

БОЙКО Н. І.

Національний університет "Львівська політехніка"

<https://orcid.org/0000-0002-6962-9363>e-mail: Nataliya.i.boyko@lpnu.ua

БЛАЖЕВСЬКИЙ С. Г.

Чернівецький національний університет імені Юрія Федьковича

<https://orcid.org/0000-0003-3396-7253>e-mail: s.blazhevskyy@chnu.edu.ua

МЕТОДИКА ВИЗНАЧЕННЯ СТРУКТУРИ МОДЕЛІ ОПТИМАЛЬНОЇ СКЛАДНОСТІ

Генетичні алгоритми відносяться до класу евристичних алгоритмів. Вони забезпечують глобальне вирішення проблем оптимізації та являються перспективним напрямом в області оптимізації та моделювання. Вдосконалення розвитку генетичних алгоритмів відтворюється у більш складних еволюційних методах, що використовують реальні цифри та статистику. Одним з таких методів є Груповий метод врахування алгоритмів. В роботі аналізуються принципи роботи генетичних алгоритмів, його логіки та порівняти їх роботу з Груповим методом врахування аргументів і дослідити задачу побудови мостів. Наведено огляд публікацій з даної проблематики, які стали поштовхом для розвитку наведеної тематики. Генетичні алгоритми та груповий метод врахування аргументів є інструментами для застосування у багатьох областях, але слід пам'ятати, що вони мають обмеження: дають лише оціночні відповіді та вимагають мало часу для обрахування, також потрібні очищені вхідні дані. Отож, для покращення підходу для реалізації реальних програм потрібно проводити їх постійні дослідження та порівняння. Розглянуто генетичний алгоритм та метод групового використання аргументів. Описано принцип роботи та побудови генетичних алгоритмів. Наведено принцип роботи методу групового використання аргументів, його спектр алгоритмів. Використано структурно параметричну ідентифікацію та прогнозування методу групового врахування аргументів. Розглянуто самоорганізації моделей, які спрямовані на визначення структури моделі оптимальної складності. Розглянуто генетичний алгоритм та груповий метод врахування аргументів на предмет їх схожості та відмінностей. Виявлено переваги та недоліки методів дослідження. Виконання задачі побудови моста саме генетичним алгоритмом, що зумовлене аналізом його з Груповим методом врахування аргументів та доведенням, що для цього завдання генетичний алгоритм являється кращим рішенням.

Ключові слова: генетичний алгоритм; метод групового використання аргументів; інтелектуальний аналіз даних; нейронні мережі; алгоритм; популяція; фітнес; кросинговер; мутація.

NATALIYA BOYKO

Lviv Polytechnic National University

STEPAN BLAZHEVSRYY

Yuriy Fedkovych Chernivtsi National University

METHOD OF DETERMINING THE STRUCTURE OF THE MODEL OF OPTIMAL COMPLEXITY

Genetic algorithms belong to the class of heuristic algorithms. They provide a global solution to optimization problems and are a promising area in optimization and modelling. Genetic algorithm development is reproduced in more sophisticated evolutionary methods that use real numbers and statistics. One such method is the Group method of accounting for algorithms. The paper analyzes the principles of genetic algorithms their logic, compares their work with the Group method of argumentation, and explores building bridges. An overview of publications on this issue, which became the impetus for the development of this topic. Genetic algorithms and group argumentation are tools for many applications, but keep in mind that they have limitations: they give only evaluative answers and require little computational time, and you need refined input. Therefore, to improve the approach to implementing existing programs, it is necessary to conduct constant research and comparison. The genetic algorithm and the method of group use of arguments are considered. The principle of operation and construction of genetic algorithms is described. The focus of the process of group use of discussions and its range of algorithms is given. Structurally parametric identification and prediction of the method of group consideration of arguments are used. Self-organizations of models are considered to determine the structure of the model of optimal complexity. The genetic algorithm and group method of viewing statements for similarities and differences are considered. The advantages and disadvantages of research methods are revealed. The task of building a bridge is performed by a genetic algorithm, which is due to its analysis with the Group method of taking into account arguments and proving that the genetic algorithm is the best solution for this task.

Keywords: genetic algorithm; method of group use of arguments; data mining; neural networks; algorithm; population; fitness; crossover; mutation.

