УДК 004.89 http://doi.org/10/15407/pp2023.03.49

А.Є. Вітюк, А.Ю. Дорошенко

ПОШУК НОВИЗНИ У МЕТОДІ НЕЙРОЕВОЛЮЦІЇ ДЛЯ ПОЗИЦІОНУВАННЯ РОБОТИЗОВАНОЇ КІНЦІВКИ

У статті досліджено використання алгоритму нейроеволюції для пошуку політик у формі нейромереж у процесі створення контролера для управління роботизованою кінцівкою. Описано основні аспекти нейроеволюційного підходу для навчання нейромережі у задачах, що не мають найкращого рішення та потребують навчання з підкріпленням. Задача позиціонування має такий ландшафт функцій пристосованості, що робить її схильною потрапляти у пастки локального оптимуму.

Досліджено підхід підвищення ефективності пошуку політики у формі нейромережі для задачі позиціонування робочої кінцівки. Зокрема, розглянуто використання пошуку новизни для поліпшення нейроеволюційного процесу навчання за допомогою алгоритму NEAT. Цей підхід покращує здатність алгоритму знаходити інноваційні та ефективні рішення, сприяючи розвитку більш універсальних і потужних архітектур нейронних мереж.

Результати навчання політики управління для двовимірної роборуки з двома ланками у симуляції середовища показали, що використання пошуку новизни дозволяє отримати ефективну політику в формі нейромережі, яка має мінімальну конфігурацію. А це дозволить підвищити швидкодію контролера, яка є критичною для роботи реальної системи. Дослідження даного підходу для вирішення задачі позиціонування в системах більшої розмірності та складних роботизованих систем становлять особливий інтерес для покращення програмних контролерів управління роботизованими кінцівками в динамічних умовах реального світу.

Ключові слова: нейронна мережа, нейроеволюція, роботизована рука, навчання з підкріпленням, NEAT, пошук новизни, функція пристосованості.

Вступ

У сфері робототехніки дослідження інтелектуальних контролерів для управління робочими кінцівками є актуальним напрямком досліджень і розробок. Навіть враховуючи активну роботу провідних дослідницьких центрів по всьому світу, рівень точності та ефективності роботизованої руки все ще не може досягнути рівня людської руки. А з огляду на те, як промисловість продовжує використовувати автоматизацію, попит на роботизовані системи, які можуть виконувати складні та точні завдання, зріс. Від виробництва до охорони здоров'я, від логістики до аерокосмічної галузі, розгортання роботизованих систем для задач маніпулювання об'єктами стало повсюдним, революціонізуючи традиційні робочі процеси та відкриваючи нові можливості. У цьому контексті розробка інтелектуальних контролерів, особливо тих, що використовують нейроеволюцію, стала критично важливим фактором для оптимізації продуктивності та адаптивності роботизованих маніпуляторів у різноманітних середовищах.

Складність, притаманна сучасним роботам, вимагає контролерів, які виходять за межі традиційних методологій, заснованих на правилах. Інтелектуальні контролери, засновані на передових технологіях, таких як штучний інтелект, машинне навчання та нейронні мережі, обіцяють подолати розрив між можливостями роботизованої руки та вимогами складних сценаріїв реального світу. Серед цих методів нейроеволюція виділяється як потужний і актуальний підхід, який використовує принципи еволюції для створення інтелектуальних контролерів, адаптованих до задач роботизованої руки.

Дана стаття розглядає метод підвищення ефективності застосування нейроеволюції для пошуку політик у формі нейромереж, які можуть бути використані для різноманітних задач управління роботизованою кінцівкою, зокрема, позиціонування та маніпуляції цільовим об'єктом. Метою дослідження є підвищення точності нейромережі для позиціонування робочої кінцівки, яка може бути навчена з допомогою

©А.Є. Вітюк, А.Ю. Дорошенко, 2023

нейроеволюційного алгоритму підходами навчання з підкріпленням, які не вимагають збору тренувального набору даних.

Задача позиціонування має такий ландшафт функцій пристосованості, що робить її схильною потрапляти у пастки локального оптимуму. Вирішення даної проблеми становить особливий інтерес для подальшого покращення програмних контролерів управління роботизованими кінцівками.

