Ускоренные по Шору модификации метода Поляка для овражных функций

Стецюк П.И.
stetsyukp@gmail.com

Институт кибернетики им. В.М. Глушкова НАН Украины, Киев

Молодежная научная школа-семинар
"Дискретные модели и методы принятия решений"
21-23 июня 2013, г.Новосибирск (Академгородок)

Шор Наум Зуселевич (1937 – 2006)

Родился 1 января 1937 года в Киеве. Окончил Киевский университет имени Тараса Шевченко в 1958 году. Работал в Институте кибернетики НАН Украины с 1958 по 2006 гг. Академик НАН Украины, профессор. Автор 10 монографий и более 200 статей. Лауреат Государственных премий УССР (1973), СССР(1981), Украины (1993, 2000), премии им. В.М. Глушкова НАН Украины (1987), премии им. В.С. Михалевича НАН Украины (1997).

О трех книгах избранных трудов Н.З. Шора

- Шор Н.З. Методы недифференцируемой оптимизации и сложные экстремальные задачи. – Кишинэу, Эврика, 2008

  http://www.aticmd.md/wp-content/uploads/2011/07/Shor-Book1.pdf

- Шор Н.З. Методы минимизации негладких функций и матричные задачи оптимизации. – Кишинэу, Эврика, 2009

  http://www.aticmd.md/wp-content/uploads/2011/07/Shor-Book2.pdf

- Шор Н.З. Алгоритмы последовательной и негладкой оптимизации. – Кишинэу, Эврика, 2012

  http://www.aticmd.md/wp-content/uploads/2011/07/Shor_book_12-03-2012.pdf

3-я конференция „Математическое моделирование, оптимизация и информационные технологии“, посвященная 75-летию со дня рождения Наума Зуселевича Шора, 19 – 23 марта 2012 года http://www.aticmd.md/stiinta/conf_mmoti-2012/

Статья к юбилею Н.З. Шора (январь, 2012)

75-летию со дня рождения Н.З. Шора посвящена статья

Сергиенко И.В., Стецюк П.И.

О трех научных идеях Н.З. Шора // Кибернетика и системный анализ. – 2012, № 1. – С. 4–22.

В статье описаны три центральные идеи Н.З. Шора:

обобщенный градиентный спуск (1962), использование линейных неортогональных преобразований пространства для улучшения обусловленности овражных функций (1969), двойственный подход к получению и уточнению оценок целевой функции в невыпуклых квадратичных моделях (1985). Приведены методы и алгоритмы, разработанные на их основе в Институте кибернетики им. В.М. Глушкова НАН Украины.

Content

Content

Б.Т. Поляк „Введение в оптимизацию“ (1983)

Борис Теодорович говорит:

на стр. 128

„Основные алгоритмы минимизации гладких функций – градиентный и Ньютона – были построены на использовании линейной и квадратичной аппроксимации функции, задаваемой первыми членами ряда Тейлора. Однако, для недифференцируемой функции эта идея неприменима – такая функция не может быть хорошо аппроксимирована ни линейной, ни квадратичной функциями.“

Б.Т. Поляк „Введение в оптимизацию“ (1983)

на стр. 129

„Поэтому разработка методов минимизации негладких функций требует привлечения новых идей. Одна из них, принадлежащая Н.З. Шору, выглядит несколько неожиданно. Пишется прямой аналог градиентного метода с заменой градиента на произвольный субградиент $g_f(x)$ функции $f(x)$:

$$x_{k+1} = x_k - \gamma_k g_f(x_k). \tag{3}$$

... Значения функции в методе (3) не могут убывать монотонно. Оказывается, однако, что при этом монотонно убывает другая функция – расстояние до точки минимума, и в этом то заключается основная идея субградиентного метода (3).“

Выпуклая функция и ее субградиент

Пусть $f(x)$ – выпуклая функция, $x \in E^n$, X^* – множество минимумов, $x^* \in X^*$ – точка минимума; $f^* = \inf f(x)$.

Определение 1. Субградиентом функции $f(x)$ в точке x_0 называется вектор $g_f(x_0)$, удовлетворяющий неравенству

$$f(x) - f(x_0) \geq (g_f(x_0), x - x_0) \quad \text{для всех} \quad x \in E^n \quad . \qquad (1)$$

Здесь $(\cdot, \cdot)$ – скалярное произведение векторов из E^n.

Примечание. Если $f(x)$ является непрерывно дифференцируемой в точке x_0, то вектор $g_f(x_0)$ определяется однозначно и совпадает с $\nabla f(x_0)$ – градиентом функции $f(x)$ в точке x_0. В общем случае (для функций с разрывным градиентом) вектор $g_f(x_0)$ определяется неоднозначно.

Геометрическое свойство субградиента

Из неравенства (1) следует, что если $f(x) < f(x_0)$, то субградиент $g_f(x_0)$ удовлетворяет неравенству

$$(-g_f(x_0), x - x_0) > 0. \tag{2}$$

Формула (2) означает, что антисубградиент в точке x_0 образует острый угол с направлением из точки x_0 в любую точку x с меньшим значением $f(x)$.

Следствие. Если X^* непусто и $x_0 \notin X^*$, то из точки x_0 в направлении $-g_f(x_0)$ существует ненулевой длины шаг, при котором расстояние до множества X^* убывает.

Этот факт лежит в основе субградиентного метода минимизации негладких функций. То, что эта идея принадлежит Н.З.Шору отметил Б.Т.Поляк (1983, Введение в оптимизацию, с. 128-129).

Субградиентный метод (определение)

Определение 2. Субградиентным методом называется

процедура построения последовательности $\{x_k\}_{k=0}^{\infty}$ по следующему правилу:

$$x_{k+1} = x_k - h_k \frac{g_f(x_k)}{\|g_f(x_k)\|}, \quad k = 0, 1, 2, \ldots, \tag{3}$$

где x_0 – начальное приближение, h_k – шаговый множитель, $g_f(x_k)$ – произвольный субградиент функции $f(x)$ в точке x_k. Если $g_f(x_k) = 0$, то x_k является точкой минимума функции $f(x)$ и процесс (3) останавливается.

Как регулировать шаг h_k? ... чтобы метод (3) сходился.

Об известных регулировках шага

(a) классические условия (Н. Шор, Ю. Ермольев, Б. Поляк)

$$h_k > 0; \quad \lim_{k\to\infty} h_k = 0; \quad \sum_{k=0}^{\infty} h_k = +\infty.$$

(b) фейеровский шаг (Б. Поляк, И. Еремин)

$$h_k = \frac{\gamma(f(x_k) - f^*)}{\|g_f(x_k)\|}, \quad \text{где} \quad 0 < \gamma < 2.$$

(c) геометрическая прогрессия (Н. Шор)

$$h_{k+1} = h_k * q_k, \quad \text{где} \quad 0 < q_k < 1, \quad k = 0, 1, 2, \dots$$

Для регулировок (a) и (b) метод (3) сходится с любого начального приближения x_0. Регулировка (c) обеспечивает сходимость метода (3) для специальных классов выпуклых функций.

