

## ПРИКЛАДНІ СИСТЕМИ КОЛЕКТИВНОГО ІНТЕЛЕКТУ В ЕКОНОМІЦІ

**Анотація.** У статті обґрунтовано напрями використання PSO-методу як окремого рішення оптимізаційних проблем в економіці, заснованих на моделюванні інтелектуальної поведінки колоній. Показано, що PSO-метод також можна застосовувати як метод навчання нейронної мережі будь-якої архітектури.

**Ключові слова:** оптимізаційні проблеми, нейронні мережі, штучний інтелект, інтелектуальна поведінка.

**Summary.** In the article grounded directions of the use of PSO-method as separate decision of optimizations problems in the economy of the colonies based on the design of intellectual conduct. It is shown that PSO-method it is also possible to apply as method of studies of neuron network of any architecture.

**Key words:** optimizations problems, neurons networks, artificial intelligence intellectual conduct.

**Постановка проблеми.** Розробка і створення інтелектуальних інформаційних систем нового покоління, зокрема інструментальних комплексів інформаційно-когнітивної підтримки процесів синтезу, інтегрування й актуалізації знань, є одним із головних напрямів фундаментальних наукових досліджень у галузі штучного інтелекту (ШІ), що характеризуються інноваційністю в багатьох галузях економіки [1].

**Аналіз останніх досліджень і публікацій.** Теоретичним підґрунтям статті стали розробки вітчизняних авторів А. А. Олійника [2] і С. А. Суботіна [3], які запропонували мультиагентний метод виділення інформативних ознак для синтезу моделей об'єктів управління й порівняльний аналіз методів оптимізації на базі методу мурашиних колоній.

**Мета статті** — обґрунтувати напрями використання PSO-методу як окремого рішення оптимізаційних проблем в економіці, заснованих на моделюванні інтелектуальної поведінки колоній.

**Виклад основного матеріалу.** Новим напрямом розвитку методів штучного інтелекту є мультиагентні методи інтелектуальної оптимізації в економіці, що моделюють колективний інтелект (КІ) суспільних тварин, комах та інших живих істот, — методи Swarm Intelligence. Цей напрям штучного інтелекту є молодим і ще мало дослідженим, проте мультиагентні методи демонструють гарні результати при вирішенні різних задач оптимізації, що говорить про перспективність його розвитку.

Головною особливістю мультиагентних методів КІ є їхня біонічна природа — аналіз методів колоній, призначених для розв'язання задач оптимізації, зокрема методу еволюційної оптимізації (у тому числі генетичні алгоритми), методів мурашиних та бджолиних колоній. При цьому слід зазначити, що ці методи моделюють поведінку груп різних соціальних тварин, комах та інших живих істот, яка дозволяє цим групам вирішувати різні важкі практичні задачі в природі, що свідчить про ефективність їхньої поведінки, а отже, й про ефективність роботи зазначених методів.

Тому актуальним є вивчення методів, можливих галузей їхнього застосування, а також розробка нових математичних моделей, заснованих на поведінці колективних тварин, в аспекті вирішення оптимізаційних задач економіки, до яких ці методи ще не застосовувалися [4].

До мультиагентних розподілених систем штучного інтелекту відносяться мультиагентні методи інтелектуальної оптимізації (методи колективного інтелекту, Swarm Intelligence). Ці методи мають біонічну природу, тобто вони засновані на моделюванні поведінки комах, птахів, тварин і т. ін, поведінка яких носить колективний характер, за рахунок чого досягається так званий колективний інтелект.

При реалізації даних методів використовуються парадигма агентно-орієнтованого програмування, заснована на моделюванні суспільного інтелекту, до якого відносяться: метод мурашиних колоній (Ant Colony Optimization, ACO) [5], метод бджолиної колонії (Bee Colony Optimization, BCO) [6], оптимізація за допомогою рою часток (Particle Swarm Optimization, PSO) [7] та інші методи. Перелічені методи вже ефективно застосовуються для вирішення різних задач: ACO — для вирішення задач комівояжера, задачі календарного планування, відбору інформативних ознак, кластеризації; BCO — для вирішення задачі календарного планування, задачі комівояжера, транспортного завдання.