1. Політики в формі нейромереж для задачі управління роботизованою кінцівкою

Більшість роботизованих завдань вимагають виконання багатокрокових цілей, для яких потрібен механізм планування. Традиційні роботизовані контролери зазвичай розробляються вручну, а розподіл завдань на підзадачі виконує людина-конструктор.

Оскільки завдання та робочі середовища для роботів ускладнюються, зростає потреба в методах навчання та пошуку, які можуть планувати вирішення задачі, не покладаючись на вже існуючу структуру підзадач або станів, розроблену людиною.

Методи навчання та еволюційного пошуку розроблені для оптимізації швидкої поведінки, яка потім використовується певним контролером вищого рівня. Нейроеволюція – це сімейство методів машинного навчання, які використовують еволюційні алгоритми для полегшення вирішення складних проблем, таких як ігри, робототехніка та моделювання природних процесів. Нейроеволюційні алгоритми імітують процес природного відбору. Кінцевим результатом нейроеволюції є оптимальна топологія мережі, яка робить модель більш ресурсоефективною та легшою для аналізу [4]

Для навчання з учителем потрібні набори ситуацій захвату та відповідної оптимальної поведінки. Оскільки оптимальна поведінка часто невідома, покрити типові ситуації – складна задача. Нейроеволюція – це метод навчання з підкріпленням, і він не потребує навчальних прикладів, де ідеальна поведінка захватного пристрою відома.

Нейроеволюція не покладається на повну інформацію про стан. Інші методи з підкріпленням часто припускають, що поточний стан системи повністю відомий. Однак, якщо конфігурація середовища динамічно змінюється, спостережувані значення реакцій не можуть бути пов’язані з правильним основним станом. Нейроеволюція вирішує проблему, розвиваючи повторювані зв’язки; повторення встановлює пам›ять, яка дає можливість розрізняти стани.

Класичною задачею для роботизованої кінцівки із захватним пристроєм є взаємодія з цільовим об’єктом, що є проблемою на стику робототехніки, комп’ютерного зору та штучного інтелекту. Стійкий захват передбачає складну координацію рухів роботизованої руки з метою точного позиціонування руки та маніпулювання об’єктами в навколишньому середовищі. Тобто цю задачу можна розділити на дві взаємозалежні частини: позиціонування руки та маніпулювання об’єктом.

Необхідність використання двох контролерів для цього завдання демонструє, що нейроеволюції легше розвинути реактивну поведінку, ніж довгострокову стратегічну поведінку.

Розглянемо детальніше першу підзадачу – позиціонування кінцівки роботизованої руки для досягнення цільового об’єкту.

Позиціонування руки означає точне розміщення кінцевого елемента роботизованої руки (наприклад, захвату або інструменту) у певному місці в середовищі. Це передбачає визначення відповідних кутів з’єднань і рухів, необхідних для досягнення бажаної мети.

Роботизована рука зазвичай складається з кількох шарнірів, які забезпечують руку ступенями свободи (DOF). DOF представляє кількість незалежних параметрів, необхідних для опису конфігурації руки. Кожен суглоб дозволяє руці обертатися або переміщатися вздовж певної осі, дозволяючи руці рухатися в кількох напрямках. Позиціонування роботизованої руки базується на використанні систем координат для визначення положення та орієнтації руки. Орієнтація руки може бути представлена за допомогою кутів Ейлера, кватерніонів або матриць обертання.

2. Алгортим NEAT

NEAT (NeuroEvolution of Augmenting Topologies) є потужним нейроеволюційним алгоритмом, призначеним для розвитку штучних нейронних мереж (ANN) зі складними та адаптивними структурами. Розроблений Кеннетом О. Стенлі та Рісто Мійкулайненом 2002 року, NEAT став відомим як інноваційний підхід до розвитку нейронних мереж для різних завдань із контролем, оптимізацією та розпізнаванням шаблонів включно.

NEAT був спеціально розроблений для вирішення проблем розвитку нейронних мереж із різними топологіями, що було обмеженням у традиційних підходах нейроеволюції. NEAT представляє механізми, які дозволяють нейронним мережам ускладнюватися з часом, дозволяючи їм вирішувати дедалі складніші завдання.

