

Міловідов Юрій Олегович

старший викладач кафедри комп'ютерних наук,

Національний університет біоресурсів і природокористування України

ORCID: <https://orcid.org/0009-0000-5705-3590>

E-mail: yurii_milovidov@nubip.edu.ua

Бородкіна Ірина Лаврентіївна

кандидат технічних наук, доцент, доцент кафедри комп'ютерних наук,

Національний університет біоресурсів і природокористування України

ORCID: <https://orcid.org/0000-0003-3667-3728>

E-mail: i.borodkina@nubip.edu.ua

ВІЗУАЛІЗАЦІЯ АЛГОРИТМУ ПОШУКУ ОПТИМАЛЬНОГО ШЛЯХУ МІЖ ДВОМА ТОЧКАМИ КЛІТИННОГО ЛАБІРИНТУ З ПЕРЕШКОДАМИ, ЩО ДИНАМІЧНО ЗМІНЮЮТЬСЯ

Анотація. У даній роботі досліджується актуальна проблема пошуку оптимального шляху в динамічних середовищах, що моделюються як клітинні лабіринти з перешкодами. Проблема пошуку оптимального шляху між двома точками клітинного лабіринту є актуальною через її універсальність, прикладну цінність і фундаментальне значення для розвитку сучасних технологій. Основна увага приділяється розробці та програмній реалізації системи візуалізації алгоритмів на графах, яка дозволяє в реальному часі спостерігати за процесом прийняття рішень автономним агентом. У роботі проведено детальний порівняльний аналіз класичних методів, таких як алгоритм Дейкстри та A^* , а також спеціалізованих інкрементальних підходів, зокрема $D^* Lite$. Обґрунтовано переваги використання теорії графів для вирішення задач природокористування, логістики та робототехніки, де середовище може змінюватися непередбачувано. Метою представленої роботи є розробка програми для візуалізації алгоритму пошуку оптимального шляху на полі, яке можна уявити у вигляді лабіринту з переборними і непереборними перешкодами. Задача полягає в тому, щоб знайти оптимальний шлях між двома точками на полі та відобразити його. Практична частина дослідження включає розробку програмного забезпечення на мові C#, яке демонструє процес перебудови маршруту при виникненні нових перешкод без необхідності повного перерахунку всієї мережі. Це критично важливо для мінімізації обчислювальних витрат у складних інформаційно-аналітичних системах. Окремий акцент зроблено на освітньому аспекті: розроблена візуалізація інтегрована у навчальний процес для викладання дисциплін «Алгоритми і структури даних» та «Об'єктно-орієнтоване програмування», що значно покращує засвоєння складних математичних концепцій студентами. Результати роботи підтверджують, що поєднання теоретичних методів пошуку шляху з інтерактивною візуалізацією забезпечує високу надійність і прозорість функціонування сучасних навігаційних та екологічних систем моніторингу. Програма візуалізації алгоритму пошуку оптимального шляху в лабіринті має величезне практичне значення і може застосовуватися у робототехніці та автономних системах для навігації роботів у реальному середовищі, у мережах і телекомунікаціях здійснює пошук оптимального маршруту передачі даних. Візуалізація алгоритму пошуку оптимального шляху в клітинному лабіринті з динамічними перешкодами дозволяє глибше зрозуміти принципи роботи алгоритмів, оцінити їх ефективність у реальному часі, експериментувати з різними стратегіями перебудови маршруту.

Ключові слова: пошук у глибину, пошук у ширину, теорія графів, пошук оптимального шляху, динамічне середовище, алгоритм Дейкстри, алгоритм A^* , алгоритм $D^* Lite^*$, візуалізація алгоритмів, клітинний лабіринт, інформаційно-аналітична підтримка.

Актуальність. Сфера природокористування сьогодні – це не просто видобування природних ресурсів або охорона лісів, це вирішення питань, пов'язаних з управлінням складними екосистемами в умовах кліматичних змін та обмеженості ресурсів. Без якісної інформаційно-аналітичної підтримки приймати ефективні рішення в галузі природокористування сьогодні практично неможливо. Одним з напрямків такої підтримки є використання теорії графів, яка являє собою один із найпотужніших математичних інструментів. Серед напрямків застосування теорії графів у сфері природокористування в першу чергу слід зупинитись на моделюванні взаємозв'язків між об'єктами. В екології та управлінні екоресурсами графом можна представити будь-яку мережеву структуру, а методи теорії графів дозволяють вирішувати при цьому найрізноманітніші задачі, як-то моделювання екологічних коридорів просування популяцій тварин (тут вершинами графа є заповідники, лісові масиви або парки, а ребрами – можливі шляхи міграції тварин між ними. Використання алгоритмів пошуку найкоротших шляхів в такому графі дозволяє зрозуміти функціональну зв'язність ландшафту, розрахувати оптимальні «зелені коридори», оскільки для тварини найкоротшим є шлях з найменшими енергетичними витратами або мінімальним ризиком. Тобто такі алгоритми дозволяють знаходити шлях, який забезпечує максимальну виживаність особини при переході між масивами). В [1] розглядається методологія використання алгоритмів найменшої вартості (LCP) для моделювання генетичних дистанцій між популяціями, а робота [2] присвячена оптимізації обчислень у великих мережах заповідників. Автори використовують алгоритм Флойда-Воршелла для побудови повної матриці зв'язності територій, що дозволяє виявити не лише прямі шляхи міграції, а й опосередковану роль кожного заповідника як транзитного вузла в масштабах цілого регіону.

