

УДК 681.306

Литвиненко В.І.

Модифікований підхід рішення безперервної задачі k -медіан на основі імунного алгоритму

Обговорюються варіанти застосування штучних імунних систем до рішення задачі k -медіан і описується модифікація запропонованого алгоритму клонального добору. У роботі запропонована нова схема використання імунних механізмів у клональному алгоритмі та експериментально оцінюється якість отриманих рішень.

Variants of application of artificial immune systems to the decision of a problem of k -medians are discussed and described updating of the offered algorithm clonal selection. In work the new scheme of use of immune mechanisms in clonal algorithm is offered and quality of the received decisions is experimentally estimated.

Вступ

Існує великий клас задач оптимізації, який зводиться до задач розбивки заданої множини певної структури на його непересічні підмножини з метою мінімізації деякого критерію якості розбивки [1, 3]. Все це різноманіття задач розбивки заданої множини на його непересічні підмножини можна розділити на два класи[2]:

1. Задача, в якій множина, підлягає розбивці, складається з кінцевого числа елементів. Такі задачі називаються дискретними задачами оптимальної розбивки.
2. Задачі, у яких множина підлягає розбивці, є континуальними. Такі завдання називають безперервними задачами розбивки.

Безперервні задачі оптимальної розбивки множини виникають при розв'язанні деяких задач про покриття, задач глобальної оптимізації для відшукання областей притягання локальних екстрімумів, при відшуканні вузлів оптимальних квадратурних формул для обчислення інтегралів, при обчисленні вирішальних правил у нелінійних задачах цілочисленого стохастичного програмування, у теорії статистичних рішень при розбивці простору ознак на непересічні класи, для визначення центрів класів при виборі початкових параметрів радіально-базисних мереж [4].

Постановка проблеми

У даній роботі розглядається один із класів задач безперервної розбивки, що мають назву безперервних задач k -медіан. Постановка такої задачі була вперше розглянута в роботах Є.М. Кисельової і найбільш повний перелік бібліографії з даної проблеми можна знайти в роботі, що недавно вийшла [2].

Необхідно відзначити, що безперервне задачі k -медіан вважається одним з найбільш складних задач комбінаторної оптимізації, і для нього невідомий точний поліноміальний алгоритм рішення. Метою даної роботи є проведення дослідження

ефективності застосування штучної імунної системи на етапі вибору початкового наближення в локально-оптимальному алгоритмі. Деякі локальні оптимуми безперервного задачі k-медіан можна розглядати як наближені рішення безперервного задачі оптимальної розбивки множини. Варто помітити, що в задачах k-медіан є й багато інших локальних оптимумів, обумовлених геометричними особливостями розташування елементів множини, що розбиває, серед яких може бути і глобальне рішення.

Математична постановка задачі

Відповідно до роботи [5], нехай задана кінцева множина $\Omega \subset E_2$. Необхідно розбити цю множину на N підмножин $\Omega_i \subseteq \Omega$ і визначити координати центрів цих медіан (або підмножин) $\tau_i \in E_2$, $i=1, \dots, N$ таким чином, щоб доставити мінімальне значення функціоналові

$$I(\tau, \Omega(\cdot)) = \sum_{i=1}^N \sum_{x \in \Omega_i} c(x, \tau_i). \quad (1)$$

Тут функція $c(x, \tau_i)$, відстань у якійсь матриці від крапки $x \in \Omega$ до медіани τ_i . Розбивку варто проводити таким чином, щоб

$$\bigcup_{i=1}^N \Omega_i = \Omega, \quad \Omega_i \cap \Omega_j = \emptyset, i \neq j. \quad (2)$$

Очевидно, що при фіксованому розташуванні центрів і відсутності обмежень на потужність підмножин умова приналежності елемента $x \in \Omega$ до підмножини Ω_i з медіаною в τ_i можна записати у вигляді:

$$c(x, \tau_i) = \min_{j=1, N} c(x, \tau_j). \quad (3)$$

Аналіз останніх досліджень і публікацій

Завдання k-медіан є одним з найбільше часто розв'язуваних в області комбінаторної оптимізації. Вона виникає безпосередньо при розміщенні proxy-серверів у мережі, при проектуванні розподілених мереж [6], серед додатків data mining, стиску і квантування даних [7]. Для рішення даного класу задач в [2] був запропонований метод рішення субградієнтного типу, заснований на наближенному обчисленні інтегралів і інтегралів у часткових похідних по формулі трапецій. Однак цей підхід приводить до появи великого числа нових екстремальних крапок, що утруднює дослідження оптимуму. У роботі [5] було запропоновано застосувати узагальнений градієнтний спуск для рішення відомого завдання геометричної кластеризації, у цій же роботі була запропонована схема використання цих оптимумів у генетичному алгоритмі, тобто додатково були використані оператори мутації і кросинговера. Фактично автори цієї роботи на першому етапі рішення завдання (1)-(3) будували локально-оптимальний алгоритм – узагальнений градієнтний спуск у з'єднанні з адаптивним вибором крокового множника. Вибір настільки простого алгоритму зроблений для того, щоб потужні оптимізатори (використовуючи, наприклад, розтягання простору) не затемнюювали роботу генетичного алгоритму. У роботі [8] ми спробували повторити описаний підхід, замінивши генетичні оператори алгоритмом клонального добору. При цьому ми виходили з того, що порівняно з генетичним алгоритмом [9] є наявність механізмів підтримки розмаїтості векторів рішень і навчання, відсутніх у випадку генетичного алгоритму. Отримані результати показали порівняно високу ефективність запропонованого методу. Однак нас не задовольняє “багатошарність” і складність

розробленого алгоритму, з іншого боку, небилиці використані всі можливості імунного алгоритму. Тому ми поставили за мету спростити і удосконалити імунний алгоритм, призначений для розв'язку даного класу задач.

Методика рішення

Формально алгоритм клональної селекції можна подати у такий спосіб [10]:

$$CLONALG = (P^l, G^k, l, k, m_{Ab}, \delta, f, I, \tau, AG, AB, S, C, M, n, d), \quad (4)$$

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 на нові.

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

$$\delta : P^l \rightarrow G^k \quad (4)$$

Функція (4) є функцією перетворення варіантів рішень із P^l у їхні внутрішні подання (G^k) у вигляді індивідуумів популяції. Ця функція інакше називається функцією експресії. Тут варто сказати, що на практиці при розробці ШС найчастіше вводять подібні перетворення з міркувань зручності застосування імунних операторів і обчислення афінності індивідуумів. Наприклад, вектор речовинних атрибутів розмірності l може бути перетворений у рядок біт довжини k , що уможливлює застосування специфічних операторів мутації та обчислення афінності за допомогою різних видів Хемінгової відстані. Незважаючи на зазначену важливість, даний аспект не розглядається у вивчених автором публікаціях, що стосуються проектування ШС. Тому терміни “генотип”, “фенотип” і “експресія” у даному описі були запозичені із близьких за функціональним призначенням і структурою еволюційних алгоритмів, хоча більше підходять до вживанню в контексті еволюції хромосом, ніж молекулярних структур антитіл. Передбачається також, що для кожного рішення $p \in P^l$ існує одне і тільки одне його подання $\delta(p) \in G^k$. При цьому в загальному випадку протилежне твердження невірне. Використовуючи узагальнене подання, можна ввести функцію афінності f :

$$f : P^l \times P^l \rightarrow \mathbb{R}^+. \quad (4)$$

При цьому завдання полягає в максимізації функції афінності.

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

$$I : G^k \times m_{Ab} \rightarrow AB(G^k). \quad (5)$$

Часто ініціалізація виробляється випадковим чином з використанням рівномірного розподілу.

Нехай Q – унарний стохастичний оператор перетворення на множини G^k , що використовує керуючу множиною 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, в якому одиничні позиції визначають номера біт індивідуума, що піддається мутації. Отже, функціональний запис оператора Q може бути поданий так:

$$Q : G^k \times K_Q \rightarrow G^k. \quad (6)$$

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

$$\forall k \in K_G : f(\mathcal{Q}(Ab_{opt}, k), Ag) \leq f(AB_{opt}, Ag). \quad (7)$$

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

$$\forall Ag \in AG : \exists Ab \in G^k \mid Ab = AB_{opt}. \quad (8)$$

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:

$$S : G^k \times K_S \rightarrow \{0, 1\}, \quad (9)$$

множина, що утвориться в результаті селекції:

$$G_S = \{Ab \in G^k \mid S(Ab, k_S) = 1\} \quad |G_S| = n. \quad (10)$$

Аналогічно виробляється добір індивідуумів у популяцію клітин пам'яті.

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

$$C : G_S \times K_C \rightarrow G_S. \quad (11)$$

Оператор мутації M з керуючою множиною K_M :

$$M : G^k \times K_M \rightarrow G^k. \quad (12)$$

Метадинаміка системи відображеня у вигляді функції заміщення гірших антитіл популяції:

$$R : G^k \times d \rightarrow AB_d(G^k). \quad (13)$$

Гірші антитіла попередньо відбираються за допомогою оператора селекції.

