

ДОСЛІДЖЕННЯ ЗБІЖНОСТІ ІМУННИХ АЛГОРИТМІВ

Стаття присвячена дослідженню властивостей збіжності імунних алгоритмів, зокрема алгоритму клонального добору CLONALG. У роботі визначені достатні умови для доведення збіжності розглянутого алгоритму. Розглянуто задачу визначення незалежності верхніх границь для числа поколінь, які гарантують пошук рішення з певною ймовірністю за аналогією з дослідженнями, виконаними для генетичних алгоритмів. Наведено результати чисельних експериментів.

Ключові слова: штучна імунна система, алгоритм клонального добору, збіжність імунних алгоритмів.

Статья посвящена исследованию свойств сходимости иммунных алгоритмов, в частности клональному алгоритму CLONALG. В работе определены условия, достаточные для доказательства сходимости рассматриваемого алгоритма. Рассмотрена задача определения независимости верхних границ для числа поколений, которые гарантируют поиск решения с определенной вероятностью по аналогии с исследованиями генетических алгоритмов. Приведены результаты численных экспериментов.

Ключевые слова: искусственная иммунная система, алгоритм клонального отбора, сходимость иммунных алгоритмов.

The study is directed towards research of properties of immune algorithms convergence, in particular to clonal algorithm CLONALG. The conditions are determined for the proof of convergence of considered algorithm. The problem of definition of independence of the upper limits for the number of generations which is considered guarantee that the solution will be found with certain probability by analogy to the research of genetic algorithms. Results of numerical experiments are given.

Key words: artificial immune system, algorithm clonal selection, convergence of immune algorithms.

ВСТУП

За останні кілька десятиліть значно зріс інтерес дослідників до використання біології як джерела нових ідей для розв'язання обчислювальних задач. Біологічні системи являють собою складні системи обробки інформації, які є головним мотивом для здобування корисних метафор, що дозволяють знаходити ефективні рішення в широкому спектрі предметних областей. Організм вищих біологічних істот наділений двома системами, що володіють здатністю запам'ятовувати, навчатися, розпізнавати образи, приймати рішення і т. ін. Це нервова й імунна системи.

На сьогодні нервова система широко використовується в інформатиці як біологічний прототип штучних нейронних мереж (ШНМ), які впроваджені в апаратну частину ЕОМ, що одержали назву нейрокомп'ютерів [1, 2].

Імунна система також має множину корисних особливостей, таких, наприклад, як властива їй паралельна децентралізована обробка інформації, здатність до розпізнавання «свій – чужий», специфічне поведіння і т. ін. Завдяки своїм унікальним властивостям імунна система є

об'єктом інтенсивних досліджень з метою розробки на її основі різних обчислювальних інструментів. Даний напрям одержав назву *штучні імунні системи* (ШІС).

Означення 1. *Штучна імунна система* (ШІС) – це адаптивна обчислювальна система, що використовує моделі, принципи, механізми та функції, описані в теоретичній імунології, які застосовуються для розв'язання прикладних задач.

Незважаючи на те, що природні імунні системи вивчені далеко не повністю, на сьогодні існують щонайменше три теорії, які пояснюють функціонування імунної системи та описують взаємодію її елементів, а саме: теорія негативного відбору [3, 4], теорія клональної селекції [5, 6] і теорія імунної мережі [6, 7-9]. Вони лягли в основу створення трьох алгоритмів функціонування ШІС. На рис. 1. представлено типові задачі, які можна вирішувати за допомогою обчислювальних методів, розроблених на основі зазначених вище біологічних теорій.

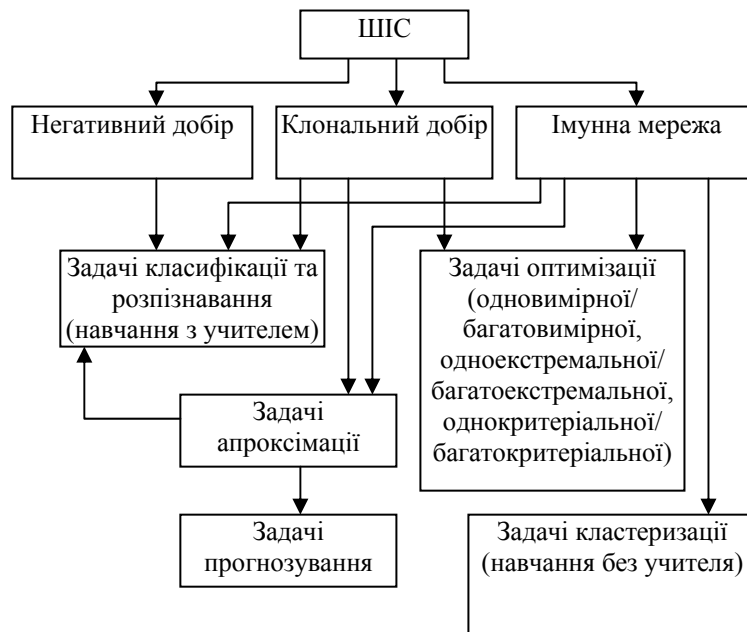


Рис. 1. Задачі, які розв'язуються за допомогою ШІС

Сфера застосування ШІС включає такі області (але не обмежується ними): методи обчислень, когнітивні моделі, виявлення аномалій і неполадок, мультиагентні системи, моделі самоорганізації, моделі колективного інтелекту, системи пошуку й оптимізації, моделі автономних розподілених систем, моделі штучного життя, системи комп'ютерної безпеки, методи здобування інформації, обробка сигналів і зображень та деякі інші.

1. АНАЛІЗ ОСТАННІХ ДОСЛІДЖЕНЬ

З огляду на стохастичну природу алгоритмів ШІС, розвиток популяції, зокрема для алгоритму клональної селекції, можна теоретично розглядати як рух у дискретному просторі станів, що відбувається відповідно до певних ймовірнісних правил. Оскільки ймовірності переходу в новий стан залежать тільки від поточного стану системи (а не від її попередніх станів), можна сказати, що поведінка ШІС може бути описане за допомогою властивостей марковських ланцюгів. Дійсно, у роботі [10, 11] використана теорія марковських ланцюгів для доказу збіжності алгоритму MISA за умови підтримки елітної множини клітин пам'яті. В іншій роботі [12], де моделювався оператор гіпермутації, пов'язаний з алгоритмом В-клітини, використано спрощений метод доведення збіжності в плані знаходження хоча б одного

глобального оптимуму з одиничною ймовірністю за час $t \rightarrow \infty$. У роботі [13] використано критерії Рудольфа для узагальненого клонального алгоритму, названого імунним алгоритмом. Цей алгоритм включає можливість вибору різних схем гіпермутації, старіння і т. ін. У цій роботі для доведення збіжності алгоритму наведені дві істотних умови.

Виконаний аналіз публікацій в області досліджень збіжності імунних алгоритмів показав, що алгоритми, досліджувані в ранніх роботах, є спрощеними варіантами клонального алгоритму, описаного в роботі [6-9], або модифікаціями генетичного алгоритму, у якому використовується оператор гіпермутації клонального алгоритму. Доведення збіжності складніших алгоритмів, таких як CLONALG, залишається відкритим завданням.

