

МОДЕЛЬ РОЗВИТКУ ПРОСТИХ ОРГАНІЗМІВ З ВИКОРИСТАННЯМ ГЕНЕТИЧНИХ АЛГОРИТМІВ ТА ГЛИБИННОГО НАВЧАННЯ

В. С. Татенко¹, О. А. Орехов¹

¹Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»,
ІН Фізико-технічний інститут

Анотація

Розглянуто задачу оптимізації моделі розвитку організмів з використанням генетичних алгоритмів та глибинного навчання. Для її вирішення було побудовано модель поведінки групи простих організмів. Підібрано оптимальні параметри та методи етапів генетичного алгоритму для заданої моделі.

Ключові слова: еволюційні алгоритми, генетичні алгоритми, нейронні мережі, популяція

Вступ

Еволюція завжди була і є цікавим механізмом для вивчення через розвиток і переваги, які вона надає, якщо проходить успішно. Її можна розглядати з багатьох сторін, однією з яких є оптимізаційні методи, як генетичний алгоритм.

Навчання є ваговою складовою еволюції. Від його успішності залежить, чи буде досягнуто розвитку та виживання організмів.

Для створення моделі розвитку простих організмів будуть використані генетичний алгоритм та нейронні мережі [1]. Будуть реалізовані такі біологічні поняття розвитку, як елітизм, селекція, схрещування, мутація.

1. Огляд літератури

1.1. Еволюційний алгоритм

Еволюційний алгоритм тісно пов'язаний з механізмами зміни форм живих організмів, їх пристосованості до виживання, та подальшого розвитку. Ті, які прагнуть розвитку, під час еволюції надають найкорисніші характеристики для нащадків. Це надає більше ймовірності на успішне виживання організму.

Процес вирішення обчислювальної задачі еволюційним алгоритмом протікає через розвиток набору екземплярів (населення), які несуть в собі рішення. Кожен з цього набору зростає по своєму, після чого завдяки методам селекції обираються кращі екземпляри, як батьки наступного покоління.[2]

Також в алгоритмі присутній оператор мутації екземпляра, який надає більше простору для огляду рішень задачі, які запропонує модель. Треба зазначити, що це надає захист від конвергенції, що тягне за собою погані наслідки при вирішенні поставленої задачі.

1.2. Генетичний алгоритм

Генетичний алгоритм є членом сімейства еволюційних алгоритмів, які є методами обчислювального пошуку, натхненими природним відбором. [3]

Оскільки еволюція впливає на сутності, наступні версії популяції, які ще називають поколіннями, розвиваються, щоб максимізувати свою придатність, тобто здатність виживати в середовищі або виконувати задане завдання, або максимізувати показник продуктивності. Подібним чином одноклітинні організми еволюціонували до homo sapiens, ЕА перетворюють невитончені тіла на складні, адаптивні сутності. Як комп'ютерна програма, весь цей процес еволюції починається зі способу представлення цих сутностей: генетичного уявлення.

1.2.1. Алгоритми генетичного алгоритму

Генетичні алгоритми, по суті, виконують п'ять основних кроків, які присутні у всіх ГА, хоча з різними реалізаціями. Спочатку створюється початкова сукупність сутностей: перше покоління. Далі описані кроки виконуються протягом вказаної кількості ітерацій. Суб'єкти поточного покоління оцінюються з точки зору пристосованості. По закінченню виконання ітерацій обираються батьки з найбільшою пристосованістю для подальшого розмноження. Розмножуючись, вони створюють потомство, яке переймає частину їх генетичної інформації. На отримані генотипи дітей впливають мутації, які збільшують генетичну різноманітність популяції. Повторення цієї процедури до тих пір, поки ми не отримаємо повністю нове покоління, являє собою крок ГА. Після створення нового покоління ми повторюємо кроки програми, поки ця еволюція не досягне певної мети або поки не буде досягнуто максимальної кількості вказаних ітерацій.

