Модели целочисленного программирования

Виктор Васильевич Лепин

Раскраска графа и карт

Раскраска графа и карт

- В какое минимальное число цветов можно раскрасить вершины заданного графа, чтобы никакие две смежные вершины не были окрашены в один цвет.

Раскраска графа и карт

- В какое минимальное число цветов можно раскрасить вершины заданного графа, чтобы никакие две смежные вершины не были окрашены в один цвет.
- Так формулируется задача о раскраске графа

- В какое минимальное число цветов можно раскрасить вершины заданного графа, чтобы никакие две смежные вершины не были окрашены в один цвет.
- Так формулируется задача о раскраске графа
- Самый знаменитый частный случай данной задачи, известный как проблема четырех красок, состоит в том,

Раскраска графа и карт

- В какое минимальное число цветов можно раскрасить вершины заданного графа, чтобы никакие две смежные вершины не были окрашены в один цвет.
- Так формулируется задача о раскраске графа
- Самый знаменитый частный случай данной задачи, известный как проблема четырех красок, состоит в том,
- чтобы определить минимальное число цветов, необходимых для раскраски политической карты так,

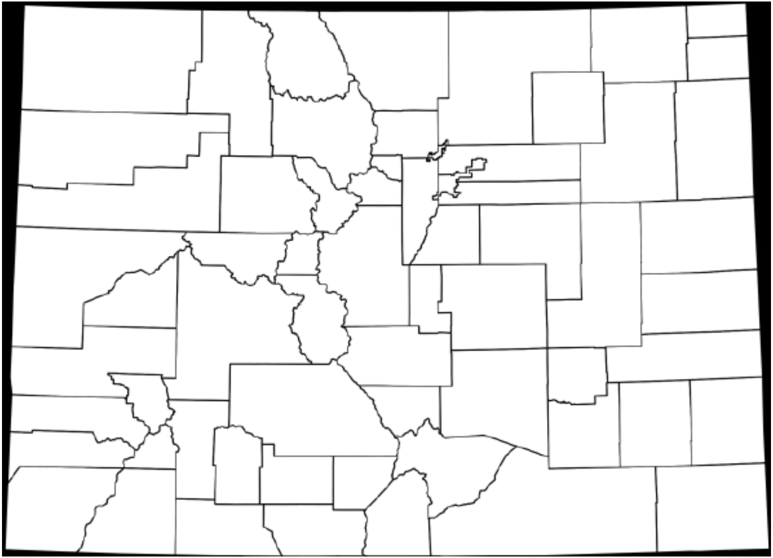
- В какое минимальное число цветов можно раскрасить вершины заданного графа, чтобы никакие две смежные вершины не были окрашены в один цвет.
- Так формулируется задача о раскраске графа
- Самый знаменитый частный случай данной задачи, известный как проблема четырех красок, состоит в том,
- чтобы определить минимальное число цветов, необходимых для раскраски политической карты так,
- чтобы никакие две страны, имеющие общую границу, не были раскрашены в один цвет.

