УДК 519.8

А.І. ПАВЛЕНКО

РОЗВ'ЯЗУВАННЯ БАГАТОКРИТЕРІАЛЬНОЇ ЗАДАЧІ ПОШУКУ ОПТИМАЛЬНОГО ШЛЯХУ В ДИНАМІЧНИХ МЕРЕЖАХ АЛГОРИТМОМ МІТОК

Запропоновано підхід до використання алгоритму міток для розв'язування задач пошуку оптимального шляху в динамічній мережі з урахуванням розкладу авіаперельотів, загальної вартості квитків на маршрут, часового вікна. Наведено опис розробленого алгоритму та оцінка ефективності з використанням реальних даних авіаперельотів.

Вступ. Пошук оптимальних шляхів у динамічних мережах знаходить застосування в багатьох сферах людської діяльності. Це створює потребу в ефективних методах для вирішення цих проблем – програмних комплексах для пошуку розв'язків у реальному часі. Такі алгоритми мають бути швидкими і забезпечувати оптимальні чи близькі до них розв'язки для будь-якого запиту користувача, що призводить до розробки алгоритмів на основі евристичних [1] або застосуванню точних алгоритмів з специфічними підходами (геометричні контейнери, скорочення ієрархій, сепаратори тощо [2]). У даній роботі розглядається застосування алгоритму міток [3] для розв'язання динамічної задачі пошуку оптимального маршруту авіаперельотів з урахуванням заданих початкового і кінцевого пунктів, часового вікна (найраніший час відправлення з початкового пункту і найпізніший час прибуття в кінцевий), загального часу і вартості квитків знайденого маршруту, а також розкладу авіаперельотів. Ключовою відмінністю мереж громадського транспорту (в тому числі авіатранспорту) від автомобільних, є наявність залежності від часу, адже деякі сегменти мережі стають досяжними тільки у визначені моменти часу. Існує два основних підходи подання розкладів у мережах – time-expanded (розширений за часом) і time-dependent (залежний від часу), stop model (модель зупинок) [4]. Ідея першого підходу полягає у створенні вершини для кожної події розкладу – прибуття або відправлення літака, що дозволяє використовувати класичні методи пошуку оптимальних шляхів без суттєвих модифікацій, але призводить до графів великої розмірності. У залежному від часу графі сегменти подорожі подаються однією дугою, а вартість подається залежною від часу функцією. Модель зупинок найчастіше використовується для топологічного подання мережі на етапах попередньої обробки даних. Для кожного підходу можна побудувати просту і реалістичну моделі. Реалістична модель зазвичай включає деталі предметної області – можливість трансферів між аеропортами, класи авіаперельотів тощо.

Постановка. Нехай N – множина вершин, що позначають аеропорти. Для подання розкладу використовується підхід залежних від часу функцій. Тобто кожній дузі, що позначає сполучення між двома вершинами, відповідає залежна від часу функція, яка повертає вартість і тривалість переходу по дузі – інформацію по рейсу. $\{v_1, v_2, \ldots, v_k\}$ – послідовність вершин, що задають маршрути слідування (routes) $R_s = \{(v_1, v_2), (v_2.v_3), .., (v_{k-1}, v_k)\}$, причому відправлення з кожної вершини v_i дозволяється у визначені моменти часу $ST_{v_i}^s := \left\{\tau_{v_i}^1, \tau_{v_i}^2, \ldots, \tau_{v_i}^\gamma\right\}$ в часовому інтервалі $[0, T]$, T – межі інтервалу планування.

Для спрощення в даній роботі вважаємо, що час подорожі залежить тільки від часу відправлення τ з вершини відправлення v_i і позначається $t_s^\tau(v_i, v_{i+1})$. Маршрут (*itinerary*) – будь-який шлях $p^\tau(v_0, v_n)$ між двома вершинами v_0 та v_n із зазначенням часу відправлення τ_{v_i} для кожної вершини.

$$p^\tau(v_0, v_n) := ([(v_0, v_1);\ \tau_{v_0}; s_0],\ [(v_1, v_2);\ \tau_{v_1}; s_1], \ldots,\ [(v_{n-1}, v_n);\ \tau_{v_{n-1}}; s_{n-1}]). \quad (1)$$

Задача планування маршруту полягає у пошуці такого маршруту, який задовольняє умовам розкладу авіаперельотів і є оптимальним за заданими критеріями.

