ТЕОРІЯ ОПТИМАЛЬНИХ РІШЕНЬ

УДК 519.8

Н.Г. ЖУРБЕНКО, Э.И. НЕНАХОВ

ОБ ОДНОМ АЛГОРИТМЕ ε-СУБГРАДИЕНТНОГО ТИПА МИНИМИЗАЦИИ ВЫПУКЛОЙ ФУНКЦИИ[1]

Предложен алгоритм построения двух отсекающих плоскостей, локализующих множество решений задачи минимизации выпуклой функции. Обеспечивается сколь угодно малый угол между отсекающими плоскостями. Алгоритм основан на процедуре одномерного спуска. Приведены результаты численных экспериментов.

Введение. В работе [1] предложен ε-субградиентный алгоритм решения задачи ε-минимизации ограниченной снизу выпуклой функции $f(x)$ в $n-$мерном евклидовом пространстве R^n:

$$f^* = \min\{ f(x) | x \in R^n \}.$$

Для краткости этот алгоритм будем называть α(ε)-алгоритмом. Он обеспечивает решение задачи минимизации выпуклых функций с заданной точностью.

Для заданного числа $\varepsilon \geq 0$ введем ε-оптимальное множество $X^*(\varepsilon)$:

$$X^*(\varepsilon) = \{x \in R^n \mid f(x) \leq f^* + \bar{\varepsilon}\}. \qquad (1)$$

Произвольную точку $\tilde{x} \in X^*(\bar{\varepsilon})$ и значение функции $\tilde{f} = f(\tilde{x})$ будем называть решением задачи ε-оптимизации (ε-решением).

Пусть задан шар $D(z,R)$ начальной локализации ε-решения: $z \in R^n -$ центр шара, $R > 0 -$ его радиус. Будем предполагать, что $D(z,R) \cap X^*(\varepsilon) \neq \varnothing$.

Будем использовать несколько отличное от классического определение ε-субградиента [2]. Вектор $g \in R^n$ называется условным $(\varepsilon, \tilde{f})$-субградиентом функции $f(x)$ в точке z, если для $\forall x \in D(z,R)$ выполняется неравенство

---

[1] Работа выполнена при частичной финансовой поддержке Украинского научно-технологического центра (грант № 1625)

$$f(x) \geq \tilde{f} + (g, x - z) - \varepsilon, \qquad (2)$$

где $f(z) \geq \tilde{f} \geq f^*; \varepsilon \in R^1$. Обычное классическое определение ε-субградиента [3] соответствует определению $(\varepsilon, \tilde{f})$-субградиента (2) для $\tilde{f} = f(z)$; $\varepsilon \geq 0$.

Обобщенный градиент функции $f(x)$ в точке z в классическом смысле [4] является $(0, f(z))$–субградиентом.

$G(\varepsilon, \tilde{f}, z), \partial f(z)$ будет обозначать множество $(\varepsilon, \tilde{f})$ – субградиентов и множество обобщенных градиентов в точке z соответственно. В дальнейшем предполагается, что имеются алгоритмы вычисления $f(z)$ и $g(z) \in \partial f(z)$ в произвольной точке z.

Пусть $g \in G(\varepsilon_1, \tilde{f}_1, z_1)$; $f^* \leq f_2 \leq f(z_2)$, тогда $g \in G(\varepsilon_2, \tilde{f}_2, z_2)$, где

$$\varepsilon_2 = \varepsilon_1 + \tilde{f}_2 - \tilde{f}_1 - (g, z_2 - z_1). \qquad (3)$$

Таким образом, $(\varepsilon, \tilde{f})$–субградиент в некоторой точке z_1 является $(\varepsilon, \tilde{f})$– субградиентом в любой другой точке z_2. Формула (3) определяет правило пересчета параметров $(\varepsilon, \tilde{f})$–субградиента. $G(\varepsilon, \tilde{f}, z)$ по своим свойствам аналогично множеству $\partial f(z)$ (выпуклость, замкнутость и т. д.).

Утверждение 1. Для заданных точки z, числа $\tau \geq 0$, направления спуска η $(|\eta|=1)$ и $\varepsilon > 0$ можно гарантировать вычисление такого ε–субградиента $g \in G(\varepsilon, \tilde{f}, z)$, что $\varepsilon \leq \varepsilon$ и $(g, \eta) \geq -\tau |g|$.

В силу ограничения объема работы, доказательство утверждения и описание соответствующего алгоритма не приводится. Отметим, что алгоритм аналогичен известным процедурам генерации субградиентов множеств [3, 5].

Пусть $\tilde{X}(\varepsilon, \tilde{f}) = \{x \in R^n \mid f(x) \leq \tilde{f} - \bar{\varepsilon}\}$.