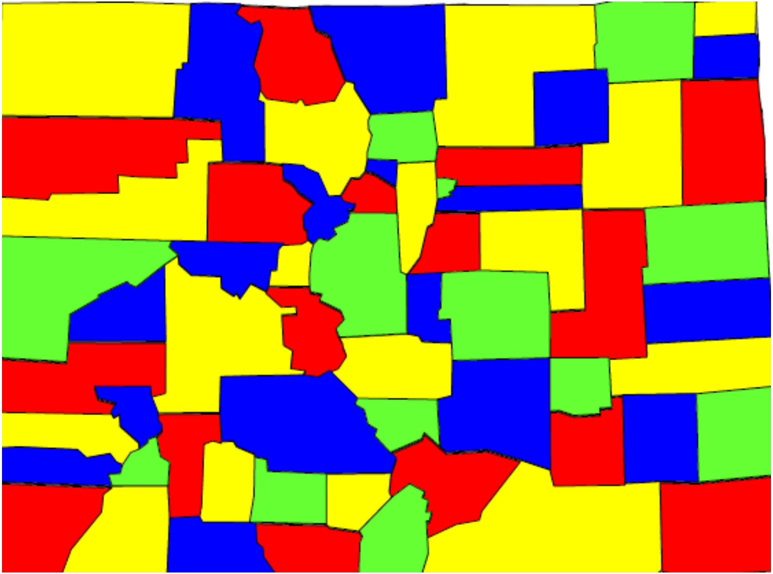
Раскраска графа и карт

- В какое минимальное число цветов можно раскрасить вершины заданного графа, чтобы никакие две смежные вершины не были окрашены в один цвет.
- Так формулируется задача о раскраске графа
- Самый знаменитый частный случай данной задачи, известный как проблема четырех красок, состоит в том,
- чтобы определить минимальное число цветов, необходимых для раскраски политической карты так,
- чтобы никакие две страны, имеющие общую границу, не были раскрашены в один цвет.
- Если представить каждую страну отдельной вершиной графа и соединить две вершины ребром, если соответствующие им страны имеют общую границу,

Раскраска графа и карт

- В какое минимальное число цветов можно раскрасить вершины заданного графа, чтобы никакие две смежные вершины не были окрашены в один цвет.
- Так формулируется задача о раскраске графа
- Самый знаменитый частный случай данной задачи, известный как проблема четырех красок, состоит в том,
- чтобы определить минимальное число цветов, необходимых для раскраски политической карты так,
- чтобы никакие две страны, имеющие общую границу, не были раскрашены в один цвет.
- Если представить каждую страну отдельной вершиной графа и соединить две вершины ребром, если соответствующие им страны имеют общую границу,
- то задача о раскраске карты представляется как задача о раскраске полученного графа.





Раскраска в 4 цвета.

Переменные

Пусть $G = (V, E)$ — граф соседства стран;
V — множество стран,
$E = \{vu \mid v, u \in V, v$ и u имеют общую границу$\}$
Введем два семейства бинарных переменных:

- $y_k = 1$, если цвет k используется;
- $x_{ik} = 1$, если стране i дан цвет k, и $x_{ik} = 0$ в противном случае.

- Цель — минимизировать количество используемых красок:
  $\min \sum_k y_k$

Формулировка

- Цель — минимизировать количество используемых красок:
  $\min \sum_k y_k$
- Каждая страна должна быть окрашена:
  $\sum_k x_{ik} = 1 \ \forall i \in V$

Формулировка

- Цель — минимизировать количество используемых красок:
  $\min \sum_k y_k$
- Каждая страна должна быть окрашена:
  $\sum_k x_{ik} = 1 \; \forall i \in V$
- Если страны i и j имеют общую границу, то их нельзя раскрашивать в один цвет:
  $x_{ik} + x_{jk} \leq 1$ для $ij \in E$ и $k \in \{1, 2, 3, 4\}$

Формулировка

- Цель — минимизировать количество используемых красок:
  $\min \sum_k y_k$
- Каждая страна должна быть окрашена:
  $\sum_k x_{ik} = 1 \ \forall i \in V$
- Если страны i и j имеют общую границу, то их нельзя раскрашивать в один цвет:
  $x_{ik} + x_{jk} \leq 1$ для $ij \in E$ и $k \in \{1, 2, 3, 4\}$
- Если страна i раскрашена в цвет k, то цвет k имеет статус использованного:
  $x_{ik} \leq y_k$ для $i \in V$ и $k \in \{1, 2, 3, 4\}$

Формулировка

- Цель — минимизировать количество используемых красок:
  $\min \sum_k y_k$
- Каждая страна должна быть окрашена:
  $\sum_k x_{ik} = 1 \ \forall i \in V$
- Если страны i и j имеют общую границу, то их нельзя раскрашивать в один цвет:
  $x_{ik} + x_{jk} \leq 1$ для $ij \in E$ и $k \in \{1, 2, 3, 4\}$
- Если страна i раскрашена в цвет k, то цвет k имеет статус использованного:
  $x_{ik} \leq y_k$ для $i \in V$ и $k \in \{1, 2, 3, 4\}$
- Все переменные бинарные:
  $x_{ik} \in \{0, 1\}$, $y_k \in \{0, 1\}$.

