

**ВИВОДИ**

В роботі пропонується новий генетичний алгоритм з штучним відбором на основі комплексного методу адаптаційної оптимізації, призначений для пошуку екстремуму произвольних функцій великого числа аргументів в умовах суттєвої неопределенності о характері цих функцій. Алгоритм має покращені характеристики порівняно з традиційними генетичними процедурами, прост в реалізації і призначений для використання в Genetic Mining великих масивів текстових документів в режимі послідовної обробки.

**ПЕРЕЧЕНЬ ССЫЛОК**

1. Desjardins G. A genetic algorithm for text mining / G. Desjardins, G. R. Godin, R. Proulx // Sixth International Conference on Data Mining, Text Mining and their Business Applications. – 2005. – P. 133–142.
2. Zhang C. Self-adaptive GA, quantitative semantic similarity measures and ontology-based text clustering / C. Zhang, W. Song, C. Li, W. Yu // 2008. – <http://eprints.rclis.org/14909/> (15.12.2008).
3. Othman R. M. Incorporating semantic similarity measure in genetic algorithm: an approach for searching the gene ontology terms / R. M. Othman, S. Deris, R. M. Illias, H. T. Alashwal, R. Hassan, F. Mohamed // International Journal of Computational Intelligence. – 2006. – № 3. – P. 257–266.
4. Khalessizadeh S. M. Genetic mining: using genetic algorithm for topic based on concept distribution / S. M. Khalessizadeh, R. Zaefarian, S. H. Nasseri, E. Ardil // Proceedings of World Academy of Science, Engineering and Technology – 2006. – 13. – P. 144–147
5. Mani I. Advances in Automatic Text Summarization / I. Mani, M. T. Maybury // Cambridge : MIT Press, 1999. – 442 p.
6. Othman R. M. Automatic clustering of gene ontology by genetic algorithm / R. M. Othman, S. Deris, R. M. Illias, Z. Zakaria, S. M. Mohamad // International Journal of Information Technology. – 2006 – 3.–№ 1. – P.37–46.

7. Rocha F. E. L. A new approach to meaningful learning assessment using concept maps: ontologies and genetic algorithms / F. E. L. Rocha, J. V. da Costa, E. L. Favero // 2004. – <http://cmc.ihmc.us/papers/cmc2004-238.pdf> (15.12.2008)
8. Holland J. H. Genetic algorithms and the optimal allocations of trails // SIAM Journal of Computing. – 1973. – 2. – P. 88–105.
9. Holland J. H. Adaptation in Natural and Artificial Systems. An Introductory Analysis with Application to Biology, Control and Artificial Intelligence. – London : Bradford Book Edition, 1994. – 211 p.
10. Батищев Д. И. Генетические алгоритмы решения экстремальных задач. – Воронеж : Воронежский государственный технический университет, 1995 – 69 с.
11. Курейчик В. М. Генетические алгоритмы. Состояние. Проблемы. Перспективы // Известия РАН. Теория и системы управления. – 1999. – 1. – С. 144–160.
12. Рутковская Д. Нейронные сети, генетические алгоритмы и нечеткие системы / Рутковская Д., Пилиньский М., Рутковский Л. – Москва : Горячая линия – Телеком, 2004. – 452 с.
13. Горский В. Г. Планирование промышленных экспериментов / Горский В. Г., Адлер Ю. П. – Москва : Металлургия, 1974. – 264 с.
14. Химмельблау Д. М. Прикладное нелинейное программирование / Химмельблау Д. М. – Москва : Мир, 1975. – 534 с.
15. Реклейтис Г. Оптимизация в технике : кн. 1 / Реклейтис Г., Рейвиндран А., Рэгсдел К. – Москва : Мир, 1986. – 349 с.