Прикладну мультиагентну систему (ПМАС) у загальному випадку можна подати у вигляді множини із трьох елементів: Агенти, Середовище, Зв'язки між Середовищем та Агентами:

$$MAC = \langle \text{Агенти}, \text{Середовище}, \text{Зв'язки} \rangle.$$

Кожен  $i$ -й Агент <sub>$i$</sub>  описується за допомогою множини чотирьох елементів: Стан <sub>$i$</sub> , Вхід <sub>$i$</sub> , Вихід <sub>$i$</sub> , Процес <sub>$i$</sub> :

$$\text{Агент}_i = \langle \text{Стан}_i, \text{Вхід}_i, \text{Вихід}_i, \text{Процес}_i \rangle,$$

де Стан <sub>$i$</sub>  — це множина змінних, що повністю визначають агента; Вхід <sub>$i$</sub>  та Вихід <sub>$i$</sub>  — підмножина

*Стану*, елементи яких пов'язані із середовищем; *Процес*, — автономний метод, що виконує відповідні зміни над *Станом*.

*Середовищем* є множина з двох елементів:

$$\text{Середовище} = \langle \text{Стан}_c, \text{Процес}_c \rangle,$$

де індекс *c* інформує про те, що ці елементи відносяться до середовища, а не до будь-якого іншого агента.

Важливою особливістю такого подання *Середовища* є те, що воно самостійно активне, оскільки містить свій власний *Процес*, котрий може змінювати *Стан*, незалежно від агентів, що входять до цього *Середовища*.

*Вхід*, і *Вихід*, різноманітних агентів пов'язані з елементами *Стану*, але середовище не відрізняє, які з елементів *Стану* знаходяться у залежності з ними. Відсутність *Входу* та *Виходу* у *Середовищі* означає, що воно, на відміну від агентів, є необмеженим. Якби до *Середовища* були додані *Вхід* і *Вихід*, то це означало б, що середовище обмежене та є фактично агентом високого рівня. Таким чином може моделюватися взаємодія агентів високого й низького рівнів.

У відповідності до наведеної концепції подання мультиагентних систем можуть створюватися різноманітні типи мультиагентних систем в економіці, при цьому така концепція організації систем може бути використана не тільки в програмних (штучних) системах, а й у природних системах. Така класифікація мультиагентних систем наведена на рис. 1.

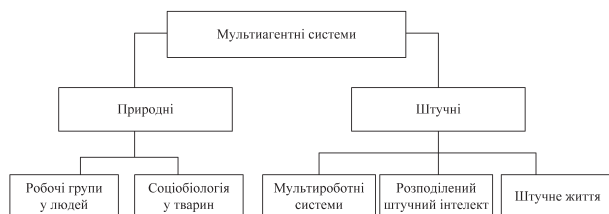


Рис. 1. Класифікація мультиагентних систем

Оптимізація з використанням рою часток (Particle Swarm Optimization, PSO) — це метод пошуку, що базується на понятті популяції і моделює поведінку птахів у зграї й косяків риб. Первісною метою використання концепції рою часток була графічна імітація гарного й непередбачуваного руху птахів або риб у зграї з метою виявлення базових принципів, завдяки яким птахи літають (риби плавають) синхронно і вміють змінювати напрямок руху із перегрупованням в оптимальні формації. Із тих часів концепція розвинулася в простий і перспективний оптимізаційний метод.

У PSO-методі особини, що називаються частками, переміщуються в багатовимірному просторі рішень. Зміни координат часток усередині простору пошуку обумовлюються природною соціаль-

но-психологічною тенденцією часток конкурувати між собою. Отже, зміни у 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 у рої часток мають дуже просту поведінку: вони прагнуть перевершити досягнення сусідніх часток і поліпшити власні. Таким чином, емергентна властивість цієї системи міститься у дослідженні оптимальних ділянок багатовимірного простору пошуку.

