

## Про одне сімейство субградієнтних алгоритмів з перетворенням простору

Микола Журбенко<sup>1</sup>, Олексій Лиховид<sup>2</sup>

<sup>1</sup>к. ф.-м. н., с.н.с, Інститут кібернетики ім. В.М. Глушкова НАН України, Проспект академіка Глушкова, 4003187, Київ, e-mail: zhurbnick@gmail.com

<sup>2</sup>н.с., Інститут кібернетики ім. В.М. Глушкова НАН України, Проспект академіка Глушкова, 4003187, Київ, e-mail: o.lykhovyd@gmail.com

*У статті наведено короткий опис результатів розробки сімейства субградієнтних алгоритмів мінімізації з використанням операторів розтягу простору змінних ( $r(\sigma)$ -алгоритмів).  $r(\sigma)$ -алгоритми є модифікаціями  $r$ -алгоритму Н.З. Шора. На відміну від  $r$ -алгоритму, значення коефіцієнтів розтягу простору в  $r(\sigma)$ -алгоритмах визначаються програмно під час виконання алгоритму. Суттєво, що в  $r(\sigma)$ -алгоритмах немає необхідності використовувати процедуру одновимірної мінімізації за напрямком – алгоритми можна використовувати з постійним кроком у перетвореному просторі змінних.*

**Ключові слова:** негладка оптимізація, субградієнтний алгоритм, перетворення простору, чисельна ефективність

**Вступ.** Понад 50 років тому було розроблено субградієнтний алгоритм мінімізації з розтягом простору у напрямку різниці двох послідовних градієнтів –  $r$ -алгоритм [1], [2]. Практика використання  $r$ -алгоритму показує, що і досі він є одним із найефективніших алгоритмів негладкої оптимізації. Проте теоретичне дослідження ефективності алгоритму не закінчено. Основна проблема теоретичного обґрунтування  $r$ -алгоритму полягає у узгодженому виборі значень коефіцієнтів розтягу простору та крокових множників. З метою подолання певною мірою цієї проблеми та спрощення чисельної схеми було розроблено сімейство модифікацій  $r$ -алгоритму з програмним вибором значень коефіцієнтів розтягу простору [3], [4], [5]. У доповіді буде надано коротку характеристику  $r(\sigma)$ -алгоритмів та запропоновано нові елементи процедури управління значеннями коефіцієнтів розтягу.

### 1. Загальна схема субградієнтних алгоритмів з перетворенням простору

Загальна схема субградієнтних алгоритмів з перетворенням простору для задачі безумовної мінімізації субдиференційованої функції  $f(x)$  в  $R^n$  полягає у наступному. Нехай  $A$  – невироджений лінійний оператор (невироджена матриця) в  $R^n$ . Нехай  $\varphi(y) = f(A^{-1}y)$ ,  $y = Ax$ ,  $g(x) \in \partial f(x)$ . Функцію  $\varphi(y)$  можна розглядати як функцію  $f(x)$  у перетвореному оператором  $A$  просторі:  $Y = AX$ . Множина субградієнтів  $\partial g_\varphi(y)$  функції  $\varphi(y)$  визначається співвідношенням  $\partial \varphi(y) = A^{*-1} \partial f(x)$ , яке визначає перетворення субградієнтів під час перетворення простору змінних [2]. Нехай на ітерації  $k$  методу отримана точка  $x_k$  та перетворення простору визначається невиродженим лінійним оператором  $A_k : Y_k = A_k X_k$ .

Точці  $x_k$  відповідає точка  $y_k$  перетвореного простору:  $y_k = A_k x_k$ . Для отримання наступного наближення  $x_{k+1}$  реалізуємо один крок субградієнтного спуску у перетвореному просторі (ітерація  $k+1$ ):

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

де  $g_\varphi(y_k) = A_k^{*-1} g_f(x_k)$ ,  $g_f(x_k) \in \partial f(x_k)$ . Застосувавши до (1) оператор  $A_k^{-1}$ , отримаємо

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

або

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

де  $B_k = A_k^{-1}$ . На ітерації  $k+1$  вибирається оператор  $T_k$  перетворення простору  $Y_k$ :  $Y_{k+1} = T_k Y_k = T_k A_k X$ . Таким чином, оператор перетворення на ітерації  $k+1$  визначається оператором  $A_{k+1} = T_k A_k$ . Обернене перетворення  $B_{k+1} = B_k T_k^{-1}$ . Ітеративна процедура (3) породжує конкретні алгоритми при вказівці послідовностей  $h_k, T_k$ , оператора  $B_0$  та початкової точки  $x_0$ .

