

ПІДХІД ДО УСУНЕННЯ КОНФЛІКТІВ У МИЛЬТИАГЕНТНИХ СИСТЕМАХ НА ОСНОВІ АЛГОРИТМУ ДЕЙКСТРИ

В.Г.Пенко, О.В.Пенко, В.В.Коган

Одеський національний університет імені І. І. Мечнікова,
вул.Дворянська,2, м.Одеса, Україна,
e-mails: vpenko@onu.edu.ua , odael.odes@gmail.com, vladislav.kogan@gmail.com

Завдання пошуку агентом маршруту є одним із базових завдань і має велике практичне значення. Як правило, це оптимізаційне завдання, яке вирішується у просторі графа станів з метою мінімізації довжини маршруту. Таке завдання стає складнішим і набуває ще більшої популярності, якщо воно вирішується на основі одного графового простору декількома агентами. В цьому випадку з'являється ще один практично важливий параметр оптимізації, який виражається кількістю конфліктів між агентами. Така абстрактна постановка завдання має численні варіанти формалізації та велику кількість підходів до її вирішення. У цієї роботі основним елементом підходу є застосування класичного алгоритму Дейкстри, що виконується послідовно кожним агентом. Під час знаходження найкоротшого маршруту кожен агент модифікує графовий простір таким чином, щоб зменшити ймовірність конфліктів з наступними агентами. Цей похід є оригінальним і для перевірки був виконаний ряд обчислювальних експериментів за допомогою розробленого для цієї мети програмного забезпечення. Експерименти, що були проведено, демонструють адекватну поведінку основного алгоритму. У роботі проаналізовано обмеження, властиві запропонованому підходу у рамках абстрактної постановки завдання. Визначено напрями подальшого розвитку базового підходу.

Ключові слова: кооперативна поведінка агентів, алгоритм Дейкстри, мультиагентні системи, конфлікти.

Вступ та огляд існуючих підходів. Розробка та дослідження мультиагентних систем (МАС) є перспективним сучасним науково-практичним напрямом, який можна віднести до категорії міждисциплінарних на стику наступних галузей: інформатика (особливо штучний інтелект), психологія (особливо соціальна психологія) та філософія. Цей список можна розширити, але й у такому вигляді зрозуміло, що від вдалих проектів МАС слід очікувати синергетичного ефекту не тільки завдяки синхронній участі кількох агентів, але й ефекту використання методів цих наукових напрямів, що їх взаємодоповнює.

Мабуть, ключовий фактор, що дозволяє досягти позитивного синергетичного ефекту – це здатність системи організувати взаємодію агентів так, щоб зменшити або навіть мінімізувати конфлікти між ними. В іншому випадку ми ризикуємо отримати зворотний ефект, коли агенти діють непродуктивно, заважаючи один одному. Звернемо увагу, що словосполучення «здатність системи» не має на увазі тут конкретної децентралізованої архітектури системи. Більш того, найбільшого ефекту слід очікувати від систем значною мірою гомогенних.

Область досліджень, якій присвячена дана робота, має досить усталене позначення – Multi-Agent Path Finding (MAPF). З цього погляду поведінка агента зводиться до знаходження маршруту у просторі станів.

Основною метою даної роботи є розробка механізму, який дозволить

знаходити баланс між двома взаємно суперечливими показниками – кількістю конфліктів та довжина маршруту. Іншими словами, ми хочемо знайти субоптимальне рішення одночасно для цих двох показників. Незважаючи на присутність показника довжини маршруту, спектр предметних областей не обмежений середовищами, в яких явно присутній метричний простір (одночасне переміщення автомобілів з автопілотами, авіаційних транспортних засобів, роботів, що виконують сервісні функції у приміщенні чи будівлі). При більш загальній інтерпретації предметна область може мати не просторового аспекту і бути системою споживання ресурсів обмеженого обсягу, які агенти використовують рои виконанні своїх завдань. Таким чином, важливим аспектом запропонованого дослідження є розробка методики, що дозволяє перетворювати концепції вихідної предметної галузі у концепції формального походу. Зазначимо також, що в цій роботі ми намагаємося запропонувати підхід, який дозволить уникнути конфлікту, а не усувати його після факту його виникнення. У термінах МАС цей метод значною мірою відноситься до розділу планування [1].