Аналіз гідрографічних мереж з використанням теорії графів перетворює хаотичне сплетіння річок, притоків та каналів на чітку математичну структуру – орієнтований ациклічний граф (DAG), в якому вершинами слід обирати витoki річок, місця злиття притоків, ГЕС, очисні споруди, а ребрами слугують русла річок із заданим напрямом. Застосування теорії графів при такому описі дозволяє моделювати поширення забруднень та визначати оптимальні точки розміщення датчиків моніторингу в річкових басейнах [3]. Питання, пов'язані з моделюванням повеней та логістикою водних ресурсів з використанням DAG-графів розглядаються в [4].

Теорія графів дозволяє здійснювати моделювання процесів зникнення одного виду і визначення впливу такого зникнення на всю мережу. Аналіз впливу зникнення одного виду на екосистему в теорії графів називається аналізом стійкості мережі (Network Robustness) або моделюванням каскадних вимирань (Secondary Extinctions). У таких моделях екосистема розглядається як граф харчової мережі (Food Web), де вузли – це види, а ребра – трофічні зв'язки. Робота [5] фокусується на тому, як видалення "центрального" вузла такого графу (тобто видів з великою кількістю зв'язків) призводить до розвалу всієї мережі. В ній описано алгоритми моделювання каскадного ефекту, коли зникнення хижака або ключової жертви викликає серію вторинних умирань. В роботі [6] автор розглядає питання застосування теорії графів для ідентифікації видів, чиє зникнення завдає найбільшої шкоди стабільності екосистеми. А в роботі [7] теорія графів використовується для моделювання різних сценаріїв вимирання окремих популяцій і оцінювання порогу, після якого екосистема втрачає здатність до самовідновлення.

При проектуванні інженерних мереж в сфері природокористування (наприклад, систем зрошення або лісових доріг) теорія графів дозволяє розв'язувати два основних завдання: мінімізацію вартості (довжина мережі) та максимізація надійності (забезпечення потоку) [8].

Отже, проблема пошуку оптимального шляху між двома точками клітинного лабіринту є актуальною через її універсальність, прикладну цінність і фундаментальне значення для розвитку сучасних інформаційних технологій. Вона зумовлена переходом від простих геометричних задач до управління надскладними системами в реальному часі. У природокористуванні та екології це стає критичним через непередбачуваність природних процесів. Для статичних середовищ, тобто таких середовищ, де перешкоди не змінюються,

актуальність змістилася з пошуку "найкоротшої лінії" до пошуку "шляху з найменшим опором". Для динамічних середовищ (кліматичні зміни, техногенні аварії) актуальність полягає в здатності алгоритмів реагувати на зміни середовища "на льоту". В таких задачах доцільно використовувати Алгоритм D*-Lite (Dynamic A* Lite) – інкрементальний алгоритм пошуку найкоротшого шляху, розроблений для динамічних середовищ, де граф маршрутів може змінюватись у реальному часі. Слід зазначити, що сучасні алгоритми перераховують шлях у графі лише для змінених ділянок, а не для всієї мережі, що економить обчислювальні ресурси. Слід зазначити також, що пошук шляху на графі в сучасних програмних застосуваннях - це інструмент, який дозволяє людині вирішувати стратегічні задачі, ефективно використовувати ресурси, мінімізуючи втручання в екосистеми, що динамічно змінюються під тиском глобальних викликів.

Аналіз останніх досліджень та публікацій. Сучасні дослідження у сфері пошуку оптимального шляху в статичних та динамічних середовищах зосереджені на вдосконаленні евристичних алгоритмів, підвищенні їхньої швидкодії та адаптивності, а також інтеграції з методами локального планування руху.

У роботі [9] проаналізовано історичний розвиток алгоритмів на основі A* та визначено напрями їх подальшої оптимізації для застосування в статичних і динамічних середовищах. Автор розглядає класичний алгоритм A* як основу більшості сучасних підходів до планування маршруту, підкреслюючи його універсальність і здатність до модифікацій.

Роботи [10], [11] присвячені інтеграції Adaptive A* з удосконаленим методом Dynamic Window Approach (DWA). Запропонований підхід поєднує глобальне планування маршруту з локальною оптимізацією руху, що дозволяє ефективно реагувати на динамічні перешкоди.

Робота [12] розглядає питання, пов'язані з пошуком шляху у динамічних середовищах, тобто в середовищах, топологія яких може змінюватись в реальному часі, часто через втручання людини.

У статті [13] розглянуто вдосконалення алгоритму D*Lite шляхом інтеграції з методом Dynamic Window Approach. Запропонований алгоритм дозволяє виконувати швидко перепланування маршруту при зміні конфігурації перешкод і забезпечує стабільну роботу в реальному часі.