О субградиентном методе Поляка

При $\gamma = 1$ фейеровский шаг дает субградиентный метод

$$x_{k+1} = x_k - h_k^* \frac{g_f(x_k)}{\|g_f(x_k)\|}, \quad h_k^* = \frac{f(x_k) - f^*}{\|g_f(x_k)\|}, \quad k = 0, 1, 2, \ldots, \qquad (4)$$

известный как метод Поляка (1969). Шаг h_k^* называют шагом Поляка или AMS-шагом (шаг Агмона-Моцкина-Шенберга).

Поляк Б.Т. Минимизация негладких функционалов // Журн. вычислит. математики и матем. физики. – 1969. – Т.9, №3. – С. 507–521.

Геометрический смысл метода (4) – следующий. Функция $f(x)$ аппроксимируется линейной $\tilde{f}(x) = f(x_k) + (f'(x_k), x - x_k)$ и шаг выбирается таким, чтобы $\tilde{f}(x_{k+1})$ стала равной f^*. Шаг h_k^* задает величину максимального сдвига в направлении нормированного антисубградиента, при котором угол между антисубградиентом и направлением из точки x_{k+1} на точку минимума будет нетупым.

Почему AMS-шаг?

Первыми AMS-шаг в 1954 году использовали С. Агмон (Agmon) и независимо Т. Моцкин (Motzkin) и И. Шенберг (Schoenberg) в релаксационном методе для нахождения хотя бы одного из решений совместной системы линейных неравенств.

Agmon S. The relaxation method for linear inequalities // Canadien Journal of Mathematics. – 1954. – 6. – P.382–392.

Motzkin T. and Schoenberg I.J. The relaxation method for linear inequalities // Там же. – P.393–404.

И.И. Еремин (1965) обобщил этот метод для систем выпуклых неравенств.

Еремин И.И. Обобщение релаксационного метода Агмона–Моцкина // УМН, 1965, т. XX, вып. 2(122).

Content

Статья в газете „Сегодня“, 28 декабря 2002 года

ПОЕХАЛ В ГРЕЦИЮ, а чемодан остался в Венгрии...

Академик НАН Украины профессор Наум Шор, родившийся 1 января 1937 года, имеет железную логику и ясный ум. Он 44 года без перерыва работает в Институте кибернетики им. В. М. Глушкова. Сегодня занимает там должность заведующего отделом методов решения сложных задач оптимизации.

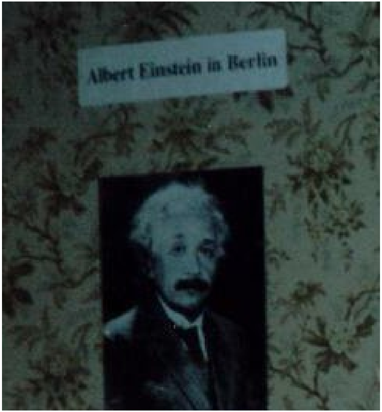
Коллеги из Швейцарии пригласили ученого в музей Эйнштейна в Берне, чтобы сделать это фото и подчеркнуть их сходство

– Что обо мне писать? — засмущался Наум Зуселевич. — Я живу: с работы домой и обратно. Все мои

комился 40 лет назад, 1 января 1963 года. Звезды расположились так, что и она Козерог, родилась 9 января. А вот дети —

Фотография (Ж.Ф. Эмменеггер, Берн, 1997)

И еще о внешней схожести, но уже помоложе …



Альберт Ейнштейн в Берлине.

Наум Шор в Киеве.

О сходстве идей ... или что общего?

между

операторoм Шора для растяжения пространства

$$R_\alpha(\xi) = I + (\alpha - 1)\xi\xi^T, \qquad \text{где} \qquad \alpha > 1, \qquad (*)$$

(I – единичная $n{\times}n$-матрица, вектор $\xi \in E^n$ такой, что $\|\xi\|{=}1$)

и

формулой Ейнштейна для энергии в релятивистской динамике

$$E = mc^2, \qquad \text{где} \qquad m = \frac{m_0}{\sqrt{1 - v^2/c^2}} \qquad (**)$$

Шар S_n и эллипсоид $\mathcal{E}_n$ в пространстве E^n

В преобразованном (т.е. в растянутом в направлении ξ с коэффициентом $\alpha = 1/\beta > 1$) пространстве $y = R_\alpha(\xi)x$ образом шара $S_n = \{x \in E^n : \|x - x_0\| \le r\}$ с радиусом r является эллипсоид $\mathcal{E}_n = \{y \in E^n : \|R_\beta(\xi)(y - y_0)\| \le r\}$.

Объемы шара S_n и эллипсоида $\mathcal{E}_n$ равны

$$\mathrm{vol}(S_n) = v_0 r^n \qquad \text{и} \qquad \mathrm{vol}(\mathcal{E}_n) = \frac{v_0 r^n}{\det R_\beta(\xi)},$$

где v_0 – объем единичного n-мерного шара.

Соотношение объемов шара S_n и эллипсоида $\mathcal{E}_n$

Отношение объемов $\mathcal{E}_n$ и S_n вычисляется по формуле

$$\frac{\mathrm{vol}(\mathcal{E}_n)}{\mathrm{vol}(S_n)} = \frac{\frac{v_0 r^n}{\det R_\beta(\xi)}}{v_0 r^n} = \frac{1}{\det R_\beta(\xi)} = \frac{1}{\beta} = \alpha > 1,$$

где v_0 – объем единичного n-мерного шара.

Его можно переписать так

$$\mathrm{vol}(\mathcal{E}_n) = \frac{\mathrm{vol}(S_n)}{\beta} = \frac{\mathrm{vol}(S_n)}{\sqrt{1-(1-\beta^2)}} \qquad (***)$$

Связь обьемов с формулой Эйнштейна

Из $(***)$ следует

$$\mathrm{vol}(\mathcal{E}_n) = \frac{\mathrm{vol}(S_n)}{\sqrt{1-(1-\beta^2)}} \quad .=. \quad \frac{\mathrm{vol}(S_n)}{\sqrt{1-v^2/c^2}}. \qquad \text{(Shor)}$$

Чем не m? в формуле Эйнштейна $(**)$ ...

$$m = \frac{m_0}{\sqrt{1-v^2/c^2}} \qquad \text{(Einstein)}$$

Об идее Шора (1969)

Идея состоит в

применении линейных неортогональных преобразований пространства для улучшения свойств минимизируемой функции в преобразованном пространстве переменных.

