

УДК 517.53

©1994 Е.А. Нурминский

УСКОРЕНИЕ ИТЕРАТИВНЫХ МЕТОДОВ ПРОЕКЦИИ НА МНОГОГРАННИК

Введение

В современной теории оптимизационных алгоритмов одной из часто встречающихся вспомогательных операций является нахождение эвклидовой или иной проекции начала координат на выпуклое множество D , заданное в виде некоторой системы линейных или нелинейных неравенств.

Такая задача типична для алгоритмов недифференцируемой оптимизации типа ϵ -кратчайшего спуска [1, 6, 8, 3], методов проектирования [7], методов отделяющих плоскостей [2] и др. Среди практических приложений этой задачи следует отметить проблему нахождения оптимального пакета ценных бумаг [17].

В случае, когда D задано как выпуклая оболочка некоторого множества вершин и лучей, задача проекции состоит в нахождении

$$\begin{aligned} r^2 = \min \|x\|^2 &= \|x^*\|^2, \\ x &= Xu + Yv, \\ u \in \Delta, v &\geq 0, \end{aligned} \quad (1)$$

где X - матрица, столбцами которой являются векторы вершины множества D , Y - матрица, столбцами которой являются направляющие векторы лучей. Множество Δ - это стандартный симплекс подходящей размерности:

$$\Delta = \left\{ u : \sum_{i=1}^m u_i = 1, u_i \geq 0 \right\}$$

Задача (1) имеет ряд неприятных особенностей с вычислительной точки зрения:

- неединственность (в терминах переменных u или v),
- неустойчивость при малых r .

Неединственность принципиальна и неизбежна в данной задаче, если количество вершин или лучей существенно превышает размерность. Источником этого является сингулярность матриц $X^T X, Y^T Y$.

Вычислительная неустойчивость при малых r - а именно эта ситуация характерна вблизи экстремумов оптимизационных задач, в ходе решения которых используется подзадача (1) - особенно существенна. Например, в методах отделяющих плоскостей [2]

требуется нормированное решение задачи (1): $\bar{x} = \gamma^{-1}x^*$, где γ либо норма x^* , либо какая-либо из компонент вектора x^* и вычислительная неустойчивость приводит к весьма отличающимся от истинного направлениям \bar{x} .

С задачей проекции специалисты в области негладкой оптимизации имеют дело достаточно давно, с первых предложений по разработке аналогов методов сопряженных градиентов [6, 9], методов ускоренного градиентного спуска [5]. Специализированные алгоритмы предлагались в [10, 5]. В программных реализациях пакетных алгоритмов [11] использовались эти алгоритмы с определенными техническими усовершенствованиями, не меняющими существа алгоритмов, в том числе не гарантирующими конечной сходимости в линейном случае. Дальнейший прогресс был связан с использованием методов линейной комлементарности [4] и методов квадратичного программирования [8]. В частности в [8] и последующих алгоритмах этой серии использовались программы известного пакета QPSOLVE, входящего в библиотеку NAG [12]. Используются также программы К. Щитковского [13], К. Кивеля [14].

Возникающие при этом проблемы связаны, во-первых, со сложностью общих программ квадратичного программирования, зачастую по своему объему во много раз превосходящих основную часть алгоритма и, во-вторых, с неизбежной вырожденностью этих задач при работе в составе методов негладкой оптимизации. Как правило, стандартные средства квадратичного программирования не рассчитаны на решения вырожденных (полуопределенных) задач, что, правда, не относится к пакету QPSOLVE, и могут заканчивать работу в неоптимальной точке. Алгоритм симплексного типа, основанный на линейной комлементарности [4], может заикливаться, что и наблюдалось в реальных расчетах и при имеющейся в данный момент реализации не слишком устойчив.

Одним из простейших способов решения задачи (1) являются итеративные методы. Привлекательность этих методов заключается в том, что они используют минимальную информацию о многограннике P в виде минимума или максимума линейных форм на этом многограннике и, следовательно, могут быть применены и в случае бесконечного количества вершин, что соответствует множеству P , заданному системой линейных неравенств.

