

УДК 519.168 (045)

О. М. Зудов, канд. техн. наук,  
О. С. Васильєв, асп.

## МЕТОД ГРАДІЄНТНОГО СХРЕЩУВАННЯ У ГЕНЕТИЧНИХ АЛГОРИТМАХ НЕПЕРЕРВНОГО КОДУ

Факультет комп'ютерних систем НАУ, e-mail: stud\_dekan@ukr.net

*Присвячено модифікації одного з методів знаходження абсолютного екстремуму функції шляхом використання генетичних алгоритмів. Виконано огляд існуючих підходів до використання генетичних алгоритмів і запропоновано підхід до цієї проблеми. Наведено математичні розрахунки для реалізації цього алгоритму і проаналізовано доцільність використання такого методу.*

**Ключові слова:** генетичний алгоритм, локальний екстремум, глобальний екстремум, градієнт, схрещування, ітерація, селекція, схрещування

**Вступ та постановка завдання.** Відомо, що всі градієнтні методи оптимізації мають спільний недолік: за наявності у цільовій функції декількох локальних максимумів (мінімумів) ці методи не гарантують знаходження глобального екстремуму (рис. 1), [1; 5].

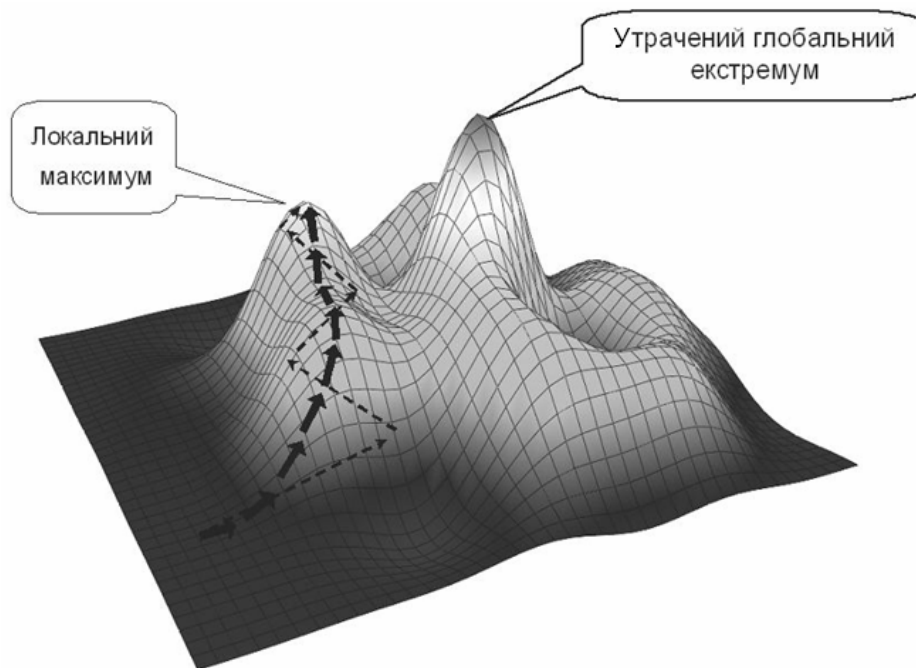


Рис. 1. Втрата глобального розв'язку в разі застосування градієнтного пошуку

Для розв'язання таких багатоекстремальних (полімодальних) оптимізаційних задач часто застосовують генетичні алгоритми, запропоновані Голандом [2], які моделюють еволюційні процеси у живій природі. Основними етапами генетичного алгоритму (рис. 2.) є:

- 1) кодування множини розв'язків задачі (фенотипу) бінарним кодом (генотипом); як правило, використовують код Грея;
- 2) генерація масиву початкових розв'язків (формування первинної популяції);
- 3) вибір пар розв'язків для схрещування (конкурентна селекція);
- 4) схрещування, тобто обмін бінарних подань батьківської пари частиною коду (обмін генами);
- 5) випадкова зміна деякої невеликої частини кодів (мутація);
- 6) новий цикл селекції серед новостворених розв'язків (конкурентна боротьба нових поколінь); далі алгоритм повторюється з п. 3).