Аналіз джерел, присвячених проблемам MAPF демонструє різноманітність методів їх вирішення. Природно, значна частина цих методів пов'язана з побудовою графа станів для розв'язуваної задачі та застосування різних різновидів алгоритмів пошуку на графах. Для підвищення ефективності часто використовують різновиди алгоритму A*. Наприклад, у роботі [2] пропонується модифікація EPEA*, у якій у процесі пошуку оптимізується кількість вузлів графа, що розкриваються на кожному кроці. У цьому напрямі робота [3] пропонує використовувати дворівневий пошуковий алгоритм, спроектований у тому, щоб перевершити традиційний алгоритм A*. Двохрівнева декомпозиція на думку авторів має забезпечити оптимальність рішення, зберігаючи при цьому прийнятні показники ефективності. Автори стверджують, що запропонований підхід забезпечує оптимальність рішення за прийнятний час у деяких випадках його застосування. Багато з пропонувананих у цьому напрямі підходів реалізують багатоагентність у формі групового агента (joint agent), у результаті вузли графового простору подають інформацію фактично про безліч агентів.

Хоча робота [4] також у основі формальної моделі використовує графові алгоритми, але у значно іншій манері. Автори зосередили зусилля на безконфліктному русі транспортних засобів на ділянках із некерованими перехрестями. Математичним підґрунтям цього підходу є алгоритм мінімального розміру клікового покриття графа, а інструментарієм дослідження – імітаційне моделювання з подальшим аналізом результатів експериментів.

Інша методика вирішення проблем MAPF пропонується у [5]. Тут як основу пошукового алгоритму автори використовують генетичний алгоритм. Недоліком генетичного алгоритму, як і інших традиційних методів штучного інтелекту, є його непрозорість, яка не дозволяє користувачеві критично аналізувати отримані рішення, спираючись на зрозумілу логіку алгоритму. З іншого боку, прикладний напрямок запропонованого автором підходу пов'язаний з управлінням авіаперевезеннями (Air Transportation Management - ATM), де такий критичний аналіз є важливим. Тому, крім самого механізму пошуку маршрутів, автори пропонують використовувати методи візуалізації для аналізу рішень.

Поданий тут невеликий перелік підходів не претендує на повноту, але дозволяє зрозуміти, що для вирішення завдання організації ефективної безконфліктної поведінки безлічі агентів застосовуються дуже різноманітні у математичних основах методи. Досить серйозною проблемою є також різноманітність формулювань розв'язуваних задач, що ускладнює порівняльний аналіз різних підходів.

Базове завдання кооперативної поведінки агентів та основний підхід до її вирішення. Предметна область даної роботи – мультиагентне середовище, у якому агенти досягають своїх різних цілей, намагаючись мінімізувати взаємні конфлікти. У цій роботі така ситуація не буде формалізована таким чином, щоб формулювання виявилось би певною мірою універсальним. Спроби зробити таке формулювання міститься в [6]. Однак зворотною стороною такої універсальності є складність здійснити на її основі реалізацію представницької множини корисних прикладних випадків.

Розглянемо наступну версію постановки завдання. Деякі агенти існують у просторі, заданому орієнтованим зваженим графом. Вагу дуги називатимемо довжиною дуги. Кожен із агентів на початку знаходиться в одному із вузлів графа (стартовий вузол). Наприкінці процесу агент повинен опинитися у вузлі, який називатимемо цільовим вузлом. Може існувати кілька маршрутів, які приводять агента зі стартового вузла до цільової. Кількість маршрутів може бути навіть нескінченною через наявність циклів у графі. Конфліктом між двома агентами у початковій постановці завдання є присутність у їх маршрутах однакової дуги. Кількість конфліктів – це кількість таких збігів. Метою завдання є знаходження таких маршрутів для всіх агентів, які одночасно мінімізують довжини маршрутів та кількість конфліктів для всієї сукупності агентів. Не завжди вдається досягти нульової кількості конфліктів за мінімізації довжини маршрутів. І тут рішення приймається виходячи з балансу між цими суперечливими показниками у конкретній задачі.

У такій постановці завдання немає поняття часу. Тому поняття конфлікту визначено безвідносно послідовності дуг у маршруті. Таку постановку завдання далі називатимемо базовим завданням кооперативної поведінки.

У такому трактуванні завдання його не можна безпосередньо сприймати як модель руху матеріальних об'єктів у метричному просторі. Важливим напрямом розвитку пропонованого підходу є його розширення до більш реалістичних варіантів завдань.

Кількість можливих маршрутів для одного агента в сукупності з необхідністю проаналізувати ці маршрути для багатьох агентів, унеможлиблює рішення базового завдання навіть для невеликого графового простору і кількості агентів.

Напрями наукових досліджень, у яких можна шукати методи розв'язання базової задачі: по-перше, це теорія машинного навчання з підкріпленням як у загальному вигляді [7], так і її похідна підобласть – мультиагентне навчання з підкріпленням [8], по друге це теорія ігор, в якій досліджуються особливості взаємодії агентів з різними цілями, які не завжди збігаються [9].

Ми використовуємо у певному сенсі зворотний підхід. Взавши за основу гіпотезу про застосування алгоритму Дейкстри в умовах постановки завдання, уточнюватимемо, як та чи інша модифікація цього алгоритму може дозволити вирішити відповідний різновид завдання.

