ограничениях, то есть когда X не совпадает с R^n , то непосредственное применение процесса (10) может привести к нарушению ограничений, даже если начальная точка $x_1 \in X$. Однако эту трудность можно преодолеть, например, если получаемую по формуле (10) очередную точку проектировать на множество X . Если обозначить операцию проектирования x на множество X через $P_x(x)$, то соответствующий итеративный процесс имеет вид

$$x_{k+1} = P_x(x_k + \alpha_k f^1(x_k)). \quad (12)$$

Полученный метод носит название метода проекции градиента. Шаг α_k в методе (12) может выбираться различными способами (например, как в методе скорейшего подъема). Стационарная точка этого процесса является решением задачи (4) в случае вогнутой функции $f(x)$, а в общем случае требуется дополнительное исследование.

Недостатком метода проекции градиента является необходимость проведения операции проектирования, которая в общем случае эквивалентна некоторой задаче поиска экстремума. Однако, когда

X является шаром, параллелепипедом, гиперплоскостью, полупространством или ортантом, задача проектирования решается просто и в явном виде.

Еще одной разновидностью градиентных методов является метод условного градиента, который также предназначен для решения экстремальных задач с ограничениями. Суть его состоит в решении вспомогательной задачи максимизации на множестве X линейной функции $\langle f^1(x_k), x - x_k \rangle$, представляющей собой главную часть приращения функции $f(x)$ в точке x_k . Эта вспомогательная задача может быть непростой, но если X задается линейными ограничениями, то она представляет собой задачу линейного программирования, которая решается за конечное число шагов стандартными методами (например, симплекс-методом). Если решение вспомогательной задачи \hat{x}_k найдено, то следующее приближение для исходной задачи строится по формуле

$$x_{k+1} = x_k + \alpha_k (\hat{x}_k - x_k). \quad (13)$$

Если множество X выпуклое, то $x_{k+1} \in X$. Шаг α_k выбирается из условия максимального роста функции $f(x)$ или любым другим способом, обеспечивающим рост $f(x)$. На практике обычно решают вспомогательную задачу не точно, а приближенно. В процессе (13) направление движения не совпадает с градиентом функции $f(x)$ в точке x_k , но определяется им, так как его компоненты берутся в качестве коэффициентов линейной целевой функции вспомогательной задачи.

Методы возможных направлений

Идея методов возможных направлений, близкая к идее градиентных методов для задач с ограничениями, состоит в следующем: на каждой итерации определяется допустимое направление на множестве X , вдоль которого функция $f(x)$ возрастает (такое направление называется возможным направлением возрастания функции $f(x)$), и по нему совершается шаг. Фактически в методе проекции градиента и в методе условного градиента мы находим такие направления. Однако там исходным было определение градиента, а допустимое направление определялось по нему однозначно. В методах же возможных направлений исходным пунктом являются описание всех допустимых направлений и выбор из них такого, вдоль которого функция $f(x)$ возрастает и желательно скорейшим образом.

Рассмотрим вариант метода возможных направлений применительно к задаче максимизации $f(x)$ на множестве (3), где $R = R^n$. Пусть мы имеем k -е приближение x_k к решению этой задачи, и для построения следующего приближения поставим вспомогательную задачу: максимизировать u при ограничениях

$$\langle f^1(x_k), a \rangle \geq u, \quad \langle g^1(x_k), a \rangle \geq u,$$

$$i \in I_k \quad |a^i| \leq 1, \quad j = 1, 2, \dots, n,$$

где

$$I_k = \{i \mid 1 \leq i \leq m, g_i(x_k) = 0\}, \quad a = (a^1, \dots, a^n).$$

Эта задача представляет собой задачу линейного программирования в $(n + 1)$ -мерном пространстве векторов (a, u) . Множество планов замкнуто, ограничено и непусто, так как $a = 0, u = 0$ является допустимым планом. Значит, вспомогательная задача имеет решение (a^k, u^k) , причем $u^k \geq 0$. Если $u^k > 0$, то нетрудно показать, что направление a^k является возможным направлением возрастания функции $f(x)$, то есть точка $x_{k+1} = x_k + \alpha_k a^k$ при достаточно малом α_k принадлежит множеству X и обеспечивает большее значение функции $f(x)$, чем x_k . Выбор пары (a^k, u^k) с возможно большим значением u^k при этом означает выбор допустимого направления, наиболее близкого к градиенту функции, то есть возможного направления с наибольшим ростом функции. Если $u^k = 0$, то получается стационарная точка процесса, которая для задачи выпуклого программирования дает решение, а в общем случае требует дополнительного исследования.