Постановка проблеми у загальному вигляді

та її зв'язок із важливими науковими чи практичними завданнями

Останнім часом розвиток в області штучного інтелекту широко розвиваються. Завданням дослідження є привертання уваги до не дуже популярних напрямків штучного інтелекту, таких як генетичні алгоритми. Вони являються швидким і потужним підходом до вирішення проблем, проте їм притаманні деякі обмеження [1, 2].

Перше, і найважливіше, врахування при створенні генетичного алгоритму – це вміння подати проблему. Мова, що використовується для визначення конкретних рішень [3], має бути надійною; тобто він повинен вміти «впізнавати» випадкові зміни, щоб вони не призвели до фатальних помилок.

Генетичні алгоритми – це цікавий спосіб дослідження просторів параметрів на основі результатів функції фітнесу. Досліджені у роботі приклади створювались під час оптимізації її показників [4, 5].

Переваги генетичного алгоритму:

- Не потрібно знати як працює система. Просто потрібно вміти вимірювати, наскільки добре вона працює. Тобто потрібно розмножити кращу групу «вхідних даних» або «рішень».
- Легко виявити глобальний оптимум та уникати захоплення локальними оптимумами.
- Паралелізм легко модифікований і пристосований до вирішення різних проблем.
- Великий та широкий вміст пошуку простору рішення.
- Процес оптимізації, оснований на знаннях, що використовується для оцінки функцією фітнес.

Хоча генетичні алгоритми виявились перспективним підходом до вирішення проблем, проте існують деякі обмеження.

Перше, і найважливіше, врахування при створенні генетичного алгоритму – це подання проблеми. Мова, що використовується для визначення конкретних рішень, має бути надійною; тобто він повинен вміти враховувати випадкові зміни, щоб вони не призвели до фатальних помилок [6, 7].

Однією із головних перешкод генетичних алгоритмів є кодування функції придатності (оцінювання), для отримання кращих рішень для вирішення проблем. Неправильний вибір функції фітнесу може призвести до критичних проблем, таких як неможливість знайти рішення проблеми, а ще гірше – повернути неправильне рішення проблеми [1, 8].

Поряд з правильним вибором функції фітнесу, інші параметри генетичного алгоритму, такі як чисельність популяції, мутація та швидкість кросоверу, також повинні вибиратися обережно [3, 9]. Невеликий розмір популяції не дає генетичному алгоритму достатнього простору рішення для отримання точних результатів. Висока частота генетичних змін або погана схема відбору призводить до зриву корисної схеми, і популяція може здійснити забагато помилок при швидкій зміні та спричинити конвергенцію.

Не доцільно використовувати генетичні алгоритми для аналітичних проблем. Хоча вони їх робота направлена на знаходження точних рішень для подібних проблем [5, 10, 11]. Традиційні аналітичні методи можуть знайти ті самі рішення за менший час і за допомогою декількох кроків обчислення.

Передчасна конвергенція – ще одне питання, яке слід враховувати в процесі генерації рішень генетичного алгоритму. Цей вид неефективності в основному спостерігається при незначних помилках, коли навіть випадкові зміни швидкості розмноження можуть призвести до того, що один генотип стане домінувати над іншими [5].

Аналіз останніх джерел

В даний час спостерігається сплеск кількості публікацій в області штучних нейронних мереж. Першими у 60-х роках, були роботи Ф. Розенблатта «Принципи нейродинаміки» [12]. Поштовхом нового буму в цій галузі стали роботи Хопфілда (нейронні мережі Хопфілда) і Хінтона [4, 13] (нейронні мережі зі зворотним поширенням помилки). Отримано обнадійливі результати застосування нейронних мереж різних типів в самих різних областях [6,8]. Це наприклад, розпізнавання дуже зашумлених сигналів, розпізнавання текстів по їх матричному зображенню.

Формулювання цілей статті

Метою роботи є проаналізувати принципи роботи генетичних алгоритмів, його логіки та порівняти їх роботу з Груповим методом врахування аргументів і дослідити задачу побудови мостів.

Виклад основного матеріалу

Один із сучасних алгоритмів для вибору ознак – генетичний алгоритм. Він є стохастичним методом оптимізації функцій, який оснований на механіці природної генетики та біологічної еволюції.

