

АНАЛІЗ АЛГОРИТМІВ ПЛАНУВАННЯ ШЛЯХУ МОБІЛЬНОГО РОБОТА

Ільєнков Г.О.

Харківський національний університет радіоелектроніки

Україна, 61166, Харків, пр. Науки 14

E-mail: hlib.ilienkov@nure.ua

В роботі розглянуто актуальне питання автоматизації процесу пошуку оптимального шляху роботизованої платформи. Проведено аналіз існуючих алгоритмів для пошуку шляху, їх актуальності та особливостей застосування, також проаналізовано переваги та недоліки цих алгоритмів.

Ключові слова: автоматизація, пошук шляху, роботизована платформа, алгоритм.

ANALYSIS OF MOBILE ROBOT PATH PLANNING ALGORITHMS

Ilienkov H.O.

Kharkiv National University of Radio Electronics

Ukraine, 61166, Kharkiv, Nauky ave. 14

E-mail: hlib.ilienkov@nure.ua

The paper considers the current issue of automating the process of finding the optimal path of a robotic platform. An analysis of existing algorithms for path finding, their relevance and application features, and an analysis of the advantages and disadvantages of these algorithms are also presented.

Keywords: automation, pathfinding, robotic platform, algorithm.

В даний момент на різних підприємствах зростає кількість роботизованих платформ відповідно до цього зростає необхідність у різних системах руху для цих платформ. Автоматизація цього процесу дозволяє знизити вплив людського фактора: коли роботизована платформа має можливість самостійно орієнтуватися у просторі і оператору не потрібно його постійно направляти, це дозволяє зменшити навантаження на людину а також це дозволяє економити ресурси та час [1].

Автономна навігація є цінним активом для мобільних роботів. Вона допомагає пом'якшити їхню залежність від втручання людини. Однак вона також спричиняє безліч завдань або проблем для вирішення, наприклад, планування шляху.

Для ефективної побудови маршруту роботизованої платформи потрібно дослідити методи та алгоритми автоматизації побудови шляху. Для цієї задачі алгоритми були класифіковані таким чином [2]:

1. Алгоритми пошуку на основі сітки, які знаходять шлях на основі мінімальної вартості переміщення на карті сітки. Їх можна використовувати для таких програм як мобільні роботи в 2D-середовищі. Однак вимоги до пам'яті реалізації алгоритмів на основі сітки можуть збільшуватися з числом вимірювань.

2. Алгоритм пошуку на основі вибірки які створюють дерево пошуку шляхом випадкової вибірки нових вузлів або конфігурації робота у просторі станів. Алгоритми на основі вибірки можуть бути придатними для багатовимірних просторів пошуку, такі як ті, які використовуються для пошуку допустимого набору конфігурацій для руки робота, щоб підібрати об'єкт. Генерація динамічно можливих шляхів для різних практичних програм робить планування на основі вибірки популярним, хоча воно і не забезпечує повного рішення.

3. Алгоритми оптимізації траєкторії які формують завдання планування шляху як завдання оптимізації, яка враховує бажану продуктивність транспортного засобу, відповідні

обмеження на динаміку транспортного засобу. Вони можуть застосовуватися для онлайн-планування шляху в невизначених умовах. Проте, залежно від складності завдання оптимізації, планування у реальному часі може бути нездійсненним.

Планування шляху, поряд із сприйняттям та системами управління, складають три основні будівельні блоки навігації для будь якої роботизованої платформи чи транспортного засобу. Планування шляху додає автономності до таких систем, як безпілотні автомобілі, роботизовані платформи, безпілотні наземні транспортні засоби та безпілотні літальні апарати. Алгоритми планування шляху зазвичай намагаються отримати найкращий шлях або принаймні допустиме наближення до нього. Найкращий шлях тут відноситься до оптимального, у тому сенсі, що результуючий шлях виходить шляхом мінімізації однієї або кількох функцій оптимізації. Наприклад, цей шлях може бути тим, який вимагає найменшої кількості часу. Це є вирішальне значення в таких місіях, як пошуково-рятувальні операції: жертви катастроф можуть звернутися за допомогою в ситуаціях, коли йдеться про життя чи смерть. Іншою функцією оптимізації, яку слід враховувати, може бути енергія роботизованої платформи.