PSO-метод керує роєм часток, і кожна частка являє собою потенційне рішення.

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

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

Початковий стан визначається у такий спосіб:  $x_i(0) = U(x_{\min}, x_{\max})$ , де  $U(a, b)$  є функцією генерації випадкових чисел з діапазону  $[a, b]$ . Формула (1) являє собою вектор швидкості й визначає сам оптимізаційний процес, а також відображає використання як отриманих знань частки, так і 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 цієї особини з іншими є *соціальним компонентом* формули швидкості.

Історично було розроблено два підходи, які фактично є різновидами базового PSO-методу: *gbest* і *lbest*; вони відрізняються ступенем зв'язаності часток у просторі пошуку.

У різновиді *gbest* PSO-методу кожна частка зв'язана з усім роєм. Частки утворюють так звану соціальну мережу, що у *gbest* відповідає топології типу «зірка» (рис. 2). Кожна частка може взаємодіяти з усіма іншими частками, і вона тяжіє до кращого рішення усього рою. Частка імітує загальне оптимальне рішення, тому її швидкість залежить від інформації, одержуваної від усіх інших. У цьому випадку соціальним компонентом швидкості є найкраща досягнута позиція рою (у просторі рішень), і вона позначається як  $y^*(t)$ .

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

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

де  $v_{ij}(t)$  — це швидкість частки  $i$  у вимірі  $j$  ( $j = 1, \dots, n_x$ ) у момент часу  $t$ ;  $x_{ij}(t)$  — це позиція частки  $i$  у вимірі

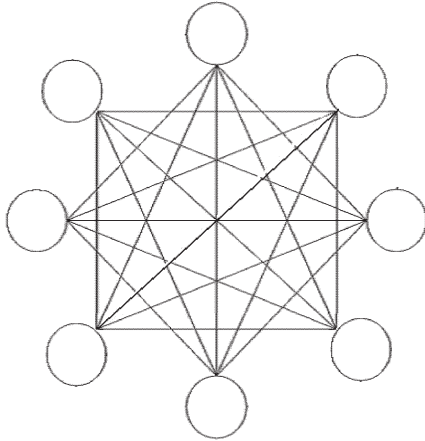


Рис. 2. Топологія типу «зірка»

$j$ ;  $c_1$  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:

$$y_i(t+1) = \begin{cases} y_i(t), & \text{якщо } f(x_i(t+1)) \geq f(y_i(t)); \\ x_i(t+1), & \text{якщо } f(x_i(t+1)) < f(y_i(t)), \end{cases} \quad (3)$$

де  $f: \mathfrak{R}^n \rightarrow \mathfrak{R}$  — фітнес-функція, де  $\mathfrak{R}^n$  — множина значень незалежних змінних,  $\mathfrak{R}$  — множина значень величини, що оптимізується. Як і в еволюційних підходах, вона є мірою близькості даного рішення до оптимального, також фітнес-функція визначає продуктивність, або якість, частки.

Глобальний найкращий оптимум  $y^*(t)$  у момент часу  $t$  визначається як:

$$y^*(t) \in \{y_0(t), \dots, y_{n_s}(t)\} | f(y^*(t)) = \min\{f(y_0(t)), \dots, f(y_{n_s}(t))\}, \quad (4)$$

де  $n_s$  — загальна кількість часток у рої. Важливо відзначити, що відповідно до формули (4)  $y^*$  — це найкраща позиція, що була знайдена кожною із часток. Глобальний оптимум також може бути розрахований на ґрунті інформації про частки із цього рою:

$$y^*(t) = \min\{f(x_0(t)), \dots, f(x_{n_s}(t))\}. \quad (5)$$

Локально найкраща позиція частки  $y_i^*$ , тобто краща позиція, знайдена в сусідстві  $N_i$  та визначається як:

$$y_i^*(t+1) \in \{N_i | f(y_i^*(t+1))\} = \min\{f(x)\}, \forall x \in N_i, \quad (6)$$

де сусідство визначається за формулою:

$$N_i = \{y_{i-n_{N_i}}(t), y_{i-n_{N_i}+1}(t), \dots, y_{i-1}(t), y_i(t), y_{i+1}(t), \dots, y_{i+n_{N_i}}(t)\}, \quad (7)$$

для сусідства розмірністю  $n_{N_i}$ .