Надійшла 16.03.2009

*Запропоновано новий генетичний алгоритм зі штучним відбором, в основі якого лежить синтез звичайного еволюційного генетичного підходу з ідеями послідовного комплекс-методу пошуку екстремуму довільних функцій багатьох змінних. Алгоритм використовується для кластеризації великих обсягів текстових документів у режимі послідовної обробки.*

*The new genetic algorithm with artificial selection is proposed. The algorithm is based on the synthesis of ordinary evolutionary genetic approach with the ideas of sequential complex-method for extremum searching arbitrary multivariable functions. The algorithm is used for a clusterization of large data collection in a data-processing mode.*

УДК 519.7:004.93

О. О. Олійник, С. О. Субботін

## ОПТИМІЗАЦІЯ НА ОСНОВІ КОЛЕКТИВНОГО ІНТЕЛЕКТУ РОЮ ЧАСТОК З КЕРУВАННЯМ ЗМІНОЮ ЇХНЬОЇ ШВИДКОСТІ

*Досліджено метод оптимізації на основі моделювання поведінки рою часток. Розроблено модифікацію дослідженого методу з керуванням зміною швидкості часток. Проведено експерименти зі знаходження глобального оптимуму багатовимірної функції на основі запропонованої модифікації.*

**ВСТУП**

Градентні методи безумовної оптимізації, що традиційно застосовуються при синтезі моделей складних об'єктів і систем, є високоітеративними та накладають певні вимоги (наприклад, унімодальність, безперервність, монотонність, диференційованість та інші.)

© Олійник О. О., Субботін С. О., 2009

до вигляду цільової функції, а також, як правило, є методами локального пошуку.

Тому актуальним є завдання розроблення нових методів оптимізації, що не накладають подібних вимог на цільові функції. У зв'язку з цим виникає інтерес до методів, заснованих на випадковому пошуку, які характеризуються більшою швидкістю роботи, а також не накладають ніяких вимог на залежності, що досліджуються. До таких методів належать мультиагентні методи інтелектуальної оптимізації, що мають біонічну природу і засновані на моделюванні поведінки колективного інтелекту соціальних живих істот (Swarm Intelligence) [1, 2].

До інтелектуальних методів мультиагентної оптимізації, заснованих на моделюванні суспільного інтелекту, відносяться: метод мурашиних колоній (Ant Colony Optimization, ACO) [3, 4], метод бджолоїної колонії (Bee Colony Optimization, BCO) [5, 6], оптимізація з використанням рою часток (Particle Swarm Optimization, PSO) [7] та інші методи [8]. Дані методи вже ефективно застосовуються для вирішення різних задач оптимізації [3–10].

## 1 ПОСТАНОВКА ЗАВДАННЯ

Метод оптимізації з використанням рою часток (Particle Swarm Optimization, PSO) традиційно використовується для вирішення задачі знаходження глобального оптимуму багатовимірної функції, де показує досить гарні результати [11, 12]. Проте метод характеризується й недоліками, пов'язаними з можливістю занадто сильної зміни швидкості часток, що призводить до збільшення часу пошуку оптимального рішення.

Метою даної роботи є розробка додаткових процедур для методу оптимізації з використанням рою часток для виключення його недоліків, пов'язаних з лавиноподібною зміною швидкості часток, що дозволяє покращити оптимізаційний процес.

Основними завданнями роботи являються:

- дослідження методу оптимізації з використанням рою часток;
- виявлення основних недоліків досліджуваного методу;
- створення методу оптимізації на основі моделювання поведінки рою часток, що враховує головні недоліки базового методу;
- розробка програмного забезпечення, що реалізує запропонований метод, та його тестування.

## 2 МЕТОД ОПТИМІЗАЦІЇ 3 ВИКОРИСТАННЯ РОЮ ЧАСТОК

В PSO-методі, особини, що називаються частками, переміщуються в багатовимірному просторі рі-

шень. Зміни координат часток усередині простору пошуку обумовлюються природною соціально-психологічною тенденцією часток конкурувати між собою. Отже, зміни в стані частки залежать від досвіду й знань її сусідів. У цьому випадку, слово «знання» є синонімом «інформації». Результат такого моделювання полягає в тому, що процес пошуку мотивує частки недетермінованим чином повертатися в оптимальні ділянки простору рішень.