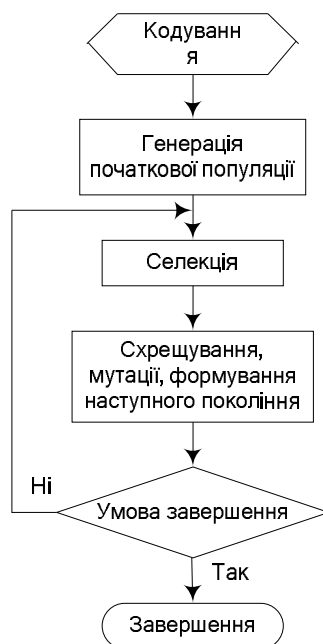


Рис. 2. Генетичний алгоритм

Суттєвим недоліком генетичних алгоритмів є велика розрядність бінарного подання, особливо для великої розмірності простору розв'язків. Так, уже для цільової функції з чотирма параметрами  $f(x_1, x_2, x_3, x_4)$ ,  $-1 < x_i < 1$ , для точності  $10^{-6}$  необхідно закодувати  $2 \cdot 10^{24}$  розв'язків, що вимагає обробляти бітовий рядок довжиною  $\log_2(2 \cdot 10^{24}) \approx 81$  біт. Крім того, такий великий простір розв'язків потребує великої чисельності популяції.

Проте в разі застосування генетичних алгоритмів виникає потреба у визначенні критеріїв схрещування, тобто у виборі генетичних ознак – параметрів, що будуть успадковані наступним поколінням проміжних розв'язків. Незалежно від методу схрещування такими ознаками часто стають спільні для оптимальних розв'язків гени (бітові комбінації), особливо для близько розміщених батьківських пар, що може зумовити передчасну збіжність алгоритму з локальним оптимумом (рис. 3).

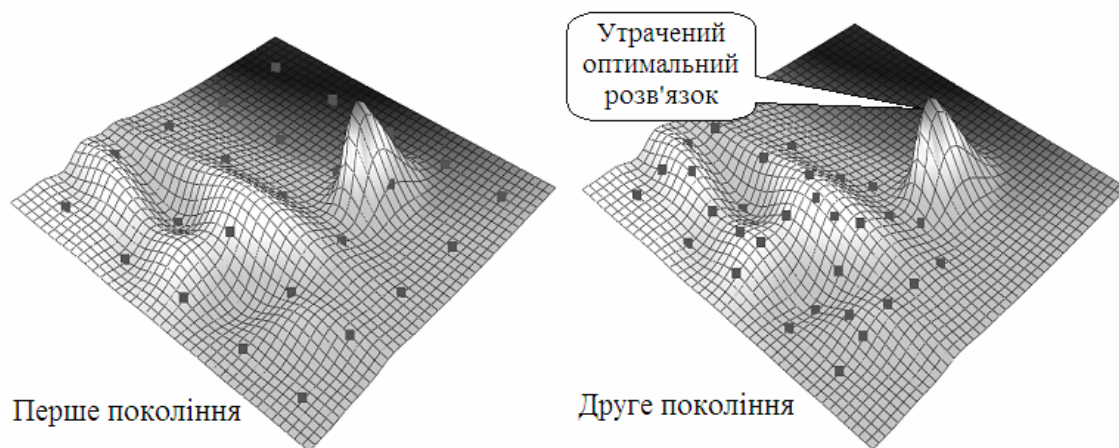


Рис. 3. Генетичний пошук максимуму

**Використання диплоїдного схрещування.** Альтернативою бінарного кодування у генетичних алгоритмах застосовують безпосереднє вираження розв'язків вектором дійсних чисел. Такі методи називають генетичними алгоритмами неперервного кодування [3], або еволюційними стратегіями [1]. Методи схрещування у неперервному генетичному алгоритмі також дають змогу отримати нащадки в околі батьків [1; 7]; тому, починаючи з деякого

покоління, алгоритм може “застрягти” на локальному екстремумі. Для уникнення ефекту у класичному генетичному алгоритмі додають механізм мутації, але параметри його (частку мутації, кількість мутованих генів тощо) вибирають довільно. У разі слабких мутацій алгоритм може не “вийти” з локального екстремуму, а за великої мутації значно збільшується час збіжності алгоритму. У праці [7] запропоновано оригінальний підхід до використання диплоїдного статевого схрещування, запозичений з біології реальних живих організмів. Цей метод дозволяє знаходити глобальний екстремум полімодальної функції, навіть якщо вона змінюється з часом (реалізований механізм мінливості й адаптації). Це досягається кодуванням параметрів двома генами – домінантним і рецесивним, як це відбувається у природі в організмах зі статевим розмноженням. Показано, що такий механізм значно ефективніший від мутацій. Цей метод, однак, придатний лише для бінарного кодування і не може бути використаний у генетичних алгоритмах неперервного коду.