Отметим, что

$\tilde{X}(\varepsilon, \tilde{f}) = \varnothing \Rightarrow \{ \tilde{f}$ является ε – решением $\}$;

$\tilde{f} \geq f^* + 2\varepsilon \Rightarrow X^*(\varepsilon) \subset \tilde{X}(\varepsilon, \tilde{f})$.

Введем обозначения: $P(z, \eta) = \{x \in R^n \mid (x - z, \eta) = 0\}$ – плоскость, проходящая через точку z с нормалью $\eta \in R^n$, $|\eta| = 1$; $P^+(z, \eta) = \{x \in R^n \mid (x - z, \eta) \geq 0\}$ – полупространство, определяемое плоскостью $P(z, \eta)$.

Пусть

$$g \in G(\varepsilon, \tilde{f}, z); \; h = (\varepsilon - \varepsilon)/|g|; \; \tilde{z} = z - hg/|g|. \qquad (4)$$

Тогда $\tilde{X}(\varepsilon, \tilde{f}) \subset P^+(z - \tilde{z}, -g/|g|)$. Этот факт соответствует обычному построению отсекающих плоскостей "со сдвигом".

Основной операцией $\alpha(\varepsilon)$-алгоритма является процедура (Localization's Algorithm) построения двух отсекающих плоскостей $P(z, \eta_1)$, $P(z, \eta_2)$. При этом угол φ между векторами η_1, η_2 должен быть достаточно велик, точнее: для заданного (сколь угодно малого) числа $\delta > 0$ выполнено неравенство $(\eta_1, \eta_2) = \cos(\varphi) \le -1 + \delta$. В работе [1] Localization's Algorithm использует операцию вычисления наименьшего по норме вектора выпуклой оболочки конечного множества векторов. Для решения такой простейшей задачи квадратичного программирования имеются достаточно эффективные алгоритмы (например [6]). Однако для больших размерностей трудоемкость такой операции становится существенной. Приведенный ниже Localization's Algorithm не использует указанную операцию и имеет существенно меньшую трудоемкость.

Localization's Algorithm.

Итерация 1 (инициализация).

Вычисляем $g_1 \in \partial f(z)$. Если $g_1 = 0$, то останов (задача минимизации $f(x)$ решена), иначе: $G_1 = \{g_1\}$; $e_1 = -g_1 / |g_1|$; $E_1 = \{e_1\}$; $p_1 = e_1$.

Пусть на итерации $k\,(k = 1,2,\ldots)$ получены множества $E_k = \{e_1, e_2, \ldots, e_k\}$, $G_k = \{g_1, g_2, \ldots, g_k\}$ и вектор

$$p_k = \frac{1}{k}\sum_{l=1}^{k} e_l \ne 0. \tag{5}$$

Итерация $(k+1)$ $(k = 1,2\ldots)$.

Используя процедуру одномерной минимизации по направлению $p_k / |p_k|$ вычисляем ε–субградиент $g_{k+1} \in G(\varepsilon, \tilde{f}, z)$, для которого выполнены условия (см. утверждение 1):

$$\varepsilon \le \varepsilon, \quad (g_{k+1}, p_k) \ge -|g_{k+1}| / k. \tag{6}$$

Если $g_{k+1} = 0$, то останов – задача ε-минимизации решена (заметим, что в результате одномерной минимизации значение $\tilde{f}$ может уменьшиться, в этом случае производится пересчет $(\varepsilon, \tilde{f})$ параметров субградиентов множества G_k согласно (3)).

Полагаем:

$$G_{k+1} = G_k \cup g_{k+1}; \; e_{k+1} = -g_{k+1} / |g_{k+1}|; \; E_{k+1} = E_k \cup e_{k+1};$$

$$p_{k+1} = \frac{1}{k+1}\sum_{l=1}^{k+1} e_l. \tag{7}$$

Если $p_{k+1} = 0$, то останов (задача минимизации $f(x)$ решена).

Так как p_{k+1} внутренняя точка выпуклой оболочки множества E_{k+1}, то согласно теоремы Каратеодори справедливо представление

$$p_{k+1} = \sum_{i=1}^{m} \lambda_i e_{l_i}, \tag{8}$$

где $l_i \in \{1,2,\ldots,k+1\}$; $n+1 \ge m \ge 2$. Пусть коэффициенты λ_i в (8) упорядочены: $\lambda_1 \ge \lambda_2 \ge \ldots \lambda_m > 0$.

Полагаем $\eta_1 = e_{l_1}$, $\eta_2 = \xi / |\xi|$, где $\xi = \sum_{i=2}^{m} \frac{\lambda_i}{1-\lambda_1} e_{l_i}$; $\cos(\varphi_{k+1}) = (\eta_1, \eta_2)$.

Если $\cos(\varphi_{k+1}) \le -1+\delta$, то останов, иначе переход на следующую итерацию.