Особини в рої часток мають дуже просту поведінку: вони прагнуть перевершити досягнення сусідніх часток і поліпшити власні. Таким чином, емергентна властивість даної системи полягає в дослідженні оптимальних ділянок багатовимірному простору пошуку.

PSO метод керує роєм часток і кожна частка являє собою потенційне рішення [28, 29]. За аналогією з еволюційними стратегіями, рій можна трактувати як популяцію, а частку як індивіда (або хромосому). Таким чином, частка «літає» у багатовимірному просторі рішень, та її позиція визначається виходячи із власного досвіду і досвіду своїх сусідів.

За допомогою  $x_i(t)$  позначимо позицію частки  $i$  у просторі пошуку у момент часу  $t$  ( $t$  позначає дискретні значення часу). Позиція частки змінюється шляхом додавання швидкості  $v_i(t)$  до поточної позиції:

$$x_i(t + 1) = x_i(t) + v_i(t + 1).$$

Початковий стан визначається в такий спосіб:  $x_i(0) = U(x_{\min}, x_{\max})$ , де  $U(a, b)$  є функцією генерації випадкових чисел з діапазону  $[a, b]$ . Дана формула являє собою вектор швидкості й визначає сам оптимізаційний процес, а також відображує використання, як отриманих знань частки, так й обмін інформацією із сусідніми частками. Власні знання самої частки, що також називаються *когнітивним компонентом* формули швидкості, прямо пропорційні поточній відстані частки від її найкращого положення, що було знайдено з моменту старту її життєвого циклу. А обмін інформацією даної особини з іншими є *соціальним компонентом* формули швидкості.

Відомі два підходи, що є різновидами базового PSO-методу: *gbest* та *lbest*, які відрізняються ступенем зв'язаності часток у просторі пошуку.

В *gbest* PSO-методі швидкість частки розраховується за формулою:

$$v_{ij}(t + 1) = v_{ij}(t) + c_1 r_{1j}(t)[y_{ij}(t) - x_{ij}(t)] + c_2 r_{2j}(t)[y_j^*(t) - x_{ij}(t)],$$

де  $v_{ij}(t)$  – швидкість частки  $i$  у вимірі  $j$  ( $j = 1, \dots, n_x$ ) у момент часу  $t$ ;  $x_{ij}(t)$  – позиція частки  $i$  у вимірі  $j$ ;  $y_j^*(t)$  – найкраща досягнута позиція рою (у просторі рішень);  $c_1$  та  $c_2$  – позитивні константи прискорення, які використовуються для варіювання ваг

когнітивного й соціального компонентів швидкості частки відповідно;  $r_{1j}(t), r_{2j}(t) = U(0, 1)$  є випадковими значеннями з діапазону  $[0, 1]$ . Ці випадкові величини привносять стохастичний елемент у роботу методу.

Величина  $y_i$  відображає найкращу позицію частки  $i$ , що вона відвідувала, починаючи з першої ітерації. Наступна оптимальна позиція частки  $i$  у момент часу  $t + 1$  розраховується за формулою

$$y_i(t+1) = \begin{cases} y_i(t), & \text{якщо } f(x_i(t+1)) \geq f(y_i(t)); \\ x_i(t+1), & \text{якщо } f(x_i(t+1)) < f(y_i(t)), \end{cases}$$

де  $f: \mathfrak{R}^{n_x} \rightarrow \mathfrak{R}$  – фітнес-функція,  $\mathfrak{R}^{n_x}$  – множина значень незалежних змінних,  $\mathfrak{R}$  – множина значень оптимізованої величини. Також, як і в еволюційних підходах, вона є мірою близькості даного рішення до оптимального, також фітнес-функція визначає продуктивність, або якість частки.

