

УДК 519.6

Гуцул Т.В.,

Київський національний університет будівництва та архітектури

МУЛЬТИАГЕНТНА ОПТИМІЗАЦІЯ ПЛАНУВАННЯ ПОТОКІВ ДОРОЖНЬОЇ МЕРЕЖІ: ОСОБЛИВОСТІ МУРАШИНОГО АЛГОРИТМУ

Об'єкт дослідження – особливості розробки алгоритму мультиагентної оптимізації дорожньої мережі на основі порівняння існуючих мурашиних методів. В дослідженні обґрунтовано алгоритм та механізм його роботи. Пояснено необхідні вхідні дані та їх результуючий формат після виконання алгоритму. Зазначено прикладні програмні засоби для його практичної реалізації.

Ключові слова: мурашині алгоритми, мультиагентна оптимізація, планування доріг.

Постановка проблеми: в останні роки для вирішення комбінаторних задач оптимізації все частіше застосовуються алгоритми обчислювального інтелекту. Означені алгоритми володіють рядом беззаперечних переваг, зокрема достатньою простотою реалізації, гнучкістю налаштувань, високою ефективністю та можливістю пошуку глобальних або близьких до них розв'язків за поліноміальний час. До числа таких алгоритмів відносять так звані мурашині алгоритми (Ant colony optimization algorithms), які моделюють поведінку колонії мурах при вирішенні загальної задачі.

Автомобільні дороги – капіталомісткі споруди, тому їх проектування повинно бути спрямоване з одного боку на досягнення високих експлуатаційних якостей, а з іншого на мінімізацію будівельних витрат. Звісно, при всій множині існуючих алгоритмів та їх модифікацій без їх попереднього порівняння складно відшукати ідеальне вирішення. Саме тому, незважаючи на численну кількість досліджень мурашиних алгоритмів мультиагентної оптимізації, питання щодо їх ефективного застосування в плануванні дорожніх потоків залишаються актуальними [1].

Вихідні передумови: наявна базова, реально існуюча дорожня мережа (представлена у вигляді векторного файлу *.shp формату), матриця висот території об'єкту дослідження (комбіноване поєднання SRTM-даних для передгірної і рівнинної частини та результатів векторизації топографічних карт для гірської), лімітуючий шар (умовна оцінка перешкод та заохочень алгоритму різноманітного походження), шар з локалізацією населених пунктів (точки обов'язкового сполучення з розв'язком).

Аналіз останніх досліджень та публікацій: мурашині алгоритми

запропоновані ще на початку дев'яностих. Перша стаття з мурашиних алгоритмів в міжнародному науковому журналі надрукована в 1996 р. Автором її був професор Марко Доріго [2]. Мурашині алгоритми також вивчаються і вітчизняними вченими Штовбою С.Д., Рудим О.М. та Колесніковим К.В., Карапетяном А.Р., Кравченко О.В [3,4].

Постановка завдання Порівняти перспективність дослідження існуючих мурашиних методів мультиагентної оптимізації для покращення наявної дорожньої мережі. Вхідними величинами для початку роботи буде картографічна основа поділена на квадрати однакового розміру. В свою чергу, кожному квадрату присвоюватиметься певна умовна оцінка, величина котрої залежатиме від вартості побудови дороги саме в даному квадраті за поточних географічних і економічних умов. В даній статті здійснюється огляд існуючих мурашиних мультиагентних методів, що можуть бути використані для покращення наявної вихідної дорожньої мережі.

Виклад основного матеріалу. Методи мультиагентної оптимізації базуються на визначенні терміну агент. В широкому сенсі агент – це частина системи, що відповідає за прийняття рішень. В більш вузькому розумінні агенти моделюють певні біологічні організми, поведінку яких симулюють для вирішення поставленої задачі. Основним агентом мурашиного алгоритму є мураха. Сама по собі мураха не здатна вирішувати жодних комплексних задач, але сукупність особин може здійснювати самоорганізовану складну поведінку. Варто зазначити, що в біологічному прототипі в мурашнику присутня королева, яка тільки відкладає яйця і фактично не займається управлінням колонії. В програмній абстракції королева відсутня.

Узагальнена схема роботи метода мурашиних колоній показана на рис. 1.