Математичну модель задачі сформулюємо так. Критерії формуються такими факторами: c_1 – загальна вартість подорожі, c_2 – час подорожі, c_3 – кількість трансферів транзитного часу c_4, де $d_{v_0}^e$ і $d_{v_0}^l$ – найраніший і найпізніший час відправлення з початкової вершини, $a_{v_n}^e$ і $a_{v_0}^l$ – найраніший і найпізніший час прибуття у кінцеву вершину, $t_s^\tau(v_i, v_{i+1})$ і $c_s^\tau(v_i, v_{i+1})$ – час транзиту і вартість перельоту рейсом s з пункту v_i в v_{i+1} в час τ.

Більш формально:

$$\tau_{v_i} + t_s^{\tau_{v_i}}(v_i, v_{i+1}) \le \tau_{v_{i+1}}, (v_i, v_{i+1}) \in p^t(v_0, v_n), \tau_{v_i} \in ST^s, \quad (2)$$

$$d_{v_0}^e \le \tau_{v_0} \le d_{v_0}^l, \quad (3)$$

$$a_{v_n}^e \le \tau_{v_{n-1}} + t_s^{\tau_{v_{q-1}}}(v_{n-1}, v_n) \le a_{v_0}^l. \quad (4)$$

Цільова функція – мінімізувати сумарну вартість подорожі:

$$f = \min c^{\tau}(v_0, v_n), \tag{5}$$

де $c^{\tau}(v_0, v_n)$ – сумарна вартість маршруту $p^{\tau}(v_0, v_n)$.

Для порівняння якості маршрутів у даній роботі використовується поняття лексикографічного порядку. Вважатимемо, що шлях $p_1^{\tau}(v_0, v_n)$ домінує над $p_2^{\tau}(v_0, v_n)$ ($p_1^{\tau} \le_L p_2^{\tau}$), якщо існує j, що $c_{k_j}(p_1^{\tau}) \le c_{k_j}(p_2^{\tau})$ і $c_{k_i}(p_1^{\tau}) = c_{k_i}(p_2^{\tau})$ для $i = 1, ..., j-1$.

Розширенням розглядуваної задачі може бути включення до маршруту обов'язкових для відвідування пунктів у певний час. У такому разі задача розбивається на кілька підзадач пошуку проміжних шляхів між початковим і проміжним пунктом у заданому часовому вікні, пошуку шляхів між проміжними пунктами, а також проміжними і кінцевим.

Опис алгоритму. Кожному рейсу (з відправленням з v_r в час τ) відповідає мітка $\underline{\lambda}^{\tau}(v_r) := (\lambda_{c_{k_1}}^{\tau}(v_r), \lambda_{c_{k_2}}^{\tau}(v_r), \lambda_{c_{k_3}}^{\tau}(v_r), \lambda_{c_{k_4}}^{\tau}(v_r))$, за якою оцінюється якість переходу по дузі. Кожний атрибут мітки $\lambda_{c_{k_i}}^{\tau}(v_r), i = 1, 2, 3, 4$ відповідає значенню критерію c_{k_i} в маршруті з вершини v_r в час τ у вершину v_n. На кожному кроці алгоритму визначається мінімальна лексикографічна мітка з покажчиком $\psi^{\tau}(v_r)$ на наступну вершину в маршруті. Розглянемо алгоритм.

1. Ініціалізувати мітки для кожного моменту часу і вершини $\lambda_{c_{k_i}}^{\tau}(v_n) = 0$, $\lambda_{c_{k_i}}^{\tau}(v_r) = \infty$, $\psi^{\tau}(v_r) = nil$, $\forall \tau \in [d_{v_0}^{e}, a_{v_n}^{l}]$.