В lbest PSO-методі швидкість розраховується за формулою:

$$v_{ij}(t+1) = v_{ij}(t) + c_1 r_{1j}(t)[y_{ij}(t) - x_{ij}(t)] + c_2 r_{2j}(t)[y_{ij}^*(t) - x_{ij}(t)],$$

де  $y_{ij}^*(t)$  – найкраща позиція, знайдена сусідами  $i$ -ї частки у вимір  $j$ . Локально найкраща позиція частки  $y_i^*$ , тобто краща позиція, знайдена в сусідстві  $N_i$ , що визначається як

$$y_i^*(t+1) \in \{N_i | f(y_i^*(t+1))\} = \min\{f(x)\}, \forall x \in N_i,$$

де сусідство визначається за формулою

$$N_i = \{y_{i-n_{N_i}}(t), y_{i-n_{N_i}+1}(t), \dots, y_{i-1}(t), y_i(t), y_{i+1}(t), \dots, y_{i+n_{N_i}}(t)\}$$

для сусідства розмірністю  $n_{N_i}$ .

Метод PSO може бути описаний у вигляді послідовності таких кроків.

Крок 1. Якщо умови закінчення виконано, тоді закінчити роботу методу; у протилежному випадку – перейти до кроку 2.

Крок 2. Створити й ініціалізувати  $n_x$ -вимірний рій.

Крок 3. Встановити:  $i = 1$ .

Крок 4. Визначення локально найкращої позиції. Якщо виконано умову  $f(x_i) < f(y_i)$ , тоді встановити:  $y_i = x_i$ .

Крок 5. Визначення глобально найкращої позиції. Встановити:  $y^* = y_i$ .

Крок 6. Встановити:  $i = i + 1$ .

Крок 7. Якщо  $i < n_s$ , тоді виконати перехід до кроку 3, у протилежному випадку – до кроку 8.

Крок 8. Встановити:  $i = 1$ .

Крок 9. Оновити швидкість частки, використовуючи формулу gbest PSO або lbest PSO.

Крок 10. Оновити позицію частки.

Крок 11. Встановити:  $i = i + 1$ .

Крок 12. Якщо  $i < n_s$ , тоді виконати перехід до кроку 9, у протилежному випадку – до кроку 1.

Закінчення виконання PSO-методу відбувається, коли виконується хоча б один із критеріїв зупинення. Існують два аспекти, які повинні бути враховані при виборі подібних мір.

1. Умова зупинення не повинна спричиняти передчасну збіжність PSO-методу. У протилежному випадку будуть отримані субоптимальні рішення.

2. При перевірці критеріїв зупинення фітнес-функція має викликатися мінімальну кількість разів, оскільки від цього сильно залежить швидкість роботи PSO-методу.

Як критерії зупинення використовують:

- досягнення максимальної кількості ітерацій;
- знаходження задовільного рішення;
- вихід при відсутності поліпшень протягом багатьох ітерацій;
- вихід за умови, що перша похідна фітнес-функції в околиці частки або часток дорівнює нулю.

### 3 МЕТОД PSO З КЕРУВАННЯМ ЗМІНОЮ ШВИДКОСТІ ЧАСТОК

Ефективний оптимізаційний метод має володіти двома важливими властивостями: повністю досліджувати простір рішень задачі, і фіксувати пошук біля потенційних оптимумів. В PSO-методі це визначається формулою швидкості часток.

В ранніх реалізаціях PSO-методу було виявлено, що швидкості можуть занадто швидко зростати, особливо швидкості тих часток, які знаходяться далеко від свого власного оптимуму або загального оптимуму своїх сусідів. У результаті, такі частки можуть залишати простір пошуку, що дуже негативно впливає на роботу методу в цілому. Для запобігання подібним ситуаціям варто контролювати діапазон зміни швидкостей часток. Якщо швидкість частки перевищує максимально допустиму, то її варто знизити до дозволеного рівня. Нехай  $V_{\max,j}$  – максимально припустима швидкість частки у вимірі  $j$ . Тоді швидкість частки пропонується змінювати у такий спосіб:

$$y_{ij}(t+1) = \begin{cases} v'_{ij}(t+1), & \text{якщо } v_{ij}(t+1) = V_{\max,j}; \\ V_{\max,j}, & \text{якщо } v'_{ij}(t+1) \geq V_{\max,j}, \end{cases}$$

де значення  $v'_{ij}$  розраховується у відповідності до gbest PSO або lbest PSO.

