Національний технічний університет України
"Київський політехнічний інститут імені Ігоря Сікорського"

Теорія складності

Дослідження $coNP$-повної задачі

Проблема негамільтонового циклу

ФІ-13 Дідух Максим

Фізико-технічний інститут

Кафедра математичних методів захисту інформації

2022

Дослідження $coNP$-повної задачі

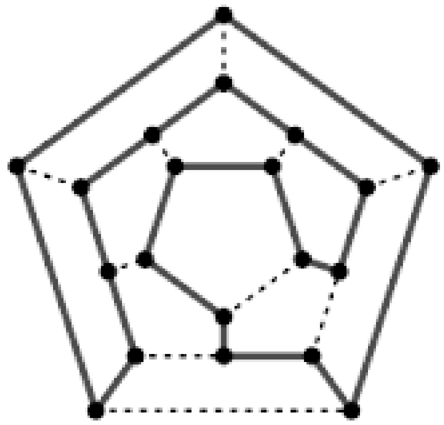
7 грудня 2022

Зміст

1 Вступ

Граф G називається *гамільтоновим*, якщо існує цикл, що містить кожну вершину рівно один раз. Такий цикл носить назву гамільтонового. Також слід зазначити, що існує поняття гамільтоновго шляху (це знадобиться при доведенні складності): граф G має *гамільтонів шлях* з вершини s у вершину t, якщо є марштрут між цими вершинами, який містить усі вершини рівно один раз.

Взагалі, це поняття пішло від Вільям Гамільтон (англ. *William Hamiltonian*), який вигадав не дуже вдалу гру під назвою "ікосіанська гра"(англ. *"icosian game"*), завданням якої був пошук гамільтонового циклу на додекаедричному графі (і можливо на його підграфах).



мал. 1 *гамільтоновий цикл на додекаедрі* change resolution

Хоча означення гамільтонового графу дуже схоже з означенням ойлерового графу, виявляється, що ці дві концепції поводяться досить по-різному. Якщо теорема Ойлера дає нам чіткий критерій ойлеровості, то для гамільтонових графів немає аналогічного твердження. Як з'ясувалось, перевірка графу на наявність гамільтонового циклу є NP-повною задачею.

1.1 Постановка задачі

Дано: неорієнтований граф $G = (V, E)$, де $V = \{v_1, v_2, \dots v_n\}$ - множина вершин, $E = \{e_1, e_2, \dots v_k\}$ - множина ребер. Перевірити, що даний граф не має циклу гамільтона. Цю задачу будемо позначати **NON-HAM-CYCLE** .

1.2 Історія виникнення та наявні її модифікації

НЕ ПОДОБАЄТЬСЯ

Проблема негамільтонівського циклу, або **NON-HAM-CYCLE** — досить відома задача, яка виникає, коли граф містить цикли, які не містять усіх вершин графа. Вперше була сформульована у 1930 році математиком Карлом Менгером і з тих пір стала важливою областю досліджень у теорії графів.

Початкова мотивація Менгера для введення проблеми полягала в тому, щоб вирішити більш загальну проблему пошуку гамільтонового циклу в графі. Тобто шлях, який відвідує кожнну вершину на графі рівно один раз, перш ніж повернутися до початкової точки. Знайти гамільтонів цикл може бути дуже важко, і Менгер хотів знайти простішу версію проблеми, яка все ще мала цікаві властивості.

Він вигадав ідею негамільтонових циклів, які є циклами, які не містять усіх вузлів на графі. Ці цикли все ще можуть містити деякі з вузлів, але не всі з них. Наприклад, трикутний графік матиме гамільтонів цикл (1-2-3-1), але також і негамільтонів цикл (1-2-1).

Спостереження Менгера відкрили цілу нову область дослідження теорії графів, у якій науковці досліджували властивості, алгоритми та застосування негамільтонових циклів. Однією з головних тем цього дослідження було визначення мінімальної кількості негамільтонових циклів, які можуть існувати в графі. Ця робота призвела до розробки проблеми комівояжера, яка передбачає пошук найкоротшого маршруту між набором міст.