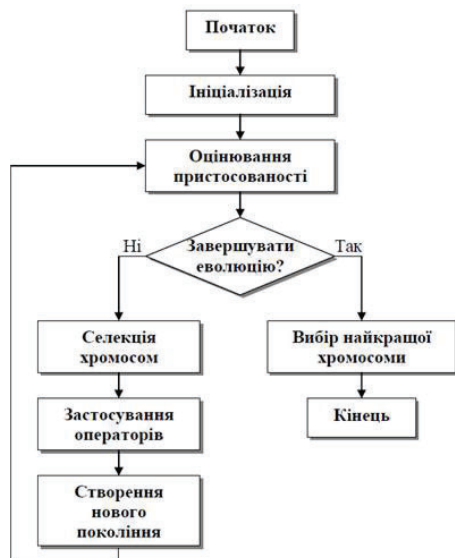


Рис. 1. Блок-схема генетичного алгоритму

2. Реалізація моделі

В нашій моделі буде реалізовано розвиток простих організмів. Задля забезпечення розвитку організмів використовуємо генетичний алгоритм. Також застосуємо нейронні мережі для реалізації «мізків» наших організмів.

2.1. Поле

Поле не має кордонів: коли об'єкт потрапляє на ліву стіну, він «телепортується» на праву стіну. Під час тестів з полем, в якого наявні кордони, вся популяція геномів мала тенденцію прилипнути до однієї зі стін, не перетворюючись на більш гнучку популяцію. Однак наявність стін без меж супроводжується проблемою, яку ще не вирішено: геноми, які знаходяться, наприклад, біля лівої стінки, не виявлять ворогів, які знаходяться близько до правої стінки, навіть якщо відстань між організмами може бути дуже малою.

На полі присутні 2 різні об'єкти: їжа та організми. Їжа нерухома і не має «мізків». Кожен шматочок їжі має статичну харчову цінність. Після того, як їжа була з'їдена, вона просто переміщається на нове місце на полі.

2.2. Організми

Організми, здатні приймати рішення про інших індивідів та їжу за допомогою нейронних мереж. Кожен геном бачитиме лише 3 найближчих організми і 3 найближчі кульки їжі в певному радіусі. Також вони повільно зменшуються в розмірі (0.2% від їх маси), коли не харчуються (або поїдаючи інші організми, або їжу). Вони можуть їсти інші організми, лише коли вони на 10% більші, і вони можуть вільно переміщатися крізь інші організми, розміри яких менше ніж на 10%. В залежності від їх маси, вони мають різний розмір та колір.

Геноми не можуть прискорюватися навмисно, вони відразу пристосовуються до швидкості залежно від їх маси, яку дає їхній мозок.

2.3. Генетичний алгоритм та нейронна мережа

Генетичний алгоритм є ядром штучного інтелекту. У першому кадрі певна кількість організмів ініціалізується з нейронною мережею як мозком. Її структура є рідкою. Вона починається з заданого пулу вузлів, а потім створюються випадкові з'єднання між ними. Мережа має 16 входів (ця кількість залежить від того, скільки об'єктів їжі та опонентів, може помітити організм) і 2 виходи (напрямок руху та швидкість).

Мозок представляє популяцію покоління. Потім ці мізки розвиваються, поміщаючи все населення в єдине поле і дозволяючи їм змагатися один з одним. Найпридатніші мізки переміщуються в наступне покоління, менш пристосовані мізки мають високі шанси бути видалені.

Описаний щойно процес має на увазі такі механізми: елітизм, селекція, схрещування, мутація.

Елітизм – це відсоток найрозвиненіших організмів, які перейдуть в наступне покоління. В моделі таких екземплярів буде 20% з поточного покоління.

Селекція – це вибір батьків для створення наступного покоління. Для успішного розвитку, мають обиратися найуспішніші індивіди, як батьки. Існує не один спосіб, як зробити цей вибір (метод колеса, стохастичний метод, рейтинговий відбір, турнірний відбір, тощо), але в даній моделі буде використовуватися рейтинговий відбір, а оцінюватись буде середня пристосованість покоління, яка буде вираховуватись за формулою:

$$\frac{\sum_{i=1}^n \text{gen.organism}_i.\text{fitness}}{n},$$

де n – кількість організмів у популяції.