Величина  $V_{\max,j}$  є досить важливою, оскільки вона контролює загальні масштаби пошуку. Більші значення змушують PSO-метод досліджувати простір за-

дачі більш ретельно. Якщо значення  $V_{\max,j}$  є досить невеликим, то PSO-метод може взагалі не покинути локальних ділянок, а також збільшується кількість ітерацій, потрібних для досягнення оптимуму.

Виходячи з зазначеного робимо висновки, що  $V_{\max,j}$  варто підбирати таким чином, щоб балансувати між швидким і повільним пересуваннями часток, і між фіксацією пошуку й загальним дослідженням простору. Пропонується розраховувати  $V_{\max,j}$  у такий спосіб:

$$V_{\max,j} = \delta(x_{\max,j} - x_{\min,j}),$$

де  $x_{\max,j}$  та  $x_{\min,j}$  – величини, які, відповідно, знизу та зверху обмежують діапазон значень виміру  $j$ , а  $\delta \in (0,1]$ . Очевидно, що величина  $\delta$  є проблемно-залежною.

Відзначимо дві важливі властивості запропонованої модифікації PSO-методу.

Величина  $V_{\max,j}$  обмежує не простір пошуку, всередині якого пересуваються частки, а тільки швидкості часток (точніше діапазон їхньої зміни за одну ітерацію).

Максимальна швидкість часток визначається для кожного виміру окремо й обумовлюється тільки його фізичною природою.

Виходячи з запропонованого підходу можна відзначити, що обмеження швидкості частки може викликати зміну напрямку її пересування, що може як позитивно, так і негативно відбиватися на пошуку оптимуму. Цей ефект зображено на рис. 1. У цьому випадку,  $x_i(t+1)$  – це позиція  $i$ -ї частки без обмежень за швидкістю. А величина  $x'_i(t+1)$  вже відображає позицію частки після того, як швидкість стала рівною  $V_{\max,j}$ .

Небажаний ефект може виникнути у випадку, якщо швидкості всіх часток стануть рівними  $V_{\max,j}$ .

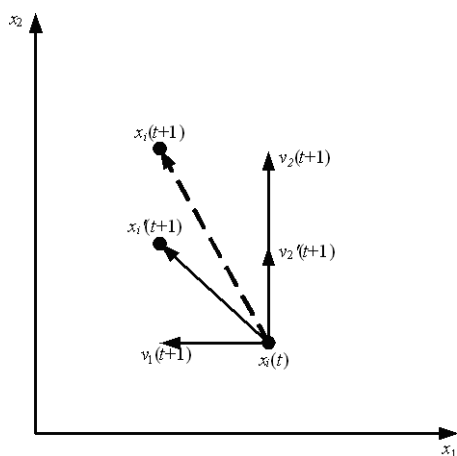


Рисунок 1 – Ефект використання обмеження швидкості частки

Для його запобігання можна запропонувати з ходом ітерацій зменшувати значення  $V_{\max,j}$ . Тоді модифікований PSO-метод буде послідовно звужувати масштаби дослідження простору пошуку, що робить оптимізаційний процес більш ефективним.

Тоді метод PSO з керуванням зміною швидкості часток може бути описаний у вигляді послідовності наступних кроків.

Крок 1. Якщо умови закінчення виконано, тоді припинити роботу методу; у протилежному випадку – перехід до кроку 2.

Крок 2. Створити й ініціалізувати  $n_x$ -вимірний рій. Також розрахувати максимально допустимі швидкості часток:

$$V_{\max,j} = \delta(x_{\max,j} - x_{\min,j}).$$

Крок 3. Встановити:  $i = 1$ .

Крок 4. Визначити локально найкращу позицію. Якщо виконано умову  $f(x_i) < f(y_i)$ , тоді встановити  $y_i = x_i$ .

Крок 5. Визначити глобально найкращу позицію. Встановити:  $y^* = y_i$ .

Крок 6. Встановити:  $i = i + 1$ .

Крок 7. Якщо  $i < n_s$ , то виконати перехід до кроку 3, у протилежному випадку – до кроку 8.

Крок 8. Встановити:  $i = 1$ .

