

Горобець Сергій,
кандидат педагогічних наук, доцент,
доцент кафедри комп'ютерних наук та інформаційних технологій,
Сидоров Данило,
здобувач другого (магістерського) рівня вищої освіти,
фізико-математичного факультету,
Житомирський державний університет імені Івана Франка,
м. Житомир, Україна

ОГЛЯД АЛГОРИТМІВ ПОШУКУ ТРАЄКТОРІЇ РУХУ

Постановка проблеми. Розвиток нових цифрових технологій поступово створює попит на складні алгоритми пошуку шляхів, які відіграють ключову роль у розробці та функціонуванні інформаційних систем та знаходять своє застосування в різних сферах – логістиці, навігації, робототехніці, іграх. Проте до цього часу актуальним питанням залишається досягнення рівноваги між ефективністю й точністю, тому не існує одного або декілька універсальних алгоритмів.

Аналіз актуальних досліджень. Розробці та удосконаленню алгоритмів для пошуку оптимальних маршрутів у різних типах просторів – як структурованих, так і неструктурованих, присвятили свої дослідження багато вітчизняних та зарубіжних науковців (Остапенко Л., Харченко В., Сіклічук А., Сенік І., Choset Н., Lynch К., Xin Р., Wang Х., Yang L., Song D. та ін.).

Метою статті є огляд найбільш популярних алгоритмів пошуку шляхів, можливостей їх застосування та особливостей використання у різних умовах.

Виклад основного матеріалу. Класичним алгоритмом, відомим своєю здатністю знаходити найкоротший шлях, перебираючи всі можливі маршрути у визначеній мережі, є алгоритм Дейкстри. Попри свою значну обчислювальну складність, він широко використовується у сферах, де точність важливіша за швидкість. Зокрема, у географічних інформаційних системах, де його комплексний підхід гарантує знаходження оптимального маршруту. Однак характер даного алгоритму робить його непрактичним для сценаріїв у режимі реального часу, особливо у великих мережах [1].

На підході Дейкстри ґрунтується алгоритм A^* , який застосовує евристику для визначення найкращих шляхів. Даний алгоритм оцінює відстань до цілі та забезпечує більш ефективну маршрутизацію. Саме тому він користується популярністю при розробці відеоігор, де неігрові персонажі (NPC) повинні

Секція 4. Технології розробки інформаційних систем

швидко адаптуватися до змін у навколишньому середовищі. Завдяки використанню евристичної функції забезпечується баланс між точністю та обчислювальною ефективністю. Це робить A* придатним як для роботи в реальному часі, так і для застосування у масштабних застосунках [1].

Алгоритм A* широко використовують у логістиці для оптимізації маршрутів доставки. За рахунок цих алгоритмів логістичні компанії можуть мінімізувати час доставки, зменшити споживання пального та швидко реагувати на дорожні умови. Amazon та UPS використовують ці системи для управління мережею з тисячами доставок. Вони враховують схеми руху транспорту, стан доріг та оновлення в режимі реального часу для забезпечення ефективної та гнучкої навігації.

Метод RRT (Rapid-exploring Random Trees) призначений для багатовимірних просторів і особливо корисний у робототехніці. Він дозволяє автономним дронам і роботам орієнтуватися в складних середовищах з перешкодами. Ця особливість вкрай важлива в таких умовах, як міські ландшафти, де безпека та ефективність є найбільшим пріоритетом. Саме RRT добре підходить для середовищ з численними та непередбачуваними змінними. Фокус алгоритму на швидкому аналізі допомагає знаходити можливі шляхи, навіть якщо оптимальний шлях не є очевидним [2].

Наприклад, дрони, що використовують RRT, можуть орієнтуватися в небезпечних умовах, таких як завали будівель або зони затоплення і допомагати в пошуково-рятувальних операціях. Здатність цих дронів швидко змінювати маршрут в обхід перешкод дає рятувальним командам можливість безпечно і ефективно охоплювати більшу територію. Під час катастроф велике значення має час, тому здатність алгоритмів швидко створювати доцільні маршрути може стати вирішальним для порятунку потерпілих.

Алгоритм Ієрархії скорочення (Contraction Hierarchies) часто використовується у великих транспортних мережах. Алгоритм значно скорочує час розрахунку маршрутів за допомогою попередньої обробки даних та створення коротких шляхів. Цей метод особливо ефективний для застосунків, що працюють з великими масивами даних, де критично важливі швидкі обчислення. Попереднє опрацювання ефективно «вирівнює» мережу, дозволяючи швидше відповідати на численні запити. Це корисно для систем, таких як громадський транспорт і сервіси спільного використання поїздок [3].