ПОСТАНОВКА ЗАДАЧІ

З огляду на велику кількість існуючих модифікацій алгоритмів даного класу, у даній роботі ставиться задача дослідження властивостей збіжності імунних алгоритмів і умов, які достатні для доведення їх збіжності до глобального оптимуму при розв'язанні задач оптимізації. При цьому як об'єкт для проведення теоретичного аналізу й визначення умов збіжності обрано класичний клональний алгоритм CLONALG, описаний у роботі [6-9].

2. ІМУННІ АЛГОРИТМИ

Імунна система має всі головні особливості штучного інтелекту: пам'ять, здатність навчатися, вміння розпізнавати й ухвалювати рішення щодо того, як розглядати чужорідний білок (антиген), що потрапив в організм, навіть якщо останній ніколи не існував на Землі. Подібно до штучних нейронних мереж, ШНС можуть накопичувати нову інформацію і, використовуючи попередньо вивчену інформацію, здійснювати розпізнавання образів децентралізованим способом.

Таким чином, імунна система виконує кілька функцій. Разом з іншими системами організму вона підтримує стійкий стан життєвих функцій, названий гомеостазом. Однак найзначнішою її роллю є захист організму від захворювань, зумовлених проникненням в організм інфекційних агентів або чужорідних речовин, що несуть чужорідну генетичну інформацію, а також знищення пошкоджених клітин. Мікроорганізми, які подібні до вірусів, бактерій, грибків і паразитів, класифікуються як хвороботворні, оскільки вони можуть викликати захворювання після вторгнення в наш організм. Першочерговим завданням, з яким зіштовхується імунна система, є розпізнавання цих хвороботворних організмів. Варто сказати, що імунна система не розпізнає весь хвороботворний організм цілком. Розпізнавання відбувається на рівні окремих молекул мікроба, які називають антигенами. Не вдаючись у подробиці будови природної імунної системи, можна сказати, що основним елементом, що забезпечує функціонування імунітету, є В – і Т-клітини (лімфоцити) [6]. Вони містять на своїй поверхні спеціальні молекули (рецептори) – антитіла. Антитіла є безпосередніми виконавцями процесу розпізнавання чужорідних антигенів. З погляду хімії процес розпізнавання антигенів антитілами зводиться до міжмолекулярної взаємодії, під час якої антитіла «хімічно» зв'язуються з активними (рецепторами) ділянками антигенів і в такий спосіб нейтралізують їх. При цьому чим краще антитіло розпізнає антиген, тим сильнішим виходить зв'язок.

Розглядаючи обчислювальні аспекти парадигми імунних систем, можна виділити такі елементи імунних алгоритмів:

- множина способів подання компонентів системи;
- множина механізмів, що дозволяють оцінити взаємодію індивідуумів з навколишнім середовищем і одного з одним;
- процедури адаптації, які управляють динамікою системи, тобто зміною її стану в часі.

Способи подання дозволяють створювати абстрактні моделі імунних органів, клітин або молекул; механізми оцінювання, які називають також функціями афінності, дозволяють кількісно оцінити взаємодії цих «штучних імунних органів», а процедури адаптації, виражені у вигляді безлічі загальних алгоритмів досягнення мети, управляють динамікою ШНС.

Для проектування структури ШС найчастіше застосовують методи, які активно використовують в інших біологічних обчислювальних парадигмах – таких як, наприклад, нейронні мережі й еволюційні алгоритми [1, 2, 14-19]. Набір оціночних функцій, узятих з даних обчислювальних парадигм, цілком може бути використано для оцінювання взаємодії індивідуумів ШС. Разом з тим еволюційні алгоритми добре підходять для керування зміною в часі станів кліток і молекул, з яких складається штучна імунна система. Із врахуванням сказаного вище, процес побудови ШС можна розділити на два основних етапи:

- вибір належної форми подання індивідуумів і міри афінності;
- застосування кожного з існуючих алгоритмів (або нового алгоритму) для керування зміною станів системи в часі.

Означення 2. Під афінністю будемо розуміти *міру взаємодії* (або *силу зв'язку*) відповідних комплементарних ділянок антигену й антитіла або двох антитіл. Афінність може бути формально представлена у вигляді однієї з метрик (наприклад, Евклідової відстані); ця міра вказує на ступінь подібності або розбіжності між відповідними атрибутами рядків таку, що $S^P \times S^P \rightarrow \mathfrak{R}^+$.

Означення 3. Безліч стереохімічних взаємодій або множину властивостей антитіл, що визначають їх міжмолекулярну афінність, будемо називати *багатовимірним простором форм P*.

При використанні простору форм передбачається можливість повного опису узагальненої форми антитіла за допомогою рядка з k параметрів. Якщо при цьому припустити, що й антиген, у свою чергу, має комплементарну антитілу область, яку можна описати за допомогою такої ж кількості параметрів, то при об'єднанні цих параметрів у вектори антитіло (Ab) і антиген (Ag) можуть бути представлені в такий спосіб, як точки у k -вимірному Евклідовому просторі.

Відображення безлічі форм у множину параметрів розв'язуваного завдання не має значення в обчислювальному змісті, а повністю визначається предметною областю використання ШС. Кожне антитіло здатне розпізнавати не єдиний антиген, а множину антигенів, чия форма доповнення (комплемента) перебуває усередині певного об'єму у просторі форм. Цей об'єм характеризується параметром ε , який називають крос-реактивним порогом.

Означення 4. *Крос-реактивним порогом* називають область у багатовимірному просторі форм, що охоплює антитіло, усередині якої проявляється його активність стосовно антигенів або інших антитіл.

Означення 5. Під *рівнем гіпермутації* будемо розуміти чисельне значення, яке використовується для визначення кількості клонів підданої мутації *клітини пам'яті*, які можна ввести в популяцію клітки.

Необхідно також відзначити, що форма подання індивідуумів і міра афінності повністю визначаються предметною областю, у якій передбачається використання ШС, і типом розв'язуваної задачі. Поводження узагальненого імунного алгоритму розглянемо на прикладі алгоритму клональної селекції.

Формально алгоритм клональної селекції можна представити в такий спосіб [20]:

$$CLONALG = CLONALG(P^l, G^k, l, k, m_{Ab}, \delta, f, I, \tau, AG, AB, S, C, M, n, d), \quad (1)$$

де P^l – простір пошуку (простір форм); G^k – подання простору; l – довжина вектора атрибутів (розмірність простору пошуку); k – довжина рецептора антитіла; m_{Ab} – розмір популяції антитіл; δ – функція експресії; f – функція афінності; I – функція ініціалізації початкової популяції антитіл; τ – умова завершення роботи алгоритму; AG – підмножина антигенів; AB – популяція антитіл; S – оператор селекції; C – оператор клонування; M – оператор мутації; n – кількість кращих антитіл, що відбираються для клонування; d – кількість гірших антитіл, що підлягають заміні новими.