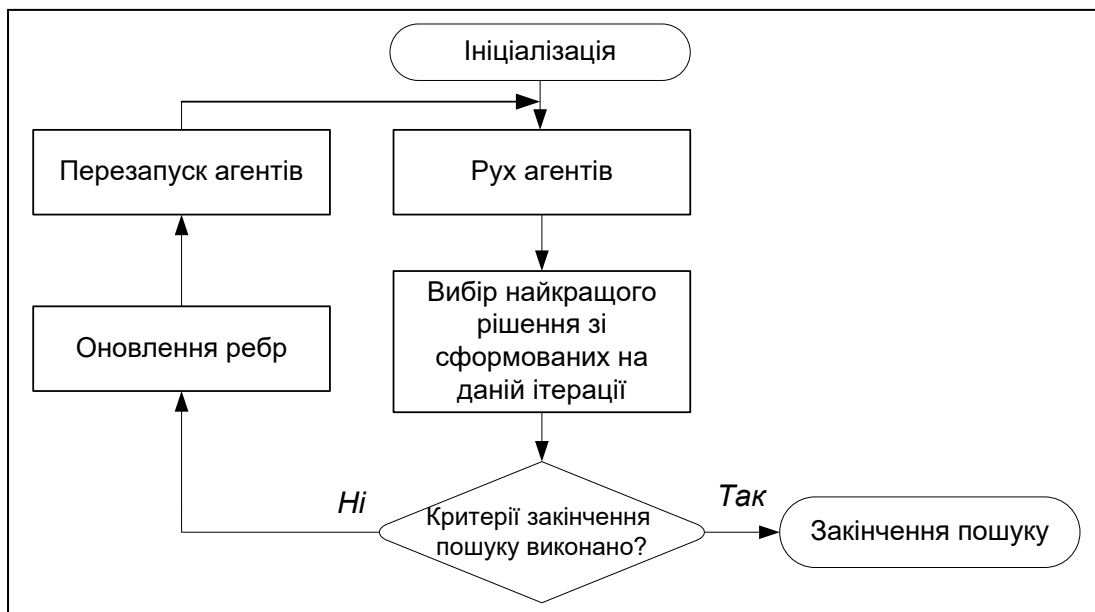


Рис. 1. Узагальнена схема роботи метода мурашиних колоній

Для підтвердження здатності колонії мурах знаходити оптимальне рішення було проведено експеримент з двома мостами [5,6].

В загальному мурашиний алгоритм можна описати таким чином:

- 1) ініціалізації мурашиної колонії
- 2) рух агентів площиною пошуку
- 3) вибір найкращого рішення сформованого на даній ітерації
- 4) перевірка критерію завершення
 - 4.1) критерій не відповідає критерію завершення:
 - 4.1.1) оновлюємо ребра
 - 4.1.2) перезапускаємо агентів і переходимо до кроку 2)
 - 4.2) критерій відповідає критерію завершення тоді переходимо до кроку 5
- 5) завершуємо пошук

Першою задачею до якої застосували мурашиний алгоритм була задача комівояжера (Travelling Salesman Problem). За її формулюванням потрібно також знаходити найкоротший можливий шлях між набором міст. Тому алгоритм природно підходив до даної задачі. В подальшому було розроблено кілька модифікацій мурашиного алгоритму для покращення його роботи. Основні відмінності полягали у тому яким способом проводити оновлення ребер графа пошуку рішень. Це важливо, оскільки джерела їжі мурах відповідають вершинам графу, а ребра відповідають шляхам, і мураха здійснюючи рух від одного джерела їжі до іншого здійснює вибір ребер. Отже, виділили три основні способи оновлення ребер: щільнісний (ant-density), кількісний (ant-quality) і циклічний (ant-cycle). У щільнісному та кількісному методах, агенти залишали феромони в процесі формування рішення, в той час, як у циклічному методі агенти залишали феромони після закінчення переміщення, тобто після формування рішення. Експериментальним шляхом було виявлено, що циклічний метод мав значно кращі результати порівняно з іншими. Тому під правилом оновлення ребер в мурашиному алгоритмі в даний час розуміють саме циклічний варіант оновлення ребер.

В процесі руху по ребру графа, мураха залишає слід із феромону. Інформація щодо кількості феромонів на ребрах графу змінюється в процесі роботи алгоритму. При цьому, кількість феромону залишеного агентом, пропорційна якості сформованого рішення: чим менше шлях, тим більше феромону буде залишено. Таким чином здійснюється пошук в напрямку знаходження кращого рішення.

Агент відвідує кожен вершину графа тільки один раз, зберігаючи при цьому номер відвіданої вершини. Після обходу всіх вершин графу розраховується пройдена відстань і ефективність маршруту. В результаті, короткий шлях матиме більш високу концентрацію феромону, ніж довший.

Потім ребра оновлюються описаним вище способом.

Існують такі розширення мурашиного алгоритму:

1) метод мурашиних систем, заснований на елітній стратегії – передбачає додаткове збільшення кількості феромонів для кращого глобального шляху при завершенні кожної ітерації. Таким чином, процедура додавання феромону до кращого шляху повторюється при цьому кількість феромону розраховується відповідно до довжини кращого шляху.