У природі, гени організмів, як правило, еволюціонують протягом наступних поколінь, щоб краще адаптуватися до навколишнього середовища [6, 14]. Генетичні алгоритми працюють на популяції особин, для отримання кращих наближень. У кожному поколінні нова популяція створюється шляхом вибору особин відповідно до рівня їхньої придатності в проблемній галузі та рекомбінації їх разом, використовуючи операторів, запозичених із природної генетики. Потомство може мутувати. Для кращого розуміння на рис. 1 представлена схема вибору ознак за допомогою генетичного алгоритму.

Цей процес призводить до еволюції популяцій індивідів, які краще підходять до визначеного оточення, а ніж індивіди, з яких вони були створені в природній адаптації. У описаному випадку кожна особина в популяції являє собою нейронну мережу, де гени є бінарними значеннями і являють собою включення або особливості в моделі. Кількість генів – це

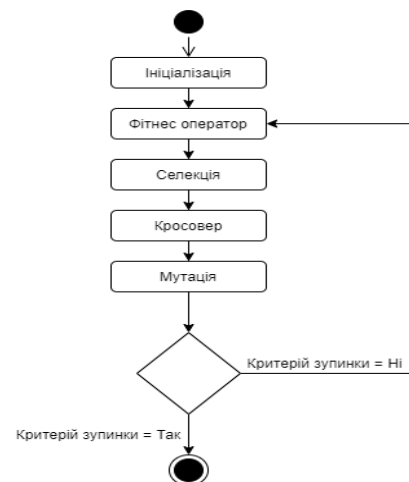


Рис. 1. Діаграма діяльності вибору ознак за допомогою генетичного алгоритму

загальна кількість вхідних змінних у наборі даних. Для кожної програми необхідно вибрати кількість осіб або кількість популяції. Зазвичай це встановлено як $10N$, що становить N – кількість функцій [8, 15].

При описі моделі слід детально описати оператори та відповідні параметри, що використовуються генетичним алгоритмом.

1. Оператор ініціалізації

Першим кроком є створення та ініціалізація особин у популяції. Оскільки генетичний алгоритм є методом стохастичної оптимізації, то гени особин зазвичай ініціалізуються навмання [6].

Для ілюстрації цього оператора слід розглянути модель прогнозування, що представлена нейронною мережею з 6 можливими ознаками (рис. 2). Якщо згенерувати популяцію з 4-х особин, то отримаємо 4 різні нейронні мережі з випадковими ознаками.

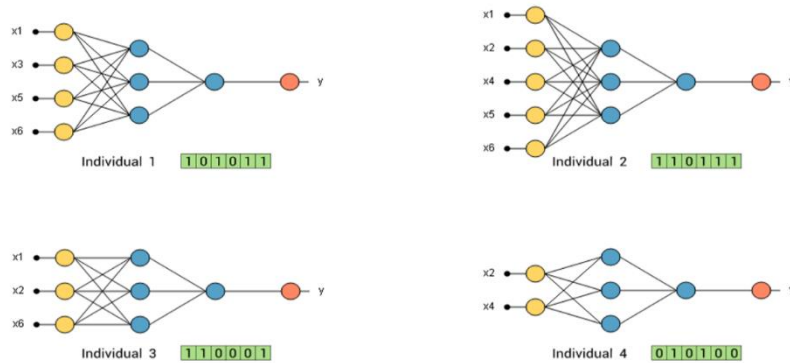


Рис. 2. Ілюстрація нейронних мереж з популяції 6-и та 4-х особин

Як видно на рис. 2, кожен індивід представлений 6-ма бінарними генами. Кожен позитивний ген означає, що відповідна особливість включена в нейронну мережу.

2. Оператор призначення фітнесу

Після ініціалізації потрібно призначити значення пристосованості кожній особині в популяції. Для цього слід натренувати кожен нейронну мережу, а потім оцінити їх помилку за допомогою екземплярів вибору [10, 15].

Очевидно, що висока помилка вибору означає низьку пристосованість. Особини з більшою пристосованістю мають високу ймовірність бути обраними для рекомбінації. Для цього слід використати метод присвоєння фітнесу.