У цій роботі передбачається розуміння канонічного алгоритму Дейкстри, опис якого можна знайти у різних джерелах [10]. З погляду базового завдання досить знати, що алгоритм Дейкстри знаходить найкоротший маршрут між двома вузлами графа і має обчислювальну складність $O(\text{nodes} * \log \text{nodes} + \text{edges} * \log \text{edges})$ для розріджених графів. Тепер уявімо, що в цьому ж графовому просторі є ще кілька таких агентів, які вирішують таке ж завдання, але з іншими початковими та кінцевими пунктами. Якщо вирішити завдання кожного з них незалежно один від одного, може виявитися, що рішення (найкоротші маршрути) мають спільні дуги. На рисунку 1 показано найкоротші маршрути, знайдені

алгоритмом Дейкстри для двох агентів незалежно один від одного. Тут sa_1 та sa_2 – стартові точки агентів, а fa_1 та fa_2 – їх кінцеві точки. Дуга від вузла a до вузла b використовується в обох маршрутах.

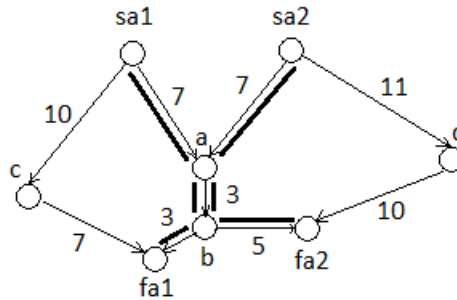


Рис. 1. Результати незалежного використання алгоритму для двох агентів.

Це не означає неминуче, що агент 1 і агент 2 виявляться під час своєї подорожі одночасно на цих дугах. Однак якщо не спробувати детально відстежувати процес переміщення агентів у часі, факт спільного використання дуг у кількох найкоротших маршрутах загалом підвищує ймовірність конфлікту у формі одночасного використання ділянки шляху (дуги).

Існують спроби такого відстеження. Зокрема, можна відзначити публікацію Д. Сільвера [11]. Однак такий підхід досить ресурсомісткий, оскільки означає підвищення розмірності розв'язуваного завдання.

Звідси впливає формулювання завдання як зниження ймовірності такого роду конфліктів кількома агентами.

Основна ідея досить проста та інтуїтивно сприймається як адекватна. Дано можливість агенту 1 побудувати свій маршрут звичайним чином. Після цього додамо до ваги всіх використаних у маршруті дуг деяку величину збільшення. Після цього на графі, що змінився, своє завдання вирішує агент 2. Якщо є наступний агент, то знову відбувається збільшення ваг дуг і т.д.

Тепер після збільшення ваг дуг логіка алгоритму Дейкстри для агента 2 з меншою ймовірністю побудує маршрут, що використовує більш «важкі» дуги. Наприклад, задача, представлена на рисунку 2 із застосуванням збільшення ваги на 10 призведе до того, що агент 2 вибере маршрут, що не має загальних дуг з маршрутом агента 1.

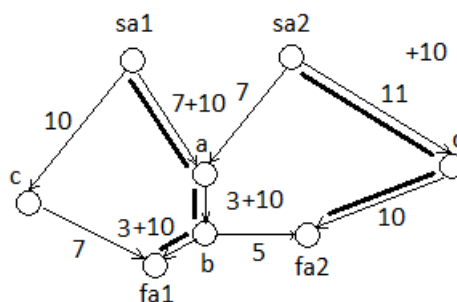


Рис. 2. Маршрути двох агентів із використанням збільшення ваги.

Цей приклад демонструє роль величини збільшення ваги дуги. Якби у наведеному прикладі збільшення було б менше, наприклад 5, маршрут агента 2 зберігся б таким же, як на рис. 1, тобто мав би місце конфлікт спільного використання дуги. Це означає, що одним із важливих аспектів подальшого дослідження є розробка механізму вибору ефективного значення величини збільшення.

Описаний вище прийом застосування алгоритму Дейкстри для знаходження найкоротшого шляху з одночасною мінімізацією конфліктів далі

називатимемо базовим підходом. Така назва обумовлена тим, що вже зараз проглядаються модифікації цього підходу для подолання деяких властивих йому обмежень.

Для експериментальної перевірки базового походу розроблено відповідне ПЗ.

План обчислювального експерименту має такий вигляд.

1. Генеруємо граф простору.
2. Розміщуємо кілька агентів у різних стартових вузлах та призначаємо їм цільові вузли.
3. Будуємо для кожного агента найкоротший маршрут без зміни дуг графів
4. Обчислюємо деякі узагальнені показники, наприклад, сумарну довжину маршрутів всіх агентів.
5. Будуємо для кожного агента найкоротший маршрут, виконуючи описану вище модифікацію дуг графів
6. Обчислюємо деякі узагальнені показники, наприклад, сумарну довжину маршрутів всіх агентів.
7. Порівнюємо параметри, обчислені на кроці 4 та 6.