Проблема негамільтонового циклу також використовувалася для моделювання комп'ютерних мереж, генетичних алгоритмів і роботизованої навігації. Проблема продовжує бути активною областю досліджень і продовжує генерувати нові ідеї в теорії графів.

2 Практичне застосування

Негамільтонові цикли мають широкий спектр застосування, оцінити вплив цього поняття та теорії, що побудувалась навколо нього неможливо, тому я спробую навести лише деякі (очевидні та не дуже) приклади застосування:

- Мережі
  - Проектування мереж: негамільтонові цикли можна використовувати для проектування надійних мереж, у яких існує кілька надлишкових шляхів між двома точками в мережі.
  - Проектування мережі: негамільтонові цикли часто використовуються при проектуванні мереж, де метою є створення шляху, який відвідує всі вузли мережі, не проходячи через жодний вузол більше одного разу.
  - Маршрутизація по мережі: негамільтонові цикли можна використовувати для маршрутизації інформації через мережу комп'ютерів. Це особливо актуально для комп'ютерних мереж, які містять вузли з різною потужністю або функціями. У цьому випадку можна використовувати негамільтонів цикл, щоб знайти шлях через мережу, який використовує переваги окремих вузлів, мінімізуючи затримку та перевантаження.
  - Маршрутизація: негамільтонові цикли зазвичай використовуються в алгоритмах мережевої маршрутизації, таких як алгоритм Беллмана-Форда, який знаходить найкоротший шлях між двома вузлами в мережі, спочатку будуючи негамільтонів цикл.
  - Проектування мереж: негамільтонові цикли можна використовувати для створення ефективних мереж, таких як комунікаційні мережі, транспортні мережі та ланцюги поставок. Використовуючи підхід негамільтонового циклу, ці мережі можуть бути більш ефективними, ніж традиційні гамільтонові цикли.
- Аналіз даних
  - Аналіз даних: негамільтонові цикли можна використовувати для виявлення закономірностей у наборах даних, допомагаючи визначити зв'язки між різними частинами даних.
  - Машинне навчання: негамільтонові цикли також використовуються в алгоритмах машинного навчання, таких як нейронні мережі та опорні векторні машини, для визначення закономірностей і прогнозування.
  - Візуалізація даних: негамільтонові цикли корисні для візуалізації даних у формі графіків, наприклад, коли використовується силово-спрямований алгоритм компонування графа *(force-directed graph layout algorithm)*. Такі алгоритми, використовують концепцію сил «відштовхування» та «притягання» для розташування вузлів у межах графа. Гамільтонів цикл не обов'язково є ідеальним для такого виду візуалізації, оскільки його ребра можуть стати занадто стиснутими, якщо є лише кілька вузлів або якщо вузли розташовані дуже близько один до одного. Негамільтонові цикли забезпечують більшу гнучкість і краще підходять для відображення складних даних у візуально цікавий та інформативний спосіб.
  - Розпізнавання шаблонів: негамільтонові цикли можна використовувати для пошуку шаблонів або трендів у великих наборах даних.
- Криптографія
  - Негамільтонові цикли також використовуються в певних криптографічних системах, таких як обмін ключами Діффі-Хеллмана та криптографія еліптичної кривої, для безпечної передачі даних через Інтернет.
- Планування та логістика
  - Планування: негамільтонові цикли можна використовувати для планування завдань у розподілених системах, оскільки вони забезпечують ефективний спосіб мінімізації загального часу, необхідного для виконання всіх завдань.
  - Планування: негамільтонові цикли можна використовувати для планування завдань на паралельному процесорі, наприклад планування завдань на роботі-мануалі на виробничому підприємстві.