## 2. Обчислювальна схема г-алгоритму та г(σ)-алгоритмів

**г-алгоритм.** У г-алгоритмі в якості оператора перетворення простору  $T_k$  використовується оператор розтягу простору за напрямком [2]:  $R_\alpha(\eta) = (\alpha - 1)\eta\eta^T + I$ , де  $I$  – одинична матриця,  $\eta \in R^n$ ,  $\alpha$  – напрямок та коефіцієнт розтягу простору,  $\|\eta\| = 1$ ,  $\alpha \geq 0$ . Зауважимо, що:  $R_\alpha^{-1}(\eta) = R_\beta(\eta)$ ,  $\beta = 1/\alpha$ . Напрямок розтягу  $\eta_{k+1} = (g_{k+1}^* - g_k^*) / \|g_{k+1}^* - g_k^*\|$ , де  $g_{k+1}^*, g_k^*$  – субградієнти на ітераціях  $k+1, k$  у перетвореному просторі  $Y_k$ . Значення коефіцієнтів розтягу простору  $\alpha_k$  (параметр г-алгоритму) вибираються однаковими на всіх ітераціях:  $\alpha_k = \alpha > 1$ . На практиці рекомендується вибирати це значення порядку 2. Розмір крокового множника визначається процедурою мінімізації за напрямком  $p_k = -B_k g_k^* / \|g_k^*\|$ . Застосовувані процедури є досить грубою реалізацією алгоритму локалізації мінімуму за напрямком  $p_k$  [1], [2]. Основною вимогою при цьому є виконання умови  $(p_k, g(x_{k+1})) \geq 0$  (ця умова забезпечує, що  $\|g_{k+1}^* - g_k^*\| > 0$ ).

**г(σ)-алгоритми.** Обчислювальна схема г(σ)-алгоритмів переважно відповідає г-алгоритму. Відмінність полягає лише в наступному. Замість оператора розтягу [2]  $R_\alpha(\eta) = (\alpha - 1)\eta\eta^T + I$ ,  $\|\eta\| = 1$  буде використовуватися наступний оператор  $\tilde{R}_\sigma(\tilde{\eta}) = \gamma_\sigma \sigma \tilde{\eta} \tilde{\eta}^T + I$ , де  $\tilde{\eta} \in R^n$ ;  $\sigma$  – нормуючий множник,  $\sigma \in R^1, \sigma > 0$ ;  $\gamma_\sigma$  – додатне число. Число  $\gamma_\sigma$  визначається таким чином, щоб значення коефіцієнтів розтягу для всіх ітерацій були обмежені:  $\alpha_k \leq \alpha_{\max}$ . Тут  $\alpha_{\max} \geq 1$  є вхідним параметром алгоритму. Параметр  $\alpha_{\max}$  у попередніх версіях г(σ)-алгоритмів був відсутній.

На відміну від оператора  $R_\alpha(\eta)$ , вектор  $\tilde{\eta}$  не нормовано, тобто виконання умови  $\|\tilde{\eta}\|=1$  не вимагається. Легко бачити, що якщо  $\tilde{\eta} \neq 0$ , то  $\tilde{R}_\sigma(\tilde{\eta}) = R_\alpha(\tilde{\eta}/\|\tilde{\eta}\|)$ , де

$$\alpha = 1 + \gamma_\sigma \sigma \|\tilde{\eta}\|^2. \quad (4)$$

Зауважимо, що якщо  $\tilde{\eta} = 0$ , то  $\alpha = 1$ ,  $\tilde{R}_\sigma(\tilde{\eta}) = I$  (незалежно від множника  $\sigma$ ).