Мета експерименту – визначити умови, у яких метод збільшення ваг дуг дає адекватні для предметної області результати.

Цей план насправді узагальнений і може мати модифікації, що мають різні прикладні інтерпретації.

Найбільш природним видається випадок, коли:

А) Граф досить поступово покриває евклідовий простір. Це може бути, наприклад, прямокутна сітка із досить невеликим розміром клітин.

В) Стартові точки агентів перебувають у відносній близькості один від одного.

С) Кінцеві точки агентів знаходяться у відносній близькості один від одного.

Дамо змістовну інтерпретацію цих умов.

Умова А) в екстремальному вигляді означає визначення простору можливих переміщень у вигляді дрібнозернистої сітки із квадратних клітин. Неформально це означає велику свободу вибору напряму руху, що дозволяє уникати більшості конфліктів. Однак на практиці таке середовище не становить великого інтересу, оскільки воно не містить перешкод і обмежень, що задаються штучно створеною і не завжди регулярною структурою графа.

Умови В) та С) означають близькість загального напрямку руху агентів, що призводить до збільшення потенційних конфліктів.

Для проведення описаних вище експериментів було розроблено відповідні програмні компоненти.

Опис обчислювальних експериментів. Уточнимо умови першого експерименту. Простір переміщень агентів задається зваженим орієнтованим графом, схематично представленим на рисунку 3. (фактично розмір графа експериментально був 10×10). Довжини (ваги) вертикальних та горизонтальних дуг дорівнюють 1, а довжини діагоналей $\sqrt{2}$, тобто моделюється стандартна евклідова сітка.

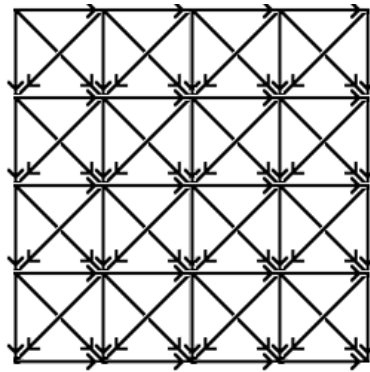


Рис.3. Граф, що використовується в експериментах (фактично 10x10).

У першій серії експериментів 10 агентів розміщувалися в лівому верхньому кутку графа, і всі вони прагнули потрапити у нижній правий кут.

Кожен експеримент у серії вирізнявся своїм значенням дельта ваги. У таблиці 1 показано, як із зміною величини збільшення ваги змінювалася сумарна відстань, пройдена агентами.

Таблиця 1

Сумарні довжини маршрутів за різних приріст ваги.

Дельта ваги	Сумарна відстань
0	127.279
0.2	144.880
0.4	152.994
0.6	159.923
0.8	165.923
1	171.923

Отже, спостерігаємо очікувану закономірність. При зростанні збільшення ваги сумарна відстань поступово зростає, оскільки агенти стають дедалі обережнішими, намагаючись уникати вже використані дуги.

Цікавим є кількість конфліктів, тобто спільно використовуваних агентами дуг графа. Однак цей показник буде мало репрезентативним, якщо ці конфлікти можуть відбуватися у різні моменти модельного часу.

Також програма, що моделює поведінку агентів, показує докладнішу інформацію про маршрути агентів таким чином:

Agent 0, best path 0:0->1:1->2:2->3:3->4:4->5:5->6:6->7:7->8:8->9:9.

Agent 1, best path 0:0->0:1->1:2->2:3->3:4->4:5->5:6->6:7->7:8->8:9->9:9.

Agent 2, best path 0:0->1:0->2:1->3:2->4:3->5:4->6:5->7:6->8:7->9:8->9:9.

Agent 3, best path 0:0->0:1->0:2->1:3->2:4->3:5->4:6->5:7->6:8->7:8->8:8->9:9.

Agent 4, best path 0:0->1:0->2:0->3:1->4:2->5:3->6:4->7:5->8:6->9:7->9:8->9:9.

Agent 5, best path 0:0->1:1->1:2->1:3->1:4->2:5->3:6->4:7->5:8->6:8->7:9->8:9->9:9.

Agent 6, best path 0:0->1:1->2:1->3:1->4:1->5:2->6:3->7:4->8:5->8:6->8:7->8:8->9:9.

Agent 7, best path 0:0->0:1->0:2->0:3->0:4->1:5->2:6->3:7->4:8->5:8->6:9->7:9->8:9->9:9.

Agent 8, best path 0:0->1:0->2:0->3:0->4:0->5:1->6:2->7:3->8:4->8:5->9:6->9:7->9:8->9:9.