Розглянемо ключові особливості та концепції алгоритму NEAT.

- Генотип і фенотип: у NEAT індивідууми представлені як генотипом, так і фенотипом. Генотип кодує структуру нейронної мережі, включаючи вузли (нейрони) і зв'язки (синапси), тоді як фенотип представляє активну мережу з фактичними вагами, призначеними зв'язкам.
- Інновації та історичні маркування: NEAT використовує історичну схему маркування для відстеження походження генів (з'єднань і вузлів) протягом поколінь. Інновації в структурі мережі зберігаються. Це гарантує, що нові функції не будуть втрачені під час еволюційного процесу.
- Ускладнення та спрощення: NEAT дозволяє нейронним мережам розвиватися та ускладнюватись протягом багатьох поколінь. Нові вузли та з'єднання можна вводити поступово, починаючи від простих мереж і розвиваючись до більш складних структур. Крім того, NEAT підтримує видалення зайвих або непотрібних структур, сприяючи спрощенню мереж, коли це виправдано.
- Видоутворення: NEAT включає видоутворення, процес поділу популяції на види на основі подібності. Це запобігає домінуванню кількох особин і дозволяє зберегти різноманітність. Специфікація заохочує дослідження різних мережевих структур і запобігає втраті багатообіцяючих інновацій.
- Спільний доступ до пристосованості: NEAT використовує обмін пристосованістю серед видів. Геноми з подібною мережевою структурою діляться значеннями фітнес-функції, сприяючи співпраці всередині виду, а не інтенсивній конкуренції. Це дає інноваційним структурам шанс розвиватися та вдосконалюватися.
- Схрещування і мутація: NEAT використовує операції схрещування та мутації. Схрещування поєднує гени двох батьків для створення нащадків із сумішшю їхніх ознак. Мутація вносить невеликі зміни в гени, полегшуючи дослідження простору рішення.

Еволюційний цикл алгоритму NEAT передбачає створення нових поколінь нейронних мереж шляхом відбору, відтворення, кросинговеру та мутації. Придатність кожної окремої мережі оцінюється на основі її продуктивності в конкретному завданні. Протягом багатьох поколінь NEAT ітеративно розвиває мережі, які демонструють покращену продуктивність і підвищену складність, що приводить до рішень, які добре підходять для їхніх завдань.

Загалом, інноваційний підхід NEAT до розвитку нейронних мереж із різними топологіями зробив значний внесок у розвиток нейроеволюційних алгоритмів та їх застосування для оптимізації рішень, що базуються на використанні нейромереж.

3. Оптимізація пошуком новизни

Хоча методи прямої оптимізації функцій пристосованості добре працюють у багатьох простих задачах, у розв'язку більш складних задач вони часто потрапляють у пастку локального оптимуму.

Пошук новизни представляє собою підхід, який може бути інтегрований в алгоритм NEAT, щоб направляти еволюційний процес до знаходження оптимальних рішень там, де ландшафти функцій пристосованості не можуть бути успішно пройдені за допомогою процесу оптимізації, заснованого виключно на вимірюванні близькості поточного рішення до цілі. Пошук нови-

зни має на меті усунути обмеження традиційної еволюції на основі цільової функції, заохочуючи дослідження нових областей простору рішень на основі новизни рішень, а не лише їх значення цільової функції [1].

У звичайній еволюції на основі цільової функції особини в популяції зазвичай оцінюються на основі їхньої продуктивності у виконанні конкретного завдання, і еволюційний процес надає перевагу особам, які мають успіх у цьому завданні. Незважаючи на ефективність, цей підхід може призвести до збіжності до локальних оптимумів, де рішення можуть застрягнути в неоптимальних областях простору рішень і не вдасться дослідити альтернативні рішення.

Пошук новизни відкриває іншу перспективу. Замість того, щоб зосереджуватися виключно на функції пристосованості, він оцінює особини на основі їх новизни, яка кількісно визначає, наскільки поведінка або характеристики особини відрізняються від інших. Новизна визначається віддаленістю або несхожістю у поведінці чи просторі ознак.