Замечание 1. Задача определения коэффициентов λ_i представления (8) - стандартная процедура линейного программирования.

Замечание 2. Процедура одномерной минимизации выпуклой функции по направлению (7) использовалась в [6].

Утверждение 2. Алгоритм Localization's Algorithm конечен. При останове алгоритма выполняется одно из следующих условий:

задача ε–минимизации $f(x)$ решена;

получены такие отсекающие плоскости $P(z,\eta_1), P(z,\eta_2)$, что $(\eta_1,\eta_2) = \cos(\varphi_{k+1}) \le -1+\delta$.

Справедлива следующая оценка значения $|p_{k+1}|$

$$|p_{k+1}| \le \sqrt{\frac{3}{k+1}};\ k = 0,1,2,\ldots\ . \tag{9}$$

Доказательство. Первая часть утверждения **2** следует из условий останова алгоритма. Для этого достаточно заметить, что $0 \in Co\{E_{k+1}\} \Leftrightarrow 0 \in Co\{G_{k+1}\}$.

Докажем конечность алгоритма. Пусть на шагах 1, 2, 3 итерации критерий останова не выполняется.

Докажем оценку (9). Учитывая определение p_{k+1} (7) и (6), получаем следующее рекуррентное соотношение

$$(k+1)^2 p_{k+1}^2 = k^2 p_k^2 + 2k(e_{k+1}, p_k) + e_{k+1}^2 = k^2 p_k^2 - 2k(g_{k+1}, p_k) | g_{k+1} | + e_{k+1}^2 \le$$

$$\le k^2 p_k^2 - 2k(g_{k+1}, p_k / | p_k |) | p_k || g_{k+1} | + 1 \le k^2 p_k^2 + 3 \le 1 + 3k < 3(k+1).$$

Отсюда следует (9).

Из определения вектора ξ следует, что $p_{k+1} = \lambda_1 e_1 + (1-\lambda_1)\xi$. Отсюда нетрудно получить, что

$$\cos(\varphi_{k+1}) = (\eta_1, \eta_2) = \frac{(p_{k+}, e_1) - \lambda_1}{\sqrt{p_{k+1}^2 - 2\lambda_1 (p_{k+}, e_1) + \lambda_1^2}}. \tag{10}$$

Так как $\sum_{i=1}^{m} \lambda_i = 1; \lambda_i > 0; m \le n+1$, то

$$\lambda_1 \ge 1/(n+1). \tag{11}$$

Из оценки (9) следует, что $\lim\limits_{k\to\infty} p_{k+1}^2 = 0.$ Но тогда из (10-11) получаем $\lim\limits_{k\to\infty} \cos(\varphi_{k+1}) = -1,$ что и доказывает конечность алгоритма.

Качественная интерпретация алгоритма [1] состоит в следующем. Алгоритм относится к классу методов с преобразованием пространства. На каждой итерации алгоритма преобразование пространства состоит в применении операторов растяжения пространства по ортогональным направлениям. Параметры ортогональных преобразований (коэффициенты и направления растяжения) определяются на основе построения множеств локализации ε-решений по информации, получаемой в результате применения процедуры одномерной минимизации по направлениям. На каждой итерации обеспечивается уменьшение объема локализации $\bar{\varepsilon}$-решения не менее чем в qVolum раз (параметр алгоритма). Параметр qVolum определяет минимальное значение используемых коэффициентов растяжения пространства. Согласно полученным в [1] оценкам, для обеспечения существенного уменьшения объема локализации ($\mathrm{qVolum} \approx 0.7$ раз) достаточно применения n^2 процедур одномерной минимизации. Значению параметра $\mathrm{qVolum} \approx 0.7$ соответствует значение коэффициента растяжения ≈ 2.4. Таким образом, согласно этой оценки, для такого коэффициента уменьшения объема локализации трудоемкость одной итерации алгоритма может быть достаточно большой. Однако эта оценка достаточно груба. Кроме того, отметим, что поведение алгоритма в отличие от метода эллипсоидов [8] существенно зависит от конкретных характеристик минимизируемой функции.

Приведем результаты численного исследования эффективности разработанной модификации α(ε)-алгоритма.

