

К.т.н., доцент Зорін Ю.М., студентка Попович К.Г.

**Національний технічний університету України
«Київський політехнічний інститут»**

**АЛГОРИТМ ІЗ ЗАБОРОНАМИ ДЛЯ РОЗВ'ЯЗКУ ЗАДАЧІ
k -РОЗФАРБУВАННЯ ГРАФА**

Abstract

*Yuri Zorin, Assoc. Prof., PhD; Kateryna Popovych, student
A Tabu Search algorithm optimization for k-coloring problem*

The paper describes improvements of a basic Tabu Search algorithm for graph k-coloring problem. The description of proposed algorithm, its advantages and comparison between basic algorithm and other improved algorithms for k-coloring are provided. The prospects for further research and modification are proposed.

Вступ

Задача розфарбування графу у k кольорів є комбінаторною задачею, що за часом виконання відносять до класу NP-складних задач [1].

Нехай $N_k = \{1, 2, \dots, k\}$, відображення $f : V \rightarrow N_k$ є розфарбуванням графу

$G = (V, E)$, де для вершини $v \in V$ число $f(v)$ є колір. Розфарбування f графу G правильне, якщо дві довільні суміжні вершини мають різні кольори.

Мінімальна кількість кольорів, для якої існує правильне розфарбування графу G , є хроматичним числом графа G й позначається $\chi(G)$.

Задача k -розфарбування полягає у правильному розфарбуванні графу у k кольорів, де k - хроматичне число, або максимально до нього наближене.

Алгоритм із заборонами [2] отримав свою назву через те, що з метою уникнення локальних екстремумів і запобігання зациклюванню він використовує спеціальний список, у якому зберігаються трансформації поточного розв'язку, виконані останніми. Ці трансформації протягом певного часу є «забороненими», тобто не можуть використовуватися в процесі пошуку розв'язку.

Постановка задачі

Метою роботи є розробка алгоритму із заборонами та його дослідження за допомогою модифікацій основних кроків: пошуку та безпосереднього перефарбування кожної наступної вершини, перезапуску алгоритму з метою виходу із локального екстремуму, а також встановлення експериментальним шляхом часу перебування комбінації під заборонаю, при якому час виконання задачі найменший.

Базовий алгоритм із заборонами

Алгоритм полягає у виконанні таких кроків.

1. Пошук вершини з максимальною кількістю конфліктів.
2. Вибір комбінації з вершини та кольору для перефарбування, що максимально зменшує загальну кількість конфліктів.

2.1 Якщо отримана загальна кількість конфліктів менша ніж поточна, результат, отриманий на даному кроці, зберігається як найкращий.

2.2 Якщо така комбінація не знайдена, виконується пошук комбінацій з вершин і кольорів серед «заборонених».

2.2.1 Якщо така комбінація не знайдена, це означає, що досягнутий локальний екстремум, тому виконується перезапуск алгоритму, який полягає в перефарбуванні випадкової вершини у випадковий колір.

3. Якщо не виконано умови завершення алгоритму, тобто загальна кількість конфліктів більша за нуль і не перевищено максимальну кількість ітерацій, повторюються пп. 1-3.

Кожна з прийнятих комбінацій заноситься на певний час у «список заборонених».

Вдосконалення алгоритму

Перефарбовування на кожному кроці тільки однієї вершини для графів з великою кількістю вершин є значним недоліком алгоритму, оскільки на таку оптимізацію витрачається багато часу і ресурсів, порівняно з досягнутим ефектом. Тому, на основі даних, отриманих після виконання пп. 1-3, доцільно перефарбовувати деяку кількість вершин одразу. Було досліджено два варіанти.

1. Перефарбування кожної з вершин, які мають максимальну на даному етапі кількість конфліктів у найперспективніший, з точки зору окремого впливу цієї вершини на загальну картину, колір.

2. Перефарбування послідовно кожної наступної вершини із максимальною кількістю конфліктів у колір, який поліпшує загальну ситуацію на даному кроці, враховуючи зміни, які відбулись при перефарбуванні попередніх.

