Kolorowania grafów

Michał Dobranowski
I LO im. Bartłomieja Nowodworskiego w Krakowie

20 lutego 2022

Streszczenie

Kolorowanie grafu jest ważnym problemem nie tylko w kontekście teorii grafów, lecz również w teorii złożoności obliczeniowej. Celem mojej pracy jest wyprowadzenie istotnych dla tych dwóch dziedzin definicji i twierdzeń. Czytelnik na początku poznaje klasy złożoności problemów, problem spełnialności formuły logicznej i problem kolorowalności grafu oraz algorytm 2-kolorowania grafu. Dalej dowodzimy „trudności" wspomnianego problemu i redukujemy do niego problem spełnialności. W kolejnej części przedstawione jest popularne twierdzenie o czterech barwach oraz dowód jego słabszej wersji (z pięcioma barwami). Dowód ten jest później rozwinięty, aby można było na jego podstawie stworzyć algorytm 5-kolorowania grafu. W ostatniej sekcji omówione są pewnie przypadki szczególne, które dla danych klas grafów mogą ułatwić problem kolorowalności.

Spis treści

1 Wstęp **2**
1.1 Notacja dużego $\mathcal{O}$ 2
1.2 Klasy złożoności problemów 2
1.3 Garść definicji z zakresu teorii grafów 3

2 Problem spełnialności **3**
2.1 k-SAT 3

3 Kolorowania grafów **4**
3.1 2-kolorowanie grafu 5
3.2 3-kolorowanie grafu 5
3.3 Kolorowanie grafu planarnego 6
3.4 Kolorowanie grafu zewnętrznie planarnego 10
3.5 Liczba chromatyczna i przypadki szczególne 11

4 Podsumowanie **11**

1 Wstęp

1.1 Notacja dużego $\mathcal{O}$

Definicja 1 (Asymptotyczne tempo wzrostu)**.** *Asymptotyczne tempo wzrostu to miara określająca zachowanie funkcji wraz ze wzrostem jej argumentów. Mówimy, że funkcja f jest co najwyżej rzędu funkcji g, co zapisujemy jako $f(x) = \mathcal{O}(g(x))$, jeśli istnieje stała rzeczywista $c > 0$ oraz stała naturalna n_0 takie, że nierówność $f(n) \leq cg(n)$ zachodzi dla każdego $n > n_0$.*

Zasadniczo przyjmuje się, że problemy „łatwe” to takie, których rozwiązanie ma złożoność czasową: stałą $\mathcal{O}(1)$, logarytmiczną $\mathcal{O}(\log n)$, liniową $\mathcal{O}(n)$ lub wielomianową $\mathcal{O}(n^k)$. Problemy „trudne” to takie, których rozwiązanie wymaga złożoności czasowej wykładniczej $\mathcal{O}(a^n)$ lub wyższej, na przykład $\mathcal{O}(n!)$.

1.2 Klasy złożoności problemów

Definicja 2 (Problem P)**.** *Problem P (deterministic polynomial) to problem decyzyjny, którego rozwiązanie można znaleźć w czasie wielomianowym.*

Definicja 3 (Problem NP)**.** *Problem NP (nondeterministic polynomial) to problem decyzyjny, którego rozwiązanie można zweryfikować w czasie wielomianowym.*

Definicja 4 (Problem NP-zupełny)**.** *Problem NP-zupełny (NP-Complete) to problem w klasie NP, do którego można zredukować dowolny problem w tej klasie.*

Pytanie, czy $\mathsf{NP} = \mathsf{P}$ jest jednym z największych problemów informatyki teoretycznej i jednym z siedmiu problemów milenijnych ogłoszonych przez Instytut Matematyczny Claya w 2000 roku.

1.3 Garść definicji z zakresu teorii grafów

Definicja 5 (Graf). *Graf $G = (V, E)$ to struktura matematyczna złożona ze skończonego zbioru wierzchołków V oraz skończonej rodziny krawędzi E, czyli dwuelementowych podzbiorów zbioru różnych wierzchołków.*

Piszemy $v \in G$, jeśli wierzchołek v należy do zbioru wierzchołków oraz $vu \in G$ jeśli krawędź $\{v, u\}$ należy do zbioru krawędzi. Będziemy również oznaczać liczbę wierzchołków jako n oraz liczbę krawędzi jako m.