Она оказалась эффективной вычислительной идеей в выпуклой оптимизации и послужила основой для создания двух семейств субградиентных методов с растяжением пространства:

- субградиентные методы с растяжением пространства в направлении субградиента;
- субградиентные методы с растяжением пространства в направлении разности двух последовательных субградиентов (r-алгоритмы).

Суть идеи Шора (1969)

Пусть на k-ой итерации субградиентного алгоритма производится замена переменных $x = B_k y$, где B_k – неособенная $n \times n$-матрица. Для субградиента имеем

$$f(x) - f(x_k) \geq (g_f(x_k), x - x_k) \quad \forall x \in E^n,$$

откуда, осуществляя замену переменных $x = B_k y$, получим

$$\varphi(y) - \varphi(y_k) \geq (B_k^T g_f(x_k), y - y_k) \quad \forall y \in E^n.$$

Вектор $g_\varphi(y_k) = B_k^T g_f(x_k)$ удовлетворяет неравенству

$$\varphi(y) - \varphi(y_k) \geq (g_\varphi(y_k), y - y_k) \quad \forall y \in E^n$$

и есть субградиентом функции $\varphi(y) = f(B_k y)$ в точке $y_k = B_k^{-1} x_k = A_k x_k$.

Суть идеи Шора (1969) .....

Для функции $\varphi(y)$ субградиентный метод в пространстве переменных $y = A_k x$ имеет вид

$$y_{k+1} = y_k - h_k \frac{g_\varphi(y_k)}{\|g_\varphi(y_k)\|} = y_k - h_k \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|}. \qquad (5)$$

Следовательно, очередное приближение $x_{k+1} = B_k y_{k+1}$ будет получено по формуле

$$x_{k+1} = B_k y_k - h_k B_k \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|} = x_k - h_k B_k \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|}. \qquad (6)$$

На формулах (5) и (6) основаны субградиентные методы с преобразованием пространства переменных.

Здесь h_k – шаг в направлении нормированного антисубградиента в преобразованном пространстве переменных $y = A_k x = B_k^{-1} x_k$.

Субградиентный метод Shor69

Пусть x_0 – начальное приближение, B_0 – $n{\times}n$-матрица. Итерация субградиентного метода с последовательным преобразованием пространства переменных имеет вид:

$$x_{k+1} = x_k - h_k B_k \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|}, \quad B_{k+1} = B_k T_k, \qquad \text{(Shor69)}$$

где h_k – шаговый множитель, T_k – $n{\times}n$-матрица.

Метод (Shor69) называют B-формой субградиентного метода с преобразованием пространства.

Его можно записать в H-форме (по типу методов переменной метрики) с помощью симметричной матрицы $H_k = B_k B_k^T$.

Об двух семействах методов с растяжением

Если в методе (Shor69) матрицы T_k подбирать так, чтобы в преобразованном пространстве поверхности овражных функций становились менее овражными, то такой метод окажется эффективнее, чем субградиентный метод.

Это подверждают созданные Н.З. Шором:

- субградиентные методы с растяжением пространства в направлении субградиента;
- субградиентные методы с растяжением пространства в направлении разности двух последовательных субградиентов (r-алгоритмы).

Первое семейство методов Шора

Субградиентным методом с растяжением пространства в направлении субградиента называется процедура построения последовательностей $\{x_k\}_{k=0}^{\infty}$ и $\{B_k\}_{k=0}^{\infty}$ по правилу:

$$x_{k+1} = x_k - h_k B_k \xi_k, \quad B_{k+1} = B_k R_{\beta_k}(\xi_k), \quad k = 0, 1, 2, \ldots, \tag{7}$$

где

$$\xi_k = \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|}, \qquad \beta_k = \frac{1}{\alpha_k} < 1. \tag{8}$$

Здесь x_0 — начальное приближение, $B_0 = I_n$ – единичная $n \times n$-матрица, h_k – шаговый множитель, α_k – коэффициент растяжения пространства, $g_f(x_k)$ – произвольный субградиент функции $f(x)$ в точке x_k. Если $g_f(x_k) = 0$, то x_k – точка минимума функции $f(x)$ и процесс (7)–(8) останавливается.

Их частный случай – метод эллипсоидов

Теорема (Шор, 1977)

Пусть $f(x)$ – выпуклая функция, определенная в E^n, и начальное приближение x_0 такое, что существует точка $x^* \in X^*$, для которой выполняется $\|x_0 - x^*\| \leq r$. Тогда, если в методе (7)–(8) принять:

$$h_0 = \frac{r}{n+1}, \; h_{k+1} = h_k \frac{n}{\sqrt{n^2-1}}, \; \alpha_k = \sqrt{\frac{n+1}{n-1}}, \quad k = 0, 1, 2, \ldots,$$

то последовательность $\{x_k\}_{k=0}^{\infty}$ удовлетворяет неравенству

$$\|A_k(x_k - x^*)\| \leq h_k(n+1), \quad A_k = B_k^{-1}, \quad k = 0, 1, 2, \ldots \quad (9)$$

r-алгоритмы (Шор, Журбенко, 1971)

r-Алгоритмом называется процедура построения последовательностей $\{x_k\}_{k=0}^{\infty}$ и $\{B_k\}_{k=0}^{\infty}$ по правилу:

$$x_{k+1} = x_k - h_k B_k \xi_k, \;\; B_{k+1} = B_k R_{\beta_k}(\eta_k), \;\; k = 0, 1, 2, \ldots, \quad (10)$$

где $$\xi_k = \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|}, \quad h_k \geq h_k^{\star} = \operatorname*{argmin}_{h \geq 0} f(x_k - hB_k\xi_k), \quad (11)$$

$$\eta_k = \frac{B_k^T r_k}{\|B_k^T r_k\|}, \;\; r_k = g_f(x_{k+1}) - g_f(x_k), \quad \beta_k = \frac{1}{\alpha_k} < 1. \quad (12)$$

Здесь x_0 — начальное приближение, $B_0 = I_n$ – единичная $n \times n$-матрица, h_k – шаговый множитель, α_k – коэффициент растяжения пространства, $g_f(x_k)$ и $g_f(x_{k+1})$ – произвольные субградиенты функции $f(x)$ в точках x_k и x_{k+1}. Если $g_f(x_k) = 0$, то x_k – точка минимума функции $f(x)$ и процесс (10)–(12) останавливается.