Агент 9, best path 0:0->1:1->2:2->3:3->4:4->5:5->6:6->7:7->8:8->9:9.

Ця форма не зручна для аналізу, оскільки конфлікти не видно явно (тут вони для зручності виділені підкресленням), тому була виконана візуалізація маршрутів агентів. На рисунку 4 показано, як виглядають маршрути агентів при

різних збільшення ваги. Конфлікти показані за допомогою ліній, що повторюються, що з'єднують вузли.

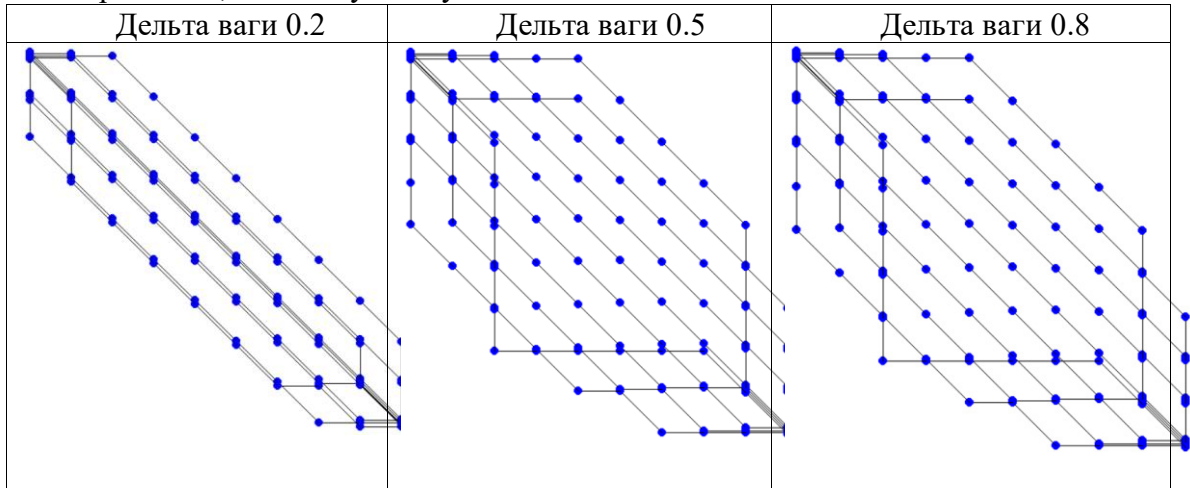
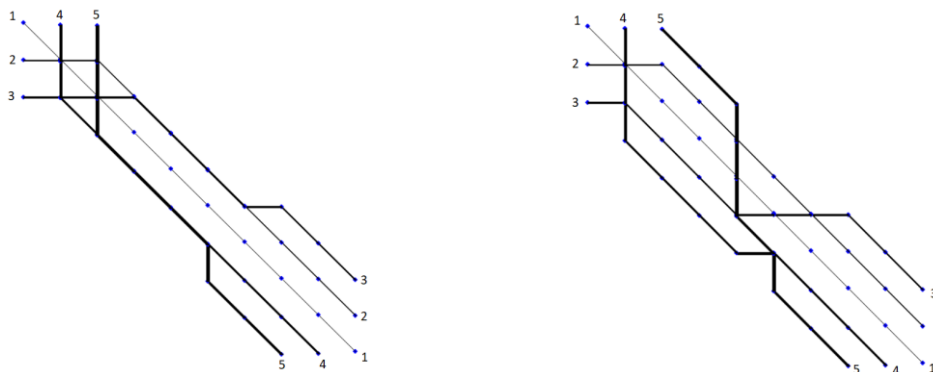


Рис. 4. Візуалізація кількох варіантів знаходження агентами найкоротших маршрутів при різних збільшеннях ваги.

У першій серії експериментів усі агенти мали однакові стартові та цільові вузли. Це зумовлює досить часті конфлікти у вигляді спільного використання дуг. При підвищенні величини збільшення ваги дуг агенти змушені ширше використовувати простір пошуку, що виглядає інтуїтивно природно.