Блок-схема алгоритму клональної селекції наведена на рис. 1.

Нижче дається 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 антигена 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 і поширення. Вибрati 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 їх афінності, тобто чим вище афінність, тим більша кількість клонів створюється, і навпаки.

Крок 4. Дозрівання афінності. Піддати мутації всі клони популяції AB_C з імовірністю обернено пропорційно 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 п. 2, одержавши матрицю афінностей D_C . Вибрati з популяції AB_C n антитіл, для яких відповідний вектор-стовпець 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 клітинами пам'яті з M_R і перейти до п. 2, доки не буде досягнуто критерію останова.

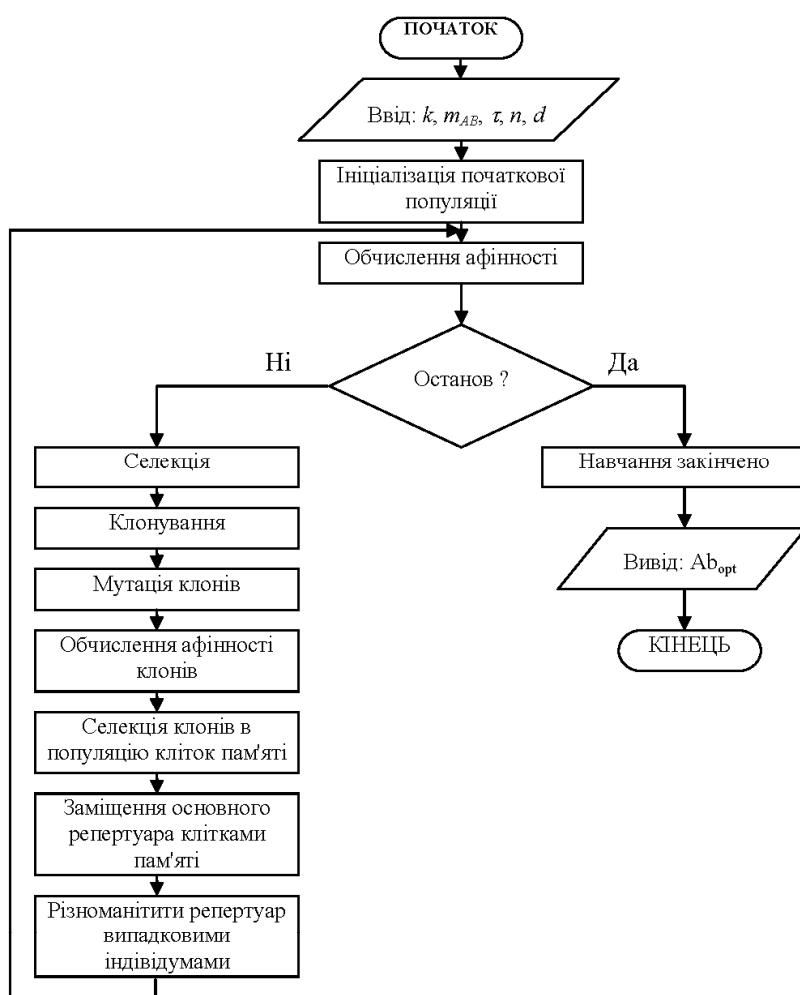


Рис. 1. Блок-схема алгоритму клональної селекції

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

Стосовно розглянутого завдання запропонованій алгоритм таких змін:

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

($AB = (\tau_1, \dots, \tau_N)$), де координата крапки $\tau_i = (\tau_i^x, \tau_i^y)$ $i = \overline{1, N}$,

що являють собою набір центрів (N) розбивки множини Ω на N підмножин.

Крок 2. Обчислення афінності. Для кожного антитіла Ab_j , $Ab_j \in AB$ обчислити його афінність стосовно створеної розбивки відповідно до набору центрів (3);

- афінність обчислюється відповідно до функціонала $I(\cdot)$ на основі розбивки за рівнянням (1)

Результати записати в матрицю афінностей $D: D = [AG | \times m_{Ab}]$ і $d_{ij} = f(AB_j, AG_i)$, $d_{ij} \in D$.

Інші кроки, 3-6, залишаються без зміни.

Отримані результати

При тестуванні алгоритму розглянуті завдання пошуку 9,16 і 25 медіан множини $\Omega \subseteq E_2$, що збігається з вузлами сітки з постійним кроком на одиничному квадраті. Для оцінки якості роботи алгоритму на цих прикладах вибиралися початкові наближення, розташовані близько від передбачених оптимумів. Як структура даних для візуалізації

отриманих результатів використовувалися діаграми Вороного, побудовані на медіанах підмножин (рис. 2).