Табл. 1. Приклад рейтингового відбору

Хромосома	Значення пристосованості	Ранг
A	0.1	6
B	-2	7
C	2	5
D	5	4
E	5.5	2
F	5.1	3
G	5.9	1

Після того, як обрались батьки покоління, відбувається схрещування, для створення нового потомства. Для цього існує багато методів (одноточкове, багатоточкове, однорідне, повна арифметична рекомбінація, тощо). Описувана модель буде передбачати одноточкове (обирається випадкова точка схрещування в хромосомі, одна половина

від одного батька, інша від іншого), двоточкове (обираються точки схрещування, які будуть розділяти хромосоми батьків на частини, після чого йде наступне: перша частина залишається, наступна міняється місцями, залишається), випадкове з двох попередніх (метод передбачає випадкову генерацію параметра, за значенням якого буде обрано, яким з двох методів буде створено нащадка).

Після створення нової популяції, деякі організми підлягають мутації, для підтримання різноманітності покоління, яке в свою чергу покращує роботу та результат роботи алгоритму. Є немала кількість

варіації того, які мутації можуть відбутись (біт-фліп, обмін, скрамбл, інверсія, тощо). В моделі буде представлено декілька варіантів, результати яких потім будуть аналізуватись та обиратись кращий з представлених.

- Мутація вагів: вона стосується вагів зв'язків у нейронній мережі. Випадковим чином буде обиратись число від -1 до 1 та додаватись до значення вагів у випадковому з'єднанні. Мутація біасу (зміщення): мутація відбувається за тим же принципом, як і мутація вагів, але їй не підлягають ноди з шару входів.
- Модифікація функції активації: випадковим чином з дозволених функцій (Logistic, Tanh, Relu, Identity, Hlim, Softsign, Sinusoid, Gaussian, Softplus, Bent Identity) обирається нова, при цьому вона не співпадає зі старою функцією. Після чого в обраній ноді (крім входів) замінюється функція активації.
- Випадково: обирається одна з вищезазначених мутацій та використовується на організмі, який мутує.

Треба зазначити, що ймовірність мутації буде підлягати аналізу, тому цей параметр визначимо пізніше.

2.4. Аналіз підібраних параметрів

2.4.1. Кількість нод у прихованому шарі

Прихований шар – це група нод, що не є у входному та вихідному шарі. Він слугує для додаткових розрвхунків, та покращенню навчання. Але існують деякі нюанси: якщо нод буде забагато, то відбудеться перенавчання. З іншої сторони монети: якщо нод замало, то відбувається нестача ресурсів для вирішення поставленої задачі адже її складність може бути не такою уж і малою.

Розглянемо декілька випадків, коли кількість нод буде рівною 0, 5, 10, 15, 20, 25 відповідно.

При проведеннях модуляцій з різними параметрами можна помітити, що організми з малою кількістю прихованих нейронів мають меншу кількість обчислень, значить їх навчання протікає швидше вони мають найменшу кількість ваг у нейронній мережі, але при цьому якість цього навчання не є високою і має багато просадок.

На графіку 2 представлено порівняння моделювання коли кількість нод рівна 0-5-10. Як можна помітити найліпший результат показала модель з десятима нейронами у прихованому шарі.

На графіку 3 порівнюються моделювання з 15-20-25 нод у прихованому шарі. Кращою виявилась симуляція з 20-ма нейронами.

Тепер порівняємо обидва результати, та як видно на графіку 4, найоптимальнішим варіантом для моделі виявилось, що найліпший результат буде при 20-ти нодах у прихованому шарі.

