КОМПЬЮТЕРНАЯ МАТЕМАТИКА

УДК 519.854

В.П. ШИЛО

# ПОДХОДЫ К РЕШЕНИЮ ЗАДАЧИ РАСКРАСКИ ГРАФА*

*Предложены два подхода к решению задачи раскраски графа. Проведено сравнительное исследование эффективности известных для данной задачи и разработанных алгоритмов, подтвердившее преимущества предложенных подходов.*



**Введение.** В данной работе предложены два подхода к решению задачи раскраски графа. Первый из них состоит в сведении двух типов таких задач к модели булевого квадратичного программирования без ограничений, для решения которой в [1] предложен эффективный подход, базирующийся на использовании метода глобального равновесного поиска (ГРП) [2]. Второй подход к решению рассматриваемых задач связан с известным утверждением – $k$-раскраске графа соответствует разбиение множества его вершин на $k$ независимых множеств. Вершины, входящие в независимые множества, окрашиваются в один цвет. Для решения задачи разбиения множества вершин графа в работе предложен приближённый алгоритм. Проведено сравнительное исследование эффективности известного из литературы [3] и данного алгоритмов, которое подтвердило преимущества последнего.

**Содержательная постановка**. Задача раскраски графа состоит в раскраске его вершин различными цветами таким образом, чтобы никакие две смежные вершины не были окрашены одинаково. Она возникает во многих важных реальных приложениях.

**Математические модели.** Выделяют две постановки задачи раскраски графа.

**Задача 1**. Пусть задан граф $G = G(V, E)$, где $V$ и $E$ – соответственно множества его вершин и ребер. Имеется $K$ цветов, в которые можно окрасить каждую вершину.

* Работа выполнена при частичной финансовой поддержке Украинского научно-технологического центра (грант № 4138).

Введем переменные $x_{ik}$, $i$=1 ,…, $|V|$, $k$=1 ,..., $K$, такие, что

$$x_{ik} = \begin{cases} 1, \text{ если } i\text{-я вершина окрашена } k\text{-м цветом,} \\ 0 - \text{ в противном случае,} \end{cases}$$

где $|V|$ – мощность множества $V$ .

Необходимо установить, можно ли раскрасить вершины графа, используя все $K$ цветов и соблюдая вышеперечисленные условия.

Математическая модель этой задачи состоит в нахождении таких значений переменных $x_{ik}$, которые удовлетворяют системе ограничений

$$\sum_{k=1}^{K} x_{ik} = 1 , \; i=1,..,|V|, \tag{1}$$

$$x_{ik} + x_{jk} \le 1 , \; (i, j) \in E , \; k=1 ,..., K, \tag{2}$$

$$x_{ik} = 0 \vee 1 , \; i=1 ,..., |V|, \; k=1 ,..., K, \tag{3}$$

или в установлении факта, что таких значений не существует.

Ограничение (1) означает, что каждая вершина графа может быть окрашена только одним цветом. Условие (2) указывает на то, что смежные вершины должны быть окрашены различными цветами.

**Задача 2**. Пусть задан граф $G = G(V, E)$, где $V$ и $E$ – соответственно множества его вершин и ребер. Имеется максимальное число $K$ цветов, в которые можно окрасить вершины графа (можно положить, например, $K = |V|$).

Введем переменные $x_{ik}$ и $y_k$, $i$=1 ,..., $|V|$, $k$=1 ,..., $K$, такие, что

$$x_{ik} = \begin{cases} 1, \text{ если } i\text{-я вершина окрашена } k\text{-м цветом,} \\ 0 - \text{ в противном случае,} \end{cases}$$

$$y_k = \begin{cases} 1, \text{ если } k\text{-й цвет используется в раскраске,} \\ 0 - \text{ в противном случае.} \end{cases}$$

Необходимо найти минимальное число цветов, которыми можно раскрасить вершины графа при выполнении вышеназванных ограничений.

