бути менша за 1. Якщо немає коренів на інтервалі збіжності, то треба застосовувати інші методи [3].

Слід зазначити, що при розв'язку нелінійних рівнянь керуються такими критеріями як: збіжність методу, діагностика кратності кореня та оцінка зупинки ітерацій. Тому, при використанні методів розв'язку нелінійних рівнянь одним із основних критеріїв є час збіжності методу, що у свою чергу визначає машинний час на розв'язання заданого рівняння.

Підсумовуючи, нелінійне рівняння може мати безліч коренів. Тому для застосування чисельного методу необхідно вказати відрізок, на якому існує тільки один корінь. Усі чисельні методи мають різну швидкість збіжності і є ефективними для певних нелінійних рівнянь з одним невідомим. Методи допомагають швидко знайти розв'язок та зробити усі розрахунки легшими. Тому усі методи важливі та ефективні, але використовуються по різному в залежності нелінійного рівняння.

Список літератури.

1. *Колесницький О.К., Арсенюк І.Р., Месюра В.І. . Загальні відомості чисельних методів та нелінійних рівнянь. 2017. 130 с.*
2. *Богач І.В. Дослідження чисельних методів. 2018. 30 с.*
3. *Волонтир Л.О., Зелінська О.В., Потапова Н.А., Чіков І.А. Чисельні методи розв'язування нелінійних рівнянь. Навчальний посібник. 2020. 320 с.*

УДК 004.6

Суліма В. К., студент 1 курсу
спеціальності 122 «Комп'ютерні науки»
Науковий керівник:
Гончар В. М., асистент
кафедри інформаційних технологій

АЛГОРИТМИ ВИЗНАЧЕННЯ МІНІМАЛЬНОЇ КІЛЬКОСТІ КОЛЬОРІВ, ДЛЯ РОЗФАРБУВАННЯ ГРАФУ

Донецький національний університет імені Василя Стуса, м. Вінниця

Розфарбування графів є одним із важливих понять у теорії графів і має широкий спектр застосування. Основна ідея полягає в присвоєнні кожної вершини графа певного кольору таким чином, щоб суміжні вершини мали різні кольори.

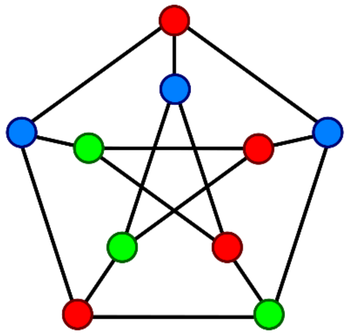

Рисунок 1 – Розфарбований граф

У математичних і комп'ютерних представленнях типово використовувати кілька перших позитивних або невід'ємних цілих чисел як «кольори», отже розфарбування графів не означає обов'язковість призначення саме кольорів для позначення вершин, а часто використовуються інші мітки, але історично ці мітки почали теж називати кольорами.

Фарбування графів є інструментом вивчення властивостей графів. Вони дозволяють виявляти та класифікувати різні структури в них, кольори вершин можуть вказувати на різні групи чи зв'язки між вершинами. Деякі класи графів, такі як планарні графи або дерева, мають особливі властивості, які можуть бути виділені за допомогою забарвлення графів, що дозволяє дослідити та краще зрозуміти структуру та властивості різних класів графів.[1]

Хроматичне число графа – одне із фундаментальних понять теорії графів. Воно позначається як χ(G) і визначається як мінімальна кількість кольорів, необхідне для розфарбовування вершин графа G таким чином, щоб жодні дві суміжні вершини не мали однакового кольору. Це число дозволяє класифікувати графи на основі їхньої розфарбовуваності. Графи з однаковим хроматичним числом мають схожі властивості та структуру, що дозволяє зробити висновки про їх комбінаторні характеристики. Це має застосування у завданнях планування, оптимізації ресурсів та розробки ефективних алгоритмів.

Існує декілька алгоритмів визначення мінімального числа кольорів, для розфарбування графа. Один з найбільш відомих алгоритмів – жадібний алгоритм – це найпростіший та інтуїтивно зрозумілий підхід задля цієї задачі. Він перебирає вершини графа в певному порядку та призначає найменший доступний колір кожній вершині. Порядок обходу вершин може впливати на кількість кольорів. Кроки цього алгоритму такі:

- Створюємо порожній набір кольорів.
- Відсортовуємо вершини графа у порядку спадання ступеня вершини (кількість суміжних вершин).
- Проходимось по відсортованими вершинами і призначаємо кожній вершині мінімальний доступний колір, який не використовується у її суміжних вершин.

- Після розфарбовування всіх вершин, кількість використаних кольорів буде мінімальною кількістю кольорів, необхідних для розфарбування графа.

Важливо, що алгоритм жадібного розфарбування не завжди дає абсолютно мінімальну кількість кольорів, але забезпечує наближене рішення.