В роботі [14] був наданий порівняльний аналіз класичних алгоритмів пошуку на графах, а в роботі [15] порівнюється ефективність алгоритмів A*, Дейкстри та Беллмана-Форда на різних топологіях графів (сітки, випадкові графи). Ви результати порівняння було зроблено висновок, що алгоритм A* залишається лідером за швидкістю при наявності гарної евристики, випереджаючи Дейкстру в 1.3–1.9 рази.

В роботі [16] детально аналізуються алгоритми пошуку в ширину (Breadth-First Search), алгоритм Дейкстри та Алгоритм A*, а робота [17] присвячена вирішенню задачі багатокритеріальної оптимізації в умовах невизначеності (стохастичності).

Результати експериментів показали зменшення часу перепланування та підвищення адаптивності системи в складних середовищах.

Важливим аспектом дослідження є візуалізація процесу пошуку. У роботі [18] розглянуто застосування теорії графів для вдосконалення та візуалізації алгоритму пошуку найкоротшого шляху в математичній моделі відеоігри.

Метою статті є розробка алгоритмів візуалізації пошуку оптимального шляху на полі, яке моделюється як лабіринт із переборними та непереборними перешкодами, та його програмна реалізація.

Матеріали і методи дослідження. *Постановка задачі.* Маємо клітинний лабіринт у вигляді двовимірної сітки розміром $N \times M$, де кожна клітинка може бути:

- вільною,
- перешкодою,
- початковою точкою (S),
- цільовою точкою (G).

Перешкоди можуть змінювати свій стан у процесі виконання алгоритму. Необхідно:

- Знайти оптимальний (найкоротший) шлях від S до G.
- Забезпечити коректну перебудову маршруту при зміні конфігурації перешкод.
- Візуалізувати процес пошуку та оновлення шляху.

Теоретичні підстави. Лабіринт представлений у вигляді матриці (двовірний масив) (рис. 1). Кожна клітинка може бути або непереборною перешкодою, або може мати певну вагу (складність проходження даного відрізка шляху).

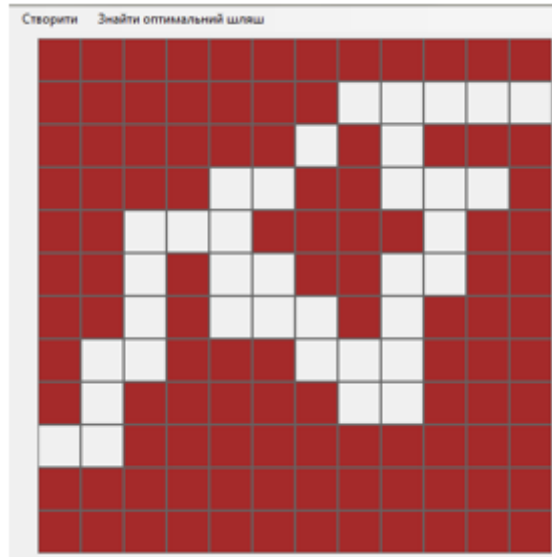


Рисунок 1 – Лабіринт, представлений у вигляді двовірний масиву

Якщо значення комірки = 0, то це непереборна перешкода, якщо комірка прохідна, то її значенням може бути дійсне число, яке відповідає складності досягнення цієї клітинки. Об'єкт може рухатися в 4-х напрямках: вгору, вниз, вліво, вправо.

Кожну комірку можна уявити як вершину графа. Якщо з неї є шлях до сусідньої комірки, то ці вершини графа пов'язані ребром відповідної ваги.

Клітинний лабіринт можна представити як граф:

- вершини – клітинки;
- ребра – можливі переходи (вгору, вниз, вліво, вправо);
- вага – зазвичай 1.

Стан клітинки опишемо такими цифровими значеннями:

- 0 – вільна,
- 1 – перешкода,
- 2 – початок,
- 3 – ціль,
- 4 – клітинка в процесі дослідження,
- 5 – фінальний шлях.

Для подання графа в пам'яті комп'ютера використовується матриця суміжності – це квадратна матриця (рис. 2), у якій кількість стовпців і рядків дорівнює кількості вершин графа.

Алгоритми пошуку оптимального шляху

Алгоритм Дейкстри (англ. Dijkstra's algorithm) – алгоритм на графах, винайдений нідерландським вченим Е. Дейкстрою в 1959 році. Гарантовано знаходить найкоротший шлях від однієї з вершин графа до всіх інших, перевіряючи всі можливі вузли навколо. Алгоритм працює тільки для графів без ребр негативної ваги. Алгоритм широко застосовується в програмуванні і технологіях, наприклад, його використовують протоколи маршрутизації OSPF і IS-IS. В екології цей алгоритм використовується для побудови екологічних мереж, де ваги ребер заздалегідь відомі (наприклад, карта опору ландшафту). Алгоритм Дейкстри застосовується для зваженого графа у разі, коли треба знайти шляху до всіх вершин у графі.