Математическая постановка этой задачи формулируется таким образом:

найти

$$\min_{x,y} \sum_{k=1}^{K} y_k \tag{4}$$

при условиях

$$\sum_{k=1}^{K} x_{ik} = 1 , \; i = 1 ,..., |V|, \tag{5}$$

$$x_{ik} + x_{jk} \le 1 , \; (i, j) \in E , \; k = 1 ,..., K, \tag{6}$$

$$x_{ik} \le y_k\ ,\ i=1\ ,...,\ |V|,\ k=1\ ,...,\ K, \tag{7}$$

$$x_{ik} = 0 \vee 1\ ,\ i=1\ ,...,\ |\ V\ |,\ k=1\ ,...,\ K. \tag{8}$$

Отметим, что задачи 1 и 2 – *NP*-сложные.

**Сведение рассматриваемых задач к известной модели.** В работе [4] предложен способ сведения задач (1)–(3) и (4)–(8) к задаче булевого квадратичного программирования без ограничений. Следуя [4], это можно сделать такими преобразованиями. Рассмотрим следующую задачу:

минимизировать

$$f(x) = xQx \tag{9}$$

при ограничениях

$$Ax = b\ , \tag{10}$$

$$x_j = 0 \vee 1,\ j=1\ ,\dots,\ n. \tag{11}$$

где $Q$ – квадратная матрица порядка $n$, $A$ – матрица размерностью $m \times n$, $x$ и $b$ – соответственно $n$-мерный и $m$-мерный векторы.

Эта модель объединяет как линейный, так и квадратичный случаи. Действительно, поскольку в силу булевости переменных справедливо равенство $x_j^2 = x_j$ , $j=1,\dots, n$, то диагональная матрица $Q$ позволяет представить линейную функцию цели в виде (9). Для произвольной положительной штрафной константы $P$ имеем

$$\tilde{f}(x) = xQx + P(Ax-b)^t(Ax-b) = xQx + xDx + c = x\tilde{Q}x + c\ .$$

Таким образом, выбрав соответствующее значение $P$ (что всегда возможно), сведем задачу булевого квадратичного программирования с линейными ограничениями вида (9)–(11) к следующей задаче:

минимизировать

$$\tilde{f}(x) = x\tilde{Q}x \tag{12}$$

при условиях

$$x_j = 0 \vee 1,\ j=1\ ,\dots,\ n. \tag{13}$$

Преобразование задачи (9)–(11) в задачу (12), (13) назовём первым преобразованием.

Пусть $E$ – некоторое множество пар индексов. Рассмотрим теперь такую задачу:

минимизировать

$$f(x) = xQx \tag{14}$$

при ограничениях

$$x_i + x_k \le 1,\ (i,k) \in E\ , \tag{15}$$

$$x_j = 0 \vee 1,\ j=1\ ,\dots,\ n. \tag{16}$$

Выбрав некоторую штрафную константу *P,* получим

$$\tilde{f}(x) = xQx + P \sum_{(i,k)\in E} x_i x_k = x\tilde{Q}x \, .$$

Назовем это преобразование вторым преобразованием. С его помощью задача (14)–(16) также сводится к задаче (12), (13).

**Подходы к решению поставленных задач.** Как видим, постановки (1)–(3) и (4)–(8) задачи раскраски графа с помощью первого и второго преобразований сводятся к задаче безусловной минимизации – квадратичного программирования с булевыми переменными вида (12), (13). Для решения последней задачи в работе [1] разработана модификация метода глобального равновесного поиска [2], использующая в качестве поискового алгоритма метод табу.

Следует отметить, что для решения задач вида (12), (13) до появления работы [1] лучшей считалась реализация алгоритма табу [5]. Результаты многочисленных экспериментальных расчётов показали, что для задач большой размерности алгоритм глобального равновесного поиска [1] лучше алгоритма табу [5] как по качеству решений, так и по времени их нахождения.