Для уникнення такого ефекту в генетичних алгоритмах застосовують схрещування за законом випадкових чисел, а також механізм “мутацій”, які змінюють невеликий відсоток параметрів (“генів”) за випадковим законом. Зазначимо, що не існує чітких кількісних критеріїв вибору частки мутацій і визначення генів, що їм підлягають, тому на практиці їх зазвичай вибирають довільним чином, “навмання”. Так само довільно вибирають метод схрещування і параметри сполук “батьківських хромосом” (бінарних кодів, що відповідають певним розв’язкам на попередній ітерації). Таким чином, механізм мутацій та існуючі методи схрещування не гарантують успіху в загальному випадку.

Іншим недоліком генетичних алгоритмів є їхня відносно мала швидкість порівняно з градієнтними. Це зумовлено потребою виконувати великі обсяги обчислень з використанням генераторів випадкових чисел і нагромадженням вибірок такого розміру, щоб оптимальний розв’язок був наявний з достатньою імовірністю. Це властиво взагалі всім стохастичним методам [4; 6]. Також у разі застосування генетичних алгоритмів невідомий час, який треба витратити на пошук оптимального розв’язку, і кількість ітерацій (поколінь).

Таким чином, класичні градієнтні методи оптимізації завдяки своїй детермінованості непридатні для пошуку глобального екстремуму багатоходової цільової функції. З іншого боку, еволюційні методи мають елементи випадковості, тому значно більше придатні для цього класу задач (хоча і не гарантують цілковитого успіху). Але недетермінованість у генетичних алгоритмах призводить до значно більшого часу обчислень (хоча й набагато менших, ніж у випадку прямого перебору).

Для подолання цього недоліку пропонується об’єднати стохастичні ознаки еволюційних стратегій і детермінованість градієнтних методів. Основна ідея методу – це застосування градієнтних напрямків цільової функції в точках батьківської пари для оцінювання їх придатності до схрещування і визначення координат точки-нащадка.

Запропонований підхід поєднання градієнтних і неперервних генетичних алгоритмів дозволяє сподіватися на подолання недоліків обох методів.

Сутність методики полягає в застосуванні градієнтної процедури схрещування. Розглянемо двовимірний випадок (рис. 4).

1. На просторі пошуку вибирають рівномірну сітку початкових значень параметрів (нульове “покоління” точок) у кількості  $N$ . Для цих точок розраховують градієнти.

2. Наступне покоління точок знаходять попарним “схрещуванням” градієнтів. Для двовимірного випадку це перетин двох променів, що виходять з кожної пари точок за напрямками градієнтів.

3. Кількість перетинів буде дорівнювати  $C_N^2$ . Із них відразу виключають точки поза межами діапазону пошуку, точки, що виникли на перетині градієнтних прямих у напрямках, протилежних градієнтам, а також близько розміщені точки (замінюють на одну). Чисельність

наступного покоління вибирають із точок, що залишилися, а також з попереднього покоління. Загальна кількість нового покоління не повинна перевищувати  $N$ .

4. На кожній ітерації (на кожному поколінні) для кращих “представників” застосовують звичайні градієнтні методи пошуку екстремумів.