Крок 9. Оновити швидкість частки, використовуючи запропонований підхід:

$$v_{ij}(t+1) = \begin{cases} v'_{ij}(t+1), & \text{якщо } v_{ij}(t+1) = V_{\max,j}; \\ V_{\max,j}, & \text{якщо } v'_{ij}(t+1) \geq V_{\max,j}. \end{cases}$$

Крок 10. Оновити позицію частки.

Крок 11. Встановити:  $i = i + 1$ .

Крок 12. Якщо  $i < n_s$ , то виконати перехід до кроку 9, у протилежному випадку до кроку 13.

Крок 13. Зменшити величину  $\delta$ :  $\delta = \delta^\alpha$ , де  $\alpha \in (0; 1)$ . Зменшення величини  $\delta$  призводить до того, що на наступній ітерації буде зменшена й величина  $V_{\max,j}$ . Перехід до кроку 1.

Розроблений метод PSO з керуванням зміною швидкості часток виключає небажаний ефект занадто великої зміни швидкості часток, що дозволяє покращити оптимізаційний процес за рахунок підвищення швидкості роботи методу.

#### 4 ЕКСПЕРИМЕНТИ Й РЕЗУЛЬТАТИ

Запропонований метод PSO з керуванням зміною швидкості часток було програмно реалізовано у середовищі пакету Matlab 7.0.

Використовуючи створене програмне забезпечення, проводилися експерименти зі знаходження оптимального значення функцій з метою дослідження ефективності запропонованого методу PSO у порів-

Таблиця 1 – Тестові функції та знайдені для них оптимуми

Функція	Фактичний оптимум	Оптимум, отриманий базовим PSO	Оптимум, отриманий запропонованим методом
$f_1 = \sum_{i=1}^n  x_i \cdot \sin(x_i) + 0,1 \cdot x_i , n = 2$ $-5 \leq x_1 \leq 5;$ $-5 \leq x_2 \leq 5$	min: $f_1^*(0; 0) = 0$	$x_1 = 0,004;$ $x_2 = 0,008;$ $f_1^* = 0,0013$	$x_1 = 0,0009;$ $x_2 = 0,005;$ $f_1^* = 0,00062$
$f_2 = 20 + e - 20 \exp\left(-0,2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right), n = 2$ $-5 \leq x_1 \leq 5;$ $-5 \leq x_2 \leq 5$	min: $f_2^*(0; 0) = 0$	$x_1 = 0,000299;$ $x_2 = 0,0002;$ $f_2^* = 0,0010$	$x_1 = 0,00005;$ $x_2 = 0,00028;$ $f_2^* = 0,000807$

