УДК 519.8

Н.Г. ЖУРБЕНКО, Б.М. ЧУМАКОВ

ПРОГРАММНОЕ УПРАВЛЕНИЕ КОЭФФИЦИЕНТАМИ РАСТЯЖЕНИЯ r -АЛГОРИТМА

Приводится описание семейства алгоритмов минимизации с использованием операции растяжения пространства в направлении разности двух последовательных субградиентов. В отличие от r-алгоритма, в предлагаемых его модификациях значения коэффициентов растяжения на каждой итерации определяются в процессе работы алгоритма. Алгоритмы не требуют использования процедуры одномерного спуска по направлению и могут применяться с постоянным шаговым множителем.

1. Рассматривается задача безусловной минимизации субдифференцируемой функции $f(x)$ в R^n. Предполагается, что в произвольной точке $x \in R^n$ для функции $f(x)$ определен субградиент – вектор $g(x) \in R^n$. Через $\partial f(x)$ будем обозначать множество субградиентов функции $f(x)$ в т. x. Эффективным алгоритмом минимизации субдифференцируемой функции является r-алгоритм – алгоритм, использующий операцию растяжения пространства в направлении разности двух последовательных субградиентов [1], [2], r-алгоритм относится к классу субградиентных алгоритмов с преобразованием пространства, общая схема которых состоит в следующем.

Пусть A невырожденный линейный оператор (невырожденная матрица) на R^n. Пусть $\varphi(y) = f(A^{-1}y)$, $\quad y_0 = Ax_0$, $\quad g(x_0) \in \partial f(x_0)$. Функцию $\varphi(y)$ можно рассматривать, как функцию $f(x)$ в преобразованном оператором A пространстве: $Y = AX$. Множество субградиентов $\partial g(y_0)$ функции $\varphi(y)$ определяется соотношением $\partial\varphi(y_0) = A^{*-1}\partial f(x_0)$, которое определяет преобразования обобщенных градиентов при преобразовании пространства переменных. Пусть на итерации k метода получено приближение x_k и преобразование исходного пространства определяется невырожденным линейным опе-

ратором. Точке x_k соответст-
вует точка y_k преобра-

зованного пространства: $y_k = A_k x_k$. Для получения следующего приближения x_{k+1} реализуем один шаг субградиентного спуска в преобразованном пространстве:

$$y_{k+1} = y_k - h_k g_\varphi(y_k), \tag{1}$$

где $g_\varphi(y_k) = A_k^{*-1} g_f(x_k), g_f(x_k) \in \partial f(x_k)$. Применив к (1) оператор A_k^{-1}, получим

$$x_{k+1} = x_k - h_k A_k^{-1} A_k^{*-1} g_f(x_k), \tag{2}$$

или

$$x_{k+1} = x_k - h_k B_k B_k^* g_f(x_k), \tag{3}$$

где $B_k = A_k^{-1}$. На итерации k выбирается оператор T_k преобразования пространства Y_k : $Y_{k+1} = T_k Y_k = T_k A_k X$. Таким образом, оператор преобразования на итерации $(k+1)$ определяется оператором $A_{k+1} = T_k A_k$. Обратное преобразование $B_{k+1} = B_k T_k^{-1}$. Итеративная процедура (3) порождает конкретные алгоритмы при указании последовательностей h_k, T_k, оператора B_0 и начальной точки x_0.

В $r-$алгоритме в качестве оператора T_k используется оператор растяжения пространства [1] $R_\alpha(\eta) = (\alpha - 1)\eta\eta^T + I$. $\eta \in R^n, \alpha -$ направление и коэффициент растяжения пространства, $|\eta| = 1, \alpha \geq 0$. Заметим, что: $R_\alpha^*(\eta) = R_\alpha(\eta)$ ($R_\alpha(\eta)$ – самосопряженный оператор); $R_\alpha^{-1}(\eta) = R_\beta(\eta)$, где $\beta = 1/\alpha$.

Вычислительная схема $r-$алгоритма применительно к задаче отыскания безусловного минимума функции $f(x)$ состоит в следующем.

0-й шаг алгоритма.