Наступний експеримент, на наш погляд, демонструє ще цікавішу поведінку запропонованого механізму. Тепер ми розміщуємо кілька агентів у різних стартових вузлах неподалік один від одного (в околиці лівого верхнього кута), а цільові вузли задаємо різними, але в околиці правого нижнього кута. Більш того, конкретні вузли ми вказуємо так, щоб змусити перетинатися маршрути різних клієнтів. Результати деяких експериментів показано рисунку 5. Для зручності візуального відстеження маршрути різних агентів показані різною товщиною ліній. Тут бачимо, що без використання механізму збільшення ваги маршрути агентів перетинаються у великій кількості дуг. Застосування дельта ваги із значенням 0.8 дає якісно іншу картину. Маршрути агентів жодного разу не перетнулися! Агенти знайшли неочевидні маршрути, щоб уникнути конфліктів. При цьому сумарна пройдена відстань збільшилася незначно (59.255 проти 58.669)



Загальний шлях 58.66904755. Дельта ваги 0.0

Загальний шлях 59.25483399. Дельта ваги 0.8

Рис. 5. Візуалізація поведінки агентів, що розташовані в околиці один від одного і мають різні цільові вузли

Порядок використання агентів у процесі застосування базового підходу. Ще одна особливість базового підходу полягає в тому, що його результати залежить від порядку використання в ньому агентів. У деяких випадках цей порядок не можна змінити, оскільки він є наслідком особливостей предметної області. Наприклад, якщо порядок використання агентів пов'язаний із деяким поняттям пріоритету їхньої участі. Якщо ж послідовність агентів є зовнішнім чинником, нав'язаним алгоритму, можна застосувати деяку процедуру аналізу різних варіантів послідовностей участі агентів в алгоритмі.

Наступний експеримент відрізняється від попереднього тим, що відстані між вузлами (ваги дуг) тепер є випадковими, які в деякій мірі відрізняються від відстаней у правильній квадратній сітці евклідового простору. Це зроблено для того, щоб збільшити різноманітність варіантів конфігурацій маршрутів. Розмір збільшення ваги 0.2. Експеримент із 5 агентами допускає всього 5! варіантів послідовностей агентів у алгоритмі. Результати такого повного комбінаторного експерименту показано у таблиці 2.

Таблиця 2

Довжини маршрутів за різних послідовностей участі агентів.

Група маршрутів	Реальна відстань	Ефективна відстань	Кіл-ть маршрутів	Група маршрутів	Реальна відстань	Ефективна відстань	Кіл-ть маршрутів
1	56.6685	57.2685	4	13	56.8140	57.1140	2
2	56.6811	57.1811	12	14	56.8284	57.1284	4
3	56.6954	57.1954	29	15	56.8810	57.0810	1
4	56.7329	57.2329	10	16	56.8962	57.1962	2
5	56.7387	57.2387	4	17	56.9314	57.3314	4
6	56.7438	57.1438	8	18	56.9355	57.1355	4
7	56.7582	57.1582	14	19	56.9427	57.3427	3
8	56.7604	57.3604	1	20	56.9622	57.1622	1
9	56.7611	57.3611	1	21	56.9894	57.2894	3
10	56.7854	57.2854	1	22	57.0058	57.1058	3
11	56.8031	57.2031	4	23	57.0140	57.1140	2
12	56.8078	57.4078	12	24	57.1766	57.4766	2

Ефективна відстань, що фігурує в таблиці 2, означає довжину маршруту агента після того, як деякі дуги в графі були збільшені в ході виконання алгоритму. Хоча ця відстань не є частиною опису початкового стану світу, у деяких випадках вона може виявитися корисною характеристикою.

Перша група з 4-х послідовностей агентів демонструє найкращий показник реального шляху 56.6685 за рахунок збільшення довжин деяких дуг (ефективна відстань – 57.2685). Цікаво, що наступна група послідовностей дає дещо більшу реальну довжину 56.6811, але меншу ефективну (57.1811). Це означає, що у другій групі було менше конфліктів. Який із варіантів слід вважати найкращим, залежить від балансу між цінністю коротких маршрутів та мінімізацією конфліктів у конкретній предметній галузі.

Деякі результати для різних послідовностей візуально показані на рисунку 6.