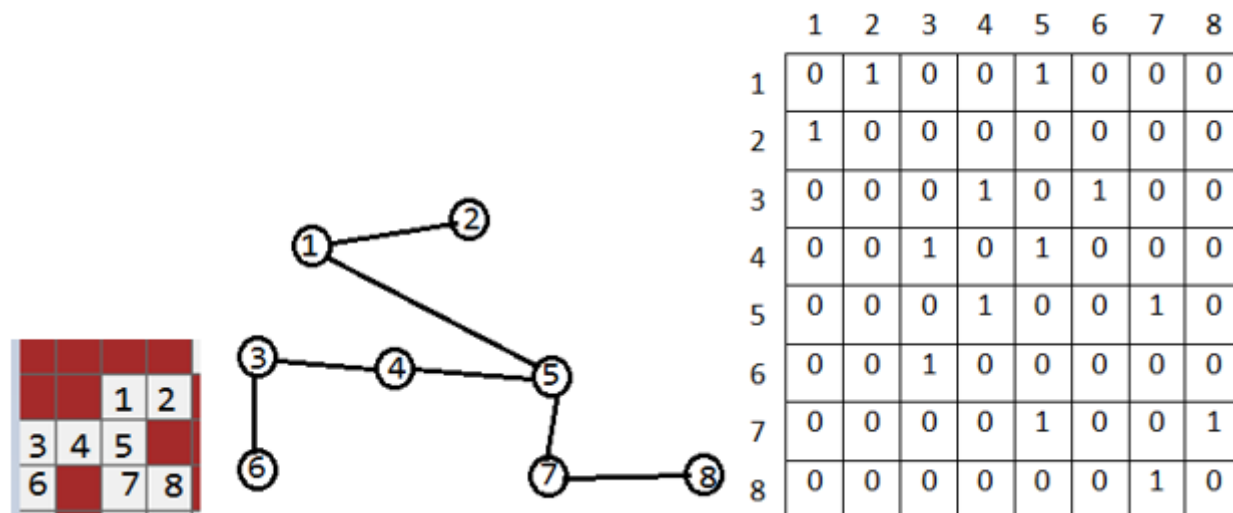


Рисунок 2 – Граф і відповідна матриця суміжності

Кожній вершині зіставимо мітку – мінімальну відому відстань від цієї вершини до a . Алгоритм працює покроково – на кожному кроці він «відвідує» одну вершину і намагається зменшувати мітки. Робота алгоритму завершується, коли всі вершини відвідані.

Переваги алгоритму полягають у тому, що він гарантує знаходження оптимального шляху та підходить для сітки з однаковими або різними вагами переходів. Тоді як до недоліків відноситься те, що не використовує евристику та може бути повільним для великих лабіринтів.

*Алгоритм A (A-Star)** – це один з найефективніших алгоритмів для пошуку шляху в сіткових структурах. Він являє собою модифікацію алгоритму Дейкстри, яка використовує прогноз відстані до цілі, що робить його значно швидшим. В екології цей алгоритм є основним стандартом для ГІС-пакетів (Linkage Mapper, ArcGIS) при моделюванні міграційних коридорів. Він використовує фактичну вартість шляху $g(n)$ й евристичну оцінку до цілі $h(n)$. Функція оцінки:

$$f(n) = g(n) + h(n).$$

Переваги алгоритму A*: швидший за алгоритм Дейкстри при використанні допустимої евристики; добре підходить для візуалізації. Серед недоліків зазначається складніша реалізація та потреба у додаткових структур даних.

Алгоритми для динамічного середовища. У випадку змін перешкод у реальному часі класичний A* потребує повного перезапуску. Для оптимізації використовують D*Lite – це інкрементальний евристичний алгоритм пошуку шляху, призначений для роботи в середовищах, що змінюються. Його основна перевага — перебудова лише змінених частин маршруту, а не повний перерахунок, як у випадку з A* search algorithm.

На відміну від A*, який будує шлях від старту до цілі, D*Lite працює у зворотному напрямку – від цілі до старту. Алгоритм підтримує дві основні оцінки для кожної вершини:

$g(v)$ – поточна найкраща відома вартість шляху;

$rhs(v)$ – однокрокова оцінка (lookahead value).

Ключова умова узгодженості:

$$g(v) = rhs(v).$$

Якщо рівність порушена – вершина вважається "неузгодженою" і потребує оновлення.

Алгоритм D*Lite використовує:

- пріоритетну чергу (відкритий список),

- евристичну функцію $h(v)$,
- ключ вершини:

$$key(v) = [\min(g(v), rhs(v)) + h(start, v) + km, \min(g(v), rhs(v))],$$

де km – коефіцієнт корекції при зміні позиції агента.

Основні етапи роботи:

Крок 1 – Ініціалізація:

- $g(v) = \infty$ для всіх вершин,
- $rhs(goal) = 0$,
- $goal$ додається в чергу.

Крок 2 – Обчислення найкоротшого шляху. Алгоритм:

- витягує вершину з мінімальним ключем,
- оновлює її значення,
- поширює зміни на сусідів.

Крок 3 – Рух агента. Агент:

- рухається до сусідньої вершини з мінімальним g ;
- при зміні середовища ініціює оновлення.

Переваги використання D*Lite:

- ефективність у великих лабіринтах,
- миттєва реакція на зміну перешкод,
- підходить для реального часу,
- мінімізує обчислювальні витрати.

Недоліки:

- складніша реалізація,
- потребує додаткових структур даних,
- вища ймовірність помилок при програмуванні.

Результати дослідження та їх обговорення. Для розробки програми обрана мова програмування C#. Програма має інтуїтивно зрозумілий інтерфейс у застосунку Windows Forms (рис. 3).