Виконаємо порівняння різновидів базового PSO-методу. Включимо в аналіз також модифікований метод, що гнучко регулює швидкості часток у просторі пошуку. Порівняльний аналіз проведемо на базі найбільш важливих і істотних розходжень, виявлених при їхньому дослідженні, а також на базі запропонованих критеріїв. Аналіз наведено у табл. 1.

Таблиця 1

Порівняльний аналіз різновидів PSO-методу

Критерії	Методи			
	Метод gbest PSO	Метод lbest PSO	PSO-метод із керуванням зміною швидкості часток	Багатокритеріальний PSO-метод
Адаптивність	Висока	Висока	Висока	Низька
Масштабованість	Середня	Середня	Середня	Середня
Гнучкість моделі	Середня	Висока	Висока	Низька
Швидкість збіжності	Висока	Низька	Середня	Сильно залежить від вирішуваної задачі
Якість оптимізації	Середня	Висока	Висока	Низька
Помилкостійкість	Висока	Висока	Висока	Низька
Кількість обчислень фітнес-функції	Середня	Низька	Середня	Висока

Різновиди *lbest* і *gbest* базового PSO-методу подібні в тому плані, що в обох випадках вони рухаються до глобального оптимуму при оновленні компонентів швидкості.

Але між цими підходами існує два важливих розходження, які впливають на збіжність:

— через більший ступінь зв'язності часток між собою (топологія «зірка») *gbest* PSO-метод

сходиться швидше; однак швидка збіжність веде до менш ретельного дослідження простору рішень;

— різновид *lbest* PSO-методу має менший шанс потрапити до локального оптимуму і знайти, таким чином, тільки субоптимальне рішення; з іншого боку, цей метод працює більш повільно, ніж *gbest*.

Розглянемо компоненти швидкості більш детально:

1. *Попередня швидкість*  $v_i(t)$  виступає в ролі пам'яті частки, тобто містить інформацію про її переміщення в минулому. Цю пам'ять можна визначити як імпульс, що запобігає стрибкоподібним змінам напрямку руху частки. Дана швидкість також є інерційним компонентом.

2. *Когнітивний компонент*  $c_1 r_1 (y_i - x_i)$  визначає продуктивність  $i$ -ї частки відносно минулих результатів і виступає у ролі індивідуальної пам'яті про найбільш оптимальні позиції даної частки. Використовуючи його, частка може повертатися до станів, які

були найкращими для неї в минулому. Це є однією з емергентних властивостей системи у цілому.

3. *Соціальний компонент*  $c_2 r_2 (y^* - x_i)$  у *gbest* PSO або  $c_2 r_2 (y_i^* - x_i)$  у *lbest* PSO визначає продуктивність частки відносно сусідніх або зв'язаних із нею. Завдяки йому частка має можливість пересуватися в оптимальні позиції, які були знайдені сусідніми частками.

Ступінь внеску когнітивного й соціального компонентів визначається випадковими величинами  $c_1 r_1$  та  $c_2 r_2$  відповідно.

Ефект впливу швидкості на поведінку частки може бути продемонстрований графічно (рис. 3).