Були досліджені наступні алгоритми [3-5]:

1. Dijkstra's Algorithm

Алгоритм Дейкстри може бути надзвичайно успішною стратегією для визначення найкоротшого шляху до цільової точки. Цей алгоритм працює ітераційним методом, обчислюючи найкоротшу відстань від початкової точки до всіх інших точок графа. Запропонований алгоритм паралелізує початковий процес: він починається з пошуку найкоротших шляхів від першого вузла до всіх інших вузлів, дозволяючи їм використовувати різні шляхи в найкращому випадку та той самий шлях до додаткових вузлів у гіршому випадку. Головним недоліком є те, що алгоритм, починає обчислення безпосередньо з центрального вузла всіх інших вузлів, і тому використовує інформацію незалежно від пройденого шляху.

Переваги:

- гарантує найкоротший шлях;
- простий для реалізації та розуміння.

Недоліки:

- повільний при великих просторах тому, що перевіряє всі можливі шляхи;
- не використовує евристик тому неефективний у складних динамічних ситуаціях.

2. A*(A-star)

A* – це алгоритм обходу графа та пошуку шляху, який використовується в багатьох галузях комп'ютерних наук завдяки своїй повноті, оптимальності та ефективності.

Його можна розглядати як розширення алгоритму Дейкстри. A* досягає кращої продуктивності в порівнянні з алгоритмом Дейкстри. Алгоритм A* – це алгоритм на основі евристичної функції для правильного планування шляху. Він обчислює значення евристичної функції у кожному вузлі робочої зони і включає перевірку надто великої кількості сусідніх вузлів знаходження оптимального рішення з нульовою ймовірністю зіткнення. Але це займає багато часу для обробки і сильно знижує швидкість роботи.

Порівняно з алгоритмом Дейкстри, алгоритм A* знаходить лише найкоротший шлях від зазначеного джерела до зазначеної мети, а не дерево найкоротших шляхів від зазначеного джерела до всіх можливих цілей. Це необхідний компроміс для використання евристики, спрямованої на конкретну мету.

Переваги:

- швидший за Дейкстра завдяки використанню евристики;
- гарантує оптимальний шлях, якщо евристика допустима;

– гнучкий та широко використовуваний в робототехніці.

Недоліки:

- якість залежить від правильно підбраної евристичної функції;
- велике споживання пам'яті при складному середовищі.

3. Artificial Bee Colony (ABC)

Основна ідея алгоритму була натхненна завданням збору меду бджолиними колоніями. Алгоритм ABC був схвалений багатьма вченими через його переваги, такі як швидка збіжність і відмінна продуктивність алгоритму при вирішенні задачі.

Алгоритм штучної колонії бджіл був розширений для вирішення багатоцільових задач, алгоритм MOABC потім часто використовувався в різних галузях. Проте про планування шляху з урахуванням багатоцільового штучного алгоритму рою рідко повідомляється.

Переваги:

- висока здатність до глобальної оптимізації;
- гнучкий і стійкий до умов середовища;
- підходить для багатокритеріальних задач.

Недоліки:

- висока складність у тонкому налаштуванні параметрів;
- не завжди стабільна продуктивність при великій кількості перешкод.

4. Rapidly-exploring Random Tree (RRT)

Класичні алгоритми планування шляху містять стратегію пошуку представлену A^* , та стратегію вибірки, представлену ймовірнісною дорожньою картою, перша покладається на поточну позицію та історичну інформацію про позицію, і таким чином, схильна до локальних оптимальних рішень. Алгоритм Rapidly-exploring Random Tree поєднує в собі характеристики пошуку та вибірки:

- його характеристики пошуку виявляються в тому, що алгоритм починається з кореневого вузла і продовжує розгалужуватися, ростячи як дерево, поки не знайде цільовий вузол;
- його характеристики вибірки виявляються у тому, що на алгоритм впливають випадкові точки вибірки у процесі розгалуження, тому алгоритм поліпшення Rapidly-exploring Random Tree має потенціал досягнення кращої продуктивності, ніж інші алгоритми.