Математическая модель

$$\min \sum_k y_k$$

$$\sum_k x_{ik} = 1 \quad \forall i \in V$$

$$x_{ik} + x_{jk} \leq 1 \ \text{ для } \ ij \in E \text{ и } k \in \{1, 2, 3, 4\}$$

$$x_{ik} \leq y_k \text{ для } i \in V \text{ и } k \in \{1, 2, 3, 4\}$$

$$x_{ik} \in \{0, 1\}, \quad y_k \in \{0, 1\}.$$

Задача о гамильтоновом цикле наименьшего веса

Задача о гамильтоновом цикле наименьшего веса

- Каждому ребру $(v, w) \in E$ графа $G = (V, E)$ приписан вес $c(v, w)$,

Задача о гамильтоновом цикле наименьшего веса

- Каждому ребру $(v, w) \in E$ графа $G = (V, E)$ приписан вес $c(v, w)$,
- нужно найти гамильтоном цикл $\Gamma = (v_0, v_1, \ldots, v_n = v_0)$

Задача о гамильтоновом цикле наименьшего веса

- Каждому ребру $(v, w) \in E$ графа $G = (V, E)$ приписан вес $c(v, w)$,
- нужно найти гамильтоном цикл $\Gamma = (v_0, v_1, \ldots, v_n = v_0)$
- наименьшего веса $c(\Gamma) = \sum_{i=1}^{n} c(v_{i-1}, v_i)$.

Задача о гамильтоновом цикле наименьшего веса

- Каждому ребру $(v, w) \in E$ графа $G = (V, E)$ приписан вес $c(v, w)$,
- нужно найти гамильтоном цикл $\Gamma = (v_0, v_1, \ldots, v_n = v_0)$
- наименьшего веса $c(\Gamma) = \sum_{i=1}^{n} c(v_{i-1}, v_i)$.
- Задача коммивояжера – это задача о гамильтоновом цикле наименьшего веса в полном графе.

Задача о гамильтоновом цикле наименьшего веса

- Каждому ребру $(v, w) \in E$ графа $G = (V, E)$ приписан вес $c(v, w)$,
- нужно найти гамильтоном цикл $\Gamma = (v_0, v_1, \ldots, v_n = v_0)$
- наименьшего веса $c(\Gamma) = \sum_{i=1}^{n} c(v_{i-1}, v_i)$.
- Задача коммивояжера – это задача о гамильтоновом цикле наименьшего веса в полном графе.
- Вершины графа представляют некоторые города, а вес $c(v, w)$ – это расстояние между городами.

Задача о гамильтоновом цикле наименьшего веса

- Каждому ребру $(v, w) \in E$ графа $G = (V, E)$ приписан вес $c(v, w)$,
- нужно найти гамильтоном цикл $\Gamma = (v_0, v_1, \ldots, v_n = v_0)$
- наименьшего веса $c(\Gamma) = \sum_{i=1}^{n} c(v_{i-1}, v_i)$.
- Задача коммивояжера – это задача о гамильтоновом цикле наименьшего веса в полном графе.
- Вершины графа представляют некоторые города, а вес $c(v, w)$ – это расстояние между городами.
- Коммивояжер, начиная из города, в котором он проживает,

Задача о гамильтоновом цикле наименьшего веса

- Каждому ребру $(v, w) \in E$ графа $G = (V, E)$ приписан вес $c(v, w)$,
- нужно найти гамильтоном цикл $\Gamma = (v_0, v_1, \ldots, v_n = v_0)$
- наименьшего веса $c(\Gamma) = \sum_{i=1}^{n} c(v_{i-1}, v_i)$.
- Задача коммивояжера – это задача о гамильтоновом цикле наименьшего веса в полном графе.
- Вершины графа представляют некоторые города, а вес $c(v, w)$ – это расстояние между городами.
- Коммивояжер, начиная из города, в котором он проживает,
- хочет посетить каждый из остальных $n - 1$ городов ровно один раз и вернуться обратно в родной город,

Задача о гамильтоновом цикле наименьшего веса

- Каждому ребру $(v, w) \in E$ графа $G = (V, E)$ приписан вес $c(v, w)$,
- нужно найти гамильтоном цикл $\Gamma = (v_0, v_1, \ldots, v_n = v_0)$
- наименьшего веса $c(\Gamma) = \sum_{i=1}^{n} c(v_{i-1}, v_i)$.
- Задача коммивояжера – это задача о гамильтоновом цикле наименьшего веса в полном графе.
- Вершины графа представляют некоторые города, а вес $c(v, w)$ – это расстояние между городами.
- Коммивояжер, начиная из города, в котором он проживает,
- хочет посетить каждый из остальных $n - 1$ городов ровно один раз и вернуться обратно в родной город,
- при этом длина его маршрута должна быть минимальной.