Выбираем начальное приближение x_0 и невырожденное линейное преобразование B_0. Вычисляем:

1) $g(x_0) \in \partial f(x_0)$ (субградиент в точке x_0);

2) $g_0^* = B_0^* g(x_0)$. (субградиент в преобразованном пространстве $Y_0 = A_0 X$).

Переходим ко второму шагу. Пусть на шаге k алгоритма $(k = 0,1,2,\ldots)$ получены определенные значения векторов x_k, g_k^* (субградиент в преобразованном пространстве) и матрицы B_k (обратная матрица преобразования пространства).

$(k+1)-$ **й шаг алгоритма** $(k = 0,1,2,\ldots)$.

Вычисляем:

1) $h_{k+1} -$ шаговый множитель, $h_{k+1} \geq 0$.

2) $x_{k+1} = x_k - h_{k+1} B_k g_k^* / | g_k^* |$;

3) $g(x_{k+1}) \in \partial f(x_k)$. (субградиент в точке x_{k+1});

4) $\tilde{g}^*_{k+1} = B^*_k g(x_{k+1})$ (субградиент в преобразованном пространстве $Y_k = A_k X$);

5) $\eta_{k+1} = (\tilde{g}^*_{k+1} - g^*_k) / | \tilde{g}^*_{k+1} - g^*_k |$ (направление растяжения пространства Y_k);

6) $\alpha_{k+1} \geq 1, \beta_{k+1} = 1/\alpha_{k+1}$ (α_k коэффициент растяжения пространства Y_k);

7) $B_{k+1} = B_k R_{\beta_{k+1}}(\eta_k)$, (обратный оператор преобразования пространства $Y_{k+1} = A_{k+1} X$);

8) $g^*_{k+1} = R_{\beta_{k+1}}(\eta_{k+1}) \tilde{g}_{k+1}$ $(g^*_{k+1} = B^*_{k+1} g(x_k)$. (субградиент в преобразованном пространстве $Y_{k+1} = A_{k+1} X$).

Переходим к $(k+2)-$у шагу алгоритма, или прекращаем работу при выполнении критерия останова.

В известных вариантах r-алгоритма выбор коэффициентов растяжения пространства и шаговых множителей определяется следующим образом. Коэффициент растяжения пространства выбирается одинаковым на всех итерациях: $\alpha_k = \alpha > 1$. Значение этого коэффициента является параметром алгоритма. На практике рекомендуется это значение выбирать порядка 2.0. Величина шагового множителя определяется выбранной процедурой минимизации (процедурой «спуска») по направлению $p_k = -B_k g^*_k / | g^*_k |$. Применяемые процедуры являются достаточно грубой реализацией алгоритма локализации минимума по направлению p_k. Основным требованием при этом является выполнение условия $(p_k, g(x_{k+1})) \geq 0$ (это условие обеспечивает, что $| \tilde{g}^*_{k+1} - g^*_k | > 0$).

Наиболее часто используется следующая процедура регулировки шаговых множителей. Пусть на $(k+1)-$й итерации помимо точки x_k, вектора спуска p_k определено значение «пробного шага» $\tilde{h}_k > 0$.

Пусть заданы числа $0 < q_1 \leq 1$, $q_2 \geq 1$ и целое число $L \geq 2$. Эти величины являются параметрами (константами) алгоритма. Они будут определять регулировку величин пробного шага. Параметр q_1 используется для уменьшения пробного шага, а q_2 и L для его увеличения.

Положим $z_0 = x_k$. Вычисляем: $z_i = z_{i-1} + \tilde{h}_k p_k$; $\tilde{h}_k := q_2 \tilde{h}_k$, если $i > L$; $g_i = g(z_i) \in \partial f(z_i)$, $i = 1,2,\ldots$, до тех пор, пока при некотором $i = l$ выполнится неравенство $(g_i, p_k) \geq 0$. Тогда $x_{k+1} = z_l$. Если $l = 1$, то $\tilde{h}_{k+1} := q_1 \tilde{h}_k$.

Наиболее часто используются следующие значения параметров регулировки пробного шага $q_1 = 0.9$, $q_2 = 1.2$, $L = 3$.