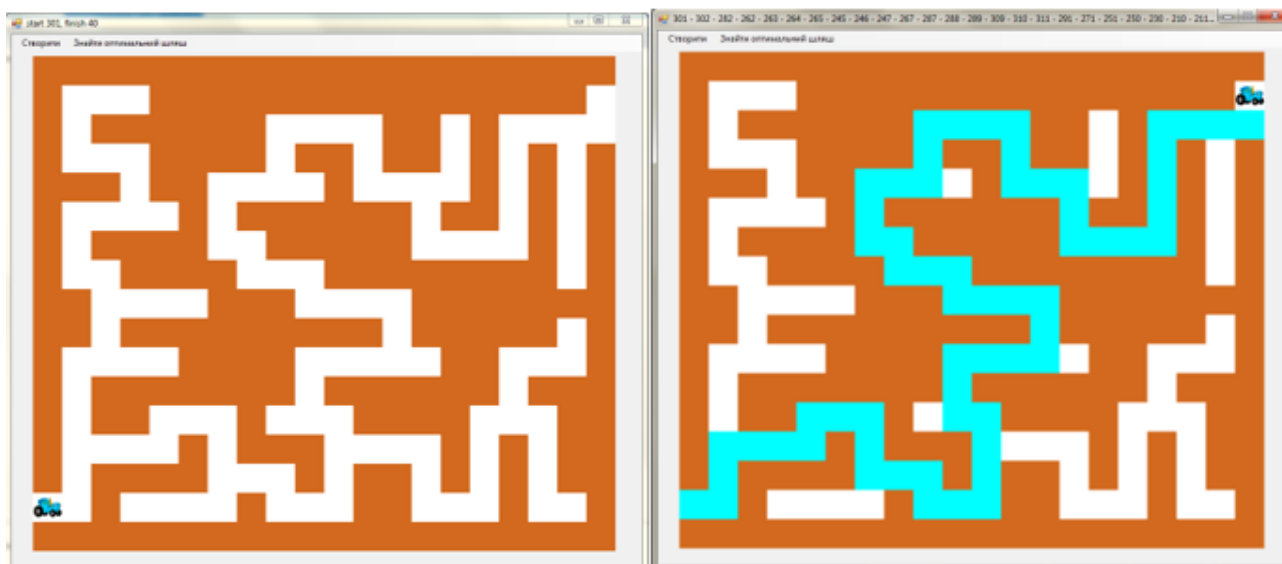


Рисунок 3 – Відображення оптимального шляху

Для пошуку оптимального шляху в лабіринті застосовані як алгоритм Дейкстри з використанням класу Dijkstra так і алгоритм D*Lite з використанням класів DstarLite (основний клас), PriorityQueue (Пріоритетна черга), Node (Модель вузла).

Діаграма класів з використанням алгоритму Дейкстри зображена на рис. 4.

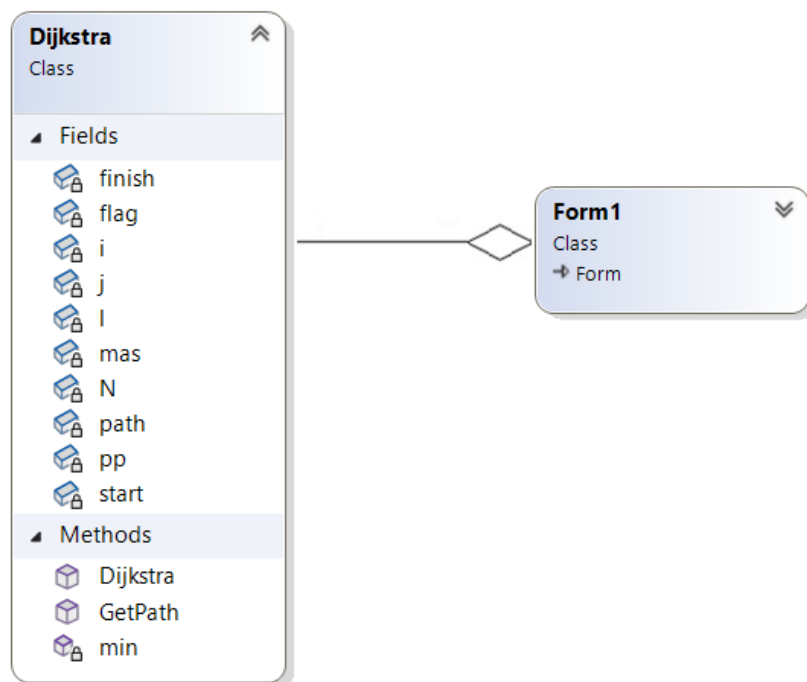


Рисунок 4 – Діаграма класів з використанням алгоритму Дейкстри

Діаграма класів з використанням алгоритму D*Lite зображена на рис. 5.