За допомогою цього методу сортуються помилки відбору всіх особин. Тоді, пристосованість, яка присвоюється кожній особині, залежить лише від її позиції в рангу особин, а не від фактичної помилки відбору.

Значення фітнесу, що присвоюється особині методом, оснований на ранзі, розраховується за формулою 1:

$$\Phi(i) = k \cdot R(i), \tag{1}$$

де $i=1, \dots, N$; k – константа, яка називається селективним тиском, і її значення фіксується між 1 і 2.

Більші значення селективного тиску роблять найбільш придатних особин з більшою ймовірністю до рекомбінації. Параметр $R(i)$ – це ранг індивідуального i .

Повертаючись до описаного вище прикладу, в таблиці 1 розрахована помилка вибору, ранг та відповідна пристосованість кожної особини.

Таблиця 1

Помилка вибору			
	Помилка вибору	Ранг	Фітнес
Мережа 1	0,9	1	1.5
Мережа 2	0,6	3	4.5
Мережа 3	0,7	2	3.0
Мережа 4	0,5	4	6,0

Для демонстрації, задано $k = 1,5$ для обчислення значення фітнесу. Отож, для обчислення пристосованості популяції слід тренувати 4-ри різні нейронні мережі.

Наведені вище значення придатності можна візуально представити на круговій діаграмі (рис. 3), де площа для кожної особини є пропорційною до її пристосованості.

З рис. 3 видно, що найбільш пристосованою мережею є четверта, тобто та, яка має найбільшу площу (40%), а найменш пристосованою особиною є перша, що має найменшу площу (10%).

3. Оператор вибору

Після виконання процесу фітнесу, оператор вибору обирає особин, які рекомбінуються для наступного покоління.

Особини, які найбільш схильні до виживання, – це більш пристосовані до навколишнього середовища. Тому оператор відбору вибирає особин відповідно до рівня їхньої пристосованості. Кількість відібраних особин становить $N/2$, що становить N чисельність популяції [7, 12].

Вибір елітарності змушує найкращих особин виживати для наступного покоління. Розмір елітизму контролює кількість особин, яких безпосередньо вибирають, і зазвичай він встановлюється на невелике значення (1,2, ...).

Одним із методів відбору є колесо рулетки, який відомий під назвою стохастичний відбір із заміною. Цей метод дозволяє розміщувати всіх особин на рулетці з областями, що є пропорційними до їх пристосованості. Потім рулетку прокручують і особин вибирають навмання. Відповідна особина вибирається для рекомбінації.

Наступний рис. 4 ілюструє процес відбору для описаного прикладу.

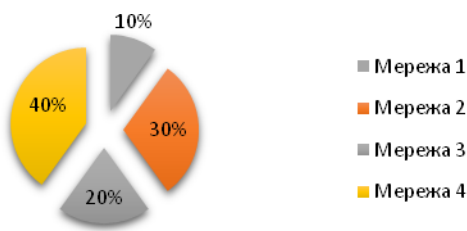


Рис. 4. Діаграма з вибором компонентів

отримання чотирьох потомств у новій популяції, поки нова популяція не досягне в кількості попередню популяцію [13, 14].

Рівномірний кросовер – це метод, який вирішує чи підходять ознаки потомству від обох батьків.

Наступний рис. 5 ілюструє метод рівномірного кросовера для описаного прикладу.

На рис. 5 представлено чотири потомства від двох батьків. Певні особливості кожної нейронної мережі відповідають одному предкові, а інші – другому. В даному прикладі чисельність потомства залишається незмінною.

5. Оператор мутації

Оператор кросовера може генерувати потомків, які дуже схожі на батьків. Це може призвести до низької різниці між поколіннями.

Оператор мутації вирішує цю проблему, замінюючи значення деяких ознак у нащадках навмання.

Щоб вирішити, чи функція мутує, то потрібно генерувати випадкове число: 0 або 1. Якщо це число є нижчим за значення, яке називається швидкістю мутації, то ця змінна повертається.

Частота мутації дорівнює $1/m$, де m – кількість ознак. За допомогою цього значення потрібно мутувати одну особливість кожної особини (статистично).

Наступний рис. 6 зображує мутацію одного з потомства нового покоління.