Інтеграція пошуку новизни у алгоритм NEAT передбачає кілька ключових кроків:

1. Визначення поведінки або особливостей. Поведінка або особливості окремих осіб повинні бути визначені так, щоб їх можна було кількісно оцінити та порівняти. Для робототехнічних завдань це може включати визначення конкретних аспектів поведінки, таких як траєкторії, стани або сенсорні моделі.

2. Розрахунок новизни. Новизна особини визначається шляхом порівняння її поведінки чи особливостей з іншими індивідами в популяції. Міра відстані або несхожості, така як евклідова відстань або метрика подібності, використовується для обчислення показників новизни.

3. Архів. «Архів новинок» підтримується для зберігання різноманітного набору поведінки або функцій, які зустрічалися раніше. Архів є еталоном для розрахунку новизни і допомагає запобігти передчасній збіжності популяції.

4. Відбір. Замість того, щоб відбирати особини виключно на основі функції пристосованості, алгоритм NEAT із використанням пошуку новизни враховує оцінки функції пристосованості та новизни. Особини з високими показниками новизни отримують шанс вижити та розмножуватися, навіть якщо їхні показники функції пристосованості нижчі. Це спонукає до дослідження нових і недосліджених регіонів простору рішень.

Пошук новизни сприяє дослідженню, заохочуючи популяцію знаходити різноманітні рішення, які спочатку можуть бути пропущені традиційним відбором на основі функції пристосованості.

Крім того, завдяки пріоритетності новизни алгоритм має меншу ймовірність застрягти в локальних оптимумах. Це запобігає передчасному переходу до неоптимальних рішень популяції.

Різноманітність рішень, виявлених за допомогою пошуку новизни, може привести до більш надійних та адаптованих рішень, оскільки алгоритм розглядає більший простір пошуку.

Пошук новизни застосовувався в задачах, де дослідження різноманітних рішень має вирішальне значення. Включаючи новизну як критерій відбору, NEAT з пошуком новизни покращує здатність алгоритму знаходити інноваційні та ефективні рішення, сприяючи розвитку більш універсальних і потужних архітектур нейронних мереж.

4. Задачі позиціонування кінцівки робота у навчальному середовищі для пошуку політики у формі нейромережі

Для задачі навчання ми розглянемо модель роботизованої руки, представлену як інтелектуальний агент в обмеженому та спрощеному середовищі.

Агент – це суб’єкт, здатний сприймати своє оточення за допомогою датчиків, а також впливати на своє оточення за допомогою приводів. Агенти зазвичай можуть сприймати власні дії, але не обов›язково є вплив цих дій на навколишнє середовище. Агент може бути математично описаний функцією агента, яка визначає дію агента на основі всієї його послідовності сприйняття.

Тож, поведінка агента може бути повністю описана шляхом визначення дії для кожної можливої послідовності сприйняття [1]. Схема взаємодії агента і середовища представлена на рис. 1.

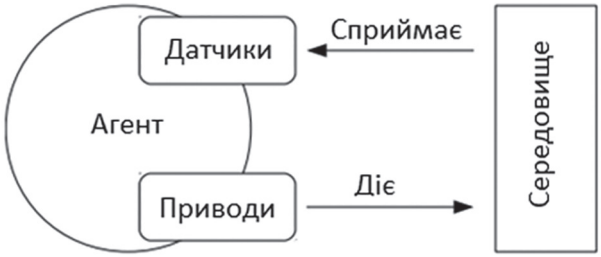

Рис. 1. Взаємодія компонентів навчального середовища

Розглянемо задачу позиціонування кінцівки роботизованої руки за допомогою середовища OpenAI Gym із двовимірною роботизованою рукою з двома суглобами. Це середовище моделювання призначене для забезпечення віртуальної платформи з метою тестування та розробки алгоритмів керування двовимірною роботизованою рукою [7]. Приклад стану навколишнього середовища представлений на рис. 2.