Значення нормуючого множника  $\sigma$  визначатиметься на підставі двох послідовних субградієнтів  $g_{k+1}^*, g_k^*$  у перетвореному просторі (обчислених на ітераціях  $k$  і  $k+1$ ):  $\sigma_{k+1} = \sigma(g_{k+1}^*, g_k^*)$ . Природною вимогою на функцію  $\sigma(g_1, g_2)$  буде виконання умови:  $\sigma(\mu g_1, \mu g_2) = \sigma(g_1, g_2) / \mu^2$ , де  $\mu \in R^1, \mu > 0$ . Ця умова забезпечує незалежність роботи алгоритму від множника на цільову функцію. Різні варіанти алгоритму будуть визначатися вибором множника  $\sigma$  та значенням параметра  $\alpha_{\max}$ . Відмітимо, що  $r(\sigma)$ -алгоритм з нормуючим множником  $\sigma_0(g_1, g_2) = 1/\|g_2 - g_1\|^2$  та  $\gamma_\sigma = 1$  фактично є  $r$ -алгоритмом з коефіцієнтом розтягу, що дорівнює 2. Зазначимо, що це значення рекомендується при використанні  $r$ -алгоритму.

Приклади нормуючих множників.

1. Алгоритм  $r(\sigma_1)$ :  $\sigma(g_1, g_2) = 1/(\|g_1\|^2 + \|g_2\|^2)$ ;  $\gamma_\sigma = (\alpha_{\max} - 1)/3$ .
2. Алгоритм  $r(\sigma_2)$ :  $\sigma(g_1, g_2) = 1/\max\{\|g_1\|^2, \|g_2\|^2\}$ ;  $\gamma_\sigma = (\alpha_{\max} - 1)/4$ .

Якщо в якості вектора  $\tilde{\eta}$  вибирається вектор  $\eta_{k+1} = g_{k+1}^* - g_k^*$  (за аналогією з  $r$ -алгоритмом), то ми отримуємо модифікації  $r$ -алгоритму з програмним визначенням значень коефіцієнтів розтягу простору (відповідно до формули (4)).

Особливий інтерес становить алгоритм з розтягом простору за напрямком різниці нормованих субградієнтів –  $r_n$ -алгоритм. У  $r_n$ -алгоритмі вектор  $\tilde{\eta}_{k+1} = g_{k+1}^* / \|g_{k+1}^*\| - g_k^* / \|g_k^*\|$ . Для цього алгоритму нормуючий множник  $\sigma$  можна покласти рівним 1:  $\sigma_n(g_1, g_2) = 1$ . При цьому  $\gamma_\sigma = (\alpha_{\max} - 1)/4$ .

Пояснимо зміст оператора  $\tilde{R}_\sigma(\tilde{\eta})$  на прикладі  $r_n$ -алгоритму. З рівності (4) випливає, що  $\alpha_{k+1} = 1 + (\alpha_{\max} - 1)(1 - \cos(\varphi_{k+1})) / 2$ , де  $\varphi_{k+1}$  – кут між векторами  $g_{k+1}^*, g_k^*$ . Таким чином, значення коефіцієнта розтягу на ітерації  $r_n$ -алгоритму має тим більше значення, чим більший кут між векторами  $g_{k+1}^*, g_k^*$ . Максимальне значення коефіцієнта розтягу ( $\alpha_{\max}$ ) досягається при куті, що дорівнює  $\pi$ . При куті, що дорівнює нулю, коефіцієнт розтягу дорівнює 1 (тобто фактично перетворення простору не виконується).

У  $r(\sigma)$ -алгоритмах значення крокового множника можна визначати як і в  $r$ -алгоритмі – з урахуванням використання процедури одномірного спуску. Однак, суттєво, що  $r(\sigma)$ -алгоритм можна використовувати без процедури спуску за напрямком, з постійним кроком у перетвореному просторі (параметр алгоритму). При цьому величина кроку у вихідному просторі на ітераціях алгоритму визначатиметься (опосередковано) тільки операторами перетворення, що використовуються. Зазначимо, що обчислювальна схема таких варіантів  $r(\sigma)$ -алгоритмів значно простіша за схему  $r$ -алгоритму.