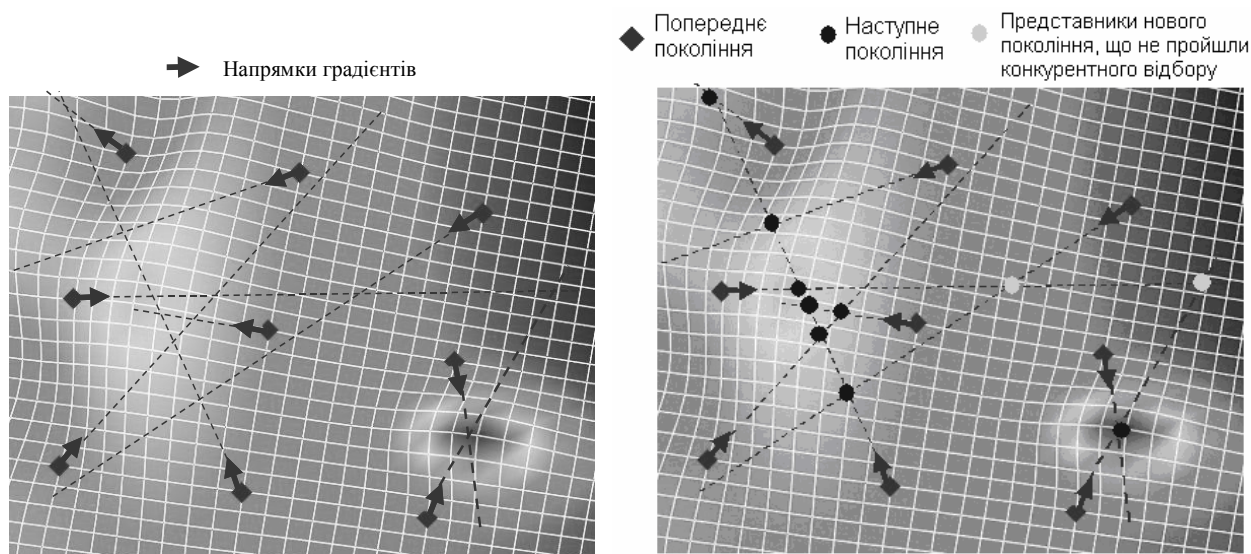


Рис.4. Процедура градієнтного схрещування

Цей підхід також дозволяє отримати нерівномірну сітку значень цільової функції, схожу на сітку з адаптивним кроком, оскільки для кожного покоління обчислюється значення цільової функції як критерії “життєздатності” точки.

**Використання в багатовимірних випадках.** Запропонований метод може бути модифікований для багатовимірного випадку. Якщо цільова функція залежить від  $n$  параметрів, то градієнтні промені у загальному випадку не перетинаються, а схрещуються у  $n$ -вимірному просторі (тобто лежать на *мимобіжних* прямих). Тоді точка-нащадок є *центром зближення* цих прямих (рис. 5). Цю точку можна обчислити як найближчу точку до цих двох прямих (середину найкоротшого відрізка, що сполучає ці прямі). Математичний вираз координат кінців такого відрізка можна отримати мінімізацією відстані між точками  $A$  і  $B$ , що належать цим прямим.

Покажемо, що у  $n$ -вимірному просторі найкоротшу відстань між мимобіжними прямими можна знаходити як розв’язок лінійної системи рівнянь, причому для двовимірного випадку розв’язок дає тотожний нуль (прямі перетинаються).

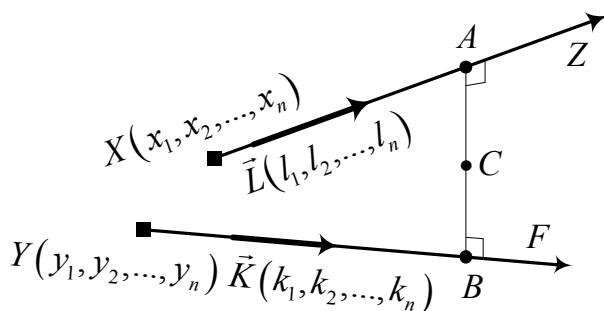


Рис. 5. Знаходження центра зближення градієнтних променів у  $n$ -вимірному просторі

Нехай прямі  $F$  і  $Z$  задано у векторно-параметричній формі:

$$Z = X + sL;$$

$$F = Y + tK,$$

де  $X(x_1, x_2, \dots, x_n)$  і  $Y(y_1, y_2, \dots, y_n)$  – відомі точки, що належать до цих прямих (зручно брати координати “батьківської пари” поточного покоління);  $L(l_1, l_2, \dots, l_n)$  і  $K(k_1, k_2, \dots, k_n)$  – відповідні вектори напрямку прямих (природно взяти для цього обраховані градієнти в точках батьківської пари);  $s$  і  $t$  – дійсні параметри; знаходження точок  $A \in Z$  і  $B \in F$  зводять до знаходження оптимальних значень цих параметрів, мінімізуювши функцію відстані  $AB$ :

$$D(s, t) = \|AB\|^2 = (X + sL - Y - tK)^2.$$

Розв’язок системи лінійних рівнянь

$$\begin{cases} \frac{\partial D}{\partial s} = 0; \\ \frac{\partial D}{\partial t} = 0 \end{cases}$$

забезпечує такі значення параметрів:

$$s_0 = \frac{(K, L)(XY, K) - \|K\|^2 (XY, L)}{\|K\|^2 \cdot \|L\|^2 - (K, L)^2};$$

$$t_0 = \frac{\|L\|^2 (XY, K) - (K, L)(XY, L)}{\|K\|^2 \cdot \|L\|^2 - (K, L)^2}$$

і відповідно координати кінців найкоротшого відрізка  $AB$ :

$$A = X + s_0 L;$$

$$B = Y + t_0 K,$$

де  $XY = (x_1 - y_1, x_2 - y_2, \dots, x_n - y_n)$ .

Координати шуканої точки зближення  $C$ , що є найближчою до мимобіжних градієнтних променів, знаходять як координати середини відрізка  $AB$ .