Робот складається з двох з'єднань, кожне довжиною 100 пікселів, і мета – досягти червоної крапки, яка випадково генерується в кожному епізоді. Простір дій представляє можливі дії, які може виконати агент. У двовимірному середовищі маніпулятора простір дій складається з керування кутами з'єднання маніпулятора робота:

- 0: залишити поточне значення кута з'єднання
- 1: приріст кута з'єднання 1
- 2: зменшення кута з'єднання 1
- 3: приріст кута з'єднання 2
- 4: зменшення кута з'єднання 2
- 5: приріст кутів з'єднань 1 і 2
- 6: зменшення кутів з'єднань 1 і 2

Простір спостережень дає інформацію про поточний стан навколишнього середовища. У середовищі двовимірної руки робота простір спостережень є безперервним і містить відповідну інформацію, необхідну для контролю та ухвалення рішень:

- цільова позиція в напрямку x, пікселі
- цільова позиція в напрямку y, пікселі
- поточне положення з'єднання 1, рад
- поточне положення з'єднання 2, рад.

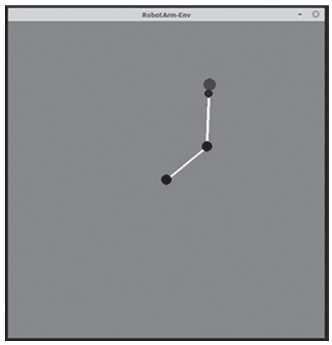

Рис. 2. Середовище OpenAI Gym 2D Robot Arm

Навколишнє середовище забезпечує винагороду для управління процесом навчання. Винагорода залежить від досягнення цільової позиції:

- робот отримає штраф -1, якщо поточна відстань між кінцівкою і цільовою позицією більша, ніж попередня відстань;
- робот отримає винагороду 1, якщо поточна відстань між кінцівкою і цільовою позицією > -Є < Є, де Є = 10 пікселів.

Крім того, визначено умови завершення для визначення кінця епізоду: поточна винагорода становить -10 або +10.

Під час експериментів із середовищем OpenAI Gym 2D Robot Arm було визначено недолік обмеженого кута обертання з'єднань, що суттєво обмежує простір пошуку, проте створює додаткові умови для демонстрації переваг нейроеволюційного підходу.

5. Розробка контролера для позиціонування робочої кінцівки симулятора

Для навчання політики у формі нейромережі для управління робочою кінцівкою в середовищі була обрана бібліотека NEAT-Python як реалізація алгоритму NEAT. Бібліотека NEAT-Python забезпечує реалізацію стандартних методів NEAT для моделювання генетичної еволюції геномів організмів у популяції. Вона містить утиліти для перетворення генотипу організму

в його фенотип (штучну нейронну мережу) і надає зручні методи для завантаження та збереження конфігурацій геному разом із параметрами NEAT. Крім того, вона надає корисні процедури, які допомагають збирати статистичні дані про хід еволюційного процесу та зберігати/завантажувати проміжні контрольні точки. Контрольні точки дозволяють нам періодично зберігати стан еволюційного процесу і пізніше відновлювати виконання процесу [1].

Для прискорення процесу навчання було використано розпаралелювання обчислень функції пристосованості між процесорами за допомогою класу ParallelEvaluator: *neat.ParallelEvaluator(mp.cpu_count(), config.eval_genome, timeout=20)*

У процесі нейроеволюції використовується двовимірний симулятор роботизованої руки для реалізації методу навчання методом спроб і помилок. Він підтримує популяцію геномів, які розвиваються з покоління в покоління, доки не буде знайдено успішний розв'язок. У процесі еволюції кожен організм у популяції перевіряється на придатність шляхом імітації руки з двома суглобами. Наприкінці симуляції організм отримує сигнал винагороди у вигляді кількості часових кроків, протягом яких він міг досягти цільової червоної точки та максимальної винагороди. Отриманий сигнал винагороди визначає працездатність організму і вирішує його долю в процесі нейроеволюції.

Спершу розглянемо реалізацію навчання з використанням звичайного алгоритму NEAT. Управління еволюційним процесом відбувається з використанням цілеорієнтованої цільової функції (goal-oriented objective function). Для розглянутого агента функція пристосованості набуває значення [-10, 10] залежно від успішності досягнення цільової позиції.

Досягнення цільової точки робочої кінцівки має локальні пастки найкращої пристосованості в областях, де значення функції буде коливатися у проміжку [-10, 10], проте глобального наближення до кінцевої позиції досягнуто не буде.