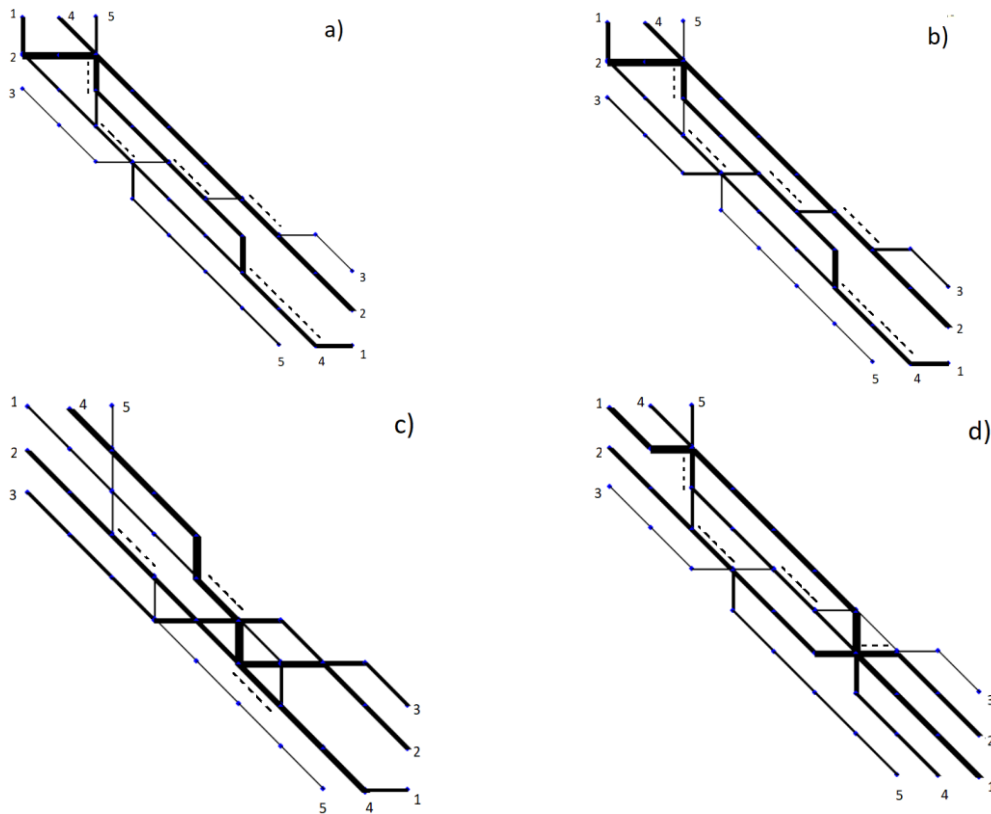


Рис. 6. Конфігурації маршрутів 5 агентів за різного порядку їхньої участі в алгоритмі (пунктиром показані конфлікти).

На рисунку 6 а) та рис. 6 б) показані результати роботи алгоритму, отримані для послідовностей агентів (3, 5, 1, 4, 2) (3, 5, 2, 4, 1). Це забезпечило найменшу загальну довжину маршрутів, однакову для двох послідовностей – 56.6684. Видно, що, незважаючи на різні послідовності агентів, їх маршрути в обох випадках однакові. Загальна кількість конфліктів (спільно використовуваних дуг) – 6 (показані пунктиром).

Рисунок 6 с) показує результат алгоритму для послідовностей агентів (2, 4, 3, 5, 1) (2, 4, 5, 3, 1) (2, 5, 3, 4, 1) (2, 5, 4, 3 1). У всіх цих випадках загальна відстань – 56.9314, що трохи гірше для найкращих послідовностей. Кількість конфліктів – 3. Це означає, що алгоритм зумів забезпечити меншу кількість конфліктів за незначного збільшення відстані. Який із варіантів вибирати залежить від того, що важливіше у конкретному випадку – мінімізація конфліктів чи відстані.

Рисунок 6 д) показує результат алгоритму для послідовностей агентів (4, 3, 0, 2, 1) та (4, 3, 1, 2, 0). Кількість конфліктів – 4. У цьому випадку загальна відстань - 57.1766. Ці послідовності програють за всіма показниками попереднім послідовностям і можуть бути відкинуті у процесі комбінаторного аналізу.

Таким чином, при невеликій кількості агентів можна провести комбінаторно вичерпний експеримент, знайшовши найкращі варіанти послідовностей агентів.

Обмеження базового підходу. Слід розуміти, що наведений вище підхід має суттєві обмеження. Це, проте, не означає його принципову прикладну обмеженість. Вже зараз проглядаються модифікації базового походу, які нівелюють деякі обмеження. У зв'язку з цим важливо сформулювати ці обмеження.

По-перше, простір прийняття рішень представлено у вигляді виваженого орієнтованого графа. Це означає, що найбільш очевидні застосування базового походу пов'язані з переміщення агентів в евклідовому просторі. Однак можлива узагальнена інтерпретація ваги дуги, що дозволяє застосовувати базовий похід для відповідних завдань без його модифікації.

Друге обмеження має на увазі, що поведінка агента в процесі проходження ним маршруту у графі здійснюється без урахування часового аспекту. Це припустимо, наприклад, коли час подолання маршрутів усіма агентами дуже малий. Умовно всі агенти долають маршрут миттєво. Ця особливість дійсно спостерігається в деяких випадках, навіть коли агенти діють у евклідовому просторі. Крім того, є предметні галузі, де відсутність тимчасового аспекту є природною. Наприклад, завдання розподілу обмеженої кількості ресурсів, необхідні виконання завдання перед її виконанням.

Третє обмеження – після того, як базовий підхід визначить для агента маршрут, цей маршрут не може бути змінений до того, як усі агенти не закінчать подорожі.

Останнє обмеження у тому, що конфліктом вважається спільне використання агентами дуг графа. У деяких випадках більш природною інтерпретацією конфлікту може бути одночасне знаходження у вузлах графа або деяке поєднання цих ситуацій.