Розглянемо простір форм (P^l) фенотипів і простір їхніх подань у вигляді антитіл (G^k) або простір генотипів. Функція

$$\delta : P^l \rightarrow G^k \quad (2)$$

перетворює варіанти розв'язків із P^l у їхні внутрішні подання (G^k) у вигляді індивідуумів популяції. Цю функцію називають ще функцією експресії. Слід зазначити, що при практичній розробці ШІС найчастіше вводять подібні перетворення з міркувань зручності застосування імунних операторів і обчислення афінності індивідуумів. Наприклад, вектор дійсних атрибутів розмірності l може бути перетворений у рядок бітів довжини k , що уможливорює застосування специфічних операторів мутації й обчислення афінності за допомогою різних видів відстані Хеммінга. Незважаючи на зазначену важливість, даний аспект не розглядався у вивчених авторами публікаціях, які стосуються проектування ШІС. Тому терміни «генотип», «фенотип» і «експресія» у даному описі запозичені із близьких за функціональним призначенням і структурою еволюційних алгоритмів, хоча більше підходять до вживання в контексті еволюції хромосом, ніж молекулярних структур антитіл. Припускається також, що для кожного розв'язку $p \in P^l$ існує одне й тільки одне його подання $\delta(p) \in G^k$. При цьому у загальному випадку зворотне твердження невірне. Використовуючи узагальнене подання, можна ввести функцію афінності f :

$$f : P^l \times P^l \rightarrow \mathfrak{R}^+ . \quad (3)$$

Завдання полягає в максимізації функції афінності.

Приймаючи початковий розмір популяції антитіл (m_{Ab}), введемо функцію ініціалізації у вигляді:

$$I : G^k \times m_{Ab} \rightarrow AB(G^k) . \quad (4)$$

Часто ініціалізація починається випадковими значеннями з використанням рівномірного розподілу.

Нехай Q – унарний стохастичний оператор перетворення на множині G^k , який використовує керуючу множину K_Q для генерації керуючих параметрів, що визначають спосіб перетворення на поточному кроці роботи алгоритму. Наприклад, у випадку мутації бітових рядків бітова маска може бути використана як керуючий параметр, у якому одиничні позиції визначають номери бітів індивідуума, що піддається мутації. Отже, функціональний запис оператора Q може бути представлено так:

$$Q : G^k \times K_Q \rightarrow G^k . \quad (5)$$

Оптимальним розв'язком $Ab_{opt} \in G^k$ стосовно оператора Q і антигена $Ag \in AG$, $AG \subset G^k$ називається індивідуум, чия афінність не може бути збільшена при подальшому впливі оператора перетворення Q , тобто

$$\forall k \in K_Q : f(Q(Ab_{opt}, k), Ag) \leq f(Ab_{opt}, Ag) . \quad (6)$$

Умова зупинки (τ) виконується, коли популяція антитіл повністю розпізнає популяцію антигенів, тобто

$$\forall Ag \in AG : \exists Ab \in G^k \mid Ab = Ab_{opt} . \quad (7)$$

Оператор селекції S формує підмножину G_S індивідуумів, чия афінність є кращою в даному поколінні. Таким чином, S разом з керуючою множиною K_S представляє функцію:

$$S : G^k \times K_S \rightarrow \{0, 1\} , \quad (8)$$

множина, якої утворюється в результаті селекції:

$$G_S = \{Ab \in G^k \mid S(Ab, k_S) = 1\} , \quad |G_S| = n . \quad (9)$$

Аналогічно відбувається відбір індивідуумів у популяцію клітинок пам'яті.

Оператор клонування C збільшує представництво елементів множини G_S у популяції і разом з керуючою множиною K_C може бути записаний так:

$$C : G_S \times K_C \rightarrow G_S. \quad (10)$$

Оператор мутації M пов'язаний з керуючою множиною K_M так:

$$M : G^k \times K_M \rightarrow G^k. \quad (11)$$

У даній роботі розглядається простий оператор багатоточкової мутації для індивідуумів, представлених у вигляді бітових рядків. Метадинаміку системи виразимо у вигляді функції заміщення гірших антитіл популяції:

$$R : G^k \times d \rightarrow AB_d(G^k). \quad (12)$$

Гірші антитіла попередньо відбираються за допомогою оператора селекції.

Блок-схема алгоритму клональної селекції представлена на рис. 2.

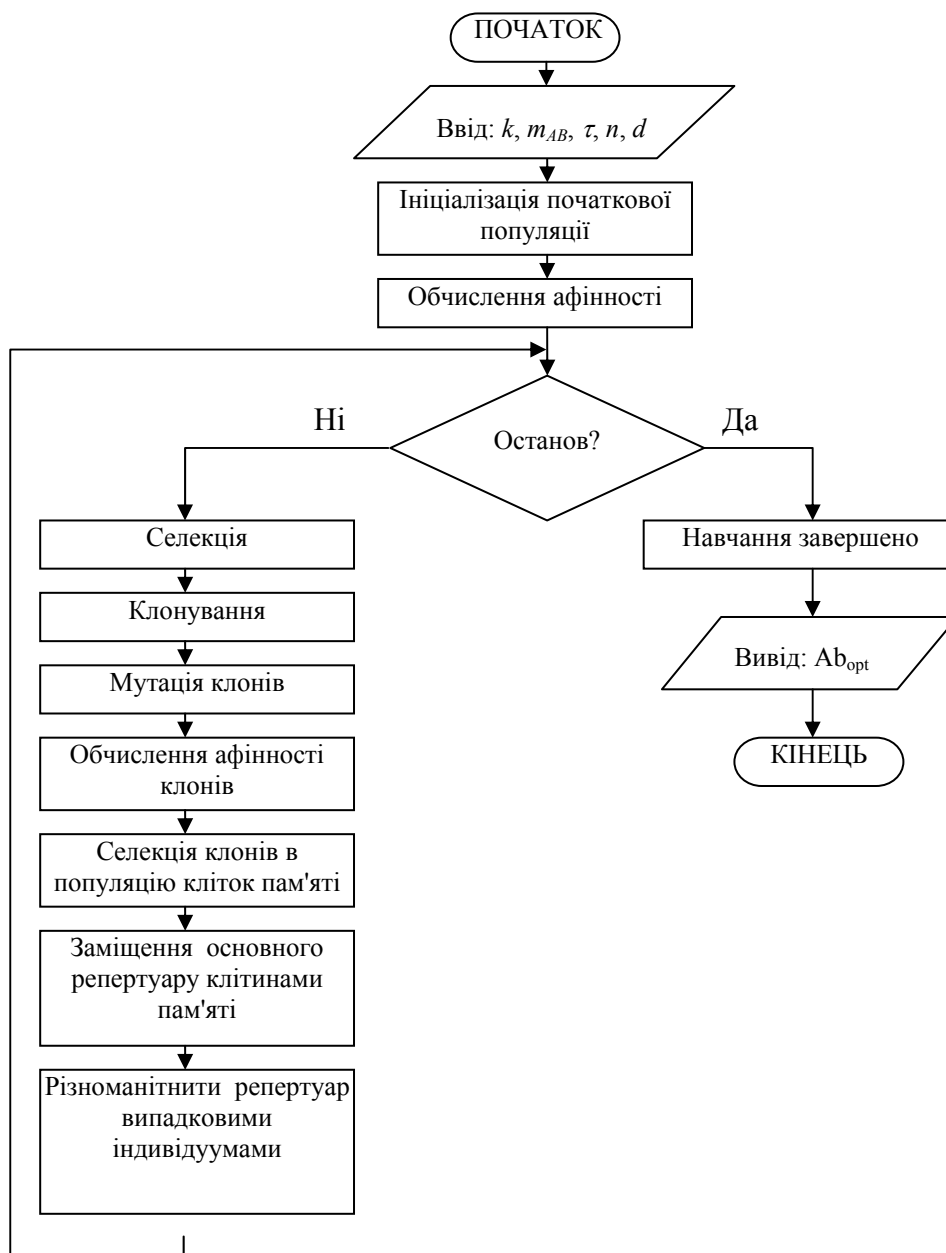


Рис. 2. Блок-схема алгоритму клональної селекції
 Нижче наведено узагальнений покроковий опис алгоритму