2. Для кожної вершини v_r, суміжної до кінцевої вершини v_n, для якої $\tau \in [d_{v_0}^{0}, a_{v_n}^{l}]$ і $a_{v_n}^{e} \le \tau + t_s^{\tau}(v_r, v_n) \le a_{v_n}^{l}$ створити нові мітки: $\lambda_{c_1}^{\tau} = c_s^{\tau}(v_r, v_n)$, $\lambda_{c_2}^{\tau} = t_s^{\tau}(v_r, v_n)$, $\lambda_{c_3}^{\tau} = 1$, $\lambda_{c_4}^{\tau} = t_s^{\tau}(v_r, v_n)$, $\psi^{\tau}(v_r) = v_n$.

3. Для кожного моменту часу k, $a_{v_n}^{l} \le k \le d_{v_0}^{e}$ (у зворотному порядку).

3.1. Для кожної вершини v_r створити мітки очікування для моменту k, базуючись на оптимальних мітках у момент $k+1$: $\lambda_{c_1}^{\tau} = \lambda_{c_1}^{\tau+1}(v_r)$, $\lambda_{c_2}^{\tau} = \lambda_{c_2}^{\tau+1}(v_r) + 1$, $\lambda_{c_3}^{\tau} = \lambda_{c_3}^{\tau+1}(v_r)$, $\lambda_{c_4}^{\tau} = \lambda_{c_4}^{\tau+1}(v_r)$, $\psi^{\tau}(v_r) = v_r$. Використовуючи лексикографічний порядок, видалити з множини міток недомінуючі.

3.2. Для кожного моменту часу $\tau = k$ з розкладу ST для сполучення (v_r, v_j) створити мітки: $\lambda_{c_1}^{\tau} = c_s^{\tau}(v_r, v_j) + \lambda_{c_1}^{\tau+1}(v_r)$, $\lambda_{c_2}^{\tau} = t_s^{\tau}(v_r, v_j) + \lambda_{c_2}^{\tau+1}(v_r)$, $\lambda_{c_3}^{\tau} = \lambda_{c_3}^{\tau+1}(v_r) + 1$, $\lambda_{c_4}^{\tau} = c_s^{\tau}(v_r, v_j) + \lambda_{c_4}^{\tau+1}(v_r)$, $\psi^{\tau}(v_r) = v_j$. Використовуючи лексикографічний порядок, видалити з множини міток недомінуючі.

Таким чином, на виході з алгоритму маємо множину пар $(\underline{\lambda}^{\tau}(v_r), \psi^{\tau}(v_r))$ міток $\underline{\lambda}^{\tau}(v_r) := (\lambda^{\tau}_{c_{k_1}}(v_r), \lambda^{\tau}_{c_{k_2}}(v_r), \lambda^{\tau}_{c_{k_3}}(v_r), \lambda^{\tau}_{c_{k_4}}(v_r))$ для кожної вершини v_r і покажчика $\psi^{\tau}(v_r)$ на наступну вершину для кожного моменту часу $\tau \in T$ (причому в множині не існує мітки, яка домінує над іншою). За отриманою множиною міток можна однозначно визначити оптимальний шлях з початкового пункту v_0 в момент часу τ_0 через покажчики на наступну вершину $\psi^{\tau}(v_r)$ у кожний наступний момент часу.

Оцінка ефективності пропонованого підходу до розробки алгоритму розв'язування здійснювалась шляхом аналізу результатів обчислювального експерименту, в якому виконувався запуск алгоритму для бази рейсів авіаліній на квітень 2017 року з кількістю аеропортів 5 – 13 і часовим інтервалом, що відповідав кількості розглядуваних днів 5 – 7. Таким чином, загальний розмір графу, що відповідає базі рейсів, складав від 34 до 8272 дуг. Для спрощення час був дискретизований за днями. Дані отримані з агрегатору пошуку авіаперельотів SkyScanner [5]. Залежність між розміром таблиці розкладу (кількості днів і кількості розглядуваних міст) показана на рис. 1. Пошук виконувався для всіх можливих комбінацій міст для кожного значення часового вікна. Залежність між кількістю запусків алгоритму і розміром часового вікна показана на рис. 2.