- Планування: негамільтонові цикли можна використовувати для створення оптимальних розкладів для складних завдань, таких як планування маршрутів польоту або оптимізація операцій ланцюга поставок.
- Оптимізація логістики: негамільтонові цикли можна використовувати для вирішення проблеми комівояжера в оптимізації логістики, де метою є знайти найкоротший маршрут для транспортного засобу доставки серед кількох зупинок.
- Планування шляху: негамільтонові цикли також можна використовувати для планування шляху, особливо коли на шляху є перешкоди. Цикл можна використовувати для уникнення цих об'єктів під час пошуку найкоротшого маршруту між двома точками.
- Маршрутизація транспортного засобу: негамільтонові цикли можна використовувати в задачах маршрутизації транспортного засобу, де метою є знайти найефективніший маршрут для руху транспортного засобу між набором місць.

• Обробка зображень

- Обробка зображень: алгоритми на основі графів з негамільтоновими циклами використовуються при сегментуванні зображень на області чи об'єкти.
- Обробка зображень: негамільтонові цикли також використовуються в задачах обробки зображень, таких як виявлення країв на зображеннях і визначення форм на цифрових фотографіях.
- Мистецтво та дизайн: негамільтонові цикли можна використовувати для створення естетично привабливих візерунків у мистецтві та дизайні.

• Робототехніка

- Роботна навігація: негамільтонові цикли створюють основу для досліджень певного середовища роботами, одночасно уникаючи перешкод.
- Робототехніка: роботи можуть використовувати негамільтонові шляхи для пересування з місця на місце, уникаючи перешкод і досліджуючи навколишнє середовище.

• Розфарбування графів

- негамільтонові цикли можна використовувати в алгоритмах розфарбовування графів, де метою є розфарбувати всі вершини графа без будь-яких двох суміжних вершин, які мають однаковий колір.
- негамільтонові цикли можна використовувати в задачах розфарбовування, щоб розділити граф на два або більше менших підграфів з оптимізованим балансом кольорів.

• Інше

- Промислові процеси: негамільтонові цикли використовуються в промислових процесах, таких як хімічне машинобудування та автоматизація. Наприклад, їх можна використовувати для оптимізації маршрутів постачання матеріалів і виробництва на заводах.
- Секвенування ДНК: негамільтонові цикли використовуються в алгоритмах секвенування ДНК. Алгоритми використовують підхід негамільтонового циклу для аналізу відносних положень нуклеотидів у молекулі ДНК.
- Проблеми оптимізації: негамільтонові цикли також можна використовувати для вирішення проблем оптимізації, таких як проблема комівояжера. На відміну від рішення, яке ґрунтується на гамільтоновому циклі, який гарантовано знаходить оптимальне рішення, але є дорогим з точки зору обчислень, негамільтонів цикл часто можна використовувати для знаходження майже оптимального рішення за менший час.

3 Доведення складності

Доведемо, що **NON-HAM-CYCLE** є $coNP$-повною. Для цього нам знадобиться довести додаткове твердження.

Твердження 1: задача перевірки графа на наявність гамільтонового циклу належить класу NP-повних задач (*далі:* **HAM-CYCLE**)

■

Спочатку покажемо, що задача перевірки орієнтованого графу на наявність гамільтоновго шляху з вершини s у вершину t (далі **D-HAM-PATH**) є NP-повною.

1. **D-HAM-PATH** ϵ NP-повна?

Нехай G — орієнтований граф. Ми можемо перевірити чи потенційний шлях $s \to \ldots \to t$ є гамільтоновим за поліноміальний час. Тепер спробуємо побудувати поліноміальне зведення **3SAT** $\leq_p$ **D-HAM-PATH** , щоб закінчити доведення повноти.

а точно 'множник'?

Нехай $\phi = \bigwedge_{i=1}^{m} \phi_i$ має n змінних і m множників. Щоб спростити доведення, припустимо, що жоден множник в ϕ не містить змінної x_i і її заперечення $\bar{x}_i$.

1. Спочатку, для кожної змінної x_i, ми генеруємо $2m + 1$ вершину з назвою $v_{i,j}$ і додаємо орієнтовані ребра $(v_{i,j}, v_{i,j+1})$ і $(v_{i,j+1}, v_{i,j})$, для $0 \leq j \leq 2m$.