Існують і точніші алгоритми, такі як алгоритм зворотного відстеження – систематично досліджує всі можливі кольори графа, поки не знайде мінімальну кількість кольорів. Він використовує стратегію пошуку в глибину (DFS) для обходу графа та призначає кольори вершинам одну за одною. Якщо він досягає точки, де не може пофарбувати вершину, не порушуючи обмежень на забарвлення, він повертається назад і пробує інший колір. Кроки цього алгоритму такі:

- Спочатку всі вершини графа не пофарбовані.
- Рекурсивно перебираються вершини графа:
  - Для кожної вершини:
  - Намагаємося пофарбувати її у кожен із доступних кольорів.
  - Перевіряємо, чи поточне фарбування є допустимим, тобто не порушується умова, що суміжні вершини не повинні мати однакового кольору.
  - Якщо фарбування допустиме, переходимо до наступної непофарбованої вершини.
  - Якщо фарбування не допустиме, повертаємось назад і перевіряємо інший колір попередньої вершини.
  - Збільшуємо кількість доступних кольорів і повторюємо процес, за необхідності.
- Якщо всі вершини були пофарбовані, то поточна кількість використаних кольорів буде хроматичним числом графа.

Алгоритм зворотного відстеження гарантує оптимальне забарвлення, але може бути обчислювально дорогим для великих графів. Різні алгоритми пропонують різні компроміси між ефективністю та оптимальністю. Вибір алгоритму залежить від розміру та характеристик графа, а також від бажаного рівня точності та доступних обчислювальних ресурсів.[2]

Але, не для кожного типу графів необхідно застосовувати алгоритми пошуку мінімальної кількості кольорів, оскільки для окремих випадків, цього можна уникнути. У циклічному графі, якщо число вершин у графі циклу парне, то його хроматичне число = 2. Якщо кількість вершин у графі циклу непарна, то його хроматичне число = 3. В планарний графах – хроматичне число будь-якого графа менше або дорівнює 4. У Повному графі – потрібно стільки різних кольорів, скільки вершин у даному графі. При роботі з дводольними графами – хроматичне число завжди буде рівно 2, а оскільки кожне дерево є дводольним графом, то число для дерева з будь-якою кількістю вершин теж 2.[3]

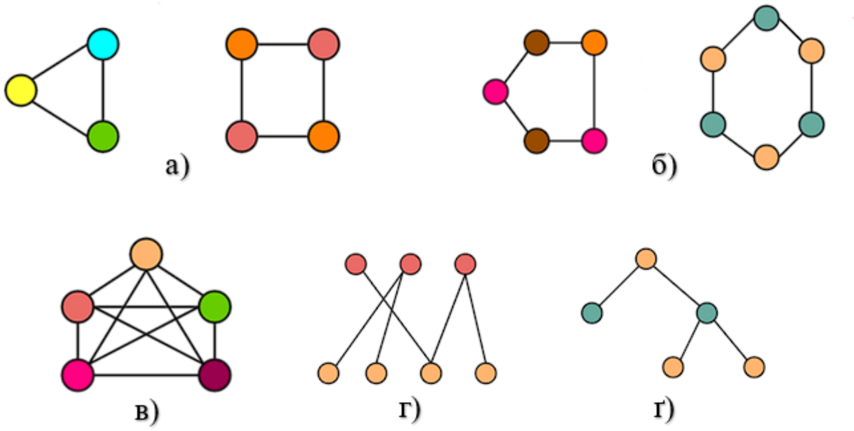

Рисунок 2 – а) розфарбований граф циклу; б) розфарбований планарний граф; в) розфарбований повний граф; г) розфарбований дводольний граф; ґ) розфарбоване дерево;

Отже, провівши дослідження було виявлено, що фарбування графів та мінімізація використання кольорів у ньому це дуже важлива задача в теорії графів, не існує повноцінних алгоритмів для виконання цієї задачі, тому використовуються ситуативні алгоритми, які ефективно можуть працювати на обмеженій кількості графів та в певних випадках можуть бути не зовсім точними. Розвиток індустрії обчислювальних машин, винайдення нових алгоритмів та їх ефективне поєднання, можуть трохи послабити цю проблему, але ці дослідження необхідні, оскільки застосування таких графів у різних галузях, таких як планування розкладу, розподіл регістрів процесора для великих задач, аналіз структури мережі, виявлення спільнот або угруповань та прогнозування взаємодій, тощо, є невід'ємними завданнями у нашому сьогоденні.

Список джерел

1. *Graph coloring and applications, Url: https://medium.com/analytics-vidhya/graph-coloring-and-applications-2157912f505d*
2. *Graph Coloring Problem, Url: https://www.interviewbit.com/blog/graph-coloring-problem/*
3. *Graph Coloring in Graph Theory | Chromatic Number of Graphs, Url: https://www.gatevidyalay.com/graph-coloring-chromatic-number/*