Методы множителей Лагранжа

Эта группа методов основана на сведениях решения задачи математического программирования к нахождению седловой точки функции Лагранжа. Методы множителей Лагранжа представляют собой различные процедуры поиска седловой точки. В качестве такой процедуры можно взять, например, итеративный процесс как в методе проекции градиента, применив его для подъема по переменной x и спуска по переменной y . Получим процесс

$$\begin{aligned} x^{k+1} &= P_R \left(x^k + \alpha_k \frac{\partial F(x^k, y^k)}{\partial x} \right), \\ y^{k+1} &= P_Y \left(y^k - \alpha_k \frac{\partial F(x^k, y^k)}{\partial y} \right), \end{aligned}$$

где

$$\begin{aligned} \frac{\partial F(x^k, y^k)}{\partial x} &= f^1(x^k) + \langle y^k, g^1(x^k) \rangle, \quad \frac{\partial F(x^k, y^k)}{\partial y} = g(x^k), \\ g(x) &= (g_1(x), \dots, g_m(x)), \quad Y = \{y \mid y \geq 0\}. \end{aligned}$$

Проекция на множество Y определяется весьма просто:

$$P_Y(y) = (Z_1, \dots, Z_m), \quad Z_i = \max(y_i, 0), \quad i = 1, 2, \dots, m.$$

Если R – неотрицательный ортант, то проекция на него определяется аналогичным образом.

Методы второго порядка

До сих пор мы рассматривали методы первого порядка, то есть методы поиска экстремума, использующие первые производные. Фактически и

этих методах производится линеаризация, связанная с заменой максимизируемой функции $f(x)$ линейным членом разложения ее в ряд Тейлора. Но если функция дважды непрерывно дифференцируема, то можно использовать для ее аппроксимизации два члена ряда Тейлора. Использование такой аппроксимизации в итеративном процессе может повысить скорость сходимости.

Наиболее известным из методов второго порядка является метод Ньютона, который состоит в следующем. На k -й итерации, имея приближение x_k , ставим вспомогательную задачу максимизации функции

$$\varphi_k(x) = \langle f^1(x_k), x - x_k \rangle + 0,5 \langle f''(x_k)(x - x_k), x - x_k \rangle$$

на множестве X . Пусть

$$\varphi_k(\tilde{x}_k) = \max_{x \in X} \varphi_k(x),$$

тогда следующее приближение строится по формуле

$$x_{k+1} = x_k + \alpha_k(\tilde{x}_k - x_k), \quad 0 \leq \alpha_k \leq 1.$$

Шаг α_k можно выбирать разными способами, в частности можно положить $\alpha_k = 1$, тогда в качестве следующего приближения принимается просто решение вспомогательной задачи, то есть $x_{k+1} = \tilde{x}_k$.

Для задачи без ограничений, когда $X = R^n$, метод Ньютона существенно упрощается. Действительно, в этом случае $\varphi_k^1(\tilde{x}_k) = 0$, то есть

$$f^1(x_k) + f''(x_k)(\tilde{x}_k - x_k) = 0. \quad (14)$$

Значит, для нахождения \tilde{x}_k надо решить систему линейных уравнений (14). Если матрица $f''(x_k)$ не вырождена, то имеем просто (при $\alpha_k = 1$)

$$x_{k+1} = \tilde{x}_k = x_k - (f''(x_k))^{-1} f^1(x_k).$$

Достоинством метода Ньютона является высокая скорость сходимости, которая с лихвой компенсирует усложнение вычислений на каждой итерации. Недостатком метода является то, что он сходится при весьма жестких предположениях.

Методы штрафных функций

Как видно из предыдущих рассмотрений, наличие ограничений существенно усложняет задачу поиска экстремума функции. Почти все изложенные методы решения экстремальных задач включали специальные сложные процедуры, учитывающие наличие ограничений (например, проектирование). В этом смысле в особую группу выделялись методы множителей Лагранжа, которые состоят в сведении задачи поиска экстремума с ограничениями к задаче поиска экстремума (точнее, седловой точки, что примерно то же самое) без ограничений или с ограничениями простого вида, учет которых не составляет труда (это видно на примере операции проектирования на неотрицательный ортант). Однако

эти методы применимы практически только в выпуклом случае. От этого недостатка избавлены методы штрафных функций, которые позволяют в весьма общем случае сводить экстремальные задачи со сложными ограничениями к задачам без ограничений или с простыми ограничениями, а в сочетании с методами поиска безусловного экстремума служат универсальным средством решения задач математического программирования (конечно, не лишенным недостатков).