$r(\alpha)$-алгоритм с адаптивным шагом

Зарекомендовал себя эффективным вариантом r-алгоритмов.

Здесь величина шага h_k адаптивно настраивается с помощью параметров h_0, q_1, n_h, q_2, где h_0 – величина начального шага (используется на 1-й итерации, на каждой последующей итерации уточняется); q_1 – коэффициент уменьшения шага $(q_1 \leq 1)$, если условие завершения спуска по направлению $(h_k > h_k^*)$ выполняется всего за один шаг одномерного спуска; q_2 – коэффициент увеличения шага $(q_2 \geq 1)$; натуральное число n_h задает число шагов одномерного спуска $(n_h > 1)$, через каждые из которых шаг в одномерном спуске будет увеличиваться в q_2 раз.

Об octave-функции ralgb5?

```
%     ralgb5 реализует $r(\alpha)$-алгоритм с адаптивным шагом,
%     использует подготовленную пользователем octave-функцию
%     function [f,g] = calcfg(x), которая вычисляет значения
%     функции $f=f(x)$ и её субградиента $g(x)$ в точке $x$.

%  Входные параметры:
%    calcfg -- имя функции calcfg(x) для вычисления f и g
%    x   -- начальная точка x0(1:n) (на выходе портится)
%    alpha -- коэффициент растяжения пространства
%    h0, nh, q1, q2 -- параметры адаптивной регулировки шага
%    epsx, epsg, maxitn -- параметры останова

%  Выходные параметры:
%    xr -- найденная точка минимума xr(n)
%    fr -- значение функции в точке минимума
%    itn -- число затраченных итераций
%    ncalls -- число вызовов функции calcfg
%    istop -- код останова (2=epsg,3=epsx,4=maxitn,5=error)
```

Octave-функция ralgb5

```
function [xr,fr,itn,ncalls,istop]=ralgb5(calcfg,x,alpha,h0,q1,
                                          q2,nh,epsg,epsx,maxitn);
itn=0; hs=h0; B=eye(length(x)); xr=x;                      # row001
ncalls = 1; [fr,g0] = calcfg(xr);                          # row002
printf("itn %4d f %14.6e fr %14.6e ls %2d ncalls %4d\n",   # row003
       itn, fr, fr, 0, ncalls);
if(norm(g0) < epsg) istop = 2;  return; endif              # row004
for (itn = 1:maxitn)                                       # row005
   dx = B * (g1 = B' * g0)/norm(g1);                       # row006
   d = 1; ls = 0; ddx = 0;                                 # row007
   while (d > 0)                                           # row008
      x -= hs * dx; ddx += hs * norm(dx);                  # row009
      ncalls ++; [f, g1] = calcfg(x);                      # row010
      if (f < fr)  fr = f; xr = x; endif                   # row011
      if(norm(g1) < epsg) istop = 2;  return; endif        # row012
      ls ++; (mod(ls,nh)==0) && (hs *= q2);                # row013
      if(ls > 500) istop = 5;  return; endif               # row014
      d = dx' * g1;                                        # row015
   endwhile                                                # row016
   (ls == 1) && (hs *= q1);                                # row017
   printf("itn %4d f %14.6e fr %14.6e ls %2d ncalls %4d\n", # row018
           itn, f, fr, ls, ncalls);
   if(ddx < epsx) istop = 3;  return; endif                # row019
   xi = (dg = B' * (g1 - g0) )/norm(dg);                   # row020
   B += (1 / alpha - 1) * B * xi * xi';                    # row021
   g0 = g1;                                                # row022
endfor                                                     # row023
istop = 4;                                                 # row024
endfunction
```

Выбор параметров для ralgb5

При минимизации негладких функций рекомендуется:

$$\alpha = 2 \div 3,\ h_0 = 1.0,\ q_1 = 1.0,\ q_2 = 1.1 \div 1.2,\ n_h = 2 \div 3.$$

Если известна априорная оценка расстояния от начальной точки x_0 до точки минимума x^*, то начальный шаг h_0 целесообразно выбирать порядка $\|x_0 - x^*\|$.
При минимизации гладких функций рекомендуется

$$q_1 = 0.8 \div 0.95.$$

При таком выборе параметров, как правило, число спусков по направлению редко превосходит два, а за n шагов точность по функции улучшается в три-пять раз.

Content

Метод эллипсоидов Юдина-Немировского-Шора

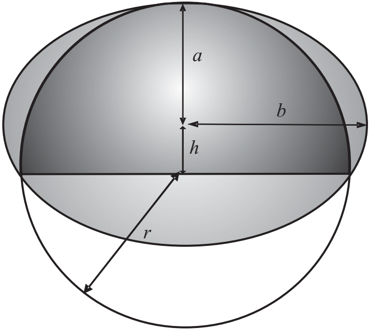
Метод эллипсоидов есть частный случай методов с растяжением пространства в направлении субградиента.

Шор Н.З.(1977) Метод отсечения с растяжением пространства для решения задач выпуклого программирования // Кибернетика. – №1. – С. 94-95.

Первыми метод эллипсоидов предложили Д. Б. Юдин и А. С. Немировский, исходя из методов последовательных отсечений.

Юдин Д.Б., Немировский А.С.(1976) Информационная сложность и эффективные методы решения выпуклых экстремальных задач // Экономика и мат. методы. – Вып.2. – С. 357-359.

Геометрия метода эллипсоидов



Эллипсоид $\mathcal{E}_n$, содержащий полушар в E^n, имеет минимальный обьем, если

$$a = \frac{n}{n+1} r, \; b = \frac{n}{\sqrt{n^2-1}} r, \; h = \frac{1}{n+1} r.$$

Чтобы преобразовать $\mathcal{E}_n$ в шар нужно растянуть пространство с коэффициентом $\alpha = \frac{b}{a} = \sqrt{\frac{n+1}{n-1}}$.