2. Вычислительная схема предлагаемых алгоритмов соответствует приведенной схеме r -алгоритма. Отличие состоит лишь в следующем. Вместо оператора растяжения $R_\alpha(\eta)$ будет использоваться следующий оператор $\tilde{R}_\sigma(\tilde{\eta}) = \sigma\tilde{\eta}\tilde{\eta}^T + I$. Здесь $\tilde{\eta}$ – вектор R^1, σ - нормирующий множитель, $\sigma \in R^1$, $\sigma > 0$. В отличие от оператора $R_\alpha(\eta)$, вектор $\tilde{\eta}$ не обязательно нормирован, то есть выполнение условия $|\tilde{\eta}| = 1$ не требуется. Различные варианты алгоритма будут определяться выбором нормирующего множителя σ. Далее будут рассмотрены несколько вариантов такого выбора.

Остановимся на простейших свойствах оператора $\tilde{R}_\sigma(\tilde{\eta})$ и его связи с оператором $R_\alpha(\eta)$. Очевидно, что $\tilde{R}_\sigma(0) = I$ (при этом значение нормирующего множителя не имеет значения). Пусть $|\tilde{\eta}| > 0$ и $\eta = \tilde{\eta} / |\tilde{\eta}|$. Тогда $\tilde{R}_\sigma(\tilde{\eta}) = R_\alpha(\eta)$, где

$$\alpha = 1 + \sigma |\tilde{\eta}|^2 . \tag{4}$$

Таким образом, если $\tilde{\eta} \neq 0$, то оператор $\tilde{R}_\sigma(\tilde{\eta})$ является оператором растяжения по направлению $\tilde{\eta} / |\tilde{\eta}|$ с указанным значением коэффициентом растяжения (4). Выбор вектора $\tilde{\eta}$, определяющего направление растяжения пространства, будет в точности соответствовать r -алгоритму: $\eta_{k+1} = \tilde{g}^*_{k+1} - g^*_k$ (разность субградиентов в преобразованном пространстве).

Значение нормирующего множителя σ будет определяться на основании субградиентов $\tilde{g}^*_{k+1}, g^*_k$: $\sigma_{k+1} = \sigma(\tilde{g}^*_{k+1}, g^*_k)$. Естественным требованием на функцию $\sigma(g_1, g_2)$ будет выполнение условия (условия «однородности») $\sigma(\mu g_1, \mu g_2) = \sigma(g_1, g_2) / \mu^2$, где $\mu \in R^1, \mu > 0$ (это условие обеспечивает независимость работы алгоритма от множителя на целевую функцию). Легко видеть, что алгоритм $r(\sigma_0)$: $\sigma_0(g_1, g_2) = 1 / |g_2 - g_1|^2$ фактически является r -алгоритмом с коэффициентом растяжения равным 2. Любопытно отметить, что именно это значение рекомендуется на практике использования r -алгоритм.

3. Приведем результаты численных исследований следующего варианта выбора нормирующего множителя: $\sigma_1(g_1, g_2) = 1 / |Nr\{g_1, g_2\}|^2$, где $Nr\{g_1, g_2\}$ – кратчайший вектор выпуклой оболочки векторов g_1, g_2. Для алгоритма $r(\sigma_1)$ значения коэффициентов не ограничены: $1 \le \alpha_{k+1} \le \infty$.

В качестве тестовых задач рассматривались задачи минимизации двух следующих функций: $f_1(x) = \sum_{i=1}^{n} \rho_n^{i-1} x_i^2$, $f_2(x) = \sum_{i=1}^{n} \rho_n^{i-1} |x_i|$, 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 $x_i = 0.0, i = 1,2,...n.$ Критерий останова: $f_k \le 10^{-6}$, где f_k - значение функции на итерации останова k .