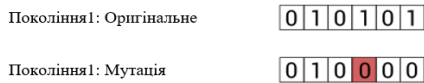


Рис. 6. Процес мутації

критерію зупинки. Кожне покоління, швидше за все, більше адаптується до навколишнього середовища, а ніж старе.

На наступному рис. 7 показана типова поведінка генетичного алгоритму. Синя крива представляє середню помилку відбору особин у кожному поколінні. Червона – являє собою помилку відбору найкращого індивіда серед усіх поколінь.

Як видно з рис. 7, середня помилка вибору на кожному поколінні зближується до мінімального значення. Розв'язання цього процесу – найкраща особина, яка коли-небудь існувала. Це відповідає мережі з найменшою помилкою

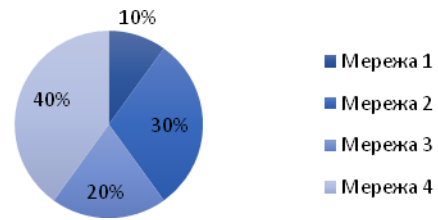


Рис. 3. Діаграма пристосованості для 4-х особин

У цьому випадку нейромережа 4 була обрана вибором елітарності, а 3 – обрана колесом рулетки. Отож, особина 2 більш придатна, аніж 3, яка не була обрана через стохастичний характер генетичного алгоритму. В даному прикладі кількість відібраних особин становить половину чисельності популяції.

4. Оператор кросовер

Після того, як оператор відбору обрав половину сукупності, то оператор кросовера рекомбінє вибраних особин для створення нової популяції. Цей оператор вибирає двох особин навмання і поєднує їх особливості для

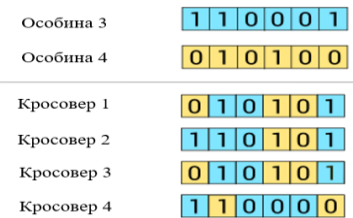


Рис. 5. Опис процесу мутації

Як видно з рис. 6, четвертий вхід нейронної мережі був вимкнений. Процес призначення, відбору, рекомбінації та мутації фітнесу повторюється до досягнення

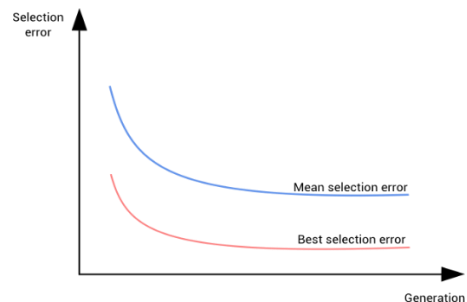


Рис. 7. Графік селекційного відбору

відбору серед усіх тих, що були проаналізовані. В наведеному прикладі найкращою особиною є подана на рис. 8 мережа.

Для початку слід розрахувати значення фітнес функції. Вихід функції повинен бути більшим, якщо результат буде більш оптимальним або наближеним до бажаної поведінки. Якщо потрібно побудувати мост, то функція фітнесу може стосуватися:

- будова структури в рівновазі.
- кількість використаних елементів.
- сума відхилення.

$$fitness = \begin{cases} m, & \text{якщо } m < 2^{n-1} \\ 2^n - m, & \text{решта значень} \end{cases}$$

де m – кількість пропусків.

Вхід фітнес-функції – ДНК. Для ДНК мосту слід використати лише 2 стани – включення або вимкнення. Можна розглядати ДНК як послідовність двійкових значень. Кожен індекс послідовності належить до елемента в мережі. Такий елемент являється стрижнем на мосту і може бути як увімкненим, так і вимкненим. Якщо він включений, то елемент буде побудований.

На рис. 9 показана сітка, яка містить можливі місця для розміщення елемента. Штрихові лінії означають, що може бути елемент, але він ще не побудований. Місце розташування елементів пронумеровано від одного до одинадцяти.

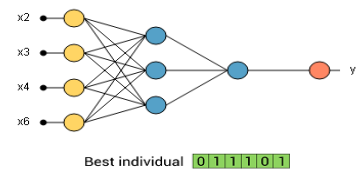


Рис. 8. Приклад відбору кращої мережі

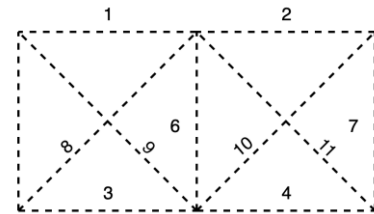


Рис. 9. Порожня сітка

Рядок ДНК, що належить цій сітці порожніх елементів, є масивом з одинадцяти нулів. На Рис. 10 показано:

- В першому рядку: послідовність ДНК.
- В другому рядку: відповідний промінь стану ДНК.