В работе [4] приводятся результаты вычислительных экспериментов, которые свидетельствуют о том, что задача раскраски графа успешно решается будучи сформулированной в виде  (12), (13). Постановка (12), (13) заключается во введении штрафов за нарушение условий (1), (2) или (5), (6). В силу простоты структуры ограничений (1) или (5) можно поддерживать их выполнение с помощью специально выбранной системы ходов локальных процедур оптимизации, используемых алгоритмом глобального равновесного поиска, стартуя при этом с начальной точки, которая удовлетворяет этим ограничениям. За нарушение ограничений (2) или (6) можно вводить штрафы, аналогичные тем, которые появляются в квадратичной модели (12), (13).

Таким образом, задачу (1)–(3) можно переформулировать в виде

минимизировать

$$\sum_{k=1}^{K} \sum_{(i,j)\in E} P\chi\left\{x_{ik} + x_{jk} > 1\right\} \tag{17}$$

при условиях

$$x_{ik} = 0 \vee 1, i = 1,\dots,|V|, k = 1,\dots, K, \tag{18}$$

$$\sum_{k=1}^{K} x_{ik} = 1, \ i = 1,\dots,|V|, \tag{19}$$

где $\chi\left\{x_{ik} + x_{jk} > 1\right\} = \begin{cases} 1, \text{ если } x_{ik} + x_{jk} > 1, \\ 0 \text{ — в противном случае;} \end{cases}$

$P$ – штрафная константа.

Для задач построения помехозащищенных кодов максимального объема представляет интерес следующая постановка:

минимизировать

$$\sum_{k=1}^{K}\left(\sum_{i=1}^{|V|} x_{ik}\right)^2 + \sum_{k=1}^{K}\sum_{(i,j)\in E} P\chi\left\{x_{ik}+x_{jk}>1\right\} \tag{20}$$

при условиях

$$x_{ik}=0\vee 1,\ i=1,...,|V|,\ k=1,...,K, \tag{21}$$

$$\sum_{k=1}^{K} x_{ik}=1,\quad i=1,...,|V|, \tag{22}$$

где $\chi\left\{x_{ik}+x_{jk}>1\right\}=\begin{cases}1, \text{ если } x_{ik}+x_{jk}>1,\\ 0 \text{ — в противном случае;}\end{cases}$

$P$ – штрафная константа.

Для решения задач (17)–(19) и (20)–(22) на базе общей схемы метода ГРП [6] разработаны оптимизационные алгоритмы.

Легко видеть, что множество вершин графа, окрашенных одним цветом, является независимым множеством. Поэтому задача раскраски графа может быть сведена к задаче разбиения множества вершин графа на независимые множества. Следовательно, подход к решению последней задачи можно использовать для решения задачи раскраски графа. Рассмотрим его.

Пусть задан граф $G=(V,E)$, где $V=\{v_1,..,v_n\}$ и $E=\{e_1,..,e_m\}$– соответственно множества его вершин и ребер, $n=|V|$, $m=|E|$. Подмножество вершин $I\subset V$ графа $G$ называется независимым, если никакие две его вершины не связаны ребром. Если для любого независимого множества $I$ выполняется соотношение $|I_{\max}|\geq|I|$, то независимое множество $I_{\max}$ называется максимальным независимым. Задача нахождения максимального независимого множества состоит в отыскании независимого множества с максимальным количеством вершин. Обозначим $\alpha(G)$ число элементов в максимальном независимом множестве $I_{\max}\left(\alpha(G)=|I_{\max}|\right)$.

Задача поиска максимального независимого множества вершин графа возникает также при нахождении кодов, корректирующих ошибки. Пусть $B^n$ – множество $n$-мерных векторов, координаты которых принимают значения 0 или 1. Рассмотрим алфавит $A=\{l_i\}, l_i\in B^k, i=1,...,2^k$, состоящий из всех двоичных векторов длиной $k$. Каждому вектору $l_i$ поставим в соответствие вершину графа $G(V,E)$. Предположим, что из-за ошибок при передаче информации вектор $l_i$ может переходить в один из элементов множества $R^i, i=1,...,2^k$.