Формулировка

- Переменные: для $i \neq j \in V$, $x_{ij} = 1$, если коммивояжер проходит по ребру ij и $x_{ij} = 0$ в противном случае.
- Цель — минимизировать пройденый путь:

$$\min \sum_{i \neq j \in V} c_{ij} x_{ij}.$$

- Из каждой вершины переходит только в одну другую вершину:

$$\sum_{j \in V, j \neq i} x_{ij} = 1 \quad \forall i \in V$$

- В каждый город входит по одной из дорог:

$$\sum_{i \in V, i \neq j} x_{ij} = 1 \quad \forall j \in V$$

- Исключаются не гамильтоновы циклы:

$$\sum_{i \in S, j \in V \setminus S} x_{ij} \geq 1 \quad \forall S \subsetneq V$$

Математическая модель

$$\min \sum_{i \neq j \in V} c_{ij} x_{ij}.$$

$$\sum_{j \in V, j \neq i} x_{ij} = 1 \quad \forall i \in V$$

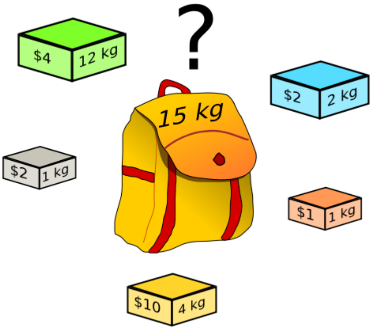
$$\sum_{i \in V, i \neq j} x_{ij} = 1 \quad \forall j \in V$$

$$\sum_{i \in S, j \in V \setminus S} x_{ij} \geq 1 \quad \forall S \subsetneq V$$

$$x_i \in \{0, 1\}.$$

Задача 0/1-рюкзак

- Рассмотрим набор предметов, где каждый предмет имеет вес и значение. Цель состоит в том, чтобы выбрать подмножество предметов, чтобы общий вес был меньше заданного предела, а общее значение было как можно большим.



0/1-рюкзак

Формализованное определение:

ВХОД: Набор предметов $S = \{1, 2, ..., n\}$. Предмет i имеет вес w_i и значение v_i. Общий весовой предел W;
ВЫХОД: Подмножество предметов с максимальной общей стоимостью и общим весом ниже W.

- Здесь, “0/1” означает, что мы должны выбрать предмет (1) или отказаться от него (0), и мы не можем выбрать часть предмета.
- Напротив, задача Дробный рюкзак позволяет выбрать дробный элемент, скажем, 0.5.

Формулировка

- Цель — максимизировать общую стоимость предметов:

$$\max \sum_i v_i x_i$$

Формулировка

- Цель — максимизировать общую стоимость предметов:

$$\max \sum_i v_i x_i$$

- Ограничение по весу:

$$\sum_i w_i x_i \leq W$$

Формулировка

- Цель — максимизировать общую стоимость предметов:

$$\max \sum_i v_i x_i$$

- Ограничение по весу:

$$\sum_i w_i x_i \leq W$$

- Все переменные бинарные:

$$x_i \in \{0, 1\}.$$

Задача 0/1-рюкзак

Математическая модель

$$\max \sum_i v_i x_i$$

$$\sum_i w_i x_i \leq W$$

$$x_i \in \{0, 1\}.$$

Задача о назначениях

Постановка задачи

Для n работников и работ, дана матрица $n \times n$, задающая стоимость выполнения каждой работы каждым работником. Найти минимальную стоимость выполнения работ, такую что каждый работник выполняет ровно одну работу, а каждую работу выполняет ровно один работник. Т.е. произвести назначение (assignment) работника на работу.

Назначение это биекция ϕ между двумя конечными множествами из n элементов.
В оптимизационной задаче нужно найти наилудшее назначение, т.е. нам нужно оптимизировать некоторую целевую функцию, которая зависит от назначения ϕ.

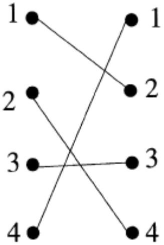
Способы представления назначения

Назначения могут быть представлены разными способами.