Для теоретичного обґрунтування збіжності  $r(\sigma)$ -алгоритмів пропонується наступний їх варіант: максимальне значення коефіцієнта розтягу простору змінюється на ітераціях, тобто визначається послідовність  $\alpha_{\max}(k)$ . Для цієї послідовності виконуються такі умови:

$\alpha_{\max}(k) \geq 1$ ;  $\alpha_{\max}(k) \rightarrow 1$  (коефіцієнти зменшуються);  $\prod_{j=1}^k \alpha_{\max}(j) \rightarrow \infty$  (загальний розтяг простору не обмежено). Приклад такої послідовності:  $\alpha_{\max}(k) = e^{c/k}$ ;  $c > 0$ . Зауважимо, що така послідовність асоціюється з послідовністю вибору крокових множників у класичному субградієнтному алгоритмі (крок прагне до нуля, сума кроків розходиться) [2].

У доповіді будуть представлені результати дослідження чисельної ефективності сімейства алгоритмів, що розглядається.

**Висновки.**  $r(\sigma)$ -алгоритми є модифікаціями  $r$ -алгоритму. Обчислювальна схема  $r(\sigma)$ -алгоритмів з постійним кроком суттєво простіше схеми  $r$ -алгоритму. Величини коефіцієнтів розтягу простору на ітераціях  $r(\sigma)$ -алгоритмів не постійні, вони обчислюються у процесі його роботи. Алгоритми можуть використовуватися з постійним кроковим множником у перетвореному просторі. Чисельні експерименти показали досить високу ефективність  $r(\sigma)$ -алгоритмів. Їхня ефективність не поступається ефективності  $r$ -алгоритму. Результати дослідження чисельної ефективності показують, що найбільш ефективним варіантом із сімейства  $r(\sigma)$ -алгоритмів є  $r_n$ -алгоритм.

Роботу частково підтримано грантом Volkswagen Foundation (грант № 97775).

### Література

- [1] Шор Н.З., Журбенко Н.Г. Метод минимизации, использующий операцию растяжения пространства в направлении разности двух последовательных градиентов. *Кибернетика*. 1971. №3. С. 51–59.
- [2] Shor N.Z. Minimization methods for non-differentiable functions. Berlin: Springer-Verlag, 1985. 178 p.
- [3] Журбенко Н.Г., Чумаков Б.М. Программное управление коэффициентами растяжения  $r$ -алгоритма. *Теорія оптимальних рішень*. Київ: Ін-т кібернетики ім. В.М. Глушкова НАН України, 2012. С. 113–118.
- [4] Журбенко Н.Г. Численная эффективность одной модификации  $r$ -алгоритма. *Теорія оптимальних рішень*. Київ: Інститут кібернетики ім. В.М. Глушкова НАН України, 2017. С. 33–38.
- [5] Журбенко Н.Г., Лиховид А.П. К численной эффективности одной модификации  $r$ -алгоритма. *Комп'ютерна математика*. К.: Ін-т кібернетики ім. В.М. Глушкова НАН України, 2019. № 1. С. 2–10.

## On one family of subgradient algorithms with space transformation

Mykola Zhurbenko, Oleksii Lykhovyd

*The article presents a brief description of the results of the development of a family of subgradient algorithms for minimization using dilation operators of space of variables ( $r(\sigma)$ -algorithms).  $r(\sigma)$ -algorithms are modifications of N.Z. Shor's  $r$ -algorithm. Unlike  $r$ -algorithm, the values of space dilation coefficients in  $r(\sigma)$ -algorithms are programmatically determined during the execution of the algorithm. It is essential that to determine the values of step coefficients in  $r(\sigma)$ -algorithms, there is no need to use the procedure of one-dimensional minimization in the direction – the algorithms can be used with a constant step in the transformed space of variables.*

Отримано 25.03.23