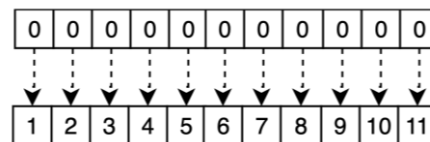


Рис. 10. Послідовність ДНК, що належить до порожньої мережі

Якщо випадковим чином змінити деякі значення ДНК з нуля на один, то активується елемент і відповідно він буде побудований на сітці. На рис. 11

показана сітка та відповідна ДНК при активації пучка № 3, 7 та 10.

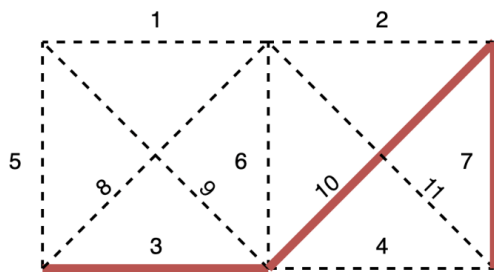


Рис. 11. Сітка з активованими елементами

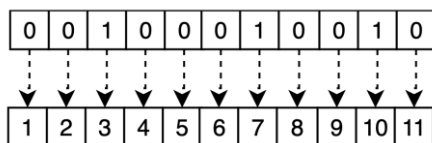


Рис. 12. ДНК, що належить до сітки з активованими елементами.

ДНК, що належить до цієї сітки, показаний нижче на рис. 12.

У першому поколінні ДНК буде генеруватися випадковим чином. Ймовірність того, що випадкова породжена послідовність покаже високий показник придатності є дуже низькою, тому слід сформувати цілу популяцію ДНК. Можна обчислити показник придатності для всієї сукупності та визначити, які послідовності ДНК будуть кращі мости. Перше покоління, ймовірно, не буде містити оптимальних структур. Тому потрібно змінити ДНК випадковим чином, зберігаючи частини ДНК, що призводять до високого показника придатності. Генетичні алгоритми можуть це зробити через селекцію, кросовер та мутацію.

Коли визначається придатність особини до

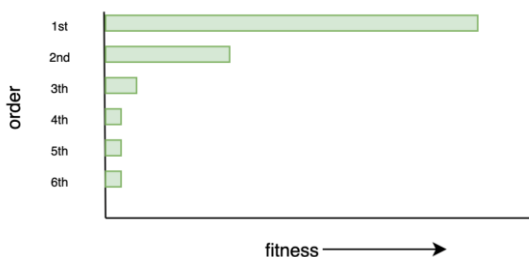


Рис. 13. Ймовірність того, що буде обрана пропорційна балу фітнесу

Маючи всю ДНК, що надходить лише від кількох батьків, і викликають невеликі коливання в сукупності, то це збільшують шанс зациклюватися на локальних максимумах. Для того, щоб підтримувати здорове потомство, шанс бути обраним буде пропорційним рангу батька, а не фітнес-балом.

На рис. 14 показано ймовірність вибору, що дорівнює рангу батьків. Можна побачити, що наступне покоління матиме більшу кількість ДНК, які надходять від батьків з найвищим балом.

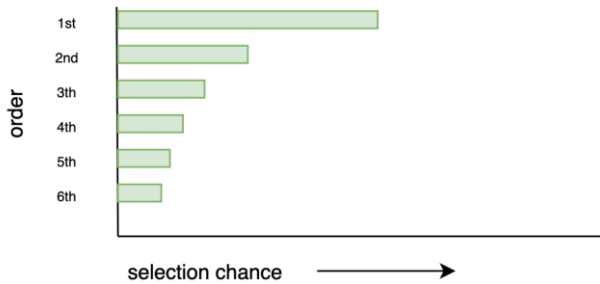


Рис. 14. Ймовірність вибору, яка пропорційна рангу батьків

у, відбору, кросовера та мутації будуть повторюватися, поки модель не дасть задовільного результату, або вона не втримається у своєму локальному оптимуму.