- Биективное отображение между двумя конечными множествами V и W может быть представлено прямым способом посредством совершенного паросочетания в двудольном графе $G = (V, W; E)$, где множества вершин V и W имеет n вершин. Ребро $(i, j) \in E$ является ребром совершенного паросочетания тогда и только тогда, когда $j = \phi(i)$, см. рис. 1.

$$\varphi = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 4 & 3 & 1 \end{pmatrix}$$

$$X_\varphi = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$



Способы представления назначения

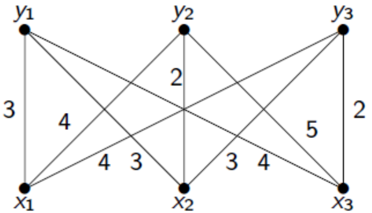
- Идентифицируя множества V и W, мы получаем представление назначения перестановкой.
- Каждая перестановка ϕ множества $N = \{1, \ldots, n\}$ уникальным образом соответствует матрице перестановок $X_\phi = (x_{ij})$ с $x_{ij} = 1$ для $j = \phi(i)$ и $x_{ij} = 0$ для $j \neq \phi(i)$. Эту матрицу X_ϕ можно рассматривать как матрицу смежности двудольного графа G, представляющего совершенное паросочетание, см. рис.1.

Матрица весов

Определение

Пусть $(K_{n,n}, w)$ — взвешенный полный двудольный граф, $w(x_i, y_j) = w_{ij}$ для всех $i, j = 1, 2, \ldots, n$. Квадратная матрица $W = (w_{ij})$ порядка n называется матрицей весов этого графа.

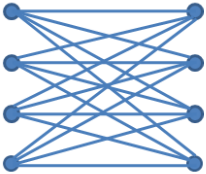
Так, матрица на рис. справа задает взвешенный граф $(K_{3,3}, w)$ на рис. слева.



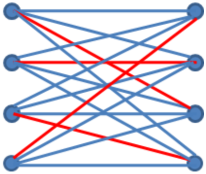
$$W = \begin{pmatrix} 3 & 4 & 4 \\ 3 & 2 & 3 \\ 4 & 5 & 2 \end{pmatrix}$$

Пример задачи о назначениях

$$\begin{pmatrix} 2 & 10 & 9 & 7 \\ 15 & 4 & 14 & 8 \\ 13 & 14 & 16 & 11 \\ 4 & 15 & 13 & 19 \end{pmatrix}$$



$$\begin{pmatrix} 2 & 10 & \boxed{9} & 7 \\ 15 & \boxed{4} & 14 & 8 \\ 13 & 14 & 16 & \boxed{11} \\ \boxed{4} & 15 & 13 & 19 \end{pmatrix}$$



Формулировка

- Цель — минимизировать стоимость выполнения работ:

$$\min \sum_{i}^{n} \sum_{j}^{n} w_{ij} x_{ij}$$

Формулировка

- Цель — минимизировать стоимость выполнения работ:

$$\min \sum_{i}^{n} \sum_{j}^{n} w_{ij} x_{ij}$$

- Ограничение: работа выполняется одним работником:

$$\sum_{i}^{n} x_{ij} = 1 \quad \forall j = 1, \ldots, n$$

Формулировка

- Цель — минимизировать стоимость выполнения работ:

$$\min \sum_{i}^{n} \sum_{j}^{n} w_{ij} x_{ij}$$

- Ограничение: работа выполняется одним работником:

$$\sum_{i}^{n} x_{ij} = 1 \quad \forall j = 1, \ldots, n$$

- Работник выполняет одну работу:

$$\sum_{j}^{n} x_{ij} = 1 \quad \forall i = 1, \ldots, n$$

Формулировка

- Цель — минимизировать стоимость выполнения работ:

$$\min \sum_i^n \sum_j^n w_{ij} x_{ij}$$

- Ограничение: работа выполняется одним работником:

$$\sum_i^n x_{ij} = 1 \quad \forall j = 1, \ldots, n$$

- Работник выполняет одну работу:

$$\sum_j^n x_{ij} = 1 \quad \forall i = 1, \ldots, n$$

- Все переменные бинарные:
  $x_{ij} \in \{0, 1\}$.

Математическая модель

$$\min \sum_{i}^{n} \sum_{j}^{n} w_{ij} x_{ij}$$

$$\sum_{i}^{n} x_{ij} = 1 \quad \forall j = 1, \ldots, n$$

$$\sum_{j}^{n} x_{ij} = 1 \quad \forall i = 1, \ldots, n$$

$$x_{ij} \in \{0, 1\}.$$