

також інтеграція штучного інтелекту для покращення поведінки НІП, що забезпечить гравцям більш захоплюючий досвід.

Перспективи розвитку також включають в себе розробку адаптивної системи штучного інтелекту для створення більш динамічного геймплею, впровадження машинного навчання для прогнозування поведінки гравців і оптимізації стратегій НІП, використання технологій хмарних обчислень для збереження та обробки великих обсягів даних ігрового процесу та результати дослідження підтверджують можливість створення сучасного ігрового проекту з високою інтерактивністю та адаптивністю, що буде цікавим для широкого кола гравців.

## **АНАЛІЗ АЛГОРИТМІВ ЗНАХОДЖЕННЯ ШЛЯХУ**

М.А. Печенов, В.В.Поліщук, к.т.н., доц. М.М. Колендовська, ХНУРЕ, м. Харків

Сучасні методи керування наземним мобільних роботом є досить складними, оскільки потрібно вирішити задачі збору та обробки даних візуального радіоелектронного контролю стану мобільних наземних роботів, а також необхідності визначення власної просторової орієнтації робота; контролю органів управління та траєкторії руху мобільного роботу, а вірогідність зустрічі з хаотично розповсюдженими перешкодами також є найвищою у порівнянні з іншими класами роботів.

Таким чином, спираючись на вищесказане, можна стверджувати, що розробка і дослідження нових методів збору та обробки даних різного радіоелектронного контролю стану мобільних наземних роботів є складним та важливим теоретичним і практичним завданням, що потребує одночасного вирішення кількох суттєвих аспектів: необхідності визначення власної просторової орієнтації робота; контролю органів управління; контролю фіксованої траєкторії руху мобільного роботу; створення засобів виявлення та розпізнавання перешкод.

Метою даної роботи є удосконалення існуючих та розробка нових методів та алгоритмів збору та обробки даних радіоелектронного контролю стану мобільних наземних роботів. Проведемо Аналіз алгоритмів знаходження шляху

Найпоширеніший спосіб визначення проблеми планування шляху (планування руху) полягає в тому, що агенту необхідно рухатися від початкової позиції до мети, уникаючи перешкод і досягаючи мінімізації витрат. Визначення вартості пошуку оптимального шляху змінюється залежно від критеріїв системи, в одних системах це час, в інших відстань, енергія тощо. Але оскільки найбільш загальний підхід полягає в мінімізації відстані (найкоротшого шляху) між початком і метою. Однак у деяких ситуаціях визначення оптимальності змінюється. Як приклад, для робота має значення час, витрачений на обчислення (кількість ітерацій і складність). Отже, коли

обчислення шляху займає надто багато часу, важче досягти безперервності завдання. Така причина змушує вибрати найбільш підходящий алгоритм на основі бажаного критерію оптимальності. У деяких випадках краще використовувати більш складні критерії – комбінувати критерії. Тому планування руху є одним із ключових завдань у робототехніці.

Під час завдання навігації можна виділити три основні гілки:

1. Завдання «Планування шляху», коли роботам надається цільова точка та/або точки інтересу, а карта навколишньої території перетворюється на режим чотиридерев за допомогою трансформації відстані відповідно до бажаної роздільної здатності. Після цього обчислюється шлях від початкової позиції через цікаві точки до мети.

2. Виконання шляху – це завдання, під час якого робот починає рухатися від початкової позиції до мети, використовуючи шлях, розрахований під час планування шляху.

3. Оновлення моделі запускається, коли робот виявляє будь-яку перешкоду в полі зору (FOV), а потім оновлює модель квадродрева оточення.

Аналіз літератури показав, що на даний час є добре розроблені математичні алгоритми для знаходження шляху в невідомому або частково відомому середовищі (оптимальний і евристичний алгоритми). Для цього зазвичай використовують дискретну математику (теорію графів) і лінійне програмування. Задачі пошуку найкоротшого шляху в графі відомі та досліджені (наприклад, алгоритми Дейкстри, Флойда-Воршелла, Прима, Крускала тощо).

Алгоритм Дейкстри— це алгоритм пошуку найкоротших шляхів між вузлами в графі, який може представляти, наприклад, дорожні мережі. Він був задуманий комп'ютерним науковцем Едгером В. Дейкстрою в 1956 році та опублікований через три роки. Алгоритм існує в багатьох варіантах. Оригінальний алгоритм Дейкстри знайшов найкоротший шлях між двома даними вузлами, але більш поширений варіант фіксує один вузол як «вихідний» вузол і знаходить найкоротші шляхи від джерела до всіх інших вузлів у графі, створюючи дерево найкоротших шляхів.

Флойд-Варшалл в інформатиці — це алгоритм для знаходження найкоротших шляхів у зваженому графі з додатними або від'ємними вагами ребер (але без від'ємних циклів). Одне виконання алгоритму знайде довжини (сумовані ваги) найкоротших шляхів між усіма парами вершин. Хоча він не повертає деталей самих шляхів, можна реконструювати шляхи за допомогою простих змін до алгоритму. Версії алгоритму також можна використовувати для знаходження транзитивного замикання відношення R або (у зв'язку з системою голосування Шульце) найширших шляхів між усіма парами вершин у зваженому графі.

A\* або A-star або A\* search є одним із добре відомих і базових евристичних алгоритмів. Це комбінація алгоритму Дейкстри. Алгоритм намагається мінімізувати функцію, сформульовану як  $f(n) = g(n) + h(n)$ ,

евристично враховуючи зв'язок між вузлами та ребрами.  $g(n)$  відноситься до вартості початкової точки або вузла, а  $h(n)$  передбачає евристичну оцінку вартості, пов'язану з залишком шляху.  $h(n)$  таким чином складають евристичну основу алгоритму.