На каждой итерации МЭ объем эллипсоида уменьшается в

$$q(n) = \frac{\mathrm{vol}(\mathcal{E}_n)}{\mathrm{vol}(S_n)} = \frac{a}{r}\left(\frac{b}{r}\right)^{n-1} = \sqrt{\frac{n-1}{n+1}}\left(\frac{n}{\sqrt{n^2-1}}\right)^n \approx 1 - \frac{1}{2n},$$

Octave-функция emshor (ellipsoid method shor)

```
%  Входные параметры:
%    calcfg -- имя функции calcfg(x): вычисляет f и g(1:n)
%    x  -- начальная точка x0(1:n) (на выходе портится)
%    rad -- радиус шара с центром в точке x0
%    epsf -- точность останова по значению функции
%    maxitn -- максимальное количество итераций
%  Выходные параметры:
%    xr -- точка минимума (рекордная)
%    fr -- значение функции (рекордное) в точке xr
%    itn -- число затраченных итераций (вызовов calcfg)
%    itnr -- номер итерации, где получено fr и xr
%    ist -- код останова (1 = epsf, 4 = maxitn)
function [xr,fr,itn,itr,ist]=emshor(calcfg,x,rad,
                                    epsf,maxitn);
dn=float(length(x)); beta=sqrt((dn-1.d0)/(dn+1.d0)); #row01
radn=rad; B=eye(length(x)); fr=1.d+100;              #row02
for (itn = 0:maxitn)                                 #row03
   [f, g1] = calcfg(x);                              #row04
   if (f < fr)  fr = f; xr = x; itr=itn; endif       #row05
   g=B'*g1; dg=norm(g);                              #row06
   if(radn*dg < epsf) ist = 1;  return; endif        #row07
   xi=(1.d0/dg)*g; dx = B * xi; hs=radn/(dn+1.d0);   #row08
   x -= hs * dx;  B += (beta - 1) * B * xi * xi';    #row09
   radn=radn/sqrt(1.d0-1.d0/dn)/sqrt(1.d0+1.d0/dn);  #row10
   printf("itn %4d itr %4d f %16.8e fr %16.8e\n",    #row11
          itn,itr,f,fr,radn)
endfor                                               #row12
ist = 4;                                             #row13
endfunction
```

Премии Фалкерсона “за метод эллипсоидов“

В августе 1982 года в Бонне (ФРГ) состоялся XI международный симпозиум по математическому программированию. Здесь премий Фалкерсона были удостоены:

- Л. Хачиян, Д. Юдин и А. Немировский (СССР) за разработку полиномиального алгоритма решения задачи ЛП с рациональными коэффициентами и метода эллипсоидов;
- М. Гретшель (ФРГ), Л. Ловаш (ВНР) и А. Схрейвер (Нидерланды) за разработку полиномиальных алгоритмов для ряда задач дискретной оптимизации (с помощью метода эллипсоидов).

Делегация от СССР и доклад Шора

Делегацию в Бонн из 6-ти советских ученых возглавлял В.С. Михалевич. Подробный отчет советской делегации об работе симпозиума представлен в статье

XI Международный симпозиум по математическому программированию // Канторович Л.В., Михалевич В.С., Рубинштейн Г.Ш., Третьяков Н.В., Шор Н.З., Якимец В.Н. – Техническая кибернетика. – Обзор. – М.: Изв. АН СССР. – 1983. – №1. – С.197-201.

Н. З. Шор сделал пленарный доклад "Методы обобщенного градиента в недифференцируемой оптимизации, использующие операции растяжения пространства".

О пленарном докладе Н.З.Шора

Абзац из статьи:

Обзорный доклад Н.З.Шора, вызвавший большой интерес у участников симпозиума, был посвящен субградиентным методам для решения задач недифференцируемой оптимизации, повышению эффективности субградиентных методов за счет использования операции растяжения пространства. Автором был рассмотрен ряд алгоритмов с преобразованием пространства, повышающих эффективность субградиентных методов (среди этих алгоритмов был и метод эллипсоидов). Докладчик описал эти алгоритмы и отметил, что их применение в комбинации со схемами декомпозиции и негладкими штрафными функциями позволяет эффективно решать задачи линейного программирования большой размерности, нелинейные минимаксные задачи, плохо обусловленные системы нелинейных уравнений, задачи нелинейного программирования общего типа.

Актуально и сегодня

Н.З.Шор, В.И.Гершович, 1982

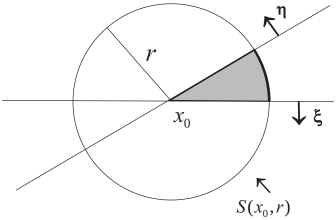
„Теория всего класса алгоритмов с растяжением пространства далека от совершенства. Нам кажется достаточно реалистичной целью – построение такого алгоритма, который по своей практической эффективности не уступал бы r-алгоритму и был столь же хорошо обоснован как метод эллипсоидов“.

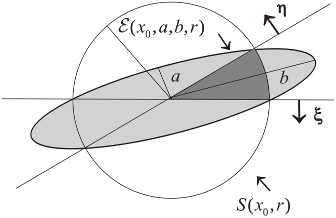
Одна из попыток сделана в статье

Стецюк П.И. r-алгоритмы и эллипсоиды // Кибернетика и системный анализ. – 1996. – № 1. – С. 113–134.

Здесь для преобразования специального эллипсоида в шар использует антиовражный прием, близкий к тому, который имеет место в r-алгоритмах.

Тело W и Специальный эллипсоид





Тело W получено как пересечение шара и двух полупространств.

Специальный эллипсоид содержит W и имеет минимальный объем.

Стецюк П.И. r-алгоритмы и эллипсоиды // Кибернетика и системный анализ. – 1996. – № 1. – С. 113–134.

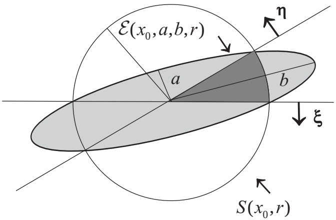
Свойства Специального эллипсоида

1. Если угол φ между векторами ξ и η тупой, то эллипсоид содержит тело W. Объем эллипсоида меньше, чем объем шара, и это уменьшение равно

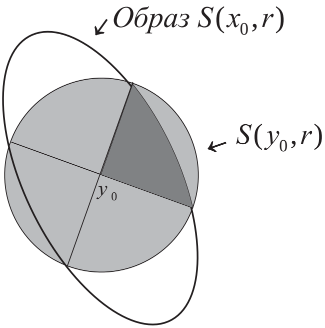
$$\sqrt{1-(\xi,\eta)^2} = \sqrt{1-(\cos\varphi)^2}.$$

2. Преобразовать эллипсоид в шар можно растяжением пространства в направлении $\frac{\xi-\eta}{\|\xi-\eta\|}$ с коэффициентом $\alpha_1 = \frac{1}{\sqrt{1+(\xi,\eta)}}$ и сжатием пространства в ортогональном направлении $\frac{\xi+\eta}{\|\xi+\eta\|}$ с коэффициентом $\alpha_2 = \frac{1}{\sqrt{1-(\xi,\eta)}}$.