Крок 1. Ініціалізація. Створення (звичайно випадковою генерацією) початкової популяції антитіл (AB).

Крок 2. Обчислення афінності. Для кожного антитіла Ab_j , $Ab_j \in AB$ обчислити його афінність стосовно кожного антигену Ag_i , $Ag_i \in AG$. Результати записати в матрицю афінностей $D: D = [|AG| \times m_{Ab}]$ і $d_{ij} = f(Ab_j, Ag_i)$, $d_{ij} \in D$.

Крок 3. Клональна селекція та поширення. Вибрати з популяції по n кращих антитіл для кожного рядка матриці D і помістити їх в окрему популяцію клонів AB_C , $|AB_C| = n \cdot |AG|$. Генерувати клони елементів популяції AB_C пропорційно до їх афінності; тобто чим вища афінність, тим більша створюється кількість клонів і навпаки.

Крок 4. Дозрівання афінності. Піддати мутації всі клони популяції AB_C з імовірністю, обернено пропорційною їх афінностям, тобто чим нижча афінність індивідуума, тим вища ймовірність його мутації. Обчислити нову афінність кожного антитіла Ab_j , $Ab_j \in AB_C$ аналогічно до кроку 2, одержавши матрицю афінностей D_C . Вибрати з популяції AB_C n антитіл, для яких відповідний вектор-стовпчик матриці D_C дає кращий узагальнений результат афінності, і перенести їх у популяцію клітин пам'яті M_R .

Крок 5. Метадинаміка. Замінити d гірших антитіл популяції AB новими випадковими індивідуумами.

Крок 6. Замінити n антитіл популяції AB клітками пам'яті з M_R і переходити до кроку 2 до тих пір, поки не буде досягнуто критерію зупинки.

Особливістю алгоритму клональної селекції є те, що він, на відміну від імунної мережі, підтримує постійний розмір популяції антитіл.

3. УМОВИ ЗБІЖНОСТІ ІМУННИХ АЛГОРИТМІВ

При доведенні збіжності імунних алгоритмів зазвичай використовують дві міри стохастичної збіжності: повна збіжність і збіжність у середньому.

Означення 6. Нехай X – випадкова змінна, а $(X_t : t > 0)$ послідовність випадкових змінних. Тоді послідовність X_t **збігається повністю** до X , якщо для будь-якого $\varepsilon > 0$:

$$\lim_{t \rightarrow \infty} \sum_{i=0}^t P(|X_i - X| > \varepsilon) < \infty$$

Або

$$\lim_{t \rightarrow \infty} \sum_{i=0}^t P(|X_i - X| > \varepsilon) \rightarrow 0. \quad (13)$$

Означення 7. Нехай X – випадкова змінна, а $(X_t : t \geq 0)$ – послідовність випадкових змінних. Тоді послідовність X_t **збігається в середньому** до X , якщо

$$\lim_{t \rightarrow \infty} E\|X_t - X\| = 0, \quad (14)$$

де $E\|\cdot\|$ – математичне очікування.

На підставі цього можна сформулювати наступні означення для збіжності популяційних алгоритмів [21].

Означення 8. Нехай $X_t : t \geq 0$ – послідовність популяцій і нехай f_{opt}^t – значення придатності кращого індивідуума в популяції на часовому кроці t . Говорять, що алгоритм **збігається повністю** до глобального оптимуму f^* задачі оптимізації, якщо невід'ємна випадкова послідовність $D_t = f^* - f_{opt}^t$ **збігається повністю до нуля**.

Означення 9. Нехай $X_t : t \geq 0$ – послідовність популяцій і нехай f_{opt}^t – значення придатності кращого індивідуума в популяції на часовому кроці t . Алгоритм **збігається в середньому** до глобального оптимуму f^* задачі оптимізації, якщо невід’ємна випадкова послідовність $D_i = f^* - f_{opt}^i$ **збігається в середньому до нуля**.

4. ДОВЕДЕННЯ ЗБІЖНОСТІ ІМУННОГО АЛГОРИТМУ

Формально одна ітерація імунного алгоритму (1) може бути описана в такий спосіб:

$$\begin{aligned} \forall i \in \{1, \dots, n\}: Ab'_i &= S(Ab_1, \dots, Ab_{m_{Ab}}) \\ \forall i \in \{1, \dots, m_C\}: Ab''_i &= M(C(Ab'_1, \dots, Ab'_n)) \\ \forall i \in \{1, \dots, n\}: Ab'''_i &= S(Ab''_1, \dots, Ab''_{m_C}) \\ (Ab^y_1, \dots, Ab^y_{m_{Ab}}) &= R(Ab'''_1, \dots, Ab'''_n, Ab_1, \dots, Ab_{m_{Ab}}) \end{aligned} \quad (15)$$

де m_C – розмір популяції клонів.

У роботі [22] наведені умови збіжності еволюційних алгоритмів (у середньому й повністю) до глобального оптимуму будь-якої задачі оптимізації:

Умова 1: Кожний індивідуум у популяції може бути змінений однократною мутацією з імовірністю $p > 0$.

Умова 2: Кращий індивідуум у популяції виживається у кожному поколінні з імовірністю $p = 1$.

Формально ці дві умови можуть бути записані так:

$$\forall Ab_i, Ab_j \in AB \quad P\{Ab_j = M(Ab_i)\} \geq \delta_{m_C} > 0 \quad (16)$$

$$P\{Ab_{opt}(R(x_1, \dots, x_k)) = Ab_{opt}(x_1, \dots, x_k)\} = 1 \quad (17)$$

$$P\{Ab_{opt}(S(x_1, \dots, x_k)) = Ab_{opt}(x_1, \dots, x_k)\} = 1. \quad (18)$$

Тут Ab_{opt} повертає кращого з популяції індивідуумів. Якщо виконується тільки умова 1, то можна довести, що еволюційний алгоритм знаходить глобальний оптимум після скінченного числа кроків з імовірністю $p = 1$ незалежно від його ініціалізації, але це не є збіжністю, тому що не можна гарантувати, що оптимум залишиться в популяції після того, як він буде знайдений. Якщо також виконується умова 2, то можна довести, що еволюційний алгоритм збігається до глобального оптимуму.