В настоящей работе мы рассмотрим, возможно, наиболее простой алгоритм, предложенный в [5] и покажем, что он может быть существенно ускорен за счет улучшения процесса выбора начальной точки и введения дополнительных (избыточных) отсечений. На его основе можно также предложить двойственный полношаговый метод, использующий процедуру генерации вершин.

Далее без специального пояснения будем пользоваться следующими обозначениями:

$\text{diam}(P) = \sup_{x, y \in P} \|x - y\|$ — диаметр множества P .

$[x, y] = \text{co}\{x, y\} = \{z = \lambda x + (1 - \lambda)y, \lambda \in [0, 1]\}$ — интервал с концевыми точками x, y .

$\text{ri}(A)$ — относительная внутренность множества A .

§1. Основной алгоритм

В этой работе ниже мы рассматриваем лишь случай ограниченного многогранника P , заданного списком своих вершин: $P = \text{co}\{\hat{x}^1, \hat{x}^2, \dots, \hat{x}^N\}$. Соответственно задача проекции имеет вид:

$$\min \frac{1}{2} \|x\|^2, \quad x = Xu, \quad u \in \Delta, \quad (2)$$

где X — матрица, столбцами которой являются векторы $\hat{x}^i, i = 1, 2, \dots, N$.

Базовая форма итеративного алгоритма проекции может быть представлена в виде следующего псевдокода:

• **Алгоритм**

Инициализация:

$$k = 0, x^0 \in P.$$

• **While** (условия оптимальности не выполнены):

Решить задачу

$$z^{i_{\min}} x^k = \min_{x \in P} x x^k = \min_{i=1,2,\dots,N} z^i x^k.$$

Определить новое приближение

$$x^{k+1} = (1 - \tau_k) x^k + \tau_k z^{i_{\min}},$$

где τ_k решает задачу

$$\min_{0 \leq \tau \leq 1} \|(1 - \tau) x^k + \tau z^{i_{\min}}\|.$$

Изменить счетчик итераций, проверить условия оптимальности и т.п.

End while

• **Конец алгоритма**

Сходимость метода следует из того, что $\|x^k\|$ монотонно убывает и, следовательно,

$$\|x^{k+1}\|^2 - \|x^k\|^2 \rightarrow 0.$$

С другой стороны

$$\|x^{k+1}\|^2 - \|x^k\|^2 = \min_{0 \leq \tau \leq 1} \|(1 - \tau) x^k + \tau z^{i_{\min}}\|^2 =$$

$$\min_{0 \leq \tau \leq 1} \|2\tau(x^k z^{i_{\min}} - \|x^k\|^2) + \tau^2 \|x^k - z^{i_{\min}}\|^2 =$$

$$\min_{0 \leq \tau \leq 1} (2\tau c_k + \tau^2 d_k),$$

где

$$c_k = x^k z^{i_{\min}} - \|x^k\|^2 = \min_{x \in P} x x^k - \|x^k\|^2 \leq 0,$$

$$d_k = \|x^k - z^{i_{\min}}\|^2 \leq \text{diam}^2(P).$$

Отсюда следует, что $c_k \rightarrow 0$ и, следовательно, для предельной точки x^* последовательности $\{x^k\}$ выполнено условие оптимальности

$$\|x^*\|^2 \leq x^* x$$

для всех $x \in P$.

§2. Процедура отсечения

Известно, что в такой простейшей форме алгоритм проекции сходится весьма медленно, и далее мы рассмотрим некоторые способы ускорения его сходимости.

Обозначим

$$P_+ = \{z \in P: zx \geq 0, \forall x \in P\}$$

или в равносильной форме

$$P_+ = \{z \in P: z\hat{x}^i \geq 0, i = 1, 2, \dots, N\}.$$

По определению кратчайший вектор $x^* \in P_+$ так, что задачи (2) и

$$\min_{x \in P_+} \frac{1}{2} \|x\|^2 \quad (3)$$

эквивалентны. Вместе с тем, множество P_+ может быть существенно уже, чем P , особенно для множеств, близких к началу координат.

Представление множества P_+ в виде выпуклой комбинации вершин может быть получено, используя процедуру отсечения, основанную на следующем результате.