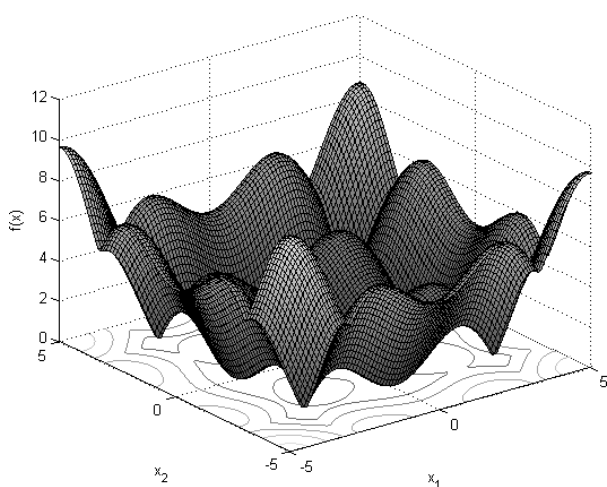


Рисунок 2 – Графік тестової функції  $f_1$

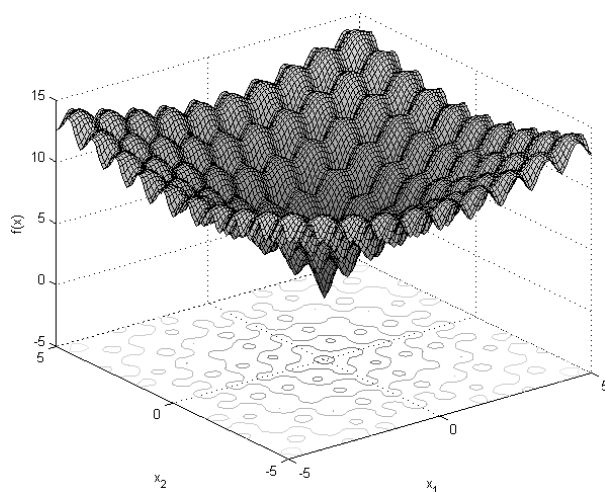


Рисунок 3 – Графік тестової функції  $f_2$

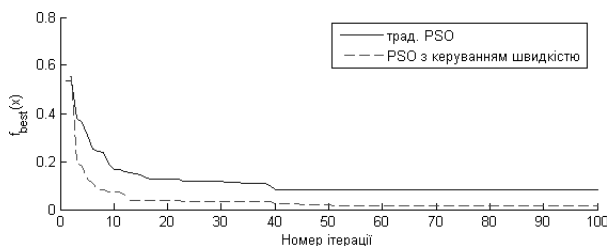


Рисунок 4 – Результати роботи для тестової функції  $f_1$

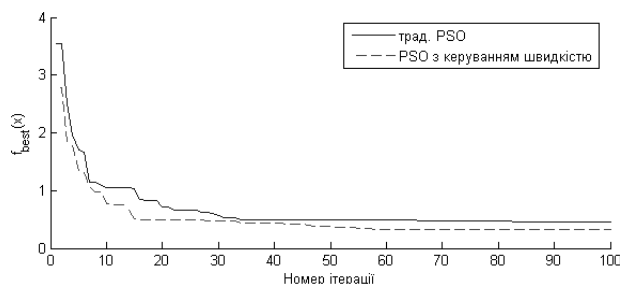


Рисунок 5 – Результати роботи для тестової функції  $f_2$

янні з традиційним методом PSO. В експериментах використовувались тестові функції, які подано в табл. 1, де також наведено отримані результати.

Головною особливістю запропонованих тестових функцій є їхня полімодальність, тобто велика кількість локальних оптимумів, що дозволяє перевірити, наскільки схильний запропонований метод до зацілювання в локальних оптимумах.

Графіки тестових функцій подано на рис. 2 та 3.

Результати роботи розробленого програмного забезпечення для відповідних функцій зображено на рис. 4 та 5.

Виходячи з наведених рисунків, можна зробити висновок, що запропонований метод знаходить більш точно точку оптимуму. При цьому знаходження оптимального значення відбувається швидше у порівнянні з базовим методом оптимізації з використанням рою часток.

## ВИСНОВОК

У роботі вирішено актуальне завдання прискорення роботи методу оптимізації з використанням рою часток.

Наукова новизна роботи полягає в тому, що метод оптимізації з використанням рою часток, на основі якого розроблено метод з керуванням зміною швидкості часток, дістав подальшого розвитку. Запропоновані процедури враховують недоліки базового методу, пов'язані з можливістю занадто великою зміною швидкості часток, що негативно впливає на оптимізаційний процес.

Практична цінність результатів роботи полягає в тому, що розроблено програмне забезпечення, яке реалізує запропонований метод оптимізації на основі моделювання поведінки рою часток з керуванням зміною швидкості часток, і дозволяє провести експерименти по знаходженню оптимуму функцій. Запропонований метод може бути використаний у будь-якій задачі оптимізації, де необхідно знайти значення змінних, за яких залежність, що досліджується, набуває оптимального значення.

Роботу виконано за держбюджетною темою «Науково-методичні основи та математичне забезпечення для автоматизації і моделювання процесів керування та підтримки прийняття рішень на основі процедур розпізнавання й еволюційної оптимізації в нейромрежному і нечіткологічному базисах» (№ держ. реєстрації 0106U008621).