Теорема 1: Розглянутий імунний алгоритм збігається повністю й у середньому до глобального оптимуму завдання оптимізації, незалежно від його ініціалізації, за умови, що до нього застосовано оператор елітного відбору.

Доведення: Щоб довести теорему, ми повинні показати, що умови 1 і 2 задовольняються даним алгоритмом.

Оскільки ні клонування, ні селекція не змінюють існуючих індивідуумів і не створюють нових, тільки два оператори здатні представити оптимальне рішення в популяції: *оператор гіпермутації* та/або *оператор заміщення*.

Нехай бітові рядки довжини k у кожній точці простору пошуку представлені вектором $\{0,1\}^k$. Якщо індивідуум популяції порівняно з рядком, що представляє оптимум, збігається в $k - c$ бітах (відповідно не збігається в c бітах), то ймовірність того, що в результаті однократного застосування оператора гіпермутації буде досягнуто глобальний оптимум дорівнює:

$$P_c^{(k)} = \frac{c!}{k^c} \frac{1}{k}. \quad (19)$$

Тут $c!$ – перестановки з c елементів по k^c можливих варіантах.

Цю ймовірність необхідно помножити на ймовірність того, що оператор насправді випадково вибирає c інвертованих бітів. Величина $\frac{1}{k}$ представляє ймовірність того, що $r = c$, де r – випадково обране число мутованих бітів. Якщо розширити вираження (19) рядками кінцевого алфавіту Ψ , де кожна точка простору пошуку представлена вектором $\{0,1,2,\dots,\Psi-1\}^k$, то ймовірність такої зміни кожного символу, тобто що він співпадає з відповідним символом оптимального рядка, дорівнює:

$$P_c^{(k)} = \frac{c!}{k^c} \frac{1}{(\Psi-1)^c} \frac{1}{k}. \quad (20)$$

Тут $\frac{1}{(\Psi-1)^c}$ – ймовірність того, що даний символ одержує правильне значення.

Оскільки $P_c^{(k)}$ завжди додатне, то умова 1 доведена. Існує також ймовірність $P_s > 0$ того, що оптимум буде випадково представлено у популяції оператором заміщення, однак $P_c^{(k)}$ досить для доказу умови 1.

Для доведення того, що умова 2 також виконується, необхідно включити в розгляд усі оператори, що впливають на популяцію, і показати, що жоден з них не здатний стати причиною втрати оптимального рішення.

Оператор клонування створює копії індивідуумів, але не змінює значення їхніх елементів і, таким чином, він не може стати причиною втрати оптимуму.

Оператор гіпермутації впливає тільки на індивідууми популяції клонів 3, але не змінює індивідуумів, створених іншими операторами, включаючи себе.

Оператор селекції вибирає кращі індивідууми в кожному поколінні і за означенням не може втратити глобальний оптимум.

Оператор заміщення позбувається від гірших індивідуумів і, таким чином, також не може втратити глобальний оптимум. У випадку, якщо в популяції існує більше одного оптимуму, то щонайменше один оптимальний індивідуум буде збережений у процесі відбору. Цього досить для доведення Умови 2 і, відповідно, для доведення теореми.

За аналогією можна довести збіжність великої кількості імунних алгоритмів. У цьому випадку досить лише довести, що використовувані ними оператори, наприклад гіпермутації, задовольняють Умову 1 і показати, що Умова 2 також виконується.

Теорема 2. Імунний алгоритм, розглянутий у даній роботі, незалежно від його ініціалізації, не збігається повністю до єдиного глобального оптимуму задачі оптимізації, якщо оператор гіпермутації видозмінює тільки окремі біти і при цьому не застосовується оператор клонального відбору.

Доведення. У Теоремі 1 виконання Умови 1 доведено для випадку, коли імунний алгоритм використовує оператори гіпермутації або заміщення, які дають можливість одержати кожної з можливих індивідуумів у просторі пошуку. Це, як обговорювалося раніше, говорить про те, що алгоритм досягне оптимуму. Важливо відзначити, що умова 2 є достатньою для збіжності, але не обов'язковою. Для доведення того, що імунний алгоритм не збігається до глобального оптимуму, досить довести, що незалежно від того, чи буде знайдений оптимум, завжди існує ітерація, в якій імунний алгоритм не містить оптимум у популяції. Якщо існує ймовірність того, що імунний алгоритм втратить оптимальне рішення, не знайшовши іншого оптимального рішення за прийнятний час, то він не збігається з одиничною ймовірністю. Існування такої ймовірності гарантується в тому випадку, якщо застосовуваний оператор селекції не використовує елітність і тому може пропустити оптимальне рішення при відборі.

Нехай у поколінні t популяція $AB^{(t)}$ складається з X_1 оптимальних індивідуумів і $X_2 = AB^{(t)} - X_1$ неоптимальних індивідуумів. У момент $t + \tau_B$ всі X_1 оптимальних рішень будуть вилучені з популяції. Доведення полягає у демонстрації того, що завжди є ненульова

ймовірність видалення всіх існуючих оптимальних рішень із популяції перед введенням у неї нового оптимального рішення незалежно від кількості представлених оптимальних рішень і значень неоптимальних рішень. Оператор клонування створює популяцію клонів, що складаються з копій X_1 оптимальних рішень і з X_2 неоптимальних рішень. Нехай кількість клонів X_2 неоптимальних рішень становить m_{X_2} . Оскільки оператор гіпермутації завжди змінює принаймні один біт, клони оптимальних рішень не будуть представляти глобальний оптимум. З іншого боку, m_{X_2} клонів можуть представити глобальний оптимум на наступному кроці з імовірністю P такою, що $P \leq P_{\rho=1}$, де $P_{\rho=1}$ – імовірність досягнення оптимуму з найбільш сприятливого стану, що перебуває на відстані $\rho=1$. Отже, ймовірність того, що жоден з них не представить глобальний оптимум, буде вищою, ніж $(1 - P_{\rho=1})^{m_{X_2}}$. Ця ймовірність мінімізується при максимізації m_{X_2} , що відбувається у тому випадку, коли в популяції $AB^{(t)}$ наявний тільки один глобальний оптимум, тобто $X_1 = 1$.

Якщо нове оптимальне рішення не було створено, оператор селекції вибере більшість із X_1 поточних оптимальних індивідуумів разом із найкращими неоптимальними індивідуумами. Крім того, оператор заміщення може згенерувати ряд нових випадкових індивідуумів. Таким чином, нижня границя ймовірності того, що оператор гіпермутації не встигне створити новий глобальний оптимум перед тим, як усі існуючі глобальні оптимуми, які були отримані раніше, покинуть популяцію, дорівнює:

$$P_{NO} \geq (1 - P_{\rho=1})^{m_{n-1} \tau_B} > 0. \quad (21)$$

Разом із розглянутою вище ймовірністю необхідно також урахувати ймовірність того, що оператор заміщення не представить глобальний оптимум протягом τ_B поколінь. Необхідно відзначити, що для нетривіальних функцій дана ймовірність буде дуже низькою. Ґрунтуючись на сказаному вище, можна стверджувати, що оптимальні рішення будуть можуть бути втрачені за скінченний час.