Definicja 6 (Graf spójny). *Graf spójny to taki graf, że każdą parę różnych wierzchołków łączy ciąg dowolnie wielu krawędzi (droga).*

Definicja 7 (Graf pełny). *Graf pełny to graf, w którym każdą parę wierzchołków łączy krawędź.*

Definicja 8 (Klika). *Klika to podgraf pełny. Liczność największej kliki w grafie G oznaczamy symbolem $\omega(G)$.*

Definicja 9 (Stopień wierzchołka). *Stopień wierzchołka to liczba sąsiadów danego wierzchołka, czyli innych wierzchołków, z którymi jest on połączony krawędziami. Oznaczamy go symbolem $\deg(v)$. Największy stopień w grafie G oznaczamy symbolem $\Delta(G)$, a najmniejszy $\delta(G)$.*

Definicja 10 (Graf planarny). *Graf planarny to graf, który można narysować na płaszczyźnie w taki sposób, by krawędzie nie przecinały się ze sobą. Będziemy oznaczać liczbę ścian (części płaszczyzny zamkniętych przez krawędzie) symbolem f.*

Definicja 11 (Graf zewnętrznie planarny). *Graf zewnętrznie planarny to graf planarny, którego wszystkie wierzchołki należą do krawędzi zewnętrznych grafu.*

2 Problem spełnialności

Problem spełnialności (SAT) to problem, który dla danej formuły logicznej określa, czy istnieje takie podstawienie wartości, że jest ona prawdziwa.

2.1 k-SAT

Definicja 12 (Literał). *Literał to zmienna logiczna lub jej zaprzeczenie.*

Definicja 13 (Klauzula). *Klauzula to alternatywa literałów.*

Definicja 14 (Problem k-SAT). *k-SAT to problem spełnialności koniunkcji klauzul mających nie więcej niż k literałów.*

Przykładem problemu 2-SAT będzie problem, czy poniższa formuła ma takie wartościowanie zmiennych logicznych a_i, że jest ona prawdziwa.

$$(\neg a_1 \vee a_2) \wedge (a_2 \vee a_3) \wedge (\neg a_3 \vee a_1)$$

Warto zauważyć, możemy z łatwością skonstruować algorytm rozwiązujący problem SAT, działający w czasie $\mathcal{O}(2^n n)$, gdzie n to liczba literałów. Będzie on po prostu testował każde możliwe wartościowanie w czasie liniowym.

Twierdzenie 2.1 (Cook-Levin, 1971)**.** *Problem SAT jest* NP*-zupełny.*

Uzasadnienie. O ile fakt, że SAT należy do NP (rozwiązanie możemy zweryfikować w czasie liniowym) jest oczywisty, to dowód jego zupełności wymagałby wprowadzenia definicji niedeterministycznej maszyny Turinga; między innymi z tego powodu pominiemy go tutaj i skierujemy do literatury [1]. Intuicyjnie możemy jednak całkiem dobrze oddać istotę dowodu: każdy problem decyzyjny w NP możemy zredukować do SAT, ponieważ formuła logiczna to matematyczny opis maszyny (złożonej z bramek logicznych), która rozwiązuje dany problem. □

Twierdzenie 2.2. *Problem 3-SAT jest* NP*-zupełny.*

Dowód. Zauważmy, że klauzula

$$a_1 \vee a_2 \vee \cdots \vee a_n$$

jest równoważna formule

$$((a_1 \vee a_2) \Leftrightarrow b) \wedge (b \vee a_3 \vee \cdots \vee a_n),$$

a z kolei

$$((a_1 \vee a_2) \Leftrightarrow b)$$

jest równoważne

$$(\neg a_1 \vee b) \wedge (\neg a_2 \vee b) \wedge (a_1 \vee a_2 \vee \neg b).$$

W ten sposób możemy zredukować rozmiar klauzuli z n do $\max(3, n-1)$. Powtarzając ten schemat wielokrotnie, dokonujemy redukcji ogólnego problemu SAT (który z twierdzenia 2.1 jest NP-zupełny) do 3-SAT. □

3 Kolorowania grafów

Definicja 15 (Kolorowanie grafu)**.** *k-kolorowanie grafu $G = (V, E)$ to funkcja $f : V \to \{0, \ldots, k-1\}$ taka, że $vu \in G \Longrightarrow f(v) \neq f(u)$.*

Definicja 16 (Liczba chromatyczna)**.** *Liczba chromatyczna grafu G to najmniejsza taka liczba $k \in \mathbb{N}$, że graf G jest k-kolorowalny (k-dzielny). Oznacza się ją symbolem $\chi(G)$.*

Definicja 17 (Problem k-COLOR)**.** *k-COLOR to problem stwierdzenia, czy dany graf jest k-kolorowalny.*

3.1 2-kolorowanie grafu

Możemy stwierdzić, że graf jest 2-kolorowalny oraz znaleźć funkcję kolorującą za pomocą algorytmu DFS (*depth first search*) w czasie $\mathcal{O}(n)$.