Переваги:

- дуже ефективний для задач з високою розмірністю (наприклад 3Д-простори);
- працює добре в середовищах з багатьма перешкодами;
- легко реалізується, можна використовувати для рухомих роботів.

Недоліки:

- шлях може бути не оптимальним;
- потребує додаткових алгоритмів для згладжування маршруту.

5. Firefly Algorithm (FA)

Алгоритм світлячка (FA) був розроблений доктором Сінг-Ше Янгом в Кембриджському університеті в 2007 році. Цей алгоритм імітує шлюбну поведінку світлячків. Світлячки спілкуються змінюючи інтенсивність свого світла. Ця інтенсивність світла також має зворотну залежність від відстані. Чим більша відстань тим нижча інтенсивність світла.

Основна концепція алгоритму, описаного, полягає у наступному :

- між світлячками немає гендерних відмінностей, а привабливість залежить тільки від інтенсивності світіння кожного світлячка;
- притягання між двома світлячками пропорційно їхній яскравості і обернено пропорційно відстані між світлячками;
- яскравість світлячка визначається значенням цільової функції.

Алгоритм починається з отримання інформації про навколишнє середовище чи завдання. Після завершення світлячок ініціалізується, і його місцезнаходження на карті записується. Передатна функція обчислює яскравість кожного світляка. Після встановлення яскравості кожен світлячок порівнюється на основі його яскравості із сусідніми світлячками. Якщо цільової функції досягнуто, ітерація закінчується. В іншому випадку він продовжить обчислювати яскравість кожного світляка і повторювати кроки знову.

Переваги:

- хороший для глобального пошуку та уникнення локальних мінімумів;
- простий у реалізації та параметризації;
- може адаптуватися до складного середовища.

Недоліки:

- повільна збіжність до точної траєкторії;
- вимагає багато ітерацій для точного результату;
- результати можуть варіюватись залежно від налаштувань

6. Particle Swarm Optimization

Алгоритм Particle Swarm Optimization починається з великої кількості можливих рішень. Ці частинки безперервно переміщуються у просторі пошуку для пошуку оптимального рішення. У цьому процесі кожна частка визначає свій рух у просторі пошуку. Вона робить це, поєднуючи свою поточну позицію з історичною найкращою позицією. Крім того вона включає деякі випадкові збурення. Коли знаходяться покращені позиції, ці позиції спрямовують рух рою. Процес повторюється багато разів у спробі знайти задовільне рішення, яке, хоч і не гарантується, як очікується, знайде глобальну або майже глобальну оптимальну відповідь наприкінці всіх циклів алгоритму.

Алгоритм можна узагальнити в чотири основні кроки, які повторюються доти, докине буде виконано умову зупинки :

- призначте початкові випадкові положення на швидкості всім частинкам у просторі пошуку;
- оцініть придатність кожної частки;
- поновіть індивідуальні та світові найкращі позиції;
- оновіть швидкість та положення кожної частки.

Алгоритм Particle Swarm Optimization успішно застосовувався у багатьох галузях, таких як планування шляху та завдання оптимізації функцій. Було показано, що алгоритм Particle Swarm Optimization забезпечує кращі результати швидше та простіше порівняно з іншими методами.

Переваги:

- швидка збіжність у простих задачах;
- простота реалізації, мінімум параметрів;
- добре підходить для неперервних середовищ.

Недоліки:

- може застрягати в локальних мінімумах;
- погано працює у складних середовищах з багатьма перешкодами;
- не гарантує оптимальності рішення.