5. ЙМОВІРНІСНІ ОЦІНКИ ЗБІЖНОСТІ ІМУННОГО АЛГОРИТМУ

Для еволюційних алгоритмів існує можливість обчислення кількості ітерацій, що гарантує одержання оптимального рішення з фіксованою ймовірністю ζ ($0 < \zeta < 1$). При цьому верхня границя $t(\zeta)$ не залежить від розв'язуваного завдання.

У роботі [23] для знаходження таких границь (для генетичних алгоритмів з бінарним кодуванням індивідуумів) використовуються марковські ланцюги; при цьому передбачається, що кожна можлива структура популяції буде досліджена алгоритмом з імовірністю, не меншу ζ .

У роботі [24] даний результат розширено для скінченних алфавітів $\Psi = 2^x$, а в роботі [25] отримано точнішу верхню границю, при якій гарантується хоча б однократне подання всіх можливих індивідуумів з ймовірністю ζ , а не всіх можливих популяцій:

$$\bar{t}_1(\delta) = INT \left[\frac{\ln(1 - \zeta)}{m_{Ab} \ln \left[1 - \min \left\{ (1 - \mu)^{k-1} \left(\frac{\mu}{\Psi - 1} \right), \left(\frac{\mu}{\Psi - 1} \right)^k \right\} \right]} \right], \quad (22)$$

де μ – ймовірність мутації.

Метою даного аналізу є знаходження верхньої границі для кількості поколінь, необхідних для подання глобального оптимуму досліджуванім імунним алгоритмом у t_1 поколіннях з імовірністю ζ .

Нехай $\{0,1,2,\dots,\Psi-1\}^k$ – вектор, що представляє простір пошуку задачі оптимізації із скінченним алфавітом Ψ і довжиною рішень k . Можна показати, що ймовірність мутації кожного символу у відповідний символ оптимального рішення дорівнює:

$$P_c^k = \frac{c!}{k^c} \frac{1}{(\Psi-1)^c} \frac{1}{k} \geq \frac{k!}{k^k (\Psi-1)^k} \frac{1}{k}. \quad (23)$$

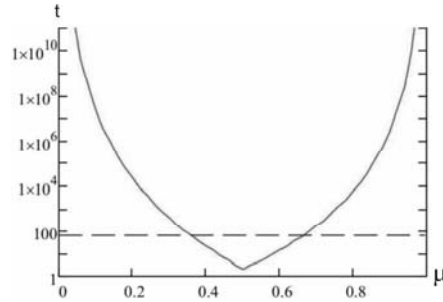


Рис. 3. Верхня границя (пряма штрихова лінія) для імунного алгоритму та верхня границя для генетичного алгоритму при різних значеннях μ

Обидва алгоритми використовують алфавіт $\Psi = 2$
(Аналіз виконано при $\zeta = 0.9, m_{Ab} = 1000$ й $k = 10$)

Доведення нерівності для $\Psi = 2$ тривіальне:

$$\frac{k!}{k^k} = \frac{c!}{k^c} \frac{(c+1)}{k} \dots \frac{(k-1)}{k} \frac{k}{k} \leq \frac{c!}{k^c}$$

для $\Psi \geq 2$, $(\Psi-1)^k \geq (\Psi-1)^c$.

Ймовірність того, що оператор гіпермутації не зможе перетворити даний індивідуум в оптимальний для популяції з m_{Ab} індивідуумів за t поколінь, становить, як мінімум, $(1 - P_{IA})^{m_{Ab}}$. Тоді ймовірність знаходження оптимуму за t поколінь дорівнює $P_t \geq 1 - (1 - P_{IA})^{m_{Ab}}$. Таким чином, можна обчислити кількість поколінь t_1 для ймовірності P_t більшої ζ : $1 - (1 - P_{IA})^{m_{Ab}} \geq \zeta$:

$$t_1 \geq \frac{\log(1 - \zeta)}{m_{Ab} \log(1 - P_{IA})} = t_{IA} \quad (24)$$

Тут t_1 – кількість поколінь, необхідних імунному алгоритму для знаходження глобального оптимуму з мінімальною ймовірністю δ . Розв'язання нерівності $t_{IA} \leq t_{GA}$ приводить до того, що для істотно більших k ($k \gg 1$) верхня границя кількості поколінь для імунних алгоритмів виявляється нижчою, ніж для генетичних алгоритмів при значеннях μ , які задовольняють таким умовам:

$$\begin{cases} \mu < \frac{1}{e} & \text{якщо } \mu \leq \frac{\Psi-1}{\Psi} \\ \mu > 1 - \frac{1}{e(\Psi-1)} & \text{якщо } \mu \geq \frac{\Psi-1}{\Psi} \end{cases} \quad (25)$$

Зокрема, для бінарного алфавіту маємо:

$$\begin{cases} \mu < \frac{1}{e} & \text{якщо } \mu \leq \frac{1}{2} \\ \mu > 1 - \frac{1}{e} & \text{якщо } \mu \geq \frac{1}{2} \end{cases} \quad (26)$$

Результати такого порівняння показано на рис. 4, що демонструє верхню границю, розраховану для імунного алгоритму (рівняння (24)), і верхню границю, отриману в роботі [25] для генетичного алгоритму при різних значеннях ймовірності мутації μ .

Значення розмірів популяції обрані однаковими для обох алгоритмів. Таким чином, оператор гіпермутації не залежить від ймовірності мутації μ , що відповідає прямій лінії на графіку.

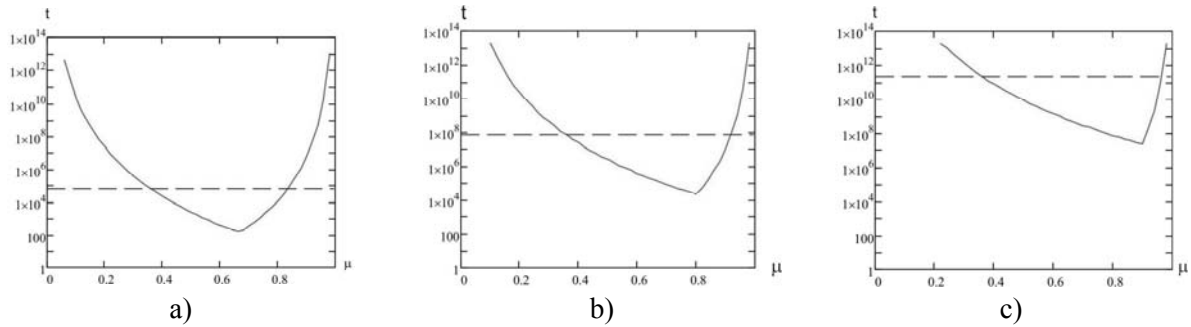


Рис. 4. Верхня границя (пряма штрихова лінія) для імунного алгоритму, і верхня границя для генетичного алгоритму при різних значеннях μ . Всі три алгоритми використовують різний алфавіт: (а) – алфавіт $\Psi = 3$; (б) – алфавіт $\Psi = 5$; (с) – алфавіт $\Psi = 10$. (Аналіз виконано при $\zeta = 0.95, m_{Ab} = 1000$ й $k = 10$)