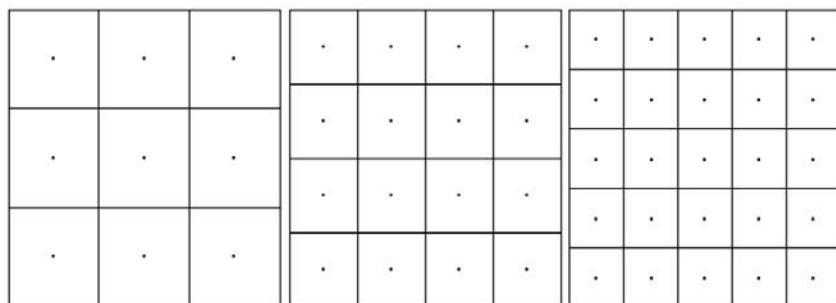


Рис. 2. Вид діаграм Вороного для передбачених оптимумів завдання

Порівняння результатів розв'язки (значення цільових функціоналів і відповідні діаграми) тестових прикладів алгоритмом, що реалізує штучну імунну систему, і мультистартовим алгоритмом GLO наведені в табл. 6.1 і на рис. 2.

Наведені результати показують, що у *всіх* проведених експериментах запропонована схема імунного алгоритму, що фактично являє собою керований випадковий пошук початкових наближень для алгоритму градієнтного типу, сходилася до рішень кращої якості.

Таблиця 1
**Порівняння результатів рішення, запропонованого алгоритмом ШІС
і двохетапним алгоритмом ШІС із GLO**

N	25x25		50x50		75x75		100x100		
	Краще рішення	Погрішність, %							
АЛГОРИТМ ШІС-II	9	0.13412297	0.000	0.133700719	0.000	0.13170068	0.000	0.13083976	0.000
	16	0.09984508	0.000	0.09828415	0.110	0.09749598	0.063	0.09705234	0.027
	25	0.07896117	1.104	0.07779670	0.002	0.07734904	-0.075	0.07711844	-0.131
АЛГОРИТМ ШІС-I	9	0.13412297	0.000	0.133700719	0.000	0.13170068	0.000	0.13095451	0.088
	16	0.10099527	1.151	0.09833996	1.001	0.09749611	0.064	0.09705307	0.028
	25	0.07915791	1.356	0.07786259	0.087	0.07735854	-0.063	0.07711883	-0.131

Цей факт можна пояснити тим, що в процесі роботи ШІС кращі локально-оптимальні рішення успадковуються наступними поколіннями.

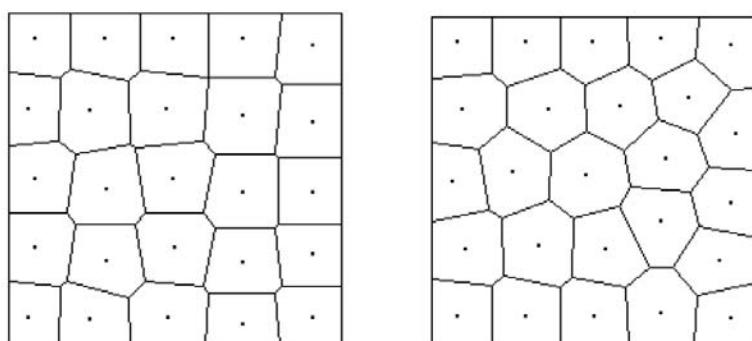


Рис. 3. а) розбивка множини на 25 областей алгоритмом ШІС-I;
б) алгоритмом ШІС-II

Відзначимо, що рішення двох задач розбивки вузлів сіток 75(75 і 100(100 на 25 підмножин краще за рішення, отримані з очевидних початкових наближень, однак не можна говорити про те, що вони глобальні. Крім того, імовірно, що різниця в ефективності алгоритмів (якості рішень) нівелюється з ростом розміру сітки. Це можна пояснити зменшенням числа локальних оптимумів при збільшенні розміру сітки.

Таким чином, при рішенні безперервної задачі *k*-медіан застосування модифікованого імунного алгоритму (ІІС-II) більш виправдане, ніж ШІС-I використовуючого генерування початкових наближень для локально-оптимальних алгоритмів (алгоритм GLO), особливо для задач великої розмірності і невеликої потужності множини.