Вибір гіперпараметрів було здійснено з застосуванням широких меж пошуку, адже маємо великий обсяг керуючих сигналів для робочої кінцівки. Початкова конфігурація фенотипу нейромережі включає 4 вхідних вузли, 7 вихідних вузлів та 1 прихований вузол. Вузли входу відповідають параметрам, що описують поточний стан системи, а вузли виходу відповідають керуючим сигналам. Прихований вузол призначений для введення нелінійності з самого початку еволюційного процесу. Структура нейромережі представлена на рисунку 3:

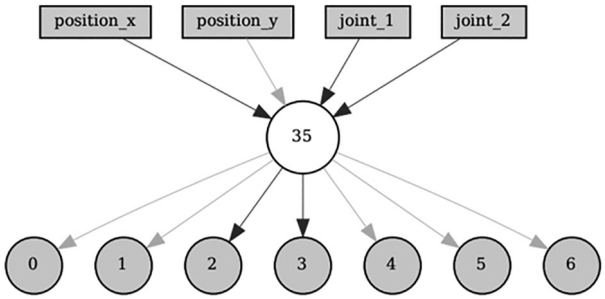

Рис. 3. Структура початкової нейромережі

Для розширення області пошуку необхідно покрити велику видову різноманітність популяції, щоб спробувати різні конфігурації геному протягом обмеженої кількості поколінь. Цього можна досягти або зниженням порогу сумісності, або збільшенням значень коефіцієнтів, що використовуються для розрахунку показників сумісності геному. Використані наступні значення параметрів:

[NEAT]
compatibility_disjoint_coefficient = 1.1
[DefaultSpeciesSet]
compatibility_threshold = 3.0

Оскільки початкова структура нейромережі має досить велику кількість виходів на невелику кількість входів, існує ймовірність перенавчання мережі, коли певні вхідні вузли не використовуватимуться і це буде приводити до субоптимального рішення, яке знаходиться в локальному мінімумі. Водночас на меті є створення оптимальної конфігурації нейромережі, яка матиме мінімальну кількість прихованих вузлів та зв'язків. Тому вірогідності додавання та видалення вузлів мають бути невеликими. Причому враховуючи початкову повно-

зв'язну модель, вірогідність видалення має бути додатково зменшена:

[DefaultGenome]
conn_add_prob = 0.3
conn_delete_prob = 0.2

За результатами експерименту навіть після 1000 поколінь еволюції успішний контролер не був знайдений. Графік видоутворення показав, що процес зійшовся до одного виду, який далі не зміг отримати ефективне рішення. Тому поріг стагнації було зменшено, а кількість видів збільшено, щоб еволюційний процес підтримувався за рахунок видів, які не показали покращення значення пристосованості і підлягають вимиранню:

[DefaultStagnation]
species_fitness_func = max
max_stagnation = 20
species_elitism = 1
[DefaultReproduction]
survival_threshold = 0.2
min_species_size = 2

Найкращий геном, отриманий з використанням алгоритму нейроеволюції, кодує непрактичну конфігурацію нейроконтролера, яка враховує лише входи 3 та 4, що відповідають за переміщення другої ланки. Такий результат показує, що алгоритм потрапив у пастку локально оптимального рішення, коли поворот ланки, найближчої до цільової точки, може дати локальний приріст цільової функції. Причому перша ланка не була використана, хоча її поворот дозволить наблизитися до кінцевої точки.

Найкраща пристосованість організму, яка була отримана, становить всього 2, що недостатньо порівняно із цільовим значенням 10 (критерієм закінчення). Структура нейромережі, що закодована цим організмом, представлена на рисунку 4.

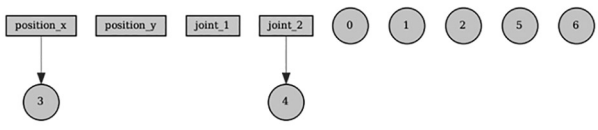

Рис. 4. Структура нейромережі, що закодована в найкращому геномі з використанням NEAT без пошуку новизни

На графі видно, що повороти другого з'єднання залежать лише від положення по одній осі та його поточного положення. Така структура призводить до спрощених рухів, декілька з яких направляють кінцівку у напрямку цільової точки, але вони є недостатніми для її досягнення.