Эллипсоид до и после растяжения



Специальный эллипсоид



в дважды растянутом пространстве становится шаром

Одноранговый эллипсоидальный оператор

ОЭО есть линейный оператор

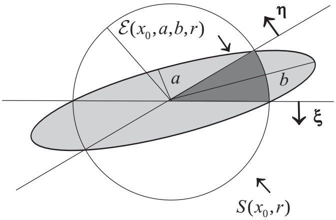
$$T_1(\xi,\eta) = I - \frac{1}{1-(\xi,\eta)^2}\left(\left(1-\sqrt{1-(\xi,\eta)^2}\right)\eta-(\xi,\eta)\xi\right)\eta^T, \quad (13)$$

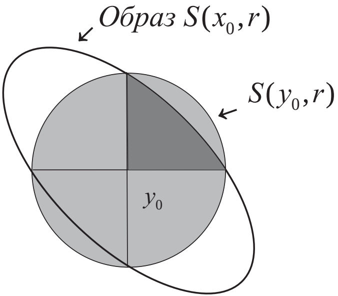
действующий из R^n в R^n. Здесь $\xi,\eta \in R^n$ – векторы, такие что $\|\xi\| = 1, \|\eta\| = 1$ и $(\xi,\eta)^2 \neq 1$, I – единичная $n\times n$-матрица.

Стецюк П.И. Ортогонализующие линейные операторы в выпуклом программировании (Часть I) // Кибернетика и системный анализ. – 1997. – № 3. – С.97–119.

ОЭО преобразует в шар Специальный эллипсоид, описанный вокруг тела W, которое получено в результате пересечения шара и двух полупространств, проходящих через центр шара. Он в два раза экономнее, чем последовательные растяжения.

Эллипсоид до и после преобразования





Специальный эллипсоид в преобразованном пространстве становится шаром

Близость к r-алгоритмам

В преобразованном пространстве эллипсоид станет шаром, а образы векторов ξ и η будут ортогональными.

Это позволяет “расширить” конус подходящих направлений убывания функции для субградиентного процесса в преобразованном пространстве переменных, аналогично тому как это делается в r-алгоритмах.

Растяжение пространства реализуется в направлении разности двух нормированных субградиентов и близким к направлению разности двух субградиентов оно будет только тогда, когда нормы субградиентов близки.

Content

Субградиентный метод Поляка

Субградиентный метод Поляка имеет вид

$$x_{k+1} = x_k - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|}, \quad h_k = \frac{f(x_k) - f^\star}{\|\partial f(x_k)\|}, \quad k = 0, 1, 2, \dots, \tag{14}$$

и решает следующую задачу

$$\text{найти} \quad x^* = \operatorname*{argmin}_{x \in R^n} f(x), \quad \text{если } f^* \text{ – известно}, \tag{15}$$

где $f(x)$ – выпуклая функция и $f^* = f(x^*) = \min_{x \in R^n} f(x)$.

Уменьшение расстояния до точки минимума

Теорема 1 (Поляк, 1969)

Последовательность $\{x_k\}_{k=0}^{\infty}$, генерируемая методом (14), удовлетворяет неравенствам

$$\|x_{k+1} - x^*\|^2 \leq \|x_k - x^*\|^2 - \frac{(f(x_k) - f^*)^2}{\|\partial f(x_k)\|^2}, \quad k = 0, 1, 2, \ldots$$

Замечание. Теорема 1 гарантирует, что в методе Поляка расстояние до точки минимума монотонно убывает.

Медленная сходимость метода Поляка

1. Овражная кусочно-линейная функция $(t > 1)$

$$f_1(x_1, x_2) = |x_1| + t|x_2|, \quad x^\star = (0, 0) \quad f^\star = 0.$$

Метод (14) сходится со скоростью геометрической прогрессии со знаменателем $q(t) = \sqrt{1 - \frac{1}{t^2}}$ (Поляк,1969).

2. Существенно овражная кусочно-квадратичная функция

$$f_2(x_1, x_2) = \max\left\{x_1^2 + (2x_2 - 2)^2 - 3, x_1^2 + (x_2 + 1)^2\right\},$$

Вырождение в точке минимума $x^\star = (0, 0)$, $f^\star = 1$.

Медленная сходимость метода Поляка

1. Овражная кусочно-линейная функция $(t > 1)$

$$f_1(x_1, x_2) = |x_1| + t|x_2|, \quad x^\star = (0, 0) \quad f^\star = 0.$$

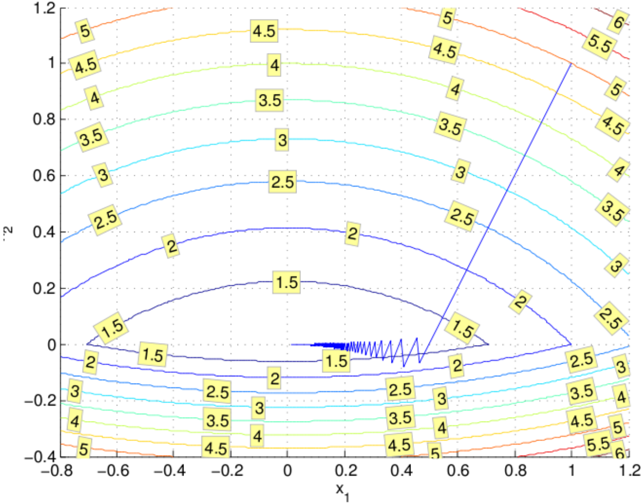
Метод (14) сходится со скоростью геометрической прогрессии со знаменателем $q(t) = \sqrt{1 - \frac{1}{t^2}}$ (Поляк,1969).

2. Существенно овражная кусочно-квадратичная функция

$$f_2(x_1, x_2) = \max\left\{x_1^2 + (2x_2 - 2)^2 - 3, x_1^2 + (x_2 + 1)^2\right\},$$

Вырождение в точке минимума $x^\star = (0, 0)$, $f^\star = 1$.

Метод Поляка для $f_2(x_1, x_2)$ (10000 итераций)



Как преодолеть овражность?

1. Медленную сходимость метода Поляка для овражных функций определяет тупой угол между двумя последовательными субградиентами. Чем ближе угол к 180 градусам, тем более медленной будет сходимость метода (14).

2. Ускорить метод (14) можно, если пространство переменных преобразовать так, чтобы тупой угол между двумя последовательными субградиентами уменьшался.