2. Далі ми з'єднуємо вершини, пов'язані з різними змінними, додаючи чотири спрямовані ребра $(v_{i,0}, v_{i+1,0})$, $(v_{i,0}, v_{i+1,2m+1})$, $(v_{i,2m+1}, v_{i+1,0})$ та $(v_{i,2m+1}, v_{i+1,2m+1})$.

3. Далі створюємо вершини під назвою c_j. Якщо x_i з'являється у множнику ϕ_j без доповнення, додаємо орієнтовані ребра $(v_{i,2j-1}, c_j)$ та $(c_j, v_{i,2j})$. Інакше, якщо x_i міститься з доповненням, то додаємо орієнтовані ребра $(v_{i,2j}, c_j)$ та $(c_j, v_{i,2j-1})$.

4. В кінці, ми додаємо дві додаткові вершини s і t. Після цього додаємо нові ребра $(s, v_{1,1})$, $(s, v_{1,2m+1})$, $(v_{n,1}, t)$ та $(v_{n,2m+1}, t)$. Граф згенерований G_ϕ (*див. рис.2*). add picture number2

Тепер, треба показати, що ϕ задовільна тоді та лише тоді, коли у графі G_ϕ є гамільтонів шлях з s у t. Припустимо, що ϕ - задовільна. Тоді ми можемо відвідати кожну вершину починаючи з s йдучи з $v_{i,0}$ до $v_{i,2m+1}$ з ліва на право, якщо x_i - позитивна, і з $v_{i,2m+1}$ до $v_{i,0}$ з права на ліво, якщо x_i - негативна, і закінчити в t після відвідування $v_{n,0}$ або $v_{n,2m+1}$. Крім того, кожна 'clause' вершина може бути відвідана, згідно з припущенням, що кожна 'clause' ϕ_i має декілька літерлів $l_i = x_k$ або $l_i = \bar{x}_i$, які є позитивними. Кожна вершина c_j може бути відвідана використовуючи ребра $(x_{k,2j-1}, c_j)$ або $(c_j, x_{k,2j})$ у першому випадку, і $(x_{k,2j}, c_j)$ і $(c_j, x_{k,2j-1})$ у другому випадку, оскільки шлях йде вправо на графі для позитивних змінних, і шлях йде вліво для негативних змінних. Тому G_ϕ має гамільтонів шлях, якщо ϕ задовільна.

І навпаки, припустимо, що G_ϕ має гамільтонів шлях (s, t), позначимо його P. Зазначимо, що G_ϕ без clause vertices є гамільтоновим, тому нам потрібно показати, що шлях не "заламається"після відвідування clause vertex. Якщо більш формальніше: потрібно показати, що якщо P відвідує $v_{i,2j-1}$, c_j, v у відповідному порядку, тоді $v = v_{i,2i}$. Нехай $v \neq v_{i,2i}$. Тоді зазначимо, що єдиними вершинами, які входять в $v_{i,2j}$ є $v_{i,2j-1}$, c_j, $v_{i,2j+1}$, а виходять з нього лише $v_{i,2j-1}$ та $v_{i,2j+1}$. Тому $v_{i,2j}$ має бути відвідана з $v_{i,2j+1}$, але тепер шлях замається і не може продовжуватись, оскільки $v_{i,2j-1}$ вже відвідано. Аналогічний аргумент показує, що P відвідує $v_{i,2j}$, c_j, v у відповідному порядку, тому $v = v_{i,2j-1}$. Отже, гамільтонів шлях в G_ϕ відвідує вершини в порядку від x_1 до x_n, чергуючи clause vertices вершини між змінними , пов'язаними з тією самою змінною. Тому, значення змінних x_i добре визначається, якщо помітити, у якому напрямку шлях йде через вершини x_i у графі G_ϕ. За побудовою це присвоєння буде задовільняти ϕ, оскільки воно робити літерал у кожній clause позитивним. fix brackets bug

Зведення відбувається за $O(mn)$ час, який є поліномом довжини вхідних даних.

2. **HAM-CYCLE** ϵ NP?