Графік видоутворення, представлений на рисунку, показує, як протягом поколінь популяції організмів розвивався процес видоутворення. Аналіз середніх показників пристосованості показав, що нейроеволюційний процес покращував показники пристосованості на перших поколіннях, проте з часом він зупинився на плато, не показуючи покращень.

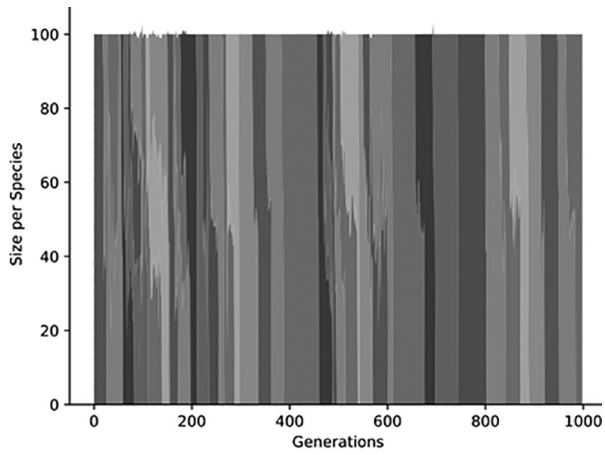

Рис. 5. Графік видоутворення еволюційного процесу з використанням NEAT без пошуку новизни

Отже, збільшення кількості поколінь не буде рішенням, необхідно використовувати механізми для покращення характеристик нейроеволюційного процесу. Тому введемо використання методу пошуку новизни, описаного раніше. Запуск здійснимо з тими ж гіперпараметрами та на тій же кількості поколінь. Після 1000 поколінь еволюції найкращий агент-контролер досяг значення функції пристосованості 6, що значно перевищує попередній результат. Крім того, аналіз конфігурації нейромережі найкращої особини (представлена на рисунку 6) показує, що дана мережа надійніша, адже має зв'язки від усіх вхідних вузлів.

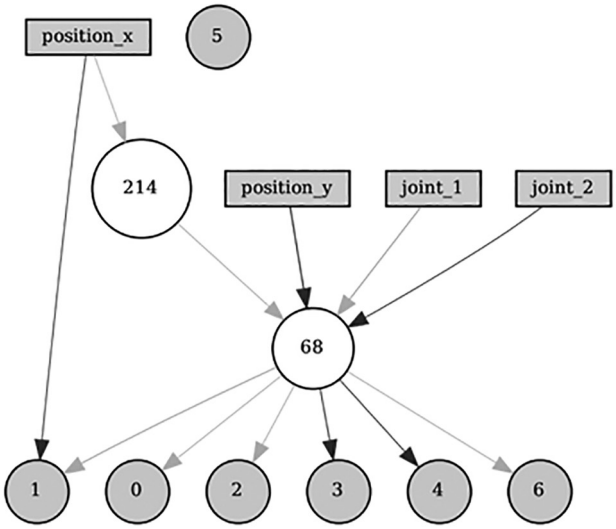

Рис. 6. Структура нейромережі,
що закодована в найкращому геномі
з використанням NEAT
з пошуком новизни

Збільшення кількості поколінь дозволить мережі дорозвинути переможця для отримання ефективного контролеру з мінімальною конфігурацією та керуванням усіма доступними управляючими сигналами. Зокрема, одночасним приростом кутів з'єднань 1 і 2 (вихідний вузол 5), який не має розвинутих зв'язків у отриманій конфігурації з найкращого геному.

Висновки

Вирішення задачі здійснення захватів роботизованою кінцівкою вимагає надійних стратегій керування, що мають враховувати варіативність умов середовища. Позиціонування робочої кінцівки має значний простір пошуку, тож класичні методи програмування контролерів дають ефективні рішення лише для чітко визначених обмежених виробничих умов. А в умовах довільного середовища мають бути використані контролери, що використовують знання, накопичені під час взаємодії з цим середовищем. Інтеграція таких методів штучного інтелекту, як машинне навчання та навчання з підкріпленням, сприяла значному прогресу у вирішенні складнощів позиціонування рук і маніпулювання об'єктами.