## ПЕРЕЛІК ПОСИЛАНЬ

1. Beni G. Swarm Intelligence / Beni G., Wang J. // Annual Meeting of the Robotics Society : proceedings of Seventh International Conference. – Tokyo : RSJ Press, 1989. – P. 425–428.
2. Bonabeau E. Swarm Intelligence: From Natural to Artificial Systems / Bonabeau E., Dorigo M., Theraulaz G. – New York : Oxford University Press, 1999. – 320 p.
3. Dorigo M. The Ant System: Optimization by a Colony of Cooperating Agents / Dorigo M., Maniezzo V., Color-

- ni A. // IEEE Transactions on Systems, Man, and Cybernetics. – 1996. – Part B, № 26(1). – P. 29–41.
4. Олейник А. А. Сравнительный анализ методов оптимизации на основе метода муравьиных колоний / Олейник А. А. // Комп'ютерне моделювання та інтелектуальні системи : збірник наукових праць / за ред. Д. М. Пізи, С. О. Субботіна. – Запоріжжя : ЗНТУ, 2007. – С. 147–159.
5. Camazine S. A Model of Collective Nectar Source by Honey Bees: Self-organization Through Simple Rules / Camazine S., Sneyd J. // Journal of Theoretical Biology. – 1991. – № 149. – P. 547–571.
6. Sumpter D. J. T. Formalising the Link between Worker and Society in Honey Bee Colonies / Sumpter D. J. T., Broomhead D. S. // Lecture Notes In Computer Science : Proceedings of the First International Workshop on Multi-Agent Systems and Agent-Based Simulation (MABS '98). – Berlin : Springer, 1998. – P. 95–110.
7. Kennedy J. Particle Swarm Optimization / Kennedy J., Eberhart R. C. // International Conference on Neural Networks : Proceedings of the IEEE. – NJ : IEEE Press, 1995. – P. 1942–1948.
8. De Castro L. N. Artificial Immune Systems. Part I. Basic Theory And Applications / De Castro L. N., Von Zuben F. J. // Technical Report No. Rt Dca 01/99. – Brazil : Feec/Unicamp, 1999. – 95 p.
9. Colorni A. Ant System for Job-shop Scheduling / Colorni A., Dorigo M., Maniezzo V., Trubian M. // Belgian Journal of Operations Research, Statistics and Computer Science (JORBEL). – 1994. – № 34. – P. 39–53.
10. Субботин С. А. Отбор информативных признаков на основе модифицированного метода муравьиных колоний / Субботин С. А., Олейник А. А., Яценко В. К. // Радиоелектроніка та інформатика. – 2006. – № 1. – С. 65–69.
11. Engelbrecht A. Computational Intelligence: an Introduction / Engelbrecht A. – Sidney : John Wiley and Sons, 2007. – 597 p.
12. Abraham A. Swarm Intelligence in Data Mining / Abraham A., Grosan G. – Berlin : Springer, 2006. – 267 p.

Надійшла 2.10.2008  
Після доробки 2.12.2008

*Исследован метод оптимизации на основе моделирования поведения роя частиц. Разработана модификация исследованного метода с управлением изменением скорости частиц. Проведены эксперименты по нахождению глобального оптимума многомерной функции на основе предложенной модификации.*

*A particle swarm optimization method has been considered. A modification of PSO-method with velocity control has been developed. The experiments on finding global optimum for multidimensional function on the basis of suggested modification has been carried out.*

УДК 621.365.036

С. Г. Удовенко, А. А. Шамраев, Г. Дибс

## ЦИФРОВОЕ НЕЙРО-НЕЧЕТКОЕ УПРАВЛЕНИЕ БЛОКАМИ ЛИНЕЙНОГО СИЛЬНОТОЧНОГО УСКОРИТЕЛЯ ЭЛЕКТРОНОВ

*Предложена структура и разработаны методы цифрового управления блоками линейного ускорителя с применением нечетких и нейросетевых моделей. Приведены результаты моделирования, подтверждающие эффективность рассмотренных методов.*

© Удовенко С. Г., Шамраев А. А., Дибс Г., 2009

## ВВЕДЕНИЕ

Линейный резонансный ускоритель электронов представляет собой сложную электрофизическую установку, в которой в течение длительного времени