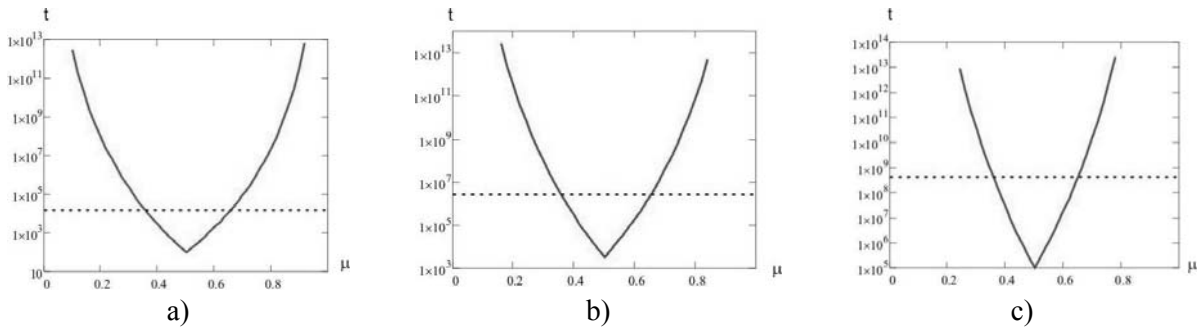


Рис. 5. Верхня границя (пряма штрихова лінія) для імунного алгоритму, і верхня границя для генетичного алгоритму при різних значеннях довжини рецептора антитіла k : (а) – довжина рецептора $k = 15$; (б) – довжина рецептора $k = 20$; (с) – довжина рецептора $k = 25$. (Аналіз виконано при алфавіті $\Psi = 2$, $\zeta = 0.95$, $m_{Ab} = 1000$)

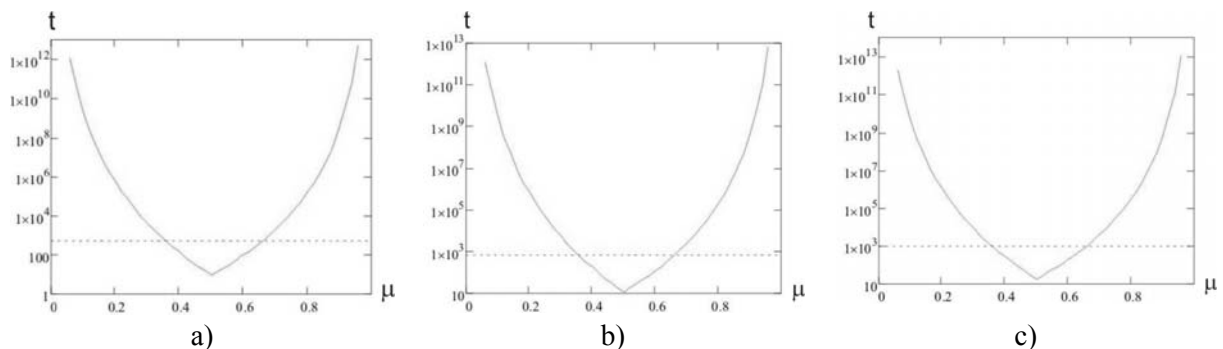


Рис. 6. Верхня границя (пряма штрихова лінія) для імунного алгоритму і верхня границя для генетичного алгоритму при різних значеннях мінімальної ймовірності знаходження глобального оптимуму ζ : (а) – мінімальна ймовірність $\zeta = 0.9$; (б) – мінімальна ймовірність $\zeta = 0.95$; (с) – мінімальна ймовірність $\zeta = 0.99$. (Аналіз виконаний при $\Psi = 2, m_{Ab} = 1000$ і $k = 12$)

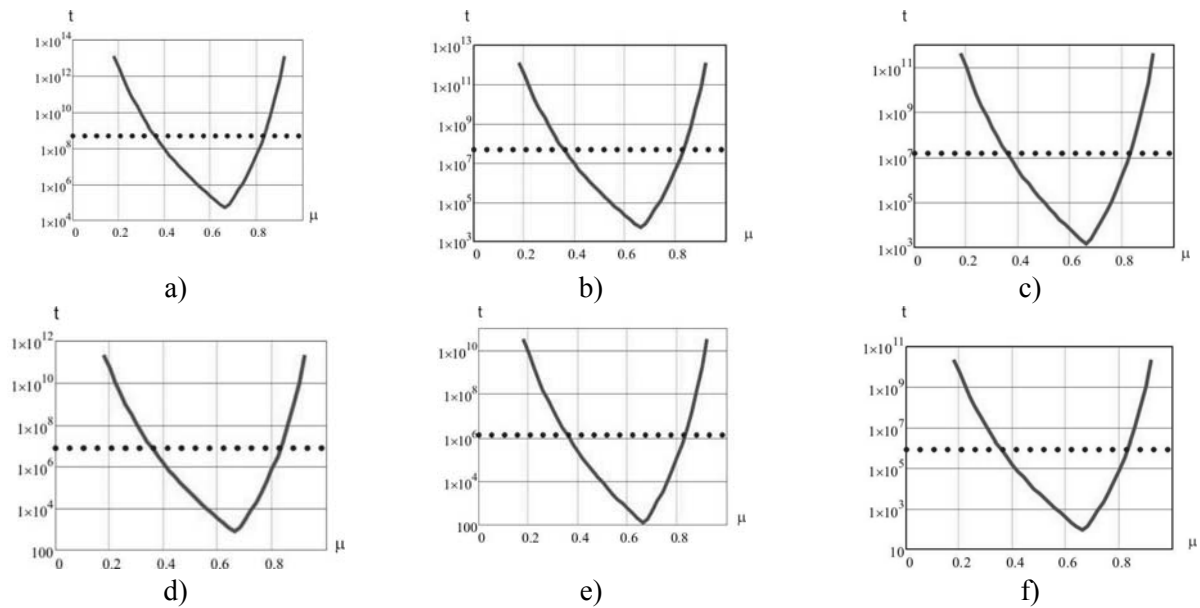


Рис. 7 Верхня границя (пряма штрихова лінія) для імунного алгоритму і верхня границя для генетичного алгоритму при різних розмірах популяції антитіл m_{Ab} :

- (а) – розмір популяції антитіл $m_{Ab} = 1\ 000$; (б) – розмір популяції антитіл $m_{Ab} = 10\ 000$;
 (с) – розмір популяції антитіл $m_{Ab} = 50\ 000$; (д) – розмір популяції антитіл $m_{Ab} = 1\ 000\ 000$;
 (е) – розмір популяції антитіл $m_{Ab} = 6\ 000\ 000$; (ф) – розмір популяції антитіл $m_{Ab} = 9\ 000\ 000$.
 (Аналіз виконано при $\Psi = 3, k = 15$ і $\zeta = 0.95$)

Незважаючи на те, що виконаний аналіз носить теоретичний характер, він залежить якоюсь мірою від типу задачі, тому що реальніші значення верхніх границь (що представляють більший практичний інтерес) можуть бути отримані тільки при врахуванні конкретного класу функцій, які оптимізуються. У роботі [23] розглядається питання, як у випадку вибору найбільш сприятливого варіанта реалізації імунного алгоритму (при $\mu = 0.5$ й мінімальному значенні границі) установити необхідне значення поколінь рівним тому, що властиве випадковому алгоритму, який діє за принципом незалежного випадкового вибору індивідумів у кожному поколінні. Крім того, при низьких значеннях μ , що представляють практичний інтерес, необхідна кількість поколінь зростає до нескінченності. Для імунного алгоритму розглянуті вище результати не актуальні, але кількість необхідних поколінь залишається вищою, ніж для випадкового алгоритму. Рис. 2 демонструє верхні границі для алфавітів $\Psi = 3$ і $\Psi = 25$ при збереженні інших параметрів, що відповідають рис. 1.