Рассмотрим общую задачу математического программирования (4). Методы штрафных функций основаны на описании множества X с помощью функций со специальными свойствами. Одна группа методов использует функции вида $\Phi(x, c)$, которые обладают следующими свойствами:

$$\lim_{c \rightarrow \infty} \Phi(x, c) = \begin{cases} 0, & x \in X, \\ -\infty, & x \notin X. \end{cases}$$

В качестве такой функции можно взять, например, квадратичную функцию штрафа

$$\Phi_0(x, c) = -c \sum_{i=1}^m [\min(0, g_i(x))]^2,$$

которая является дифференцируемой, если дифференцируемы функции ограничений $g_i(x)$. Можно показать, что если $f(x)$ и $g_i(x)$, $i = 1, 2, \dots, m$, непрерывны на замкнутом, ограниченном множестве R и $X \neq \emptyset$, то

$$\max_{x \in X} f(x) = \lim_{c_n \rightarrow \infty} \max_{x \in R} [f(x) + \Phi_0(x, c_n)]. \quad (15)$$

При этом если $x^0(c_n)$ реализует максимум в правой части (15), то любая предельная точка последовательности $\{x^0(c_n)\}$ есть решение задачи (4).

Таким образом, с помощью метода штрафных функций общую задачу математического программирования можно свести приближенно (при достаточно большом c_n) к задаче без ограничений или с простыми ограничениями (на простом множестве R).

Недостатком методов штрафных функций является ухудшение свойств вспомогательных задач (в правой части (15)) при больших значениях c_n , в частности чувствительность к ошибкам вычислений. Эти недостатки преодолены в некоторых вариантах штрафных функций, в которых можно ограничиться конечными значениями штрафных констант c_n , а также в близких к ним методах модифицированной функции Лагранжа [2].

Более подробно с задачами и методами нелинейного программирования можно познакомиться в [3–5].

Если говорить о практической реализации методов решения задач математического программирования, то надо иметь в виду, что ни один из них не

обладает универсальностью, позволяющей применять его эффективно к разным классам задач. Поэтому методы следует выбирать в зависимости от вида максимизируемой функции и функций ограничений. Еще более эффективными являются использование комбинаций методов и применение их в определенной последовательности, если брать полученное одним методом приближенное решение в качестве начального приближения для следующего метода. Такая идея реализуется в существующих пакетах и системах оптимизации, и для практического решения задач главное — научиться грамотно пользоваться ими (по этому вопросу см., например, [6]).

ЗАКЛЮЧЕНИЕ

В современной теории оптимизации рассматриваются и более сложные, чем (4), классы экстремальных задач, которые возникают в процессах принятия решений в условиях многокритериальности (векторная оптимизация), неопределенности (стохастическое программирование, минимаксные задачи и т.д.), конфликта (теория игр). Эти разделы оптимизации вместе с математическим программированием составляют новую научную дисциплину — теорию исследования операций (см., например, [7]). Вместе с тем математический аппарат исследования операций широко использует рассмотренные выше подходы, которые составляют основу современных методов оптимизации.

ЛИТЕРАТУРА

1. Ашманов С.А. Линейное программирование. М.: Наука, 1980. 340 с.
2. Гольштейн Е.Г., Третьяков И.В. Модифицированные функции Лагранжа: Теория и методы оптимизации. М.: Наука, 1989. 400 с.
3. Акулич И.Л. Математическое программирование в примерах и задачах. М.: Высш. шк., 1993. 336 с.
4. Нурминский Е.А. Численные методы выпуклой оптимизации. М.: Наука, 1991. 167 с.
5. Мишу М. Математическое программирование: Теория и алгоритмы. М.: Наука, 1990. 488 с.
6. Евтушенко Ю.Г., Мазурик В.П. Программное обеспечение систем оптимизации. М.: Знание, 1989. 48 с.
7. Горелик В.А., Ушаков И.А. Исследование операций. М.: Машиностроение, 1986. 286 с.

* * *

Виктор Александрович Горелик, доктор физико-математических наук, профессор кафедры информатики и дискретной математики Московского педагогического государственного университета, ведущий научный сотрудник Вычислительного центра РАН, действительный член Международной академии информатизации. Область научных интересов: математическое программирование, теория игр, теория оптимального управления. Автор около 100 научных работ, в том числе двух научных и трех учебных монографий.