2.4.2. Методи схрещування

На даному моменті ми будемо оцінювати пристосованість популяції в залежності від вибору методу

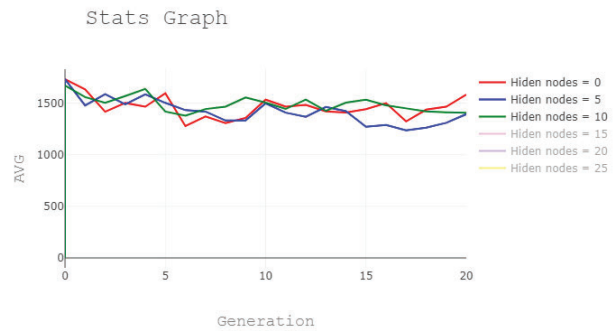


Рис. 2. Графік результатів симуляції для 0-5-10 прихованих нейронів

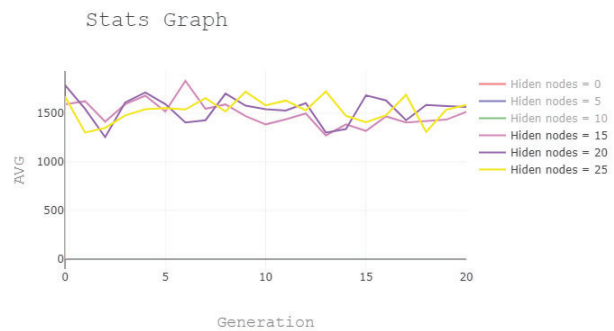


Рис. 3. Графік результатів симуляції для 15-20-25 прихованих нейронів

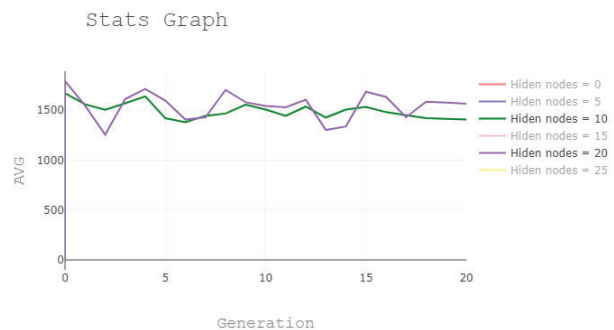


Рис. 4. Графік результатів симуляції для 10-20 прихованих нейронів

схрещування при створенні нового покоління. Проведемо 3 експерименти, в яких змодельуємо наступне:

- покоління отримуються лише завдяки одноточковому схрещуванню;
- покоління отримуються лише завдяки двоточковому схрещуванню;
- покоління отримуються лише завдяки випадковому вибору між одно-двоточковим схрещуванням.

На графіку 5 представлено результат роботи моделі. Як можна помітити, кращий результат отримано при симуляції з двоточковим схрещуванням.

Такий результат можна пояснити тим, що при такому методі схрещування популяція досягає найбільшого різноманіття, через це в нього є більше можливостей розвитку. Також треба відмітити, що навіть

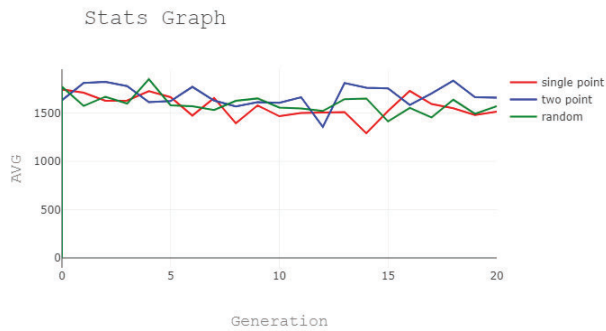


Рис. 5. Графік результатів симуляції методів схрещування

випадкового схрещування недостатньо враховуючи те, що в цьому випадку теж присутнє двоточкове схрещування.

2.4.3. Методи мутації

Для підбору кращого методу мутації, необхідно зробити чотири симуляції:

- При мутації змінюється лише значення вагів зв'язків нейронної мережі;
- При мутації змінюється лише значення біасу(зміщення) нейрону з всіх шарів окрім вхідних;
- При мутації змінюється лише значення функції активації нейрону з всіх шарів окрім вхідних;
- Випадковим чином обирається одна з вищезазначених мутацій.