Якщо довільна задача, належить класу NP, тоді, маючи «сертифікат», який є розв'язком зіеї задачі та екземпляр проблеми (граф G і додатне ціле k, у цьому випадку), ми зможемо верифікувати (перевірити, чи надане рішення

правильне чи ні) сертифікат за поліноміальний час. Сертифікат — це послідовність вершин, що утворюють гамільтонів цикл у графі. Ми можемо верифікувати розв'язок, перевіривши, що всі вершини належать графу, і, що кожна пара вершин, що належать розв'язку — суміжна.
Це можна зробити за поліноміальний час, тобто $O(V+E)$, ось псевдокод верифікації сертифікату для графу $G(V,E)$:

```
value = 1 fix spacing

для кожної пари {u, v} у підмножині V‘:

  перевірити, що між цими вершинами є ребро

  якщо ребра немає, то повернути 0 і завершити цикл

якщо value = 1:

  то повернути 1 (розв'язок коректний)

інакше:

  повернути 0 (розв'язок неправильний)
```

3. **HAM-CYCLE** NP-складна?
Щоб довести, що **HAM-CYCLE** є NP-складною, ми повинні звести вже відому NP-складну задачу до даної. Побудуємо зведення від **D-HAM-PATH** до **HAM-CYCLE** . Кожен екземпляр задачі **D-HAM-PATH** містить граф $G = (V, E)$, який може бути конвертований в задачу **HAM-CYCLE** , яка містить граф $G' = (V', E')$. Граф G‘ будемо будвати таким чином:

1. $V‘ = V \cup \{v_{new}\}$, де v_{new} - це додаткова вершина, яка поєднана ребрами з усіма вершинами графу G. $E_{new} = \{\,(v_{new},\, v_i)\,|\,\forall v_i\,\epsilon\,V\}$ - множина усіх ребер, що поєднують вершину v_{new} з графом G.
2. $E‘ = E \cup E_{new}$
3. Припустимо, що граф G містить гамільтонів шлях P, який починається у випадковій вершині, позначимо її як v_{start} та закінчується у вершині v_{end}. Тепер, оскільки ми поєднали усі вершини з множини V з вершиною v_{new}, ми можемо розширити гамільтонів шлях P о гамільтонового циклу, використовуючи ребра $v_{end}\,v_{new}$ та $v_{new}\,v_{start}$ відповідно. Тепер граф G‘, який містить усі вершини рівно один раз.
4. Ми припускаємо, що граф G‘ містить гамільтонів цикл, який проходить через усі вершини, включаючи v_{new}. Тепер, щоб перетворити цей цикл на гамільтонів шлях, ми видаляємо усі ребра (у циклі), що містять вершину v_{new}. Отриманий шлях буде покривати усі вершини з множини V і будуть робити це рівно один раз. (... рис 3 ...)

Тому ми можемо стверджувати, що граф G' містить гамільтонів цикл, якщо граф G містить гамільтонів шлях. Отже, довільний екземпляр задачі **HAM-CYCLE** зводить до екземпляру задачі **D-HAM-PATH** . З чого можна зробити висновок, що **HAM-CYCLE** є NP-складною.

Підсумовуючи все вище сказане, маємо, що **HAM-CYCLE** належить класу NP і є NP-складною, тому можна зробити висновок, що **HAM-CYCLE** є NP-повною.

■

Тепер, довівши *Твердження 1*, можемо виконати фінальні викладки. Маємо, що **HAM-CYCLE** належить класу NP та **NON-HAM-CYCLE** є доповненням задачі **HAM-CYCLE** , отже задача **NON-HAM-CYCLE** належить класу $coNP$. Також, врахувавши, що **HAM-CYCLE** є NP-повною, можна зробити висновок, що **NON-HAM-CYCLE** є $coNP$-повною.

4 Розв'язок

4.1 Наявні методи розв'язку

4.2 Наявні ефективні часткові розв'язки задачі та модифікацій

5 Список використаної літератури

чи доречно лишати лінки?

Література

[1] Adrian She, *Hamiltonian Path is NP-Complete*, 2020.

[2] джерело 1

[3] джерело 2