Определим, что в графе $G(V, E)$ ребро $(v_i,\ v_j) \in E$ тогда и только тогда, когда $R^i \cap R^j \neq \varnothing$. Очевидно, что если найдено независимое множество в $G(V, E)$, то тем самым найден код, корректирующий ошибки.

Разбиением множества вершин графа на независимые множества является множество $(IS_1,...,IS_k)$ таких независимых множеств графа $G\ (V, E)$, что каждая вершина графа принадлежит только одному независимому множеству.

Особый интерес представляют следующие задачи разбиения:

- с минимальным числом $k$ разбиений,
- с максимальным значением $\sum_{i=1}^{k} |IS_i|^2$ нормы.

В первом случае задача разбиения множества вершин графа эквивалентна задаче раскраски графа. Второй случай важен [3] при построении помехозащищенных ассиметрических кодов ($Z$-канал). Поскольку эти задачи являются $NP$-сложными, важной проблемой является разработка эффективных приближенных алгоритмов их решения.

Рассмотрим один из возможных подходов к получению приемлемого разбиения множества вершин графа $G\ (V, E)$ на независимые множества. Вначале находится максимальное независимое множество $IS_1$ исходного графа $G\ (V, E)$. Затем из графа $G\ (V, E)$ исключаются вершины максимального независимого множества $IS_1$ с принадлежащими им ребрами, т.е. новый граф имеет вид $G_{V \setminus IS_1}(V \setminus IS_1, E_{V \setminus IS_1})$. Далее находится максимальное независимое множество $IS_2$ нового графа. Процесс продолжается до тех пор, пока не будет получено разбиение.

С нашей точки зрения этот подход можно улучшить следующим образом. Изменение состоит в нахождении (если это возможно) на каждом этапе заданного числа максимальных независимых множеств соответствующего графа. Далее строится специальный граф $N$, вершины которого соответствуют найденным максимальным независимым множествам. Между двумя вершинами этого графа имеется ребро тогда и только тогда, когда соответствующие максимальные независимые множества имеют общие вершины. Затем находится несколько максимальных независимых множеств графа $N$ и по некоторому правилу среди них выбирается лучшее (например, по количеству ребер, исходящих из соответствующих максимальных независимых множеств).

Схему предлагаемого алгоритма можно представить в виде следующих процедур:

while($|V| > 0$ )
{ for $j$ = 1 to $m$
{ нахождение $IS_j$

$\text{if} \, | IS_j | < | IS_{j-1} | \, \text{break} \ \}$

формирование графа $N$

нахождение максимального независимого множества $MIS$ графа $N$

$G = G_{V \setminus MIS}(V \setminus MIS, E_{V \setminus MIS}) \ \}$

Заметим, что только при $j = 1$ число итераций будет максимальным, так как для $j > 1$ уже известна мощность максимального независимого множества.

**Результаты экспериментальных расчетов.** С целью проверки эффективности предложенного алгоритма был проведен вычислительный эксперимент по решению задачи разбиения множества вершин графа на независимые множества для графов, возникающих при построении помехозащищенных кодов. Использовались графы из семейств, которые возникают при построении кодов, корректирующих

- удаление одного бита – ldc,
- удаление двух битов – 2dc,
- единичную ошибку в $Z$-канале – lzc.

Подробнее с этими графами можно ознакомиться на сайте http://www.research.att.com/~njas/doc/graphs.html. Для поиска максимального независимого множества вершин графа использовался алгоритм, описанный в [6].

В столбце $\alpha(\bar{G})$ приведены наибольшие известные числа $\alpha(\bar{G})$.