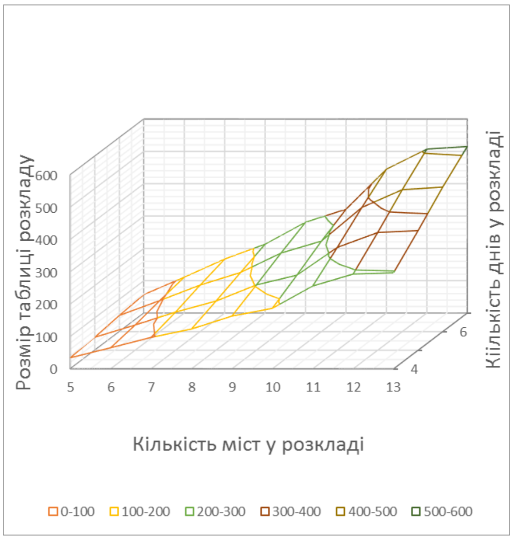

РИС. 1. Залежність між розміром таблиці розкладу авіаперельотів від розміру часового вікна і кількості міст

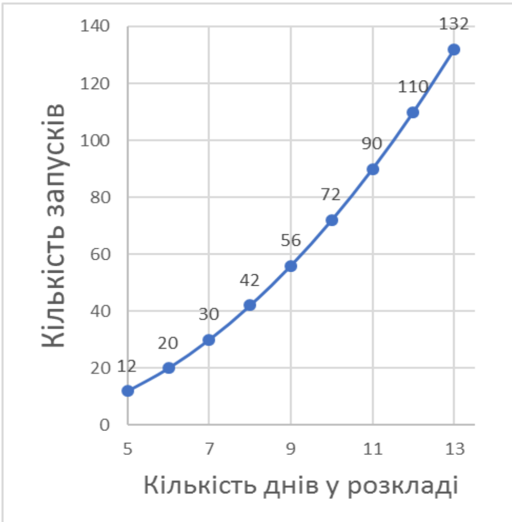

РИС. 2. Залежність між розміром часового вікна і кількістю запуску алгоритму

На рис. 3 і 4 показано середній час роботи алгоритму в мс для одного запуску в залежності від розміру таблиці розкладу. Середній час визначався з урахуванням кількості запусків алгоритму (див. рис. 2).