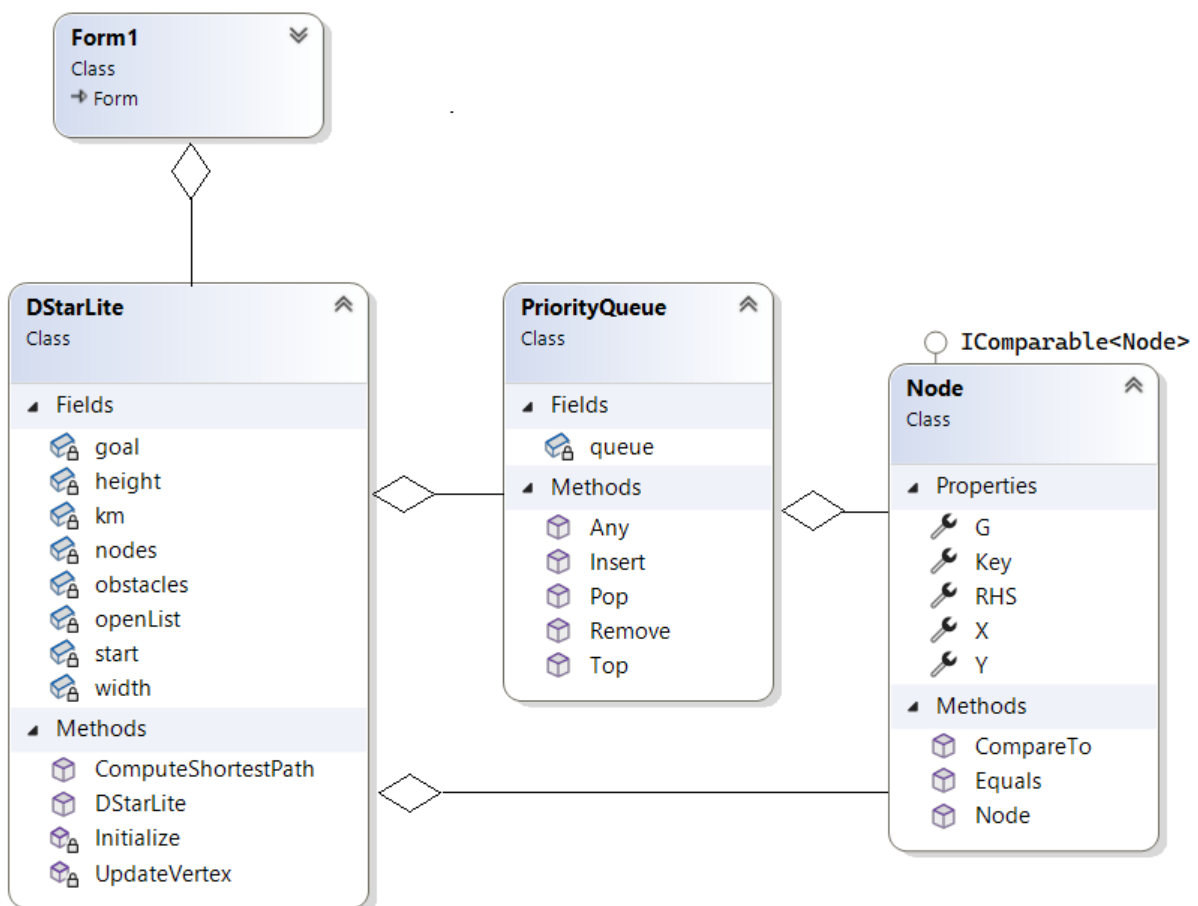


Рисунок 5 – Діаграма класів з використанням алгоритму D*Lite

Порівнюючи два алгоритми можна зробити висновок, що реалізація з використанням алгоритму Дейкстри набагато простіше, ніж реалізація з використанням алгоритму D*Lite, тому незважаючи на зазначені вище недоліки (може бути повільним для великих лабіринтів) добре підходить для візуалізації невеликих лабіринтів, таких як на рис. 3. Якщо маємо великі лабіринти і потрібна миттєва реакція на зміну перешкод, доцільно застосовувати алгоритм D*Lite, незважаючи на суттєво складнішу реалізацію і необхідність застосування додаткових структур даних.

Візуалізація алгоритмів пошуку шляху – це потужний інструмент, який поєднує теорію з практикою. Вона широко використовується в сучасних технологіях від роботів до навігаційних систем і допомагає як розробникам, так і дослідникам краще розуміти та вдосконалювати алгоритми. Програма візуалізації алгоритму пошуку оптимального шляху в лабіринті має величезне практичне значення і може застосовуватися у робототехніці та автономних системах для навігації роботів у реальному середовищі, допомагає уникати перешкод і знаходити найкоротший шлях, дозволяє інженерам бачити, як робот «приймає рішення» в реальному часі.

В навігаційних системах візуалізація допомагає аналізувати маршрути, оптимізувати трафік, демонструвати альтернативні шляхи. У логістиці та складських системах це оптимізація маршрутів для працівників, роботів, доставки товарів. У мережах і телекомунікаціях здійснює пошук оптимального маршруту передачі даних. Візуалізація дозволяє аналізувати затримки, знаходити перевантажені вузли. У медичних та біологічних задачах аналогії з лабіринтами використовуються в нейронних мережах, аналізі ДНК. В освіті та навчанні допомагає студентам зрозуміти як працюють алгоритми. Інтерактивні симуляції значно покращують засвоєння матеріалу.