ЛЕММА 2.1 Пусть Q — некоторое множество, заданное своими вершинами

$$Q = \text{co}\{\hat{x}^1, \hat{x}^1, \dots, \hat{x}^N\}$$

и A — гиперплоскость

$$A = \{x : ax = \beta\},$$

делящая пространство на два замкнутых полупространства $H_+ = \{x : ax \geq \beta\}$ и $H_- = \{x : ax \leq \beta\}$. Тогда

$$Q_+ = Q \cap H_+ = \text{co}\{[\hat{x}^i, \hat{x}^j] \cap H_+, i, j = 1, 2, \dots, N\} \quad (4)$$

Доказательство. Обозначим

$$\xi(x) = \min m : x = \sum_{k=1}^m \lambda_k \hat{x}^{i_k}.$$

Заметим, что функция $\xi(x)$ субаддитивна: $\xi(x+y) \leq \xi(x) + \xi(y)$.

Покажем индукцией по $m = \xi(x)$, что $x \in Q_+$ влечет

$$x \in Q'_+ = \text{co}\{[\hat{x}^i, \hat{x}^j] \cap H_+, i, j = 1, 2, \dots, N\}.$$

Так как обратное включение тривиально, это докажет лемму.

Если $m = 1$, то $x = \hat{x}^i$ для некоторого i и

$$x \in [\hat{x}^i, \hat{x}^i] = [\hat{x}^i, \hat{x}^i] \cap H_+ \subset Q'_+.$$

Если $m = 2$, то

$$x = \lambda_x \hat{x}^i + (1 - \lambda_x) \hat{x}^j$$

для некоторых \hat{x}^i, \hat{x}^j и, если $x \in Q \cap H_+$, то по крайней мере одна из вершин \hat{x}^i, \hat{x}^j принадлежит H_+ . Пусть для определенности $\hat{x}^i \in H_+$. Тогда $\hat{x}^i \in Q'_+$ и

$$[\hat{x}^i, \hat{x}^j] \cap H_+ = [\hat{x}^i, \bar{x}^j],$$

где

$$\bar{x}^j = \bar{\lambda}_x \hat{x}^i + (1 - \bar{\lambda}_x) \hat{x}^j, \quad (5)$$

причем

$$\bar{\lambda} = \min_{0 \leq \lambda \leq 1} \{ \lambda: \lambda \hat{x}^i + (1 - \lambda) \hat{x}^j \in H_+ \} \leq \lambda_x.$$

Из (5) следует

$$\hat{x}^j = \frac{1}{1 - \bar{\lambda}} \bar{x}^j - \frac{\bar{\lambda}}{1 - \bar{\lambda}} \hat{x}^i$$

и соответственно

$$x = \lambda_x \hat{x}^i + (1 - \lambda_x) \left(\frac{1}{1 - \bar{\lambda}} \bar{x}^j - \frac{\bar{\lambda}}{1 - \bar{\lambda}} \hat{x}^i \right) = \left(\lambda_x - \frac{\bar{\lambda}}{1 - \bar{\lambda}} \right) \hat{x}^i + \frac{1 - \lambda_x}{1 - \bar{\lambda}} \bar{x}^j = \lambda' \hat{x}^i + (1 - \lambda') \bar{x}^j,$$

где $0 \leq \lambda' \leq 1$. Поскольку $\hat{x}^i \in Q'_+$ по предположению, а $\bar{x}^j \in Q'_+$ по построению, то $x \in Q'_+$, что доказывает предположение индукции для $m = 2$.

Предположим, что утверждение доказано для всех x таких, что $\xi(x) \leq m, m \geq 2$ и покажем, что оно верно для x таких, что $\xi(x) = m + 1$.

Пусть $\xi(x) = m + 1$ и пусть после соответствующей перенумерации вершин

$$x = \sum_{i=1}^{m+1} \lambda_i \hat{x}^i = \lambda_{m+1} \hat{x}^{m+1} + (1 - \lambda_{m+1}) z,$$

где

$$z = \frac{1}{(1 - \lambda_{m+1})} \sum_{i=1}^m \lambda_i \hat{x}^i = \sum_{i=1}^m \lambda'_i \hat{x}^i.$$

Очевидно, что $\lambda' = (\lambda'_1, \lambda'_2, \dots, \lambda'_m) \in \Delta$. Возможны следующие варианты:

1. $\hat{x}^{m+1} \in Q_+, z \in Q_+$.
2. $\hat{x}^{m+1} \notin Q_+, z \in Q_+$.
3. $\hat{x}^{m+1} \in Q_+, z \notin Q_+$.