3. Тупой угол между векторами ξ и η можно преобразовать в прямой с помощью „однорангового эллипсоидального оператора“. Это делается в ускоренном субградиентном методе Поляка.

amsg2p – ускоренный метод Поляка

Имеется стартовая точка $x_0 \in R^n$ и $n \times n$-матрица $B_0 = I_n$.

$$x_{k+1} = x_k - h_k B_k \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}, \quad h_k = \frac{f(x_k) - f^*}{\|B_k^T \partial f(x_k)\|}, \tag{16}$$

$$B_{k+1} = \begin{cases} B_k + (B_k \eta)\xi_{k+1}^T, & \text{if } \mu_k < 0, \\ B_k, & \text{иначе}, \end{cases} \qquad k = 0, 1, 2, \dots, \tag{17}$$

где

$$\mu_k = (\xi_k, \xi_{k+1}), \quad \xi_k = \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}, \quad \xi_{k+1} = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|},$$

$$\eta = \left(\frac{1}{\sqrt{1 - \mu_k^2}} - 1 \right) \xi_{k+1} - \frac{\mu_k}{\sqrt{1 - \mu_k^2}} \xi_k,$$

Уменьшение расстояния до точки минимума

Теорема 2 (Стецюк, 1997)

Let $A_k = B_k^{-1}, A_{k+1} = B_{k+1}^{-1}$. Последовательность $\{x_k\}_{k=0}^{\infty}$, генерируемая методом (16)–(17), удовлетворяет неравенствам

$$\|A_{k+1}(x_{k+1}-x^*)\|^2 \leq \|A_k(x_k-x^*)\|^2 - \frac{(f(x_k)-f^*)^2}{\|B_k^T \partial f(x_k)\|^2}, \quad k = 0, 1, 2, \ldots$$

Замечание. Теорема 2 гарантирует, что в ускоренном методе Поляка расстояние до точки минимума монотонно убывает. Расстояния вычисляются в последовательно преобразованных пространствах переменных.

Об ускоренной сходимости метода (16)–(17)

Для овражных функций детерминант матрицы B_k убывает, а это означает, что убывает объем эллипсоида, локализующего точку минимума x^*.

Действительно, если на k-м шаге реализуется преобразование пространства, то

$$\det(B_{k+1}) = \det(B_k)\sqrt{1-\mu_k^2} = \det(B_k)\sqrt{1-\cos^2\varphi_k}.$$

где φ_k – тупой угол между двумя последовательными субградиентами.

Это обеспечивает ускоренную сходимость метода (16)–(17) для овражных негладких функций в сравнении с методом (14).

Кусочно-линейная функция от двух переменных

Для функции

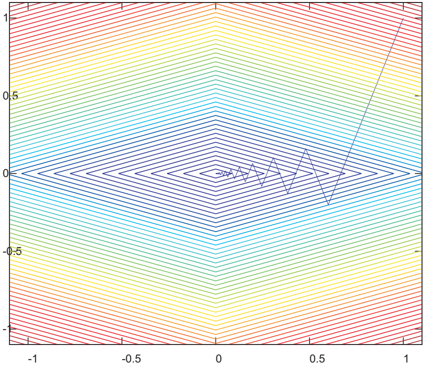
$$f_1(x_1, x_2) = |x_1| + t|x_2|, \ \forall t > 1, \ \forall x_0 = (x_0^{(1)}, x_0^{(2)})$$

метод (16)–(17) находит точку минимума $x^* = (0, 0)$ не более, чем за три итерации:

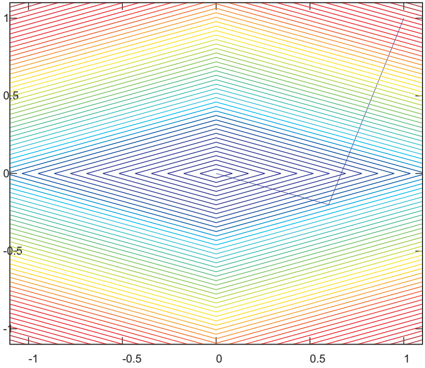
1) одна итерация, если $|x_0^{(2)}| = t|x_0^{(1)}|$. Нет преобразований.
2) две итерации, если $|x_0^{(2)}| < t|x_0^{(1)}|$. Одно преобразование.
3) три итерации, если $|x_0^{(2)}| > t|x_0^{(1)}|$. Одно преобразование.

Если $|x_0^{(2)}| \neq t|x_0^{(1)}|$, то метод Поляка сходится со скоростью геометрической прогрессии со знаменателем $q(t) = \sqrt{1 - 1/t^2}$ и требует существенного количества итераций при больших значениях t.

Тестовый пример $f_1(x_1, x_2) = |x_1| + 10|x_2|$



Метод Поляка



Метод amsg2

Пример функции от двух переменных

Существенно овражная кусочно-квадратичная функция

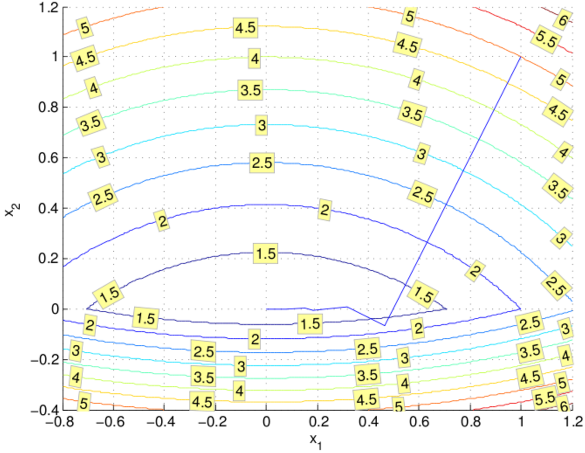
$$f_2(x_1, x_2) = \max\left\{x_1^2 + (2x_2 - 2)^2 - 3, x_1^2 + (x_2 + 1)^2\right\},$$

Вырождение в точке минимума $x^\star = (0, 0)$, $f^\star = 1$.

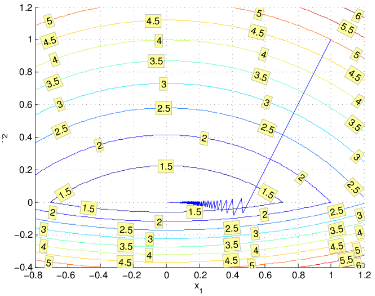
Если $x_0 = (1, 1)$, то метод (16)–(17) находит за:

- 16 итераций – точку x_{16}, где $f_2(x_{16}) \leq 1 + 10^{-6}$;
- 31 итерацию – точку x_{31}, где $f_2(x_{31}) \leq 1 + 10^{-10}$.

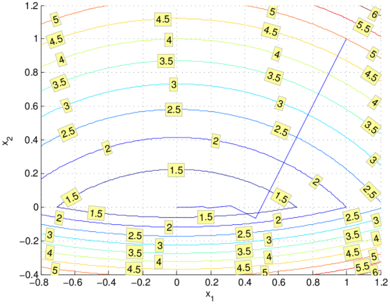
Метод (16)–(17) для $f_2(x_1, x_2)$ (31 итерация)



Сравнение методов для $f_2(x_1, x_2)$



Метод Поляка (10000 итер.)