Висновки та перспективи. Проведені в роботі дослідження дозволяють сформулювати такі висновки щодо використання алгоритмів пошуку оптимального шляху та їх візуалізації:

Задача пошуку оптимального шляху в лабіринтах із динамічними перешкодами є критично важливою для сучасного природокористування, екології, робототехніки, логістики та навігаційних систем. Використання теорії графів дозволяє ефективно моделювати складні екосистеми, наприклад, екологічні коридори для міграції тварин або гідрографічні мережі.

Проведене в роботі порівняння алгоритмів дозволило зробити такі висновки.

Алгоритм Дейкстри гарантує знаходження найкоротшого шляху і добре підходить для візуалізації невеликих лабіринтів завдяки простішій реалізації, хоча може бути повільним на великих об'єктах.

Алгоритм A* є ефективнішим за алгоритм Дейкстри в статичних середовищах завдяки використанню евристичних оцінок відстані до цілі.

Алгоритм D*Lite є оптимальним для динамічних середовищ, оскільки дозволяє перераховувати лише змінені частини маршруту замість повного перезапуску, що критично для роботи в реальному часі.

Програмна візуалізація роботи алгоритмів пошуку оптимального шляху на графі є потужним інструментом, що дозволяє бачити процес «прийняття рішень» автономними системами в реальному часі, аналізувати затримки та оптимізувати маршрути.

Інтерактивні симуляції та розроблена програма на мові C# довели свою ефективність при практичному використанні, допомагаючи наочно зрозуміти принципи роботи алгоритмів пошуку оптимального шляху на графі.

Практична цінність запропонованого в роботі підходу сприяє мінімізації витрат ресурсів та антропогенного втручання в екосистеми, забезпечуючи при цьому максимальну надійність і наочність результатів.

Основними перспективними напрямками для подальшого розвитку запропонованих в роботі підходів можна вважати розробку гібридних моделей, де класичні алгоритми пошуку шляху (A*, D* Lite) поєднуюватимуться з нейронними мережами для прогнозування появи перешкод у динамічних середовищах на основі історичних даних, та розширення можливостей візуалізації шляхом впровадження 3D-візуалізації та технологій доповненої реальності (AR)

для наукового моделювання рішень автономних систем безпосередньо на місцевості (наприклад, для навігації сільськогосподарських роботів на складних рельєфах).

Список використаних джерел

1. Minor, E.S., Urban, D.L. (2008), A Graph-Theory Framework for Evaluating Landscape Connectivity and Conservation Planning. *Conservation Biology*, 22: 297-307. <https://doi.org/10.1111/j.1523-1739.2007.00871.x>.
2. Shanu, S., Wattasseril, J. I., Qureshi, Q., & Jhala, Y. V., Bhattacharya S. (2016). A graph theoretic approach for modelling wildlife corridors. arXiv. <https://doi.org/10.48550/arXiv.1603.01939>
3. Choi, S., Ullah, Z., & Son, M. (2026). A graph-based machine learning framework for river water quality management under data limitations. *Journal of Environmental Management*, 398, 128575. <https://doi.org/10.1016/j.jenvman.2026.128575>.
4. Aho, K., Kriloff, C., Godsey, S. E., Ramos, R., Wheeler, C., You, Y., Warix, S., Derryberry, D., Zipper, S., Hale, R. L., Bond, C. T., & Kuehn, K. A. (2023). Non-perennial stream networks as directed acyclic graphs: The R-package streamDAG. *Environmental Modelling & Software*, 167, 105775. <https://doi.org/10.1016/j.envsoft.2023.105775>.
5. Dunne, J. A., & Williams, R. J., Martinez N. D. (2002). Food-web structure and network theory: The role of connectance and size, 99 (20) 12917-12922. <https://doi.org/10.1073/pnas.19240769>
6. Longjas, A. ; Tejedor, A. ; Foufoula-Georgiou, E. (2017). Graph Theory Approach for Studying Food Webs. American Geophysical Union, Fall Meeting 2017, abstract #IN33B-0119. <https://ui.adsabs.harvard.edu/abs/2017AGUFMIN33B0119L/abstract>.
7. Evans D.M., Pocock M.J. O., Memmott J. (2013). The robustness of a network of ecological networks to habitat loss. *Ecology Letters*, 16(7). <https://doi.org/10.1111/ele.12117>.
8. Ormsbee, L. E., & Lansley, K. E. (2024). Graph theory applications in water distribution system design and optimization. *Journal of Water Resources Planning and Management*, 150(4), 04023012. <https://doi.org/10.1061/JWRMD5.WRENG-6001>.
9. Zheng, Y. (2025) Historical Evolution and Future Optimization of A*-Based Path Planning in Static and Dynamic Environments. *Highlights in Science, Engineering and Technology*. <https://hsetdata.com/index.php/ojs/article/view/838>.
10. Liao, T., Chen, F., Wu, Y., Zeng, H., Ouyang, S., Guan, J. (2024) Research on Path Planning with the Integration of Adaptive A-Star Algorithm and Improved Dynamic Window Approach *Electronics* 13(2), 455; <https://doi.org/10.3390/electronics13020455>.
11. Sinkevych, O., Boyko, Y., Sokolovskyy, B., Rechynskiy, O. (2024) Study of Pathfinding Approach Based on A* With Adaptive Occupancy Grid. *ACPS*. 9(2)6, 95–100, <https://doi.org/10.23939/acps2024.02.095>.
12. Stawarz, P., Ozog, D., Łabuński, W. (2024) Supported Influence Mapping for Mobile Robot Pathfinding in Dynamic Indoor Environments *Sensors*. 24(22), 7240, <https://www.mdpi.com/1424-8220/24/22/7240>.
13. Gao, Y., Han, Q., Feng, S., Wang, Z., Meng, T., Yang, J. (2024) Improvement and Fusion of D*Lite Algorithm and Dynamic Window Approach for Path Planning in Complex Environments *Machines*, 12(8), 525; <https://doi.org/10.3390/machines12080525>.
14. Chen, L., & Wang, Y. (2025). Time and memory trade-offs in shortest-path algorithms for dynamic road networks. *Journal of Applied Mathematics and Computing*, 71(3), 442–459. <https://doi.org/10.1007/s12190-025-01842-x>.
15. Aldhafferi N. (2025) Time and Memory Trade-Offs in Shortest-Path Algorithms Across Graph Topologies: A*, Bellman–Ford, Dijkstra, AI-Augmented A* and a Neural Baseline. *Computers*, 14(12), 545; <https://doi.org/10.3390/computers14120545>.
16. Patel, A. (n.d.). Introduction to the A* algorithm. Red Blob Games. Retrieved May 22, 2024, <https://www.redblobgames.com/pathfinding/a-star/introduction.html>.
17. Ji, Y., Zuo, T., & Hu, L. (2023). Pareto optimal path generation algorithm in stochastic transportation networks with reliability constraints. *Transportation Research Part C: Emerging Technologies*, 148, Article 104039. <https://doi.org/10.1016/j.trc.2023.104039>.