Найкоротшу відстань між променями  $\|AB\|$  можна використовувати для критерію доцільності схрещування проміжних розв’язків. Якщо ця відстань велика (більша або сумірна з відстанню між схрещуваними точками), то дуже ймовірно, що градієнти вказували напрямки у бік різних локальних екстремумів. У цьому випадку схрещувати цю пару точок не має сенсу. Такий критерій поліпшує генетичний алгоритм, оскільки характеризує не окремий розв’язок на життєздатність і ранг у своєму поколінні, як це робиться у звичайних генетичних алгоритмах [1; 2; 8], а є показником пари розв’язків, що претендують на спільний нащадок.

Іншим важливим показником є знак параметрів  $s$  і  $t$ , оскільки від’ємність цих параметрів вказує на те, що точки  $A$  або  $B$  лежать на прямих у напрямках, протилежних градієнтам.

Якщо градієнтний промінь перетинає край області пошуку, то задача генетичного пошуку максимуму може бути виконана також на цьому краю, тобто на просторі меншої розмірності.

У місцях “скупчення” представників поточної популяції доцільно здійснити пошук максимуму традиційним градієнтним методом, оскільки це “скупчення” вказує на високу ймовірність близького розміщення екстремуму.

Програмне забезпечення для реалізації запропонованого методу зручно реалізовувати із застосуванням об’єктно-орієнтованого підходу, оскільки кожна точка має набір ознак (координати, життєздатність, градієнт), методів (градієнтний пошук, перевірка на приналежність діапазону), а також дружніх функцій (“схрещування”, конкурентне порівняння).

**Висновки.** Синтезовано гібрид двох різних підходів до розв'язання оптимізаційних задач – градієнтних методів і еволюційних алгоритмів. Порівняльний аналіз цих методів дозволяє авторам виокремити позитивні ознаки обох концепцій і об'єднати їх.

Прикладну реалізацію запропонованої технології узагальнено для простору параметрів довільної розмірності. Обґрунтовано вищу ефективність запропонованого методу порівняно з використанням звичайних генетичних алгоритмів із схрещуванням за випадковими законами або за механізмом мутацій

### Список літератури

1. Субботін С. О. Неітеративні, еволюційні та мультиагентні методи синтезу нечіткологічних і нейромережних моделей: монографія / С. О. Субботін, А. О. Олійник, О. О. Олійник // під заг. ред. С. О. Субботіна. – Запоріжжя: ЗНТУ, 2009. – 375 с.
2. Holland J. H. Genetic algorithms and the optimal allocations of trials / J. H. Holland // SIAM Journal Computing. – 2(2). – 1973. – P. 88 – 105.
3. Herrera F. Tackling real-coded Genetic algorithms: operators and tools for the behaviour analysis / F. Herrera, M. Lozano, J. L. Verdegay // Artificial Intelligence Review, Vol. 12, No. 4, 1998. – P. 265 – 319.
4. Жиглявский А. А. Методы поиска глобального экстремума / А. А. Жиглявский, А. Г. Жилинкас – М.: Наука, Физматлит, 1991. – 248 с.
5. Гилл Ф. Практическая оптимизация / Ф. Гилл, У. Мюррей, М. Райт; пер. с англ. — М.: Мир, 1985. – 510 с.
6. Емельянов В. В. Теория и практика эволюционного моделирования / В. В. Курейчик, В. М. Курейчик – М.: Физматлит, 2003. – 432 с.
7. Вороновский Г. К. Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности / Г. К. Вороновский, К. В. Махотило, С. Н. Петрашев, С. А. Сергеев. – Х.: Основа, 1997. – 112 с.
8. Goldberg David E. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Publishing Company, Inc. 1989. – 412p.
9. Сабанин В. Р. Модифицированный генетический алгоритм для задач оптимизации в управлении / Смирнов Н. И., Репин А. И. Exponenta Pro. – 2004. – №3 – 4 (7– 8). – С. 78 – 85.

О. Н. Зудов, А. С. Васильев

### Метод градиентного скрещивания в генетических алгоритмах непрерывного кода

Посвящена модификации одного из методов нахождения абсолютного экстремума функции путем использования генетических алгоритмов. Сделан обзор существующих подходов к использованию генетических алгоритмов и предложен подход к этой проблеме. Приведены математические расчеты для реализации этого алгоритма и проанализована целесообразность использования такого метода.

O. N. Zudov, O. S. Vasylyev

### Gradient crossing method on genetic algorithm of non-breaking code

Article is devoted to modifying one of the methods of finding the absolute extremum function by using genetic algorithms. A review of existing approaches to the use of genetic algorithms and offer their approach to this problem. The mathematical calculations for the implementation of this algorithm and perform analysis on the advisability of using this method.