Algorytm 1 2-kolorowanie grafu, *DFS*

```
function DFS(v, k)
    if v nie został wcześniej odwiedzony then
        oznacz v jako odwiedzony i pokoloruj go na kolor k
        for all u, gdzie vu ∈ E do
            if DFS(u, ¬k) = False then
                return False
        return True
    else if v został odwiedzony i ma kolor ¬k then
        return False
v ← 1                    ▷ Możemy zacząć od dowolnego wierzchołka
k ← 0                    ▷ Oznaczamy kolory 0 oraz 1, zmieniamy je znakiem ¬
DFS(u, k)
```

Powyższy algorytm decyduje, czy graf jest 2-kolorowalny oraz koloruje go we właściwy sposób. Złożoność obliczeniowa wynika z faktu, że każdy wierzchołek odwiedzimy maksymalnie raz. Łatwo wywnioskować, że warunkiem dostatecznym i koniecznym dwudzielności grafu jest brak cykli o nieparzystej długości w tym grafie.

Znajdując algorytm wielomianowy (a dokładniej liniowy) pokazaliśmy, że problem 2-COLOR jest w klasie P.

3.2 3-kolorowanie grafu

Twierdzenie 3.1. *Problem 3-COLOR jest NP-zupełny.*

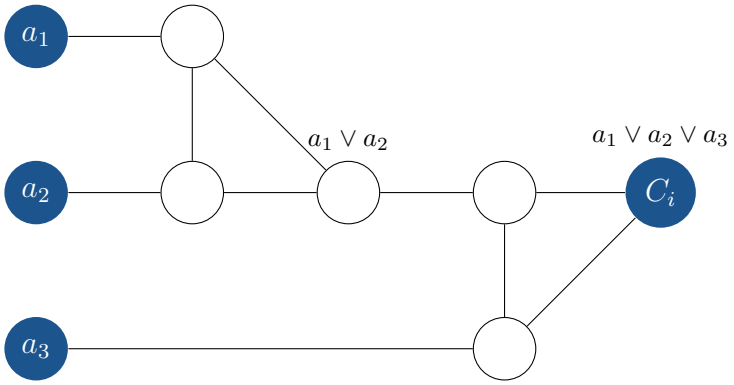
Dowód. Możemy zweryfikować odpowiedź poprzez sprawdzenie każdej krawędzi grafu w czasie $O(m) = O(n^2)$. To dowodzi, że dany problem jest w klasie NP.

Korzystając z twierdzenia 2.2, wystarczy teraz zredukować problem 3-SAT do 3-COLOR. Niech $C = C_1 \wedge C_2 \wedge \cdots \wedge C_k$, gdzie C_i jest klauzulą złożoną z trzech literałów ze zbioru $\{a_1, a_2, \ldots, a_n\}$. Musimy więc stworzyć taki graf G, że jego możliwość jego 3-kolorowania będzie równoznaczna z możliwością wartościowania zmiennych $a_1, a_2, \ldots, a_n$ tak, aby C było prawdziwe.

Na początku niech graf G będzie grafem pełnym złożonym z wierzchołków F, T oraz S. Bez starty ogólności, możemy przyjąć, że ich kolory to odpowiednio $0, 1, 2$. Następnie dodajmy do grafu G wierzchołki $a_1, a_2, \ldots, a_n$ oraz ich zaprzeczenia. Utożsammy kolor 0 z fałszem oraz 1 z prawdą; kolor 2 będzie neutralny — w tym celu połączmy wszystkie wierzchołki a_i oraz $\neg a_i$ z wierzchołkiem S. Poprawne kolorowanie zagwarantuje, że będą miały kolor 0 lub 1.