2) метод мурашиних систем заснований на ранжируванні – є розширенням попереднього методу. Агенти сортуються за довжиною складених ними шляхів, після чого глобально кращому з них феромони збільшуються з певним коефіцієнтом w . Для решти k агентів ребра оновлюються тільки ті, які увійшли в глобально кращий розв'язок причому з коефіцієнтом $(w - k)$.

3) метод системи мурашиних колоній – поліпшує метод мурашиних систем шляхом використання інформації, отриманої попередніми агентами, для вивчення простору пошуку. Даний процес забезпечують два механізми: використовується сувора елітна стратегія при оновленні феромонів на ребрах, а по-друге, агенти вибирають наступний вузол для переміщення псевдовипадковим чином (використовуючи значення феромонів і певну евристичну інформацію). Кращим агентом може бути, як агент, який отримав найкраще рішення на даній ітерації, так і взагалі глобально кращий агент. Крім того, агенти оновлюють кількість феромонів на ребрах в процесі складання рішення. Такий підхід зменшує ймовірність вибору однакових шляхів всіма агентами. За рахунок цього зменшується ймовірність зациклення в локальному оптимумі.

4) максі-мінний метод мурашиних систем – вводить нижню і верхню межі для можливих значень феромонів на ребрах. Також даний метод визначає кількість феромонів на ребрах при ініціалізації і відповідає нижній границі. В цьому методі використовується елітарна стратегія і оновлюється глобально кращий маршрут.

Для розв'язання задачі знаходження дорожньої мережі модифікуємо мурашиний алгоритм таким чином:

1) встановимо певний рівень феромону на етапі ініціалізації

2) оновлюватимемо кількість феромонів на ребрах відповідно до елітарної стратегії (кращим агентом вважатимемо глобальний і відповідно оновлюватимемо тільки його)

3) для того, щоб забезпечити зв'язок всіх населених пунктів із всіма, розмістимо в кожному із них по окремій мурашиній колонії. Кожна така колонія споживатиме певний тип їжі, який знаходитиметься у всіх містах

окрім заданого. Всі колонії використовуватимуть один феромон для позначення пройдених маршрутів. Таким чином знання про кращі маршрути будуть доступні всім колоніям

4) критерієм зупинки буде (в залежності від того яка умова швидше виконається) або фіксована кількість ітерацій або неможливість покращення поточного розв'язку протягом певної кількості ітерацій.

5) кожна мурашина колонія працюватиме паралельно з іншими оновлюючи спільні значення феромонів для знайдених ребер

Реалізувати даний алгоритм можна засобами Java. Java – сучасна мова програмування з напівавтоматичним управлінням пам'яттю. Мова широко використовується в розробці корпоративних систем різного ступеня складності. Завдяки цьому існує велика кількість бібліотек, які постійно оновлюються і вдосконалюються. Мурашині алгоритми на Java реалізовані в бібліотеці MYRA (<https://github.com/febo/myra>) Для того, щоб зберігати дані про наявність феромонів на пройдених маршрутах використаємо ChronicalMap (<http://chronicle.software/products/chronicle-map/>). Це дозволить зберігати і оновлювати значення феромонів більш ефективно порівняно із стандартними колекціями Java. Для відображення самих маршрутів і процесу їх знаходження скористаємося бібліотекою Swing, що входить в стандартний пакет Java.

Вхідними параметрами програми буде файл з координатами центрів міст (що відповідатимуть мурашникам), причому кожне місто займатиме один квадрат, а координати відповідатимуть координатам квадрата; файлу в якому для кожного квадрата вказуватиметься вартість побудови дороги; файлу, який відображатиме рельєф місцевості по вказаним квадратам (для того щоб виявити квадрати в яких знаходяться перешкоди (наприклад річки, обриви і т. ін.); файлу з конфігураціями роботи мурашиного алгоритму (на зразок кількості операцій для знаходження оптимального шляху і т. ін.).

На першому етапі обчислень із введених користувачем файлів згенеруємо одну абстрактну числову характеристику для кожного квадрата. Для прикладу, якщо в певному квадраті згідно рельєфу є перешкода (заболочена місцевість, стрімкий ухил, водна перешкода), то збільшимо вартість побудови дороги на вартість усунення перешкод (наприклад побудови мосту або осушення болотяної місцевості). Якщо перешкоду усунути неможливо з технічних причин, то тоді такому квадрату присвоїмо велике число як значення абстрактної характеристики. Таким чином, вибір даного квадрата агентом стане менш привабливим через його високу вартість, що сприятиме погіршенню самого маршруту і пошуку оптимальнішого шляху.