Метод amsg2 (31 итерация)

Кусочно-линейная функция (10 переменных)

$$f(x) = \sum_{i=1}^{10} q^{(i-1)/9}|x_i - 1|, \qquad q = 3,\ 9,\ 27, \qquad x_0 = (0, \ldots, 0)$$

eps	itn1	itn2	itn1	itn2	itn1	itn2
1.e-01	31	15	220	37	1645	64
1.e-02	72	24	458	44	3257	73
1.e-03	113	29	695	49	4871	78
1.e-04	155	38	933	54	6481	80
1.e-05	196	43	1170	59	8083	84
1.e-06	237	50	1407	62	9633	91
1.e-07	279	54	1642	66	10000	93
1.e-08	320	59	1874	74	10000	100
1.e-09	362	62	2101	78	10000	108
1.e-10	403	65	2322	85	10000	113

itn1 – число итераций метода Поляка, а **itn2** – метода amsg2.

Метод amsg2p

Субградиентный метод с преобразованием пространства

$$x_{k+1} = x_k - h_k B_k \frac{B_k^T \partial f(x_k)}{\|B_k^T \partial f(x_k)\|}, \quad h_k = \frac{\gamma(f(x_k) - f^*)}{\|B_k^T \partial f(x_k)\|}, \qquad (18)$$

$$B_{k+1} = B_k T^{-1}(\xi, \eta) \quad \text{или} \quad B_{k+1} = B_k, \quad k = 0, 1, 2, \ldots, \qquad (19)$$

„ams“ указывает на способ регулировки шага

„g2p“ указывает на использование AMS-шага в пространстве переменных, преобразованном с помощью двух последних субградиентов (g2) и агрегатного вектора (p)

Алгоритм amsg2p

На итерации $k{=}0$ имеем начальное приближение $x_0 \in R^n$ и достаточно малое $\varepsilon_f{>}0$. Вычислим $f(x_0)$ и $\partial f(x_0)$. Если $f(x_0){-}f^*{\le}\varepsilon_f$, то $x^*_\varepsilon{=}x_0$, $k^*_\varepsilon{=}0$ и окончание работы алгоритма.

Иначе положим $h_0{=}\dfrac{\gamma(f(x_0){-}f^*)}{\|\partial f(x_0)\|}$, $\xi_0{=}\dfrac{\partial f(x_0)}{\|\partial f(x_0)\|}{\in}R^n$, $p_0{=}0{\in}R^n$, $B_0{=}I_n$ – единичная матрица размера $n{\times}n$.

Перейдем к следующей итерации.

Пусть на k-й итерации получены $x_k \in R^n$, h_k, $\xi_k \in R^n$, $p_k \in R^n$, B_k – матрица $n{\times}n$. Для $(k+1)$-й итерации выполним пп. 1–5.

1. Вычислим очередное приближение

$$x_{k+1} = x_k - h_k B_k \xi_k.$$

Алгоритм amsg2p: п. 2-3

2. Вычислим $f(x_{k+1})$ и $\partial f(x_{k+1})$. Если $f(x_{k+1}) - f^* \leq \varepsilon_f$, то $x_\varepsilon^* = x_{k+1}$, $k_\varepsilon^* = k+1$ и окончание алгоритма. Иначе положим

$$\xi_{k+1} = \frac{B_k^T \partial f(x_{k+1})}{\|B_k^T \partial f(x_{k+1})\|}, \quad h_{k+1} = \frac{\gamma(f(x_{k+1}) - f^*)}{\|B_k^T \partial f(x_{k+1})\|}.$$

3. Вычислим $\lambda_1 = -p_k^T \xi_{k+1}$ и $\lambda_2 = -\xi_k^T \xi_{k+1}$. Положим

$$p_{k+1} = \begin{cases} \dfrac{\lambda_1}{\sqrt{\lambda_1^2 + \lambda_2^2}} p_k + \dfrac{\lambda_2}{\sqrt{\lambda_1^2 + \lambda_2^2}} \xi_k, & \textit{если } \lambda_1 > 0 \textit{ и } \lambda_2 > 0, \\ p_k, & \textit{если } \lambda_1 > 0 \textit{ и } \lambda_2 \leq 0, \\ \xi_k, & \textit{если } \lambda_1 \leq 0 \textit{ и } \lambda_2 > 0, \\ 0, & \textit{если } \lambda_1 \leq 0 \textit{ и } \lambda_2 \leq 0. \end{cases}$$

Алгоритм amsg2p: п. 4-5

4. Вычислим $\mu_k = p_{k+1}^T \xi_{k+1}$. Если $-1 < \mu_k < 0$, то вычислим

$$B_{k+1}=B_k+(B_k\eta)\,\xi_{k+1}^T,\text{ где }\eta=\left(\frac{1}{\sqrt{1-\mu_k^2}}-1\right)\xi_{k+1}-\frac{\mu_k}{\sqrt{1-\mu_k^2}}p_{k+1}.$$

и пересчитаем

$$h_{k+1} = \frac{h_{k+1}}{\sqrt{1-\mu_k^2}}, \quad p_{k+1} = \frac{1}{\sqrt{1-\mu_k^2}}(p_{k+1} - \mu_k \xi_{k+1})$$

Иначе положим $B_{k+1} = B_k$ и $p_{k+1} = 0$.

5. Перейдем к новой итерации с x_{k+1}, h_{k+1}, ξ_{k+1}, p_{k+1}, B_{k+1}.

amsg2p для квадратичных функций, $n = 200$

epsf	itn(Q_1)	itn(Q_2)	itn(Q_3)	itn(Q_4)	itn(Q_5)
1.00e-003	11	36	84	361	773
1.00e-005	15	46	99	405	826
1.00e-007	18	56	113	430	868
1.00e-009	22	65	128	461	916
1.00e-011	25	73	142	493	947
1.00e-013	29	81	154	517	979
1.00e-015	32	89	167	541	1006
1.00e-017	35	96	180	560	1027
1.00e-019	39	102	189	574	1042
1.00e-020	41	105	196	585	1048

$$Q_1 = 10; \quad Q_2 = 100; \quad Q_3 = 1000, \quad Q_4 = 10^6, \quad Q_5 = 10^9$$

Заключение

На основе однорангового эллипсоидального оператора можно построить ускоренные варианты субградиентных методов и для других способов регулировки шага.

Замечательной чертой таких методов есть автоматический выбор параметров преобразования пространства.

Вопросы?

СПАСИБО ЗА ВНИМАНИЕ!