В галузі штучного інтелекту все більш важливим стає навчання з використанням «підкріплення». Цей тип машинного навчання допомагає системам вчитися на минулому досвіді, що може бути дуже корисним у галузі логістики та робототехніки. Навчання з підкріпленням допомагає системам пошуку шляхів адаптуватися до нових умов і вдосконалюватися протягом тривалого часу. Це підвищує їх ефективність і здатність реагувати на складні, динамічні сценарії. В свою чергу, адаптивні алгоритми допомагають автономним транспортним засобам, дронам і роботам швидко реагувати на непередбачувані зміни. Вони застосовуються в таких галузях, як сільське господарство, реагування в надзвичайних ситуаціях та у «розумних» містах. Ці алгоритми

Секція 4. Технології розробки інформаційних систем

допомагають системам вивчати оптимальні шляхи на основі мінливих умов та зменшують потребу в постійному втручанні людини.

Не менш значну роль відіграє пошук шляху для навігації в реальному світі. Такі додатки, як Google Maps, використовують ці алгоритми для створення оптимальних маршрутів на основі даних у реальному часі. Користувачі можуть підлаштовуватися під затори на дорогах, їх перекриття та будь-які інші перешкоди. Поєднання з даними в реальному часі гарантує, що ці програми надають актуальні і точні вказівки щодо маршрутів.

Одним з найпривабливіших застосувань технології пошуку шляхів є автономні транспортні засоби. Компанії, зокрема Tesla і Waymo, вкладають значні ресурси в інтеграцію машинного навчання з алгоритмами RRT і A^* , щоб забезпечити можливість автомобілям швидко реагувати на дорожні умови, передбачати дії інших водіїв і підлаштовуватися до них. Особливо виділяється підхід Tesla завдяки системі на основі комп'ютерного зору, яка використовує кілька камер, радари та нейронні мережі для інтерпретації дорожньої ситуації. Застосування глибокого навчання дозволяє автомобілям Tesla постійно вдосконалювати свою здатність орієнтуватися в складних дорожніх умовах і ефективно реагувати на несподівані перешкоди. Для навчання нейронних мереж система FSD («повне самостійне керування») використовує величезні обсяги відеоданих з автопарку Tesla [4].

Незважаючи на значний прогрес у сфері алгоритмів пошуку шляхів, вони досі мають певні проблеми з точки зору обчислювальної складності та адаптивності. Зі збільшенням обсягів даних потреба в обчисленнях у реальному часі створює значне навантаження на системи, особливо ті, що мають обмежені обчислювальні можливості (зокрема мобільні додатки). Баланс між швидкістю і точністю є ще однією серйозною проблемою – хоча алгоритми на зразок Дейкстри і забезпечують точні результати, вони можуть бути занадто повільними для застосування у реальному часі. Так само алгоритм A^* може потребувати оптимізації для забезпечення ефективності без зниження точності. З ростом кількості застосунків, які використовують особисті дані, також необхідно враховувати питання безпеки та конфіденційності. Щоб захистити дані користувачів, розробники повинні забезпечити захист цих алгоритмів від кібератак.

Висновки та перспективи подальших досліджень. Алгоритми пошуку шляхів мають широкий спектр застосувань. Проте для того, щоб відповідати вимогам програм, які працюють в режимі реального часу, необхідно удосконалювати алгоритми у напрямку збільшення швидкості обчислень, кращої гнучкості та підвищення заходів безпеки. Варто приділити увагу можливості розробки гібридних алгоритмів, які будуть поєднувати ефективність та точність.

Список використаних джерел та літератури

1. Дейкстра проти A^* – пошук шляху. URL: <https://www.baeldung.com/cs/dijkstra-vs-a-pathfinding>
2. Порівняльне дослідження алгоритмів A^* , RRT та RRT* для планування

Секція 4. Технології розробки інформаційних систем

маршрутів у 2D просторі. URL: <https://shorturl.at/k1GHS>

3. Ієрархії скорочень: Ілюстрований посібник. URL: <https://jlazarsfeld.github.io/ch.150.project/>

4. Як Tesla використовує та вдосконалює свій штучний інтелект для автономного водіння. URL: <https://www.aiwire.net/2023/03/08/how-tesla-uses-and-improves-its-ai-for-autonomous-driving/>