Результаты решения тестовых задач минимизации функций $f_1(x)$, $f_2(x)$ приведены в таблицах 1 и 2 соответственно, где приняты следующие обозначения: $r(\sigma_1)$ алгоритм с регулировкой шаговых множителей r-алгоритма; $r^*(\sigma_1)$ - алгоритм с постоянным шагом; n – размерность пространства переменных; k – номер итерации, на которой алгоритм прекратил работу; k_g – количество вычислений обобщенного градиента; $\alpha_{\max}$ – максимальное значение коэффициента растяжения; α_{avrg} – среднее значение коэффициента растяжения:

ТАБЛИЦА 1

Параметры / Алгоритм	n	k	k_g	$\alpha_{\max}$	α_{avrg}
$r(\sigma_1)$	100	678	931	83.6	5.6
$r^*(\sigma_1)$	100	858	859	35.2	4.4
$r(\alpha)$	100	582	683	2.0	2.0
$r(\sigma_1)$	300	984	1272	31.6	5.0
$r^*(\sigma_1)$	300	2239	2240	28.6	4.4
$r(\alpha)$	300	892	1053	2.0	2.0
$r(\sigma_1)$	1000	1458	1966	14.1	4.1
$r^*(\sigma_1)$	1000	7621	7622	54.2	4.2
$r(\alpha)$	1000	2190	3258	2.0	2.0

ТАБЛИЦА 2

Параметры / Алгоритм	n	k	k_g	$\alpha_{\max}$	α_{avrg}
$r(\sigma_1)$	100	670	689	9.5	3.8
$r^*(\sigma_1)$	100	1125	1126	5.9	3.8
$r(\alpha)$	100	938	1017	2.0	2.0
$r(\sigma_1)$	300	1462	1620	8.8	3.6
$r^*(\sigma_1)$	300	3559	3560	5.1	3.7
$r(\alpha)$	300	2534	3050	2.0	2.0

$r(\sigma_1)$	1000	3817	4373	6.9	3.6
$r^*(\sigma_1)$	1000	12385	12386	4.6	3.6
$r(\alpha)$	1000	9364	11532	2.0	2.0

Численные эксперименты показали достаточно высокую эффективность $r(\sigma)$ -алгоритмов. Их эффективность не уступает эффективности r -алгоритма.

Заключение. Предложенные субградиентные алгоритмы с преобразованием пространства можно рассматривать как модификации r -алгоритма. Величины коэффициентов растяжения пространства на итерациях $r(\sigma)$ -алгоритмов не постоянны, они вычисляются в процессе его работы. Алгоритмы не требуют использования процедуры одномерного спуска по направлению. Они могут использоваться с постоянным шаговым множителем. Вычислительная схема $r(\sigma)$ -алгоритмов с постоянным шагом в некотором смысле проще схемы r -алгоритма. Это позволяет авторам надеяться на возможность теоретического исследования эффективности $r(\sigma)$ -алгоритмов.

М.Г. Журбенко, Б.М. Чумаков

ПРОГРАМНЕ УПРАВЛІННЯ КОЕФІЦІЄНТАМИ РОЗТЯГУ r -АЛГОРИТМУ

Наводиться опис сімейства алгоритмів мінімізації з використанням операції розтягу простору в напрямку різниці двох послідовних субградієнтів. На відміну від r-алгоритму, в пропонованих його модифікаціях значення коефіцієнтів розтягування на кожній ітерації визначаються в процесі роботи алгоритму. Алгоритми не вимагають використання процедури одновимірного спуску у напрямку і можуть застосовуватися з постійним кроковим множником.

N.G. Zhurbenko, B.M. Chumakov

PROGRAM CONTROL OF DILATION COEFFICIENTS OF r -ALGORITHM

The description of a family of minimization algorithms using space dilation operation along the direction of the difference of two successive subgradients is given. In contrast to r-algorithm, in the proposed modifications the values of dilation coefficients at each iteration are calculated in the process of algorithm. The algorithms do not require usage of the one-dimensional descent procedure along direction and can be used with a constant step size.

1. *Шор Н.З.* Методы минимизации недифференцируемых функций и их приложения. – Киев: Наук. думка, 1979. – 199 с.
2. *Шор Н.З., Журбенко Н.Г.* Метод минимизации, использующий операцию растяжения пространства в направлении разности двух последовательных градиентов // Кибернетика. – Киев: Наук. думка, 1971. – № 3. – С. 51–59.

Получено 17.05.2012