Висновки та перспективи. Незважаючи на перелічені обмеження, базовий похід демонструє перспективну поведінку у процесі вирішення базового завдання MAPF. Результати описаних вище експериментів демонструють здатність знаходити неочевидні, але цілком адекватні маршрути для сукупності агентів.

Базовий підхід характерний наявністю основного гіперпараметра – величини збільшення ваги дуги. Зміна цього значення дозволяє знаходити під час обчислювального експерименту відповідний компроміс між довжиною маршруту та кількістю конфліктів.

Перспективи та напрямки розвитку базового підходу:

- модифікація поняття конфлікту, що дозволяє уникати перебування кількох агентів у вузлі;
- облік чинника часу, що дозволить перебування кількох агентів на дузі чи вузлі, якщо це відбувається у різні моменти часу;
- розробка автоматизованої процедури трансформації проблем з конкретних предметних областей у терміни базового завдання, тобто перетворення предметної області конкретної задачі на графовий простір з вагами дуг та вершин, що відповідають семантиці задачі.

Список літератури

1. Russell S., Norvig P. Artificial intelligence: a modern approach. Pearson, 2021. 1152 p.
2. Felner A., Goldenberg M., Sharon G., Stern R., Beja T., Sturtevant N., Schaeffer J., Holte R. Partial-Expansion A* with Selective Node Generation. *Twenty-Sixth AAAI Conference on Artificial Intelligence*. 2012. Vol.26. No.1
3. Sharon G., Stern R., Felner, Sturtevant N. (). Conflict-based search for optimal multi-agent pathfinding. *Twenty-Sixth AAAI Conference on Artificial Intelligence*. 2012.Vol.26. No.1
4. Chen C., Xu Q., Cai M., Wang J., Xu B., Li K. Conflict-free Cooperation Method for Connected and Automated Vehicles at Unsignalized Intersections: Graph-based Modeling and Optimality Analysis. *IEEE Transactions in Intelligent Transportation Systems*. 2022. Vol.14. No.8.

5. Hurter C., Dega A., Guibert N. Usage of more transparent and explainable conflict resolution algorithm: air traffic controller feedback. *34th Conference of the European Association for Aviation Psychology Pages*. 2022. P.270–278
6. Тарасов В.Б. От многоагентных систем к интеллектуальным организациям: философия, психология, информатика. М.: Эдиториал УРСС, 2002. 352 с.
7. Sutton R., Barto A. Reinforcement Learning: An Introduction. Massachusetts: MIT Press, 2018. 548 p.
8. Busoniu L., Babuska R., De Schutter B., Multi-agent reinforcement learning: A survey. *Proceedings of the 9th International Conference on Control, Automation, Robotics and Vision (ICARCV 2006)*. Singapore, 2006. P.527–532.
9. Shoham Y., Leyton-Brown K. Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations. Cambridge University Press, 2008. 532 p.
10. Dijkstra E.W. A note on two problems in connexion with graphs. *Numer. Math. Springer Science+Business Media*. 1959. Vol.1, Iss.1. P.269–271.
11. Silver D. Cooperative Pathfinding. *Proceedings of the First Conference on Artificial Intelligence and Interactive Digital Entertainment*. 2005. P.117-122.

AN APPROACH TO THE ELIMINATION OF CONFLICTS IN MULTI-AGENT SYSTEMS BASED ON DIJKSTRA'S ALGORITHM

V.G.Penko, O.V.Penko, V.V.Kogan

Odesa National University named after I. I. Mechnikov, 2 Dvoryanska St.,
Odesa, Ukraine,

e-mails: vpenko@onu.edu.ua, odael.odes@gmail.com, vladislav.kogan@gmail.com

The task of path finding by an agent is one of the basic tasks and is of great practical importance. As a rule, this is an optimization task that is solved in the space of the state graph in order to minimize the length of the path. Such a task becomes more difficult and becomes even more popular if it is solved on the basis of one graph space by several agents. In this case, another practically important optimization parameter appears, which is expressed by the number of conflicts between agents. Such an abstract formulation of the task has numerous variants of formalization and a large number of approaches to its solution. In this paper, the main element of the approach is the application of the classic Dijkstra algorithm, which is executed sequentially by each agent. When finding the shortest route, each agent modifies the graph space in such a way as to reduce the probability of conflicts with subsequent agents. This approach is original and a number of computational experiments were performed to verify it using software developed for this purpose. The experiments that were conducted demonstrate the adequate behavior of the main algorithm. The work analyzes the limitations inherent in the proposed approach within the framework of the abstract formulation of the task. The directions of further development of the basic approach have been determined.

Keywords: cooperative behavior of agents, Dijkstra's algorithm, multiagent systems, conflicts.