Подивимось на результати вказаних симуляцій на графіку 6. Найкращий результат модель показала при мутації вагів, найгірший – мутація функції активації, що пояснюється необхідності приділити особливої уваги до підбору цієї функції для кожної моделі окремо. Також треба зазначити, що випадкова мутація дала дуже близькі результати до симуляції з мутацією вагів.

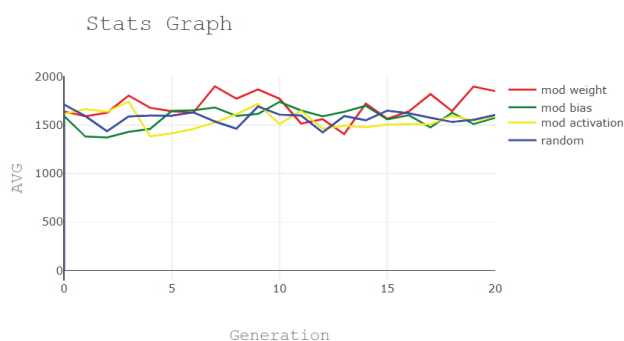


Рис. 6. Графік результатів симуляції методів мутації

Можна зробити висновок, що мутація вагів нейронної мережі найбільше впливають на пристосованість організму та подальшого успішного розвитку.

2.4.4. Частота мутації

Частота мутації є доволі вагомим параметром через те, що в залежності від нього, буде залежати

різноманітність покоління, яка, в свою чергу, може забезпечити кращий розвиток. Але слід відмітити, що надмірна кількість мутацій призведе до негативних наслідків, які будуть вповільнювати навчання.

Тож буде відбуватись 3 моделювання з різними параметрами частоти мутації, а саме: 0.1, 0.2 0.3 відповідно.

На графіку 7 видно, що при 0.3 модель демонструє найгірші результати, що запевняє в тому, що це дуже великий показник для цього параметру. При цьому моделі з 0.1 та 0.2 демонструють доволі близькі результати, але все ж судячи з порівняння середньої пристосованості цих ситуацій можна зробити висновок, що симуляція з 0.1 продемонструвала себе краще.

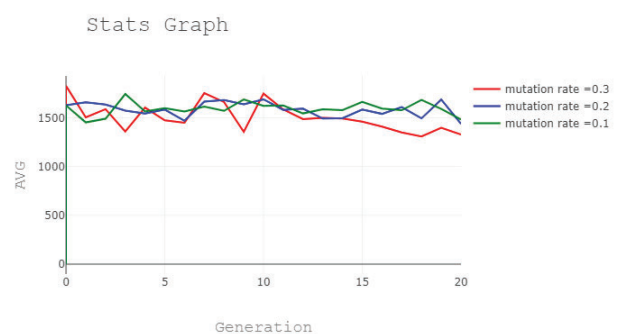


Рис. 7. Графік результатів симуляції частоти мутації

Висновки

В ході роботи була отримана модель розвитку простих, в якій нейронна мережа, яка відповідає за мозок організму, оптимізується з плинном генетичного алгоритму.

Для даної моделі було підібрано оптимальні методи схрещування(двоточкове) та мутації(мутація вагів), а також підібрані параметри для кількості нейронів у прихованому шарі (20) та частоти мутації (0.1).

Перелік використаних джерел

1. Голдберг Д. Е. Генетичні алгоритми оптимізації та машинне навчання.
2. Andrew N. Sloss, Gustafson Steven. Evolutionary Algorithms Review. — 2019. — Access mode: <https://arxiv.org/pdf/1906.08870.pdf>.
3. e Aymeric Vi', Alissa M. Kleinnijenhuis, Doyne J. Farmer. Qualities, challenges and future of genetic algorithms: a literature review. — 2021. — P. 5–15. — Access mode: <https://arxiv.org/pdf/2011.05277.pdf>.