Первый случай не представляет особенного интереса, так как по построению $\xi(z) \leq m$ и, следовательно, по предположению индукции можно считать $z \in Q'_+$. Так как $\xi(\hat{x}^{m+1}) = 1$, то тем более $\hat{x}^{m+1} \in Q'_+$, после чего $x \in Q'_+$ по выпуклости.

Случай 2 более интересен. Пусть $\hat{x}^{m+1} \notin Q_+$. Тогда

$$x = \lambda_{m+1} \hat{x}^{m+1} + \lambda_m \hat{x}^m + \lambda z',$$

где $\hat{x}^m \in Q \cap H_+, \lambda_m > 0$ и $\lambda = 1 - \lambda_m - \lambda_{m+1} \geq 0$.

Если $\lambda = 0$ то

$$x = \lambda_{m+1} \hat{x}^{m+1} + \lambda_m \hat{x}^m, \quad \lambda_{m+1} + \lambda_m = 1,$$

следовательно, $x \in [\hat{x}^{m+1}, \hat{x}^m]$ и $x \in H_+$, т.е.

$$x \in [\hat{x}^{m+1}, \hat{x}^m] \cap H_+ \subset Q'_+.$$

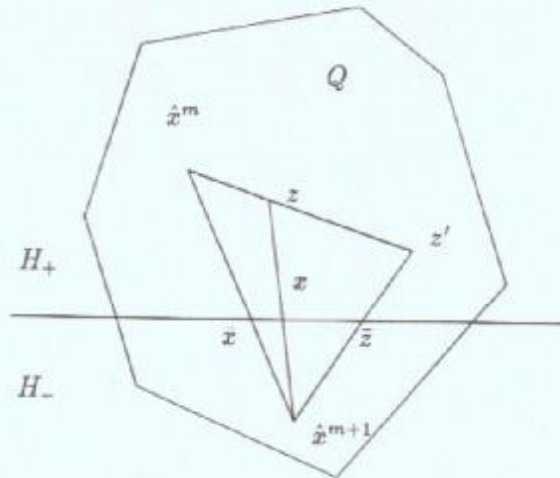


Рисунок 1. Случай $z' \in H_+$

Если $\lambda > 0$, то

$$z' = \sum_{i=1}^{m-1} \frac{\lambda_i}{\lambda} \hat{x}^i = \sum_{i=1}^{m-1} \lambda'_i \hat{x}^i,$$

где $\lambda' \in \Delta$.

Таким образом, $x \in \text{co}\{\hat{x}^{m+1}, \hat{x}^m, z'\} = T$, где $z' \in \text{co}\{\hat{x}^{m-1}, \hat{x}^{m-2}, \dots, \hat{x}^1\}$, т.е. $\xi(z) \leq m-1$. Соответствующая ситуация показана на рис. 1. Если $z' \in H_+$, то представим T в виде объединения двух множеств:

$$T = \text{co}\{\hat{x}^{m+1}, \bar{x}, z\} \cup \text{co}\{\hat{x}^m, \bar{x}, \bar{z}, z'\} = T' \cup T'',$$

где $\bar{x} \in [\hat{x}^{m+1}, \hat{x}^m]$, $\bar{z} \in [\hat{x}^{m+1}, z']$, и выберем $\bar{x} \in [\hat{x}^{m+1}, \hat{x}^m] \cap A$, $z \in [\hat{x}^{m+1}, z'] \cap A$ как показано на том же рис.1.

При этом $T' \subset H_-$ и, значит, $x \in T''$. Заметим, что

$$\xi(\bar{z}) \leq \xi(z') + \xi(\hat{x}^{m+1}) \leq m-1+1 = m$$

и, следовательно, из $\bar{z} \in Q_+$ (по построению) следует $\bar{z} \in Q'_+$. Таким образом, все вершины выпуклого четырехугольника T'' принадлежат Q'_+ и, тогда, $x \in T'' \subset Q'_+$.