6. ВИСНОВКИ

У роботі виконано теоретичний аналіз клонального алгоритму, визначені умови збіжності алгоритмів такого класу й наведено приклади того, як оцінити ці умови стосовно до узагальненого імунного алгоритму. Отримано ймовірнісні оцінки збіжності імунного алгоритму. Показано, що для еволюційних алгоритмів існує можливість обчислення кількості ітерацій, що гарантує одержання оптимального рішення з фіксованою ймовірністю ζ ($0 < \zeta < 1$). При цьому верхня границя $t(\zeta)$ не залежить від розв'язуваного завдання. Знайдено вираз для числа поколінь, необхідних *імунному алгоритму* для знаходження глобального оптимуму з мінімальною ймовірністю ζ . У подальших дослідженнях доцільно виконати узагальнення отриманих результатів на інші класи обчислювальних алгоритмів.

ЛІТЕРАТУРА

1. Горбань А. Н. Обучение нейронных сетей. – М.: СП ПараГраф, 1990. – 160с.
2. Горбань А.Н., Россиев Д.А. Нейронные сети на персональном компьютере. – Новосибирск: Наука, 1996. – 470 с.
3. Dasgupta D. Immunity-Based Intrusion Detection Systems: A General Framework. In the proceedings of the 22nd National Information Systems Security Conference (NISSC), October 18-21, 1999, pp. 345-356.
4. Dasgupta D., Forrest, S. An anomaly detection algorithm inspired by the immune system. In Dasgupta, D./Editor, Artificial Immune System and Their Applications. – Berlin: Springer-Verlag, 1999, pp. 262-277.
5. Dasgupta D. Artificial Immune Systems and Their Applications, Publisher: Springer-Verlag, Inc., January 1999. – 380 p.
6. De Castro L. N., Timmis J. Artificial Immune Systems as a Novel Soft Computing Paradigm // Soft Computing Journal, vol. 7, Issue 7, July 2003, pp. 457-469.
7. De Castro L.N., Von Zuben F.J. An Evolutionary Immune Network for Data Clustering. (Full version, pre-print). In Proceedings of the IEEE SBRN'00 (Brazilian Symposium on Artificial Neural Networks), 2000, pp. 84-89.
8. De Castro L.N., Von Zuben F.J. Artificial Immune Systems: Part I – Basic Theory and Applications. Technical Report – RT DCA 01/99, FEEC/UNICAMP, Brazil. – 95 p.
9. De Castro L.N., Von Zuben F.J. Artificial Immune Systems: Part II – A Survey of Applications. Technical Report – RT DCA 02/00, FEEC/UNICAMP, Brazil. – 64 p.
10. Villalobos-Arias, M., Coello, C.A.C., Hernandez-Lerma, O.: Convergence analysis of a multiobjective artificial immune system algorithm. In: Nicosia et al. [11], pp. 226-235.
11. Nicosia, G., Cutello, V., Bentley, P.J., Timmis, J. (eds.): Proceedings of the 3rd International Conference on Artificial Immune Systems (ICARIS 2004), LNCS 3239. Springer (2004), pp. 189-197.
12. Clark, E., Hone, A., Timmis, J.: A markov chain model of the b-cell algorithm. In: Jacob et al. [63], pp. 318-330.
13. Cutello, V., Nicosia, G., Oliveto, P., Romeo, M.: On the convergence of immune algorithms. In: Proc.of Foundations of Computational Intelligence, pp. 409-416 (2007).
14. Вороновский Г.К. Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности / Вороновский Г.К., Махотило К.В., Петрашер С.Н., Сергеев С.А.. — Харьков: Основа, 1997. – 256 с.
15. Литвиненко В.И., Фефелов А.О., Гринавцев О.И. Библиотека классов C++ для решения задач оптимизации с использованием генетического алгоритма // Искусственный интеллект № 1, 2003, с. 117-124.
16. Руденко О. Г., Бодянский Е. В. Основы теории искусственных нейронных сетей. – Харьков: ТЕЛТЕХ, 2002. – 317 с.
17. Скурихин А.Н. Генетические алгоритмы //Новости искусственного интеллекта, 1995, № 4, с. 6-46.
18. Goldberg, D. E. Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, 1989. – 580 p.
19. Holland, J.H. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence, University of Michigan Press (second edition: MIT Press, 1992). – 490 p.
20. Бідюк П.І., Литвиненко В.І., Фефелов А.О. Формалізація методів побудови штучних імунних систем // Наукові вісті Національного технічного університету України «Київський політехнічний інститут», 2007, № 1, 2007, с. 29-41.
21. Back T. and Fogel D. B. and Michalewicz Z.. Handbook of Evolutionary Computation. – Bristol, UK: IOP Publishing, 1997. – 700 p.
22. Rudolph G. Finite Markov Chain Results in Evolutionary Computation: A Tour d'Horizon Fundamenta Informaticae, 1998, vol. 35, pp. 67-89.
23. Aytug H. and Koehler G. J. Stopping criteria for finite length genetic algorithms. INFORMS J. on Comp., 1996, vol. 8, no. 2, pp. 183-191.
24. Aytug H. and Bhattacharya S. and Koehler G. J. A Markov chain analysis of genetic algorithms with power of 2 cardinality alphabets. European J. Oper. Res. 1996, vol. 96, pp. 195-201.
25. Greenhalgh D. and Marshall S. Convergence Criteria for Genetic Algorithms, SIAM J. Comput., 2000, vol. 30, № 1, pp. 269-282.
26. Safe M. and Carballido J. A. and Ponzoni I. and Brignole N. B. On Stopping Criteria for Genetic Algorithms., Advances in Artificial Intelligence – SBIA 2004. 17th Brazilian Symposium on Artificial Intelligence. Proceedings (Lecture Notes in Artificial Intelligence Vol. 3171), 2004, pp. 405-13.

Рецензенти: д.т.н., проф. Кондратенко Ю.П.,
д.т.н., проф. Данілов В.Я.

© Литвиненко В.І., Бідюк П.І.,
Фефелов А.О., 2009

Стаття надійшла до редакції 15.09.09