Другим етапом буде знаходження оптимальних маршрутів шляхом виконання мурашиного алгоритму для кожного із міст. Паралельність пошуку

сприятиме швидкому знаходженню оптимальних шляхів, а елітарне оновлення феромонів на ребрах дозволить підтримувати найкращі із них і попередить зациклення програми в точках локальних екстремумів.

Результатом виконання програми будуть знайдені маршрути, які можна буде порівняти із сучасними для оцінки рівня їх оптимальності. Власне, одержаний результат являтиме собою файл формату *.csv, із ідентифікаторами центроїдів квадратів розміщених на однаковій відстані один від одного, де можливе проходження дорожньої мережі за заданим алгоритмом. Файл розв'язку в подальшому може бути відкритий в середовищі будь-якої ГІС та інтегрований з її векторною просторовою основою для здійснення картометричних та аналітичних операцій. Таким чином можливе застосування утиліти для послідовного з'єднання обраних точок у вигляді полілінії. Автоматичне повернення довжин полілінії дозволить здійснити їх порівняння з вихідними даними реальної дорожньої мережі.

Висновки. Мурашиний алгоритм є ефективним метаевристичним методом розв'язування задачі оптимізації планування потоків. За рахунок своєї фізичної природи – моделювання поведінки мурах – він дає змогу швидко знаходити оптимальні маршрути. Проте ефективність його роботи залежить від значень керуючих параметрів, і потреба їх підбору є недоліком, що ускладнює його впровадження в програмні продукти для масового застосування

Перспективи дослідження. Найперспективнішими напрямками подальших досліджень в даній предметній області варто вважати аналіз способу вибору налаштовуваних параметрів алгоритмів. В останні роки пропонуються різноманітні способи адаптації параметрів алгоритмів «на льоту» [6]. Оскільки від вибору параметрів залежить поведінка мурашиних алгоритмів, саме до цієї проблеми звернуто найбільшу увагу дослідників на даний момент.

Ще одним перспективним шляхом покращення мурашиних алгоритмів на думку дослідників С.Д. Штовби та О.М. Рудого є їх гібридизація з іншими методами природних обчислень, наприклад, з генетичними алгоритмами. Така гібридизація реалізується у вигляді острівної схеми, коли робота мурашиного і генетичного алгоритмів здійснюється окремо – кожен на своєму острові, з обміном найкращими розв'язками протягом визначеного часу [4].

Література:

1. Гуцул Т.В. Огляд існуючих методів мультиагентної оптимізації / Т.В. Гуцул. // Містобудування та територіальне планування. – 2016. – №60. – С. 99–105.
2. Кузьмін І.А. Мурашині алгоритми / І.А. Кузьмін, В.В. Лотиш. // Комп'ютерно-інтегровані технології : освіта, наука, виробництво. – 2012. –

С. 61–66.

3. Колесніков К.В. Синектика натурних методів маршрутизації потоків даних у автономних системах телекомунікації / К.В. Колесніков, А.Р. Карапетян, О.В. Кравченко. // Вісник ХНУ. – 2010. – С. 72–74.

4. Штовба С.Д. Мурашині алгоритми оптимізації / С.Д. Штовба, О.М. Рудий. // Інформаційні технології та комп'ютерна техніка. – 2004. – С. 62–69.

5. Maniezzo V. The ant system applied to the quadratic assignment problem / V. Maniezzo, A. Colomi, M. Dorigo. – Bruuxeele: Universite Libre de Bruxelles, 1994. – 24 p.

6. Bullnheimer B. An improved ant system algorithm for the vehicle routing problem / B. Bullnheimer, R. F. Hartl, C. Strauss. – Vienna: Institute of Management Science, 1997. – 28 p.

7. T. Stützle, M. López-Ibáñez, P. Pellegrini, M. Maur, M. de Oca, M. Birattari, Michael Maur, M. Dorigo, “Parameter Adaptation in Ant Colony Optimization” // Technical Report, IRIDIA, Université Libre de Bruxelles, 2010 г.

Аннотация

Объект исследования – особенности разработки алгоритма мультиагентной оптимизации планирования потоков в дорожной сети. В исследовании обосновано алгоритм и механизм его работы. Указано о прикладных программных средствах для его практической реализации.

Ключевые слова: муравьиные алгоритмы, мультиагентная оптимизация, планирование дорог.

Annotation

The object of study – the features of ant colony algorithm in multiagent optimization of road network planning in comparison with the existing ant colony methods. This research contains algorithm definition and its flow. Defined input and output data. Indicated application software for its practical implementation. The results can be used in solving combinatorial problems in the area of planning of road networks.

Keywords: ant colony algorithm, multiagent optimization, road planning.