У статті досліджено підхід до покращення ефективності навчання політики управління роботизованою кінцівкою у формі нейромережі з допомогою використання пошуку новизни для нейроеволюційного алгоритму NEAT. Задача позиціонування має такий ландшафт функцій пристосованості, що робить її схильною потрапляти у пастки локального оптимуму. Експерименти з моделлю двовимірної роборуки з двома ланками показали, що використання пошуку новизни дозволяє отримати ефективну політику в формі нейромережі. Адже NEAT з пошуком новизни покращує здатність алгоритму знаходити інноваційні та ефективні рішення, сприяючи розвитку більш універсальних і потужних архітектур нейронних мереж.

Розробка оптимальних політик управління із застосуванням нейроеволюційного підходу може забезпечити не лише високу точність позиціонування, а й мінімальну конфігурацію. А це підвищує швидкодію системи, яка є важливою характеристикою для роботизованої системи. Майбутні дослідження даного підходу для вирішення задачі позиціонування в системах більшої розмірності та складних роботизованих систем становлять особливий інтерес для покращення програмних контролерів управління роботизованими кінцівками в динамічних умовах реального світу.

Література:

1. I. Omelianenko, Hands-On Neuroevolution with Python: Build high-performing artificial neural network architectures using neuroevolution-based algorithms, Packt Publishing, 2019
2. R. Mahjourian, R. Miikkulainen, Neuroevolutionary Planning for Robotic Control, Department of Computer Science The University of Texas at Austin Austin, 2018
3. Kenneth O. Stanley and Risto Miikkulainen, Evolving Neural Networks Through Augmenting Topologies, Evolutionary Computation 10 (2): 99-127, 2002.
4. Застосування засобів нейроеволюції в технічних системах автоматизації керування / А.Ю. Дорошенко, І.З. Ашур

// Проблеми програмування. – 2021. – № 1. – С. 16-25
5. Альона Вітюк, Анатолій Дорошенко. Програмний пакет для оцінки похибка калібрування стереокамери в системі комп'ютерного зору. Проблеми програмування, 2022. № 3-4. С. 469–477. https://doi.org/10.15407/pp2022.03-04.469
6. M. W¨urtinger, Neuroevolution for Robot Control, Test Framework and Experimental Evaluation, Institut f¨ur Informatik Lehrstuhl f¨ur Programmierung und Softwaretechnik, 2011. URL: https://www.pst.ifi.lmu.de/Lehre/Abschlussarbeiten/vorlagen/thesis-wuertinger_2011-12-19.pdf
7. OpenAI Gym 2D Robot Arm Environment, URL: https://github.com/ekorudiawan/gym-robot-arm
8. Huang, Pei-Chi, Sentis et al., Tradeoffs in Neuroevolutionary Learning-Based Real-Time Robotic Task Design in the Imprecise Computation Framework. ACM Transactions on Cyber-Physical Systems. 3. 1-29. 10.1145/3178903, 2019

Одержано: 30.08.2023

Про авторів:

Вітюк Альона Євгеніївна,
аспірант НТУУ «КПІ імені Ігоря Сікорського».
Кількість наукових публікацій в українських виданнях – 4.
https://orcid.org/0000-0002-1445-9598

Дорошенко Анатолій Юхимович,
доктор фізико-математичних наук, професор, завідувач відділу теорії комп›ютерних обчислень Інституту програмних систем НАН України, професор кафедри автоматики і управління в технічних системах НТУУ «КПІ імені Ігоря Сікорського».
Кількість наукових публікацій в українських виданнях – понад 200.
Кількість наукових публікацій в іноземних виданнях – понад 80.
Індекс Гірша – 6.
http://orcid.org/0000-0002-8435-1451,

Місце роботи авторів:

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», проспект Перемоги 37
та Інститут програмних систем НАН України, 03187, м. Київ-187, проспект Академіка Глушкова, 40.
e-mail: alyonavityuk@gmail.com, doroshenkoanatoliy2@gmail.com