Порівняльний аналіз імунного алгоритму проводився при таких параметрах штучної імунної системи:

- кількість антитіл у популяції = 64;
- кількість антитіл, які відбирають, (*n*) = 16;
- кількість антитіл, які видаляють, (*d*) = 32;
- множинний коефіцієнт, що визначає кількість створюваних клонів = 0,05;
- імовірність гіпермутації = 0,7;
- кількість поколінь імунної популяції до зупинки алгоритму (count) = 200.

Висновки

При тестуванні алгоритму розглянуті завдання пошуку 9,16 і 25 медіан множини $\Omega \subseteq E_2$, що збігається з вузлами сітки з постійним кроком на одиничному квадраті. Для оцінки якості роботи алгоритму на цих прикладах вибиралися початкові наближення, розташовані близько від передбачених оптимумів. Як структура даних для візуалізації отриманих результатів використовувалися діаграми Вороного, побудовані на медіанах підмножин.

Проведені порівняльні дослідження рішення даного завдання за допомогою модифікованого імунного алгоритму клонального добору (ШІС-II) і двохетапного рішення (ШІС-I) завдання (локально-оптимальний алгоритм). Отримані результати показують, що у *всіх* проведених експериментах запропонована схема імунного алгоритму сходилася до рішень кращої якості. Цей факт можна пояснити тим, що, по-перше, в процесі роботи ШІС кращі локально-оптимальні рішення успадковуються наступними поколіннями, по-друге, відмітною рисою імунного алгоритму є наявність механізмів підтримки розмаїтості векторів рішень.

Таким чином, запропонований варіант реалізації імунного алгоритму має набір важливих властивостей, включаючи відтворення кандидатів на рішення розподілених на всьому просторі параметрів, масове вироблення методів проб і помилок різних напрямків пошуку навколо найбільш підходящих початкових крапок, а також ефективний паралельний пошук, що стартує з набору крапок, що запам'ятовуються навчальною функцією.

Література

1. Киселева Е.М., Шор Н.З. Исследование алгоритмов решения одного класса непрерывных задач оптимального разбиения // Кибернетика и системный анализ.-1994.-№1.-С.84-96.
2. Киселева Е.М., Шор Н.З. Непрерывные задачи оптимального разбиения множеств: теория, алгоритмы, приложения: Монография.– К: Наукова думка, 2005. – 564 с.

3. Киселева Е.М. Математические методы и алгоритмы решения непрерывных задач оптимального разбиения множеств и их приложения: Автореф.дис. ... докт. физ.-мат. наук. – К., 1991. – 33 с.
4. Лебідь О.Ю. Застосування методу оптимального розбиття множин для підбору початкових параметрів радіальної мережі // Питання прикладної математики і математичного моделювання: Зб. наук. пр. / Ред. кол.: О.М.Кісельва (відп. редактор) та ін. – Дніпропетровська: Вид-во Дніпропетр. ун-ту, 2004. – С. 132-138.
5. Кузнецов К.А., Витязь О.Ю., Оголь А.А. Об эффективности генетических операторов в решении непрерывной задачи k-медиан // Питання прикладної математики і математичного моделювання: Зб. наук. пр. / Ред. кол.: О.М.Кісельва (відп. редактор) та ін. – Дніпропетровськ: Вид-во Дніпропетр. ун-ту, 2003. – С. 122-132.
6. Guyton J. Schwartz M. Locating nearby copies of replicated internet servers // ACM SIGCOMM'95. – 1995. – Р. 313-322.
7. Jain A.K., Murty M.N., Flynn P.J. data clustering: A review // ACM Computing surveys. – 1999. – Vol. 31. – №3. – Р. 264-323.
8. Литвиненко В.І., Горавский С.П., Фефелов А.О. Реалізація рішення безперервної задачі k-медіан за допомогою клонального алгоритму // Збірка наукових праць. У 5 т. Міжнародної наукової конференції “Інтелектуальні системи прийняття рішень та прикладні аспекти інформаційних технологій ISDMIT’2005”. – Євпаторія-2005. – Т. 3. – Секція 3. – С. 85-89.
9. T. Fukuda, K.Mori, M.Tsukiyama Parallel Search for Multi-Modal Function Optimization with Diversity and Learning of Immune Algorithm // Artificial Immune Systems and Their Applications / Edited by D.Dasgupta, Springer. – p. 306-1999, pp. 210-220.
10. Грицик В.В., Бидюк П.И., Литвиненко В.И., Фефелов А.А. Общие принципы построения искусственных иммунных систем // Збірка наукових праць: У 5 т. Міжнародної наукової конференції “Інтелектуальні системи прийняття рішень та прикладні аспекти інформаційних технологій ISDMIT’2006”. – Євпаторія-2005. – Розділ 2. – С. 241-252.