В качестве тестовых задач рассматривались задачи минимизации двух следующих функций:

$$f_1(x) = \sum_{i=1}^{n} \rho_n^{i-1} x_i^2, \qquad f_2(x) = \sum_{i=1}^{n} \rho_n^{i-1} \left| x_i \right|,$$

где параметр ρ_n выбирался в зависимости от размерности задачи n по формуле $\rho_n = 10^{6/(n-1)}$. Таким образом, степень вытянутости линий уровня («овражности») функций определяется значением параметра $\rho_n^{n-1} = 10^6$, она одинакова для всех функций независимо от числа переменных. Начальная точка $x_i = 1.0, i = 1,2,...n.$ Параметр точности решения по функционалу $\bar{\varepsilon} = 10^{-6}$. Результаты решения тестовых задач минимизации функций $f_1(x)$, $f_2(x)$ приведены в табл. 1 и 2 соответственно, где приняты следующие обозначения:

nVarbl – число переменных;

qVolum – значение параметра уменьшения объема локализации

$\bar{\varepsilon}$-решения на итерации ;

nIter – число итераций;

nLStep_Avrg – среднее число применения алгоритма одномерной минимизации на одной итерации;

Alph_Avrg - среднее значение коэффициента растяжения пространства.

ТАБЛИЦА 1

NVarbl	Qvolum	nIter	nLStep_Avrg	Alph_Avrg
5	0.7	36	2.25	3.624
10	0.99	107	1.364	1.661
10	0.7	56	3.214	3.035
20	0.99	195	1.297	1.621
20	0.7	86	3.686	2.724
30	0.99	283	1.254	1.606
30	0.7	109	3.872	2.848
40	0.99	360	1.236	1.611
40	0.7	134	4.127	2.761
50	0.99	435	1.205	1.6
50	0.7	153	4.255	2.759
100	0.99	711	1.136	1.592
100	0.7	243	4.407	2.744

ТАБЛИЦА 2

Nvarbl	Qvolum	nIter	nLStep_Avrg	Alph_Avrg
5	0.99	142	1.204	2.524
5	0.7	67	2.179	4.748
10	0.99	413	1.165	1.855
10	0.7	133	3.015	3.38
20	0.99	1274	1.095	1.552
20	0.7	289	4.173	2.842
30	0.99	2164	1.081	1.52
30	0.7	445	5.231	2.69
40	0.99	1930	1.09	1.508
40	0.7	374	6.035	2.607
50	0.99	2594	1.076	1.465
50	0.7	455	6.868	2.601
100	0.9	4062	2.597	1.755
100	0.7	1559	9.201	2.547

Заключение. Результаты численных экспериментов показывают достаточно высокую эффективность $\alpha(\varepsilon)$-алгоритма и выявляют следующие интересные его особенности. Среднее число применения алгоритма одномерной минимизации на одной итерации оказывается малым в сравнении с приведенной гарантированной его оценкой (n^2). Даже если параметр уменьшения объема локализации $\bar{\varepsilon}$-решения на итерации задается малым (0.99), тем не менее, среднее значение коэффициента растяжения пространства оказывается существенно больше единицы.

М.Г. Журбенко, Е.І. Ненахов

ПРО ОДИН АЛГОРИТМ ε-СУБГРАДІЄНТНОГО ТИПУ МІНІМІЗАЦІЇ ОПУКЛОЇ ФУНКЦІЇ

Запропоновано алгоритм побудови двох відтинаючих площин, які локалізують множину розв'язків задачі мінімізації опуклої функції. Алгоритм гарантує досягнення задовільно малого кута між відтинаючими площинами. Він заснований на процедурі одномірного спуску. Наведені результати обчислювальних експериментів.

N.G. Gurbenko, E.I. Nenakhov

ON ONE ε-SUBGRADIENT ALGORITHM OF CONVEX FUNCTION MINIMIZATION

An algorithm of constructing two cutting planes localizing the set of solutions to the problem of convex function ε-minimum. This algorithm provides for as small as needed angle between cutting planes. It is based on the procedure of one-dimensional descent. The results of numerical experiments are given.

1. *Журбенко Н.Г.* Об одном ε-субградиентном алгоритме минимизации // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 2002. – С. 111–118.
2. *Журбенко Н.Г.* Об одном классе методов минимизации с преобразованием пространства // Методы решения экстремальных задач. – Киев: Ин-т кибернетики им. В.М.Глушкова НАН Украины, 1996. – С. 68–80.
3. *Lemarechal C., Mifflin K.* Nonsmooth Optimization. Oxford: Pergamon Press, 1978. – 180 p.
4. *Шор Н.З.* Методы минимизации недифференцируемых функций и их применение. – Киев: Наук.думка, 1979. – 200 с.
5. *Ржевский С.В.* Монотонные методы выпуклого программирования. – Киев: Наук.думка, 1993. – 319 с.
6. *Wolfe P.* Finding the nearest point in a polytope // Math. Progr. – 1976. – Vol. 11. – № 2. – P. 128 – 149.
7. *Ненахов Э.И.* Об одном методе отсечений для минимизации выпуклых функций // Теория оптимальных решений. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 1992. – С. 14–19.
8. *Немировский А.С., Юдин Д.Б.* Сложность задач и эффективность методов оптимизации. – М. Наука, 1979. – 384 с.

Получено 15.07.2004