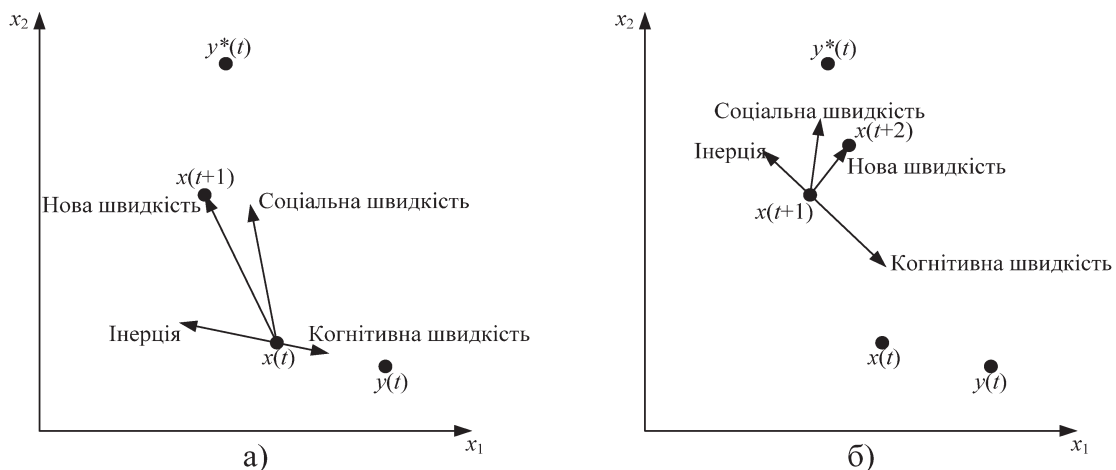


Рис. 3. Геометрична ілюстрація зміни швидкості й позиції частки: а) момент часу  $t$ ; б) момент часу  $t+1$

**Висновки.** Перші напрями використання PSO-методу варто віднести до навчання нейронних мереж (НМ) прямого поширення. Відтоді у цілому ряді робіт було наведено, що PSO-метод також можна застосовувати як метод навчання НМ будь-якої архітектури. У подібних розробках варто визначити потрібний вигляд фітнес-функції, що може бути подана як функція помилки, яку потрібно мінімізувати. Кожна частка у PSO-методі являє собою окреме рішення оптимізаційної проблеми, у цьому випадку — це ціла нейронна мережа. А кожна координата частки у просторі пошуку — це значення відповідного вагового коефіцієнту.

#### Література

1. Camazine S. A model of collective nectar source by honey bees: self-organization through simple rules / S. Camazine, J. Sneyd // *Journal of Theoretical Biology* — 1991. — № 149. — P. 547–571.
2. Subbotin S. Modifications of antcolony optimization method for feature selection / S. Subbotin, A. Oleynik // *The experience of designing and application of CAD systems in Microelectronics : Proceeding sof the IX International Conference CADSM–2007*. — Lviv : Publishing house of Lviv Polytechnic, 2007. — P. 493–494.
3. Олейник А. А. Мультиагентный метод выделения информативных признаков для син-

теза моделей объектов управления / А. А. Олейник, С. А. Субботин // *Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій : тези доповідей Міжнародної науково-практичної конференції, м. Запоріжжя, 13–15 квітня 2006 р.* / [під заг. ред. Д. М. Пізи]. — Запоріжжя : ЗНТУ, 2006. — С. 181–183.

4. Beni G. *Swarm intelligence* / G. Beni, J. Wang // *Annual Meeting of the Robotics Society: Proceedings of Seventh International Conference*. — Tokyo : RSJ Press, 1989. — P. 425–428.

5. Олейник А. А. Сравнительный анализ методов оптимизации на основе метода муравьиных колоний / А. А. Олейник // *Комп'ютерне моделювання та інтелектуальні системи : збірник наукових праць* / [за ред. Д. М. Пізи, С. О. Суботина]. — Запоріжжя : ЗНТУ, 2007. — С. 147–159.

6. Kennedy J. *Particle swarm optimization* / J. Kennedy, R. Eberhart // *International Conference on Neural Networks: Proceedings of the 1995 IEEE*. — NJ : IEEE Press, 1995. — P. 1942–1948.

7. Braendler D. The suitability of particle swarm optimization for training neural hardware / D. Braendler, T. Hendtlass // *Proceedings of the Fifteenth International Conference on Industrial and Engineering, Applications of Artificial Intelligence And Expert Systems*. — Cairns : Springer, 2002. — P. 190–199.