Teraz dodajmy do grafu G wierzchołki C_i, które będą oznaczały klauzule. Chcemy, aby każdy z nich miał kolor 1 (był prawdziwy), więc łączymy je z wierzchołkami S oraz

F. Jedyne co zostało do zrobienia, to takie połączenie każdego wierzchołka C_i z jego trzema literałami, żeby zasymulować alternatywę logiczną. Okazuje się, że nietrudno jest to zrobić:



Łatwo sprawdzić (chociażby podstawiając wszystkie 8 możliwości), że istotnie, jeśli kolor każdego z wierzchołków a_1, a_2, a_3 jest równy 0, to kolor wierzchołka C_i też jest równy 0, a jeśli chociaż jeden z wierzchołków a_1, a_2, a_3 ma kolor 1, to również C_i ma kolor 1. W taki sposób łączymy wszystkie wierzchołki reprezentujące klauzule C_i z odpowiednimi literałami. Dana redukcja 3-SAT do 3-COLOR dowodzi, że 3-COLOR jest problemem NP-zupełnym. □

Zauważmy, że redukcja w odwrotną stronę niż w dowodzie powyższego twierdzenia jest raczej trywialna. Dla każdego wierzchołka v w grafie tworzymy trzy zmienne logiczne, oznaczające, że wierzchołek v kolorujemy na dany kolor. Łączymy zmienne w takie klauzule, które są równoważne stwierdzeniom, że wierzchołek nie może dostać wielu kolorów i sąsiednie wierzchołki nie mogą dostać takiego samego koloru. Z tego wynika, że gdybyśmy znaleźli algorytm, który szybko znajdowałby 3-kolorowanie dowolnego grafu, to rozwiązalibyśmy wszystkie problemy w klasie NP.

3.3 Kolorowanie grafu planarnego

Twierdzenie 3.2 (Euler, 1750)**.** *W każdym spójnym grafie planarnym zachodzi równość* $n - m + f = 2$.

Dowód. Zauważmy, że jeśli graf G jest drzewem (grafem bez cykli), to mamy $m = n - 1$ oraz $f = 1$. Wtedy $n-(n-1)+1 = 2$, więc twierdzenie jest prawdziwe dla każdego drzewa. Korzystając z zasady indukcji matematycznej względem liczby krawędzi grafu G, załóżmy prawdziwość tezy dla grafu spójnego z liczbą krawędzi m i wykażmy ją dla $m + 1$. Niech więc graf G nie będzie drzewem (to znaczy, że zawiera co najmniej jeden cykl). Weźmy dowolną krawędź e, która należy do pewnego cyklu. Graf $G \setminus \{e\}$ na podstawie założenia indukcyjnego spełnia tezę. Jeśli dodamy tę krawędź, to liczba m zwiększy się o 1 oraz liczba f zwiększy się o 1, więc równanie dalej będzie spełnione. □

Lemat 3.3. *Jeśli w spójnym grafie planarnym G liczba wierzchołków $n \geq 3$, to liczba krawędzi $m \leq 3n - 6$.*

Dowód. Weźmy graf planarny G o maksymalnej liczbie krawędzi. Skoro graf ma mieć jak najwięcej krawędzi, ale jednocześnie jest ograniczony przez wzór Eulera (twierdzenie 3.2), to musi mieć jak najwięcej ścian, więc w najbardziej optymalnym przypadku każda ściana będzie trójkątem.

Zauważmy, że każda krawędź dzieli dwie ściany, więc $2m = 3f$. Wtedy $m = n+f-2 = n + \frac{2}{3}m - 2$, więc maksymalna liczba krawędzi to $3n - 6$. □

Lemat 3.4. *Każdy graf planarny G ma wierzchołek o stopniu co najwyżej 5, co znaczy, że $\delta(G) \leq 5$.*

Dowód. Bez straty ogólności możemy założyć, że dany graf jest spójny i ma co najmniej 6 wierzchołków. Jeśli każdy wierzchołek w takim grafie miałby 6 lub więcej krawędzi, to zawierałby co najmniej $3n$ krawędzi, co jest sprzeczne z lematem 3.3. □

Twierdzenie 3.5 (O pięciu barwach)**.** *Każdy graf planarny jest 5-kolorowalny.*

Dowód. Metodą indukcji matematycznej względem liczby wierzchołków n grafu G przeprowadźmy następujące rozumowanie. Jeśli $n \leq 5$, to twierdzenie jest oczywiście prawdziwe, dlatego załóżmy, że $n > 5$. Z lematu 3.4 wiemy, że istnieje wierzchołek v o stopniu co najwyżej 5.