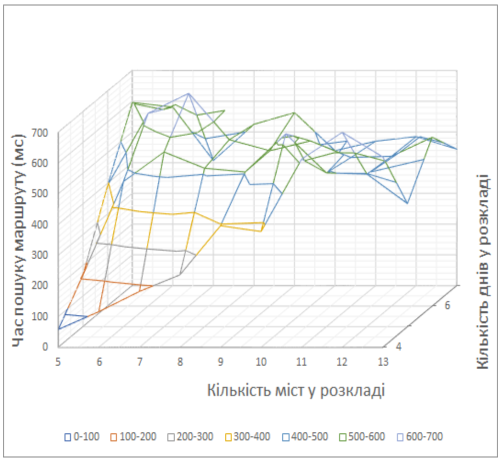

РИС. 3. Залежність між розмірами таблиці розкладу і часом пошуку маршруту

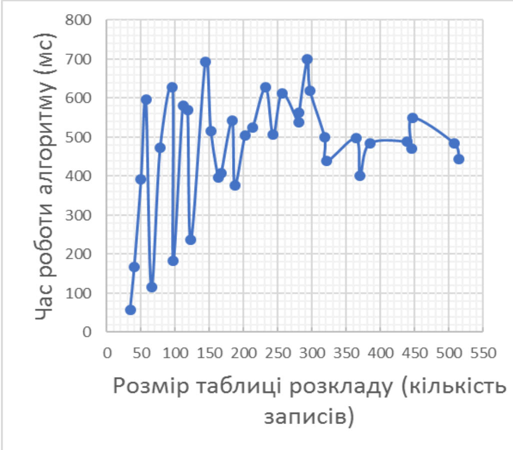

РИС. 4. Залежність між розмірами таблиці розкладу і часом пошуку маршруту

Результати розрахунків показали, що з ростом розміру бази значно збільшується час розрахунку, але не завжди прямо пропорційно (оскільки структура графу і кількість рейсів на дугу впливають на кількість розглядуваних міток). Обчислення здійснювались на комп'ютері з центральним процесором Intel Core i7, частотою 3.60 Ghz, 4 ядрами, RAM – 32 Gb DDR3 і операційною системою Windows 10. Алгоритм реалізовано на мові C#.

Висновки. В роботі описано алгоритм міток для розв'язування динамічної задачі пошуку оптимального маршруту при заданому розкладі авіаперельотів з урахуванням часового вікна і кількості пунктів відправлення/прибуття у розкладі. Результати обчислень показали, що швидкість росту часу розрахунків не є прийнятною для розв'язування задач у реальному часі для практичних обсягів бази авіаперельотів, але алгоритм може застосовуватись на етапі попередньої обробки даних і порівняння з іншими наближеними алгоритмами розв'язання.

В подальших планах – розробка більш складних метаевристичних алгоритмів розв'язування [6], таких як міметичний алгоритм і алгоритм оптимізації мурашиними колоніями з урахуванням описаних умов виділеного типу задач, а також дослідження та порівняння точності отримуваних результатів і швидкодії алгоритмів.

А.И. Павленко

РЕШЕНИЕ МНОГОКРИТЕРИАЛЬНОЙ ЗАДАЧИ ПОИСКА ОПТИМАЛЬНОГО ПУТИ В ДИНАМИЧЕСКИХ СЕТЯХ АЛГОРИТМОМ МЕТОК

Предложен подход к использованию алгоритма меток для решения задач поиска оптимального пути в динамической сети с учетом расписания авиаперелетов, общей стоимости билетов маршрута, временного окна. Приведено описание разработанного алгоритма и оценка эффективности с использованием реальных данных авиаперелетов.

A.I. Pavlenko

MULTICRITERIAL OPTIMAL PATH FINDING PROBLEM IN TIME-DEPENDENT NETWORKS USING LABEL SETTINGS ALGORITHM

The article considers an approach of using label settings algorithm to solve the problem of finding path in dynamic network taking into account air flights schedule, costs of tickets and time window. Given a description of the developed algorithm and performance evaluation on real data of air flights.

1. *Гуляницький Л.Ф.* Динамічна задача пошуку найкоротшого шляху з додатковими умовами при побудові маршруту авіаперельотів. *Математичні та інформаційні моделі в економіці*. 2015. № 2. С. 39 – 50.
2. *Павленко А.І.* Застосування залежних від часу задач пошуку найкоротшого шляху. *Журнал обчислювальної та прикладної математики*. 2015. № 1. С. 24 – 37.
3. *Zografos K.G., Androutsopoulos K.N.* Algorithms for Itinerary Planning in Multimodal Transportation Networks. *IEEE Transactions on Intelligent Transportation Systems*. 2008. Vol. 9, Issue 1. P. 175 – 184.
4. *Гуляницький Л.Ф.* Моделювання залежних від часу проблем часу проблем пошуку оптимальних маршрутів: огляд *Математичне моделювання в економіці*. 2017. № 1.
5. *Пошукова система SkyScanner* [Електронний ресурс] / Режим доступу: https://www.skyscanner.com (дата звернення 25.02.2017 р). – Назва з екрана.
6. *Гуляницький Л.Ф., Мулеса О.Ю.* Прикладні методи комбінаторної оптимізації: навч. посіб. К.: Видавничо-поліграфічний центр "Київський університет", 2016. 142 с.

Одержано 15.02.2017