Генетичний алгоритм (GA) є іншим популярним евристичним підходом. Цей метод імітує еволюційний процес і намагається отримати найкращих особин (хромосому), які представляють оптимальний шлях для цього типу проблеми з огляду на визначену цільову функцію. Стратегія кросинговеру застосовується до батьків з метою створення нових особин. На додаток до цього, процес мутації, який перешкоджає сходженню алгоритму на локальних мінімумах, є однією з ключових частин, так що різноманітність простору рішень завжди зберігається. GA та подібні евристичні методи є альтернативним способом вирішення задачі оптимального шляху, і вони часто використовуються.

Швидке дослідження випадкового дерева (RRT) — ще один ймовірнісний алгоритм, вдосконалений Лаваллем і Куффнером. Він спрямований на вирішення задач планування траєкторії, які мають неголономні обмеження. Алгоритм, як правило, дає ефективні результати в неопуклих і багатовимірних просторах. Дерево, яке передбачає простір рішень, будується поступово, і в усій схемі використовуються дві різні функції, тобто створення та розширення дерева. Розширення дерева є одностороннім, тобто від початку до кінцевої точки. З іншого боку, ідея двостороннього розширення дерева покращує часову ефективність усього рішення. Ця техніка називається двонаправленим RRT (bRRT або B-RRT), і дерево встановлюється як у початковій, так і в цільовій точках, а потім розширюється до частин, які не враховуються. Процес пошуку припиняється, коли знайдено точку з'єднання цих двох дерев і цей шлях, який є виходом алгоритму, є оптимальним маршрутом.

Ймовірнісна дорожня карта (PRM) — це один із ймовірнісних алгоритмів планування останнього шляху, який використовується в цьому дослідженні. Оптимальний шлях визначається обчисленням відстані між ребрами, які є зв'язками між випадково створеними вузлами на карті. Перш за все, вузли генеруються шляхом вибірки з точок без перешкод на карті, а потім ці вузли зв'язуються один з одним, і, нарешті, оцінюється вартість шляху. Стратегія швидкого випадкового кластера може забезпечити швидші результати, ніж інші алгоритми, але не гарантує найкоротший шлях.

Сімейство алгоритмів помилок вирішує проблему навігації, зберігаючи лише мінімальну кількість маршрутних точок, але не створюючи повної карти навколишнього середовища. Якщо шлях вирішення не існує, алгоритм здатний розпізнати цю ситуацію та припиняє повідомляти про те, що ціль недосяжна, замість того, щоб нескінченно блукати.

Модель Bug робить три припущення щодо робота. По-перше, робот є точковим об'єктом. По-друге, робот має ідеальну здатність до локалізації. По-третє, у робота ідеальні датчики. Ці три припущення нереалістичні для

реальних роботів, і тому алгоритми помилок зазвичай не застосовуються безпосередньо для практичних навігаційних завдань, але можуть розглядатися як контрольний компонент вищого рівня системи, яка включає всі три припущення. Алгоритми помилок можна розглядати як перший логічний крок до вирішення роботизованої двовимірної навігаційної задачі

Загалом ці алгоритми вирішують завдання планування руху самостійного робота в групі. Очевидно, що обмін даними між  $n$  роботами в групі є хорошим інструментом для отримання додаткової інформації. Це може служити для більш ефективної реалізації всіх перерахованих вище методів. Основна ідея полягає в тому, щоб якнайшвидше дати кожному окремому роботу в групі більше знань про сектор.

### **СТВОРЕННЯ ІНТЕРНЕТ-МАГАЗИНУ ГІТАР З ВИКОРИСТАННЯМ ПРОГРАМНОЇ ПЛАТФОРМИ REACT ТА РЕАЛІЗАЦІЄЮ 3D КОНСТРУКТОРА**

О. М. Пивошенко, НТУ «ХПІ», м. Харків

Інтернет магазин з продажу гітар, який має технологію конструктора для створення та потім замовлення створеної гітари, буде дуже актуальним серед професійних музикантів. Онлайн-конструктор відкриває можливість обрати індивідуальні налаштування, які можуть бути недоступними у звичайних магазинах, а також дає можливість експериментувати без необхідності фізичного перебування в магазині.

На даному етапі розроблено сайт для продажу гітар. Сайт дозволяє не тільки замовити ту гітару, яку ви бажаєте, але і прослухати її звучання у вигляді запису прямо на сайті. Запис виконано без сторонніх ефектів та аудіо-фільтрів. Це зроблено для того, щоб музиканти розуміли, як буде звучати їх музичний інструмент у запису. Із проблем, які поки не мають рішення, є саме 3D конструктор. При дослідженні було виявлено, що 3D об'єкти на сайті займають багато ресурсів та роблять роботу сайту набагато повільніше. Для вирішення цієї проблеми необхідний подальший аналіз можливих джерел ресурси, за допомогою яких можна буде розробити 3D конструктор, який би не навантажував роботу інтернет-магазину. В якості можливого вирішення проблеми розглядається застосування веб-технологій та хмарних мікросервісів.

Ідея дослідження полягає в розробці власного інтернет-магазину з продажу гітар, використовуючи програмну бібліотеку React, та реалізації 3D конструктора, за допомогою якого можна буде зручно замовити гітару. У ході дослідження важливо досягти ефективного результату у роботі створеного сайту, досягти грані оптимізації 3D об'єктів.

Під час дослідження було розглянуто приклади інтернет-магазинів для продажу гітар. Серед безлічі бібліотек та фреймворків було обрано React, що є