Що стосується перезапуску, то перефарбовування тільки однієї вершини призводить до того, що деякий час алгоритм залишається в околі вже досягнутого локального мінімуму. А при повному перефарбуванні не використовується результат, отриманий на попередніх ітераціях. Тому були розглянуті наступні варіанти перезапуску.

1. Перефарбування одразу декількох випадково вибраних вершин, кількість яких залежить від числа поточних конфліктів, у довільний колір.

2. Перефарбування одразу декількох випадково вибраних вершин у дозволені на даному кроці кольори згідно «списку заборонених комбінацій».

3. Перефарбовування тільки конфліктних вершин із випадково вибраних.

Час знаходження комбінації у «списку заборонених» визначається поточною кількістю конфліктів, помноженою на сталу величину λ . В базовій реалізації $\lambda=0.5$ для всіх графів. Тому для кожного графу розглядаються всі λ із діапазону від нуля до одиниці з кроком 0.05, з метою знаходження величини, при якій загальна кількість ітерацій на розфарбування найменша.

Дослідження ефективності запропонованих модифікацій

Обидва варіанти збільшення кількості вершин, що перефарбовуються на одній ітерації, привели до зменшення часу виконання алгоритму за рахунок більш інтенсивного «спуску».

Також було виявлено, що вибір кольору з найменшим номером серед тих, що зменшують кількість конфліктів для конкретної вершини при послідовній обробці декількох вершин з однаковою кількістю конфліктів, зменшує час виконання задачі. Причина полягає в збільшенні щільності кольорів на етапі «активному спуску». Це дає більшу різноманітність варіантів для перефарбовування при наближенні до локального екстремуму.

На кроці примусового перезапуску випадкове перефарбування вершин, кількість яких залежить від загальної кількості конфліктів, прискорило виконання алгоритму. Число вершин при цьому вибиралося із діапазону від 1 до поточної кількості конфліктів, поділеної на два. Якщо кількість конфліктів близька до нуля, до верхньої границі діапазону додавалася величина, що дорівнює кількості вершин, поділений на чотири.

Врахування «списку заборонених комбінацій» при перезапуску негативно відобразилось на кількості ітерацій, необхідних для правильного розфарбування. Це пояснюється тим, що не змінюється область, що досліджувалась останньою.

Варіант з перефарбуванням тільки конфліктних вершин також негативно вплинув на кінцевий результат.

Експерименти зі змінення тривалості знаходження комбінації під забороною показали, що величина λ , при якій середня кількість ітерацій на розв'язання задачі найменша, для кожного графу своя, і залежить від ступеня зв'язності графа.

Висновки

В процесі тестування, результати якого відображені в табл.1, використані графи з набору тестів DIMACS[3].

Запропонований алгоритм, достатньо швидко знаходить хроматичні числа всіх досліджуваних графів з невеликою кількістю вершин (менше за 200) та графів з великою кількістю (більше ніж 200), в яких середній ступінь вершин менший за хроматичне число.

Таблиця 1

Результати розфарбування графів DIMACS

G	Кіл-сть вершин	Кіл-сть ребер	χ (G)	Результат для χ (G)	k_{\min}
DSJC125.9	125	13922	44	+	44
DSJC250.5	250	31366	28	-	30
myciel7	191	2360	8	+	8
miles1000	128	3216	42	+	42
mulsol.i.1	197	475	49	+	49
queen8_8	64	728	9	+	9

В подальшому планується розробка гібридних алгоритмів, які можуть інтегрувати в себе запропонований алгоритм, дослідження залежності довжини «списку заборонених комбінацій» від зв'язності графа, пошук оптимальних параметрів перезапуску.

Література

1. *Карнаух Т.О., Ставровський А.Б.* Теорія графів у задачах: Навчальний посібник. – К.: Видавничо-поліграфічний центр "Київський університет", 2004. – 76 с.
2. *Glover F.* Tabu Search, Part I. //ORSA Journal on Computing №1, 1989, pp. 190-206.
3. DIMACS Graphs: BenchmarkInstances : <http://info.univ-angers/pub/porumbel/graphs/index.html>