Мости будують симетрично. ДНК містить інформацію про ліву частину моста, права – його дзеркальне зображення. Для кожного моста, опори розмішують на зовнішніх вузлах. Наведений рис. 16 демонструє найкращий результат для сітки розміром 10x1, 10 одиниць довжини у напрямку x та 1 одиниця довжини у напрямку y.

Після того, як процес відбору пройшов успішно, то відбувається обмін ДНК – кросовер. Його можна зробити, об'єднавши два масиви ДНК випадковим чином. Цей процес обміну ДНК може здійснюватися різними способами. У дослідженні використано метод, який дозволяє обраним батькам обміняти ДНК повністю випадково [17]. На рис. 15 показано, як міг би виглядати обмін ДНК, якщо випадково поміняти місцями ДНК.

Відповідно під час копіювання ДНК є ймовірність випадкової мутації. Ця випадкова мутація дає змогу досліджувати нові стани ДНК, які недоступні генофонду популяції. Етапи фітнес

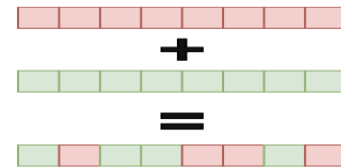


Рис. 15. Випадкова стратегія кросоверів

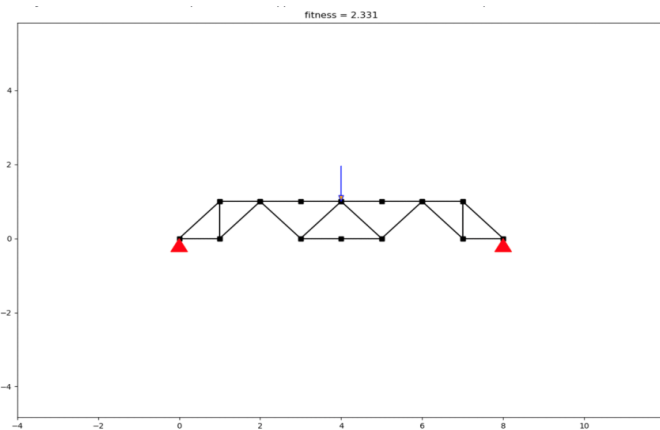


Рис. 16. Результат на сітці 10x1 та двох навісних опорах

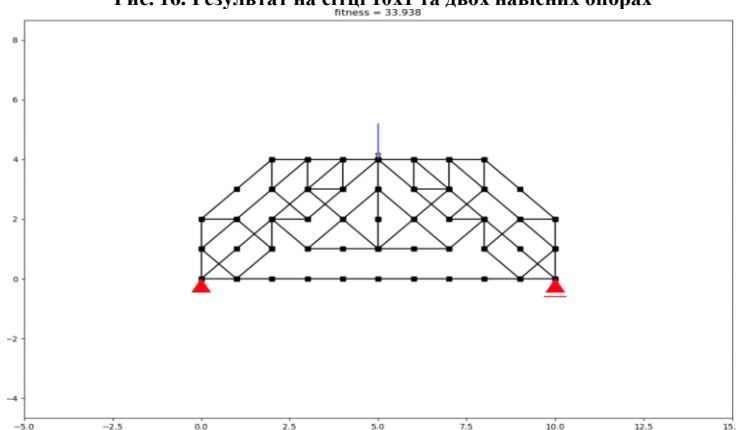


Рис. 17. Результат на сітці 10x5

На рис. 16 можна бачити, що він втрачає деякі елементи. Він також буде трикутні форми. Оскільки немає згинальної ємності та сітка недостатньо висока для побудови дуги стиснення або натяжної мотузки. Це здається єдиним правильним способом отримати жорстку конструкцію. Натяжні прутти, які зазвичай знаходяться в нижній частині гнучої балки, не йдуть до опор. Це краща структура щодо цього фітнесу, тому що опори навішуються з обох боків, що дозволяє побудувати арку стиснення.

На рис. 17 побудований найкращий міст популяції. Оскільки підшипники не можуть підтримувати вагу арки, то він створює натяжний стрижень (горизонтальні елементи на дні).