В табл. 1 содержатся результаты решения задачи разбиения множества вершин для графов, являющихся дополнениями графов семейств 1dc, 2dc.

Получение разбиений множества вершин на независимые множества для графа lzc имеет важное значение, так как эти данные могут быть использованы для нахождения нижних оценок объема помехозащищенных кодов для $Z$-канала.

В табл. 2 приведены данные, полученные в [3]. (В табл. 2 и 3 для графов lzc512 значение $\alpha(G) = 62$, для графов lzc1024 – $\alpha(G) = 112$).

ТАБЛИЦА 1. Полученные разбиения

| Граф $\bar{G}$ | Количество | | $\alpha(\bar{G})$ | Разбиение |
|---|---|---|---|---|
| | узлов | ребер | | |
| 1dc512 | 512 | 121089 | 10 | 23(10),4(9),6(8),10(7),5(6), 6(5),10(4),7(3),2(2),3(1) |
| 1dc1024 | 1024 | 499713 | 11 | 37(11),8(10),11(9),13(8), 12(7),13(6),10(5),18(4), 9(3),7(2),9(1) |
| 2dc512 | 512 | 75221 | 58 | 58,51,3(46),38,37,36,28, 26,2(22),16,14,11,6,5,4 |
| 2dc1024 | 1024 | 354614 | 72 | 72,64,5(56),53,2(49),44,42, 39,2(37),32,29,28,26,22,21, 17,2(15),14,11,9,7,5, 2(3),1 |

ТАБЛИЦА 2. Разбиения, полученные в [3]

| Граф | Количество | | Разбиение | Норма | $k$ |
|---|---|---|---|---|---|
| | узлов | ребер | | | |
| lzc512 | 512 | 123904 | 4(62),58,54,51,44,34, 19,4 | 27726 | 11 |
| | | | 4(62),2(56),52,44,30, 20,6 | 27624 | 11 |
| lzc1024 | 1024 | 507135 | 112,2(110),109,104,9 8,95,87,75,61,46,14,3 | 97306 | 13 |

Табл. 3 содержит данные, полученные при решении задач предложенным алгоритмом. Звёздочка в колонке “Норма” этой таблицы указывает на то, что разбиения найдены при решении задачи (20)–(22).

ТАБЛИЦА 3. Разбиения, найденные предложенными алгоритмами

| Граф | Количество | | Разбиение | Норма | $K$ |
|---|---|---|---|---|---|
| | узлов | ребер | | | |
| 1 | 2 | 3 | 4 | 5 | 6 |
| lzc512 | 512 | 123904 | 4(62),58,56,53,46, 31,16,4 | 28034 | 11 |
| | | | 3(62),61,58,56,53, 46,29,18,5 | 27868 | 11 |
| | | | 4(62),58,56,53,43, 32,16,6 | 27850 | 11 |
| | | | 3(62),61,58,56,52, 46,31,17,5 | 27848 | 11 |
| | | | 4(62),59,56,49,41, 38,18,3 | 27852* | 11 |
| | | | 4(62),58,56,52,43, 33,17,5 | 27832 | |
| | | | 4(62),58,56,54,42, 31,15,8 | 27806 | 11 |
| lzc1024 | 1024 | 507135 | 112,2(110),108,105, 101,98,88,77,60, 40,13,2 | 98284* | 13 |
| | | | 112,2(110),109,108, 100,95,87,75,60, 44,13,1 | 98214* | 13 |