Jeśli stopnień ten jest mniejszy niż 5, to z założenia indukcyjnego graf $G \setminus \{v\}$ jest 5-kolorowalny, a wierzchołek v możemy pokolorować na kolor, którego nie ma żaden z jego maksymalnie czterech sąsiadów.

Jeśli stopień wierzchołka v jest równy 5, to możemy znaleźć taką parę jego sąsiadów u, w, że $uw \notin G$ (gdyby każdą parę jego sąsiadów łączyła krawędź, to klika K_5 na mocy lematu 3.3 nie byłaby planarna). Teraz stwórzmy graf G' przez usunięcie z grafu G wierzchołka v oraz ściągnięcie wierzchołków u, w do jednego wierzchołka x, który będzie połączony krawędziami ze wszystkimi sąsiadami wierzchołków u, w. Z założenia indukcyjnego wiemy, że graf G' jest 5-kolorowalny. Ponadto możemy pokolorować wierzchołki u, w na ten sam kolor, który ma x (jako że nie są one połączone krawędzią). Zauważmy, że 5 sąsiadów wierzchołka v używa teraz tylko 4 kolorów, więc graf G również jest 5-kolorowalny. □

Dowód tego twierdzenie niestety nie dostarcza nam efektywnego algorytmu 5-kolorowania grafu planarnego. Aby taki algorytm znaleźć, musimy udowodnić jeszcze jeden ważny fakt. Algorytm pochodzi z [2].

Lemat 3.6. *Dany jest graf planarny G, w którym $\delta(G) = 5$. Istnieje wówczas wierzchołek o stopniu 5, którego dwaj niepołączeni krawędzią sąsiedzi mają stopień* $\deg \leq 11$.

Dowód. Aby udowodnić tezę, wykażemy, że istnieje wierzchołek o stopniu 5, który ma co najwyżej jednego sąsiada o stopniu większym niż 11. Oznaczmy przez n_d liczbę wierzchołków o stopniu d. Na mocy założenia i lematu 3.3 mamy

$$2m = \sum_d dn_d$$

oraz

$$m \leq 3n - 6 = 3\sum_d d_n - 6.$$

Z tego możemy wywnioskować, że

$$\sum_d dn_d \leq 6 \sum_d d_n - 12$$

$$0 \geq \sum_d (d-6)n_d + 12$$

$$n_5 \geq \sum_{d \geq 7} (d-6)n_d + 12.$$

Załóżmy przeciwnie do tezy, że każdy wierzchołek stopnia 5 ma co najmniej dwóch sąsiadów o stopniu większym niż 11. Wtedy

$$2n_5 \leq \sum_{d \geq 12} dn_d.$$

Łącząc dwie powyższe nierówności otrzymujemy

$$\sum_{d \geq 12} dn_d \geq 2n_5 \geq \sum_{d \geq 7} (2d-12)n_d + 24$$

$$\sum_{d \geq 7} (2d-12)n_d - \sum_{d \geq 12} dn_d + 24 \leq 0$$

$$\sum_{7 \leq d \leq 11} (2d-12)n_d + \sum_{d \geq 12} (d-12)n_d + 24 \leq 0,$$

co jest sprzecznością. □

Wierzchołek, który jest opisywany przez lemat 3.6 lub którego stopień jest mniejszy od 5 nazwiemy *redukowalnym*. Graf $G = (V, E)$ przekazany do algorytmu 2 jest listą list sąsiadów $N(v)$ dla każdego wierzchołka v (tzw. listą sąsiedztwa).

Algorytm 2 5-kolorowanie grafu planarnego

```
procedure CONSIDER(v)
    if deg(v) ≤ 4 then usuń v z kolejki R, dodaj to kolejki Q
    else if deg(v) = 5 then
        if v jest redukowalny then dodaj v do kolejki R
        else usuń v z kolejki R
    else if deg(v) = 11 then
        for all u ∈ N(v) do
            if deg(v) = 5 oraz x jest redukowalny then dodaj x do kolejki R
procedure REDUCE(v)
    dodaj do stosu S wierzchołek v
    usuń v z grafu
    for all u ∈ N(v) do CONSIDER(u)
```

ciąg dalszy algorytmu 2

procedure IDENTIFY(u, w)
  utwórz wierzchołek t, $N(t) \leftarrow N(u) \cup N(w)$
  dodaj do stosu S trójkę wierzchołków (t, u, w)
  usuń u, w z grafu
  CONSIDER(t)
  for all $z \in N(t)$ **do** CONSIDER(z)