ВИСНОВКИ. Таким чином була зібрана та структурована інформація щодо алгоритмів для розроблення автоматизованої системи визначення мінімальної траєкторії руху роботизованої платформи. Після проведеного аналізу існуючих алгоритмів для розроблення автоматизованої системи визначення мінімальної траєкторії руху роботизованої платформи а також проведеного аналізу переваг та недоліків перелічених алгоритмів. Було виявлено, що для поставленої задачі більш за все підходять наступні алгоритми :

- Firefly Algorithm (FA);
- Rapidly-exploring Random Tree (RRT);
- A*(A-star).

ВИКОРИСТАНА ЛІТЕРАТУРА

1. 3rd International Conference on Innovations in Automation and Mechatronics Engineering, ICIAME 2016 Time-Efficient A* Algorithm for Robot Path Planning Akshay Kumar Guruji, Himansh Agarwal, D. K. Parsediya* https://www.sciencedirect.com/science/article/pii/S2212017316300111?ref=pdf_download&fr=RR-7&rr=933f1b1d785a1907 (дата звернення 19.04.2025)
2. Mobile Robot Path Planning Based on Improved Particle Swarm Optimization and Improved Dynamic Window Approach <https://onlinelibrary.wiley.com/doi/10.1155/2023/6619841> (дата звернення 19.04.2025)
Recent advances in Rapidly-exploring random tree: A review <https://www.sciencedirect.com/science/article/pii/S2405844024084822> (дата звернення 20.04.2025)
3. Firefly Algorithm: Bio-Inspired Decision Making Algorithm <https://medium.com/@shantanuparab99/firefly-algorithm-bio-inspired-decision-making-algorithm-d25941725d14> (дата звернення 21.04.2025)
4. Невлюдов І. Ш., Андрусевич А. О., Євсєєв В. В., Новоселов С. П., Демська Н. П. Проектування мобільних маніпуляційних роботів: Монографія. – Х. :, 2022.
5. Path Planning for Autonomous Mobile Robots: A Review by José Ricardo Sánchez-Ibáñez <https://www.mdpi.com/1424-8220/21/23/7898> (23.04.2025)
6. Yevsieiev, V., Abu-Jassar, A., Maksymova, S., & Demska, N. (2025). Development of a model for recognizing various objects and tools in a collaborative robot workspace. *ACUMEN: International journal of multidisciplinary research*, 2(1), 224-239.
7. Yevsieiev V. Comparison of Functional Capabilities of Classic Manipulator Robots and Collaborative Robots / V. Yevsieiev, N. Demska // Digital innovation & sustainable development 2024 : Proceedings of I st International Conference, November 15, 2024. - Kharkiv, 2024. - P.16-17.
8. Yevsieiev V. Using Contouring Algorithms to Select Objects in the Robots' Workspace / V. Yevsieiev, S. Maksymova, N. Demska // Technical science research in Uzbekistan. – 2024. – Vol. 2(2). – P. 32–42.
9. Nevliudov, I., Yevsieiev, V., Maksymova, S., Demska, N., Starodubcev, N., & Klymenko, O. (2023, September). Monitoring System Development for Equipment Upgrade for IoT. In *2023 IEEE 5th International Conference on Modern Electrical and Energy System (MEES)* (pp. 1-5). IEEE.
10. Nevliudov, I., Yevsieiev, V., Maksymova, S., Demska, N., Kolesnyk, K., & Miliutina, O. (2023, September). Mobile Robot Navigation System Based on Ultrasonic Sensors. In *2023 IEEE XXVIII International Seminar/Workshop on Direct and Inverse Problems of Electromagnetic and Acoustic Wave Theory (DIPED)* (Vol. 1, pp. 247-251). IEEE.
11. Chala, O., Yevsieiev, V., Maksymova, S., & Abu-Jassar, A. (2025). USING THE HUMAN FACE RECOGNITION METHOD BASED ON THE MOBILENETV2 NEURAL NETWORK IN AUTHENTICATION SYSTEMS. *Multidisciplinary Journal of Science and Technology*, 5(3), 882-895.

Науковий керівник: Демська Наталія Павлівна, доцент, кандидат технічних наук, доцент кафедри КІТАР Харківського національного університету радіоелектроніки