*Окончание табл. 3*

| 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|
| | | | 112,110,2(109),104, 103,95,89,75,61, 41,14,2 | 98004 | 13 |
| | | | 112,2(110),108,105, 101,96,90,76,57, 41,17,1 | 97946 | 13 |
| | | | 112,2(110),109,105, 100,99,88,75,59, 37,16,4 | 97942 | 13 |
| | | | 112,2(110),109,105, 101,96,87,77,60, 38,15,4 | 97850 | 13 |
| | | | 112,2(110),108,106, 99,95,89,76,60, 43,15,1 | 97842 | 13 |
| | | | 112,2(110),108,105, 100,96,88,74,65, 38,17,1 | 97828 | 13 |
| | | | 112,2(110),108,106, 103,95,85,76,60, 40,15,4 | 97720 | 13 |
| | | | 112,2(110),108,106, 101,95,87,75,61, 40,17,2 | 97678 | 13 |
| | | | 112,110,109,108,105, 101,96,86,78,63, 36,17,3 | 97674 | 13 |

Табл. 4 содержит результаты решения предложенными алгоритмами задачи le450_5a, взятой из сайта http://mat.gsia.cmu.edu/COLOR/instances.html. Звёздочкой в колонке "Время" отмечено среднее время решения задачи по 10 просчётам.

ТАБЛИЦА 4. Время решения задачи le450_5a

| Количество | | Алгоритм | Время, с | *K* |
|---|---|---|---|---|
| узлов | ребер | | | |
| 450 | 5714 | Данные взяты из [4] | 1027 | 5 |
| | | ГРП для задачи (17)–(19) | 337* | 5 |
| | | ГРП для задачи (20)–(22) | 171* | 5 |

**Заключение.** Найдены новые окраски графов lzc512, lzc1024, возникающих при построении помехозащищенных кодов, с нормой, большей чем в [3, 7]. Это позволило, в свою очередь, улучшить нижние оценки для кодов длиной $m$ = 19, 21, 22, 24.

*В.П. Шило*

ПІДХОДИ ДО РОЗВ'ЯЗАННЯ ЗАДАЧІ РОЗФАРБУВАННЯ ГРАФА

Запропоновано два підходи до розв'язання задачі розфарбування графа. Проведено порівняльне дослідження ефективності відомих для даної задачі і розроблених алгоритмів, яке підтвердило переваги запропонованих підходів.

*V.P. Shylo*

APPROACHES TO THE GRAPH PAINTING PROBLEM SOLVING

Two approaches to the Vertex Coloring Problem solving are proposed. The comparative investigation of the efficiency of the proposed one and other methods known for this problem are provided, which confirmes the advantages of the approach proposed.

1. *Pardalos P.M., Prokopyev O.A., Shylo O.V., Shylo V.P.* Global equilibrium search applied to the unconstrained binary quadratic optimization problem // Optimization Methods and Software. –2008. – **23**. – P. 129 – 140.
2. *Шило В.П.* Метод глобального равновесного поиска // Кибернетика и системный анализ. – 1999. – № 1. – С. 74–81.
3. *Etzion T., Ostergard P. R. J.* Greedy and heuristic algorithms for codes and colorings // IEEE Trans. Inform. Theory. – 1998. – **44**, N 1. – P. 382–388.
4. *Kochenberger G. A. , Glover Fred., Alidaee B., Rego C.* An Unconstrained Quadratic Binary Programming Approach to the Vertex Coloring Problem // Ann. Oper. Res. – 2005. – **139**, N 1. – P. 229 – 241.
5. *Palubeckis G.* Multistart tabu search strategies for the unconstrained binary quadratic optimization problem // Annals of Oper. Res. – 2004. – **131**. – P. 259–282.
6. *Сергиенко И.В., Шило В.П.* Задачи дискретной оптимизации: проблемы, методы решения, исследования. – Киев: Наук. думка, 2003. – 264 с.
7. *Шило В.П.* Новые нижние оценки объема помехозащищенных кодов для *Z*-канала // Кибернетика и систем. анализ. – 2002. – № 1. – С. 19 – 23.

Получено 14.04.2009

***Об авторе:***

*Шило Владимир Петрович*,

доктор физико-математических наук, ведущий научный сотрудник
Института кибернетики имени В.М. Глушкова НАН Украины.
e-mail: v.shylo@gmail.com