1: $S \leftarrow$ pusty stos
2: $Q \leftarrow$ kolejka zawierająca wszystkie wierzchołki deg ≤ 4
3: $R \leftarrow$ kolejka zawierające wszystkie redukowalne wierzchołki deg $= 5$
4: **while** $|V| > 5$ **do**
5: **if** $|Q| > 0$ **then**
6: $v \leftarrow$ wierzchołek z kolejki Q
7: REDUCE(v)
8: **else** ▷ Na mocy lematu 3.6 kolejka R nie jest pusta
9: $v \leftarrow$ wierzchołek z kolejki R
10: REDUCE(v)
11: IDENTIFY(u, w)
12: pokoloruj graf G ▷ trywialne, skoro $|V| \leq 5$
13: **while** $|S| > 0$ **do**
14: zdejmij x ze stosu S
15: **if** x jest wierzchołkiem **then**
16: odtwórz go i pokoloruj na kolor, którego nie ma żaden z jego sąsiadów
17: **else** ▷ x to trójka t, u, w
18: $t, u, w \leftarrow x$
19: odtwórz wierzchołki u oraz w (zamiast t) i pokoloruj je na kolor wierzchołka t

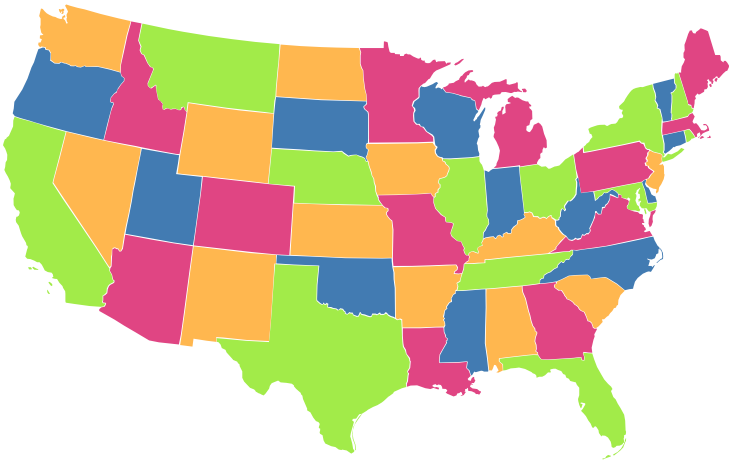
Dodawanie i usuwanie elementów z kolejek i stosu ma stałą złożoność czasową, dzięki czemu procedura CONSIDER działa w $\mathcal{O}(1)$. Procedura REDUCE jest wywoływana tylko dla wierzchołków, których stopień deg ≤ 5, a IDENTIFY deg ≤ 11, więc obie również działają w czasie $\mathcal{O}(1)$.

Inicjalizacja obu kolejek dokona się w czasie $\mathcal{O}(n)$. Linie od 4 do 11 wykonają się $\mathcal{O}(n)$ razy, podobnie linie od 13 do 19, dlatego cały algorytm 5-kolorowania grafu planarnego ma liniową złożoność obliczeniową.

Twierdzenie 3.7 (O czterech barwach)**.** *Każdy graf planarny jest 4-kolorowalny.*

Uzasadnienie. Jest to jedno z najstarszych twierdzeń, których prawdziwość udowodniono komputerowo. Chociaż dowód ten został później uproszczony, to dalej trzeba było wspierać się pracą komputera. Opis tej historii oraz kilka ciekawych uwag na temat sposobu, w jaki komputer wykonywał obliczenia, można znaleźć w [3]. □

W popularnej wersji twierdzenie brzmi następująco: każdą mapę polityczną da się pokolorować czterema kolorami w taki sposób, aby żadne dwa sąsiadujące państwa nie były tego samego koloru (zakładamy, że państwa nie posiadają eksklaw).



Rysunek 1: 4-kolorowana mapa Stanów Zjednoczonych.

Na podstawie dowodu twierdzenia o czterech barwach [4], który pojawił się w 1989 roku, stworzono algorytm 4-kolorowania działający w czasie $\mathcal{O}(n^4)$. W 1996 napisano nowy dowód [5], który pozwolił na poprawienie algorytmu do złożoności czasowej $