Висновки з даного дослідження і перспективи подальших розвідок у даному напрямі

Отож, порівнюючи роботу генетичного алгоритму та групового методу використання аргументів можна навести схожість між алгоритмами, але вони застосовуються для різних типів задач.

Генетичні алгоритми – це цікавий спосіб дослідження просторів параметрів на основі результатів фітнесу. Наведені вище приклади були створені під час оптимізації показників фітнесу на основі кількості відхилень та кількості елементів. Недоліком цього методу являється постійне визначення нової сітки, жорсткість згину, умову опори чи будь-яке інше обмеження. Алгоритм вимагає певного обчислення для того, щоб знайти (оптимальне) рішення.

Перевагою вибору генетичного алгоритму є модель, яку потрібно тренувати лише один раз, а згодом можна миттєво нанести обмеження на мостові конфігурації.

References

1. Hopfield J. Neural Networks and Physical Systems with Emergent Collective Computational Abilities. Proceedings of the National Academy of Sciences. 1982. P. 2554-2558. DOI: 10.1073/pnas.79.8.2554
2. Bartz-Beielstein T. SPOT: An R package for automatic and interactive tuning of optimization algorithms by sequential parameter optimization. Technical Report arXiv:1006.4645. CIOP Technical Report 05--10, Cologne University of Applied Sciences. 2010.
3. Bartz-Beielstein T., Lasarczyk C., and Preuss M. The sequential parameter optimization toolbox. In Bartz-Beielstein et al., editors, Experimental Methods for the Analysis of Optimization Algorithms, Springer, Berlin, Heidelberg, New York. 2010. P. 337-360. DOI: 10.1109/CEC.2005.1554761
4. Bischl B. The mlr package: Machine learning in R. <http://mlr.r-forge.r-project.org>, accessed 14.04.2011.
5. Hansen N. The CMA evolution strategy: a comparing review. In J. Lozano, P. Larranaga, I. Inza, and E. Bengoetxea, editors, Towards a new evolutionary computation. Advances on estimation of distribution algorithms. 2006. P. 75-102. DOI: 10.1007/3-540-32494-1_4
6. Jurecka F. Automated metamodeling for efficient multi-disciplinary optimization of complex automotive structures. In 7th European LS-DYNA Conference, Salzburg, Austria. 2009. P. 205-218
7. Konen W., Bartz-beielstein T., Koch P., Friese M., Flasch O., Naujoks B. Tuned data mining: A benchmark study on different tuners. Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation (GECCO '11). ACM New York, NY, USA. 2011. P. 1995-2002.
8. Farzi S. A New Approach to Polynomial Neural Networks based on Genetic Algorithm. International Journal of Computer Systems Science and Engineering. 2008. P.180-187.
9. Nariman-zadeh N., Jamali A. Pareto Genetic Design of GMDH-type Neural Networks for Nonlinear Systems. In Proceedings of International Workshop on Inductive Modeling. 2007. P. 96-103.
10. Koshulko O.A., Koshulko A.I. Adaptive parallel implementation of the Combinatorial GMDH algorithm. In Proceedings of International Workshop on Inductive Modelling (IWIM-2007), Prague, 2007. P. 71-77.
11. Taušer J., Buryan P. Exchange Rate Predictions in International Financial Management by Enhanced GMDH Algorithm. Prague Economic Papers No.3, University of Economics, Prague. 2011. P. 232-249.
12. Kondo T., Pandya A. GMDH-type Neural Networks with a Feedback Loop and their Application to the Identification of Large-spatial Air Pollution Patterns. In proceedings of the 39th SICE Annual Conference International Session Papers, 112A-4, Iizuka, Japan. 2000. P. 1-6.
13. Clinton Sh. Genetic Algorithms with Python. Paperback. 2016. P. 452.
14. Andrakhanov, A. Tyryshkin. Robot Control on basis of Inductive Method of Selforganization of Models. International Journal of Advanced Robotic Systems. 2014. P. 131-150.
15. Atashkari K., Nariman-zadeh N., Darvizeh A., Yao X., Jamali A., Pilechi A. Genetic Design of GMDH-type Neural Networks for Modelling of Thermodynamically Pareto Optimized Turbojet Engines. WSEAS Transactions on COMPUTERS, Issue 3, Volume 3. 2004. 1109-2750.