Если $z' \notin H_+$, то представим T в виде объединения двух множеств:

$$T = \text{co}\{\hat{x}^m, \bar{x}, \bar{z}\} \cup \text{co}\{\hat{x}^{m+1}, \bar{x}, \bar{z}, z\} = T' \cup T'',$$

где $\bar{x} \in [\hat{x}^{m+1}, \hat{x}^m]$, $z \in [\hat{x}^m, z']$, и выберем $\bar{x} \in [\hat{x}^{m+1}, \hat{x}^m] \cap A$, $\bar{z} \in [\hat{x}^m, z'] \cap A$, как это показано на рис. 2. При этом $x \in T'$ и аналогично предыдущему

$$\xi(z) \leq \xi(z') + \xi(\hat{x}^m) \leq m-1+1 = m,$$

что влечет $\bar{z} \in Q'_+$. Следовательно, все вершины треугольника T' принадлежат Q'_+ и, значит, $x \in T' \subset Q'_+$.

Случай 3 эквивалентен 2 с переменной местами H_+ и H_- .

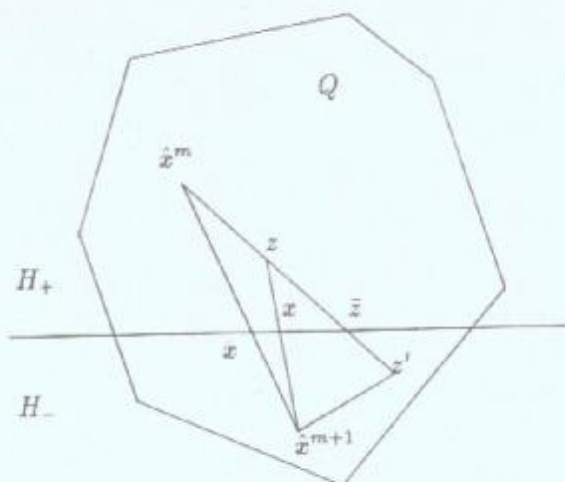


Рисунок 2. Случай $z' \in H_-$.

От представления Q'_+ в виде (4) можно легко перейти к стандартному. Действительно, для любого интервала $[\hat{x}^i, \hat{x}^j]$ пересечение $[\hat{x}^i, \hat{x}^j] \cap H_+$ либо пусто, либо представляет собой некий новый интервал $[\bar{x}^k, \bar{x}^l]$, где $\bar{x}^1, \bar{x}^2, \dots, \bar{x}^N$ — список концевых точек всех таких интервалов.

Тогда

$$Q_+ = \text{co}\{[\bar{x}^k, \bar{x}^l], k, l = 1, 2, \dots, \bar{N}\} = \text{co}\{x^k, k = 1, 2, \dots, N\}.$$

Приведенная лемма даст возможность построить P_+ , последовательно пересекая множество P с полупространствами

$$H_i = \{x : x \hat{x}^i \geq 0\}, i = 1, 2, \dots, N.$$

Эту процедуру будем называть *процедурой отсечения*.

С помощью процедуры отсечения можно получить описание P_+ в виде выпуклой комбинации своих вершин и затем применить для решения задачи (3) описанные выше алгоритмы.

§3. Двойственный метод

Однако, существует и другая возможность более эффективно использовать свойства множества P_+ . Рассмотрим задачу, двойственную (2), которая, как легко убедиться, имеет вид:

$$\min_z \left\{ \frac{1}{2} \|z\|^2 + \max_{i=1,2,\dots,N} z \hat{x}^i \right\}, \quad (6)$$

где $\hat{x}^i, i = 1, 2, \dots, N$ — вершины P . Интересно отметить, что решение прямой (3) или (2) и двойственной (6) задач совпадают с точностью до знака:

$$z^* = -x^*,$$

что позволяет решать только (6).