18. Vanin, V., Zalevska, O., Jablonskiy, P. (2020). The application of graph theory for improving and rendering algorithm for finding the shortest path in the mathematical models of video games [Zastosuvannia teorii hrafiv dlia udoskonalennia ta vizualizatsii alhorytmu poshuku naikorotshoho shliakhu v matematychnii modeli videoihry]. *Applied Geometry and Engineering Graphics*. 97, 23–28, <https://doi.org/10.32347/0131-579x.2020.97.23-28>.

Milovidov Yurii

*Senior Lecturer of the Department of Computer Science,
National University of Life and Environmental Sciences of Ukraine*
ORCID: <https://orcid.org/0009-0000-5705-3590>
E-mail: yurii_milovidov@nubip.edu.ua

Borodkina Iryna

*Candidate of Engineering Sciences, Associate Professor, Department of Computer Science,
National University of Life and Environmental Sciences of Ukraine*
ORCID: <https://orcid.org/0000-0003-3667-3728>
E-mail: i.borodkina@nubip.edu.ua

VISUALIZATION OF THE ALGORITHM FOR FINDING THE OPTIMAL PATH BETWEEN TWO POINTS IN A GRID MAZE WITH DYNAMICALLY CHANGING OBSTACLES

Abstract. *This paper investigates the current problem of finding the optimal path in dynamic environments modeled as cellular labyrinths with obstacles. The problem of finding the optimal path between two points of a cellular labyrinth is relevant due to its versatility, applied value and fundamental importance for the development of modern technologies. The main attention is paid to the development and software implementation of a system for visualizing algorithms on graphs, which allows real-time observation of the decision-making process of an autonomous agent. The paper provides a detailed comparative analysis of classical methods, such as Dijkstra's algorithm and A*, as well as specialized incremental approaches, in particular D* Lite. The advantages of using graph theory for solving problems in environmental management, logistics and robotics, where the environment can change unpredictably, are substantiated. The purpose of the presented work is to develop a program for visualizing the algorithm for finding the optimal path on a field, which can be represented as a labyrinth with surmountable and insurmountable obstacles. The task is to find the optimal path between two points on the field and display it. The practical part of the research includes the development of software in C#, which demonstrates the process of rerouting the route when new obstacles arise without the need for a complete recalculation of the entire network. This is critically important for minimizing computational costs in complex information and analytical systems. A special emphasis is placed on the educational aspect: the developed visualization is integrated into the educational process for teaching the disciplines "Algorithms and Data Structures" and "Object-Oriented Programming", which significantly improves the assimilation of complex mathematical concepts by students. The results of the work confirm that the combination of theoretical methods of pathfinding with interactive visualization provides high reliability and transparency of the functioning of modern navigation and environmental monitoring systems. The program for visualizing the algorithm for finding the optimal path in a maze has great practical importance and can be used in robotics and autonomous systems for robot navigation in a real environment, in networks and telecommunications it searches for the optimal data transmission route. Visualizing the algorithm for finding the optimal path in a cellular maze with dynamic obstacles allows for a deeper understanding of the principles of operation of algorithms, to assess their effectiveness in real time, and to experiment with different strategies for rerouting the route.*

Keywords: *Depth First Search, Breadth First Search, Graph theory, Pathfinding, Dynamic environment, Dijkstra's algorithm, A-star algorithm (A)*, D Lite algorithm*, Algorithm visualization, Grid-based maze, Decision support systems.*