Двойственная задача может быть рассмотрена как общая задача квадратичного программирования

$$\min_z \left\{ \frac{1}{2} \|z\|^2 + w \right\}$$

$$w - z\hat{x}^i \leq 0, \quad i = 1, 2, \dots, N \quad (7)$$

и для ее решения применены общие методы. Однако в практике недифференцируемой оптимизации число N и тем более N_+ , весьма велико, а общие методы весьма критичны к количеству ограничений. Хотя применительно к задаче (7) и сообщалось [15] об успешных вычислительных экспериментах, применение общих процедур квадратичного программирования требует значительных вычислительных ресурсов.

С точки зрения негладкой оптимизации функция

$$f(z) = \frac{1}{2} \|z\|^2 + \max_{i=1,2,\dots,N} z\hat{x}^i = \frac{1}{2} \|z\|^2 + \max_{x \in P} zx \quad (8)$$

представляет собой конечную негладкую выпуклую функцию, субдифференциал которой имеет вид

$$\partial f(z) = z + \text{co}\{\hat{x}^i, i \in I(z)\},$$

где $I(z)$ - множество таких индексов i , что

$$z\hat{x}^i = \max_{i=1,2,\dots,N} z\hat{x}^i.$$

Для решения этой задачи можно применить простейшие градиентные методы, которые оказываются весьма близки к прямому алгоритму, рассмотренному выше.

Действительно, градиентный метод для задачи (8) имеет вид:

$$z^{k+1} = z^k - \lambda_k(z^k + \hat{x}^{i_k}) = (1 - \lambda_k)z^k + \lambda_k \hat{x}^{i_k}, \quad (9)$$

где \hat{x}^{i_k} максимизирует $z^k x^i$:

$$\hat{x}^{i_k} z^k = \max_{i=1,2,\dots,N} x^i z^k.$$

Градиентный метод (9) совпадает с прямым методом, рассмотренным выше, за исключением ограничения на шаговый множитель λ_k , который в данном случае не обязан принадлежать интервалу $[0, 1]$.

Для определения шагового множителя λ_k , могут быть применены различные подходы, вплоть до полношаговых градиентных методов, которые, благодаря простой структуре целевой функции, будут сходиться.

Степень негладкости функции (8) определяется максимальным углом между субградиентами $f(z)$ или, что эквивалентно, максимальным углом между векторами из P . Применение процедуры отсечения, описанной выше, позволяет сделать этот угол менее $\pi/2$ и за счет этого применить полношаговые методы с вычислением

$$\min_{\lambda \geq 0} f(z^k - \lambda(z^k + \hat{x}^{i_k})) = f(z^k - \lambda_k(z^k + \hat{x}^{i_k})).$$

В идеале, если процедура отсечения приводит к тому, что $P_+ = \{x^*\}$, где x^* решает (2), то

$$f(z) = \frac{1}{2} \|z\|^2 + zx^* = \frac{1}{2} \|z + x^*\|^2 - \frac{1}{2} \|x^*\|^2$$

и один единственный шаг метода (9) приводит к решению.

При угле между субградиентами менее $\pi/2$ направление анти-субградиента является релаксационным и полношаговая процедура ведет к существенному приближению к экстремуму. Следует отметить, что при этом необходимо использовать специальные процедуры одномерной минимизации типа [16], учитывающие негладкость целевой функции.

Кроме этого заметим, что приближенное решение \bar{z} задачи (6) мажорирует снизу расстояние до многогранника P :

$$f(\bar{z}) = \frac{1}{2} \|\bar{z}\|^2 + \max_{i=1,2,\dots,N} \bar{z} \hat{x}^i \geq \frac{1}{2} \|z^*\|^2 + \max_{i=1,2,\dots,N} z^* \hat{x}^i = -\frac{1}{2} \|x^*\|^2$$

и поэтому может быть использовано для глубоких отсечений множества P полупространствами

$$H_k^\beta = \{x: x \hat{x}^k \geq \beta = -2f(\bar{z})\}, \quad k = 1, 2, \dots, N,$$

что повышает эффективность метода.

§4. Генерация вершин

Как уже упоминалось, прямое применение процедуры отсечения затрудняется быстрым ростом числа точек, образующих P_+ . Хотя этот рост и во многом фиктивен, так как значительная доля генерируемых точек не является вершинами P_+ , точные оценки количества вершин после сечения многогранника гиперплоскостью автору неизвестны. Непосредственная же проверка всех генерируемых точек с отбором крайних достаточно трудоемка, но, как оказывается, в ней нет и необходимости.

Альтернативой этому подходу может быть идея генерации вершин P_+ в сочетании с методами недифференцируемой оптимизации.

Заметим, что нетривиальная часть субградиента $f(z)$ определяется решением задачи

$$\max_{i=1,2,\dots,N} z \hat{x}^i = \max_{x \in P} z x \quad (10)$$

для исходной задачи и

$$\max_{i=1,2,\dots,N} z \hat{x}_+^i = \max_{x \in P_+} z x$$

для модифицированной задачи. Последняя отличается от (10) наличием дополнительных линейных ограничений и может быть сформулирована как задача линейного программирования

$$\max z x$$

$$x = \sum_{i=1}^N \lambda_i \hat{x}^i, \lambda \in \Delta, \quad (11)$$

$$x \hat{x}^i \geq 0, i = 1, 2, \dots, N$$

или, если применяются глубокие отсечения,

$$\max z x$$

$$x = \sum_{i=1}^N \lambda_i \hat{x}^i, \lambda \in \Delta,$$

$$x \hat{x}^i \geq \beta, i = 1, 2, \dots, N,$$

для решения которых существуют высокоэффективные методы.

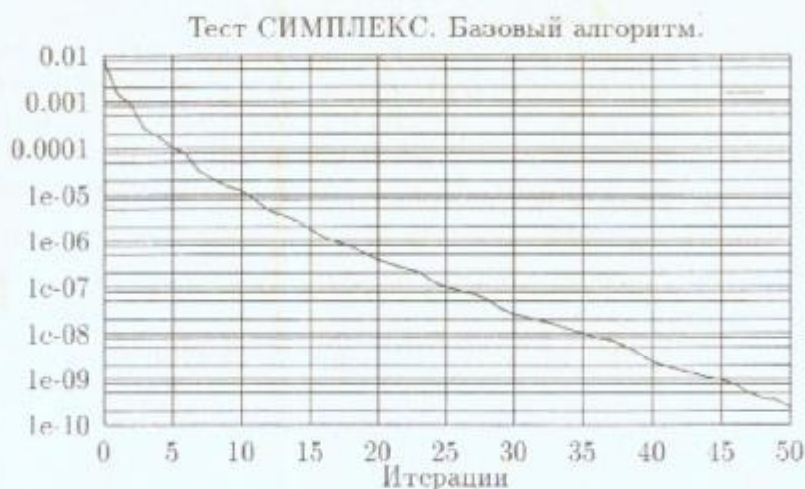


Рисунок 3. Тест СИМПЛЕКС. Основной алгоритм. Размерность $n = 10$

§5. Численные эксперименты

В этом разделе описаны вычислительные эксперименты с одним из характерных тестов.

Тест СИМПЛЕКС. Этот тест содержит набор $n+1$ векторов $n+1$ -мерного пространства E^{n+1} , $P = \{z^i, i = 1, 2, \dots, n+1\}$, причем первые n компонент этих векторов являются вершинами стандартного n -мерного симплекса, сдвинутого так, чтобы содержать начало координат, а последняя координата каждого вектора одинакова: $z_{n+1}^i = \epsilon > 0$.

Таким образом, $z^i = (\bar{z}^i, \epsilon)$, где $\bar{z}^i \in E^n, i = 1, 2, \dots, n+1$,

$$\bar{z}^i = \begin{cases} e^i - a, & i = 1, 2, \dots, n, \\ a, & i = n+1, \end{cases}$$

где e^i — i -й орт пространства E^n , а a — вектор сдвига с положительными коэффициентами, не превосходящими $1/n$.

Простейшая реализация основного алгоритма, рассмотренная выше, при размерности пространства переменных n и количестве вершин многогранника m требует $mn + 3n$ умножений и $(m+2)(n-1) + n$ сложений на итерацию.

Учитывая то, что базовый алгоритм, не отличаясь большой эффективностью, требует весьма большого числа итераций, суммарные затраты могли бы быть уменьшены за счет предварительного вычисления матрицы попарных скалярных произведений. Последнее увеличивает, однако, затраты памяти на величину, пропорциональную m^2 . При большом количестве векторов, образующих многогранник P , этот фактор может стать определяющим.

Анализируя вычислительные эксперименты можно отметить, что сходимость алгоритма замедляется с уменьшением ϵ , служащим некоторым параметром, который определяет вырожденность задачи — при уменьшении ϵ начало координат приближается к многограннику, что ухудшает соотношение между длиной минимального вектора и диаметром многогранника.

Подробнее характер сходимости основного алгоритма показан на рис. 3 для одного из тестов, соответствующему $\epsilon = 0.00001$. На этом рисунке показана сходимость к нулю расстояния до известного в этом тесте кратчайшего вектора, при этом хорошо виден линейный характер сходимости.

На рис. 3 показаны результаты применения двойственного метода с полношаговой одномерной минимизацией для той же задачи. В этом случае было проведено предварительное отсечение с помощью процедуры раздела 3, уже давшее достаточно хорошее приближение к решению задачи. Последующее применение двойственного градиентного алгоритма демонстрирует, как видно из графика, более быструю сходимость, чем линейную. Следует также отметить существенное уменьшение числа итераций.

Данные предварительно отмасштабированы по оси y умножением на $\exp(50)$, что соответствует итоговой реальной точности порядка 10^{-14} .

Список литературы

1. Демьянов В.Ф., Малозёмов В.И. Введение в минимакс. М.: Наука, 1978.
2. Нурминский Е.А. Численные методы выпуклой оптимизации. М.: Наука, 1991.
3. Ржевский С.В. E-субградиентный метод решения задачи выпуклого программирования // Ж. вычисл. матем. и матем. физики. 1981. Т. 21, N 5. С. 1126-1132.
4. Нурминский Е.А. Предварительные исключения в алгоритмах решения комбинентарных задач. Кибернетика. 1986. N 5. С. 121-122.
5. Вопросы теории и элементы программного обеспечения минимаксных задач / Демьянов В.Ф., Малозёмов В.И. Л.: Изд-во ЛГУ, 1977.
6. Wolfe P. A method of conjugate subgradients for minimizing nondifferentiable functions // Mathematical Programming Study. . N 3. P. 145-173.
7. Lemarechal C., Nemirovski A., Nesterov Yu. New variants of bundle methods. Rep. 1508. INRIA, Paris, France, 1991.
8. Schramm H., Zowe J. A version of the bundle idea for minimizing a nonsmooth function: idea, convergence analysis, numerical results, SIAM J. Optimization // 1992. V. 2, N 1. P. 121-152.
9. C. Lemarechal Constructing bundle methods for convex optimization, Mathematics for Optimization / Hiriart-Urruty J.-B.). North-Holland. 1986. P. 201-240.
10. Wolfe P. Finding the nearest point in a polytope Mathematical programming // 1976. V. 11, N 2. P. 128-149.
11. C. Lemarechal Nondifferentiable optimization / A.H.G. Rinnooy Kan, G.L. Nemhauser and M.J. Todd, editors, Handbook in OR and MS, Vol. 1, 529-572, North-Holland, Amsterdam, 1989.
12. NAG Fortran library reference manual, 1981. Numerical Algorithms Group Ltd, Oxford, England.

13. *K. Schittkowski, J. Stoer* A factorization method for the solution of constrained linear least squares problems allowing data changes // Numer. Math. 31, 431-463.
14. *K. Kiwiel* Methods of descent for nondifferentiable optimization // Lecture Notes in Mathematics, Springer-Verlag, 1133, 1985.
15. *A. Dax* On computational aspects of bounded linear least squares problems // ACM Transac. on Math. Software. 1991. V. 17, N 1. P. 64-73.
16. *E.A. Nurminski* Quadratically convergent linesearch algorithm for piece-wise smooth convex optimization // Optimization Methods and Software, to appear.
17. *H. Markovitch* Optimal Portfolio Selection // J. of Finances. 1952. V. 7, N 1. P. 77-91.

Ин-т прикладной математики
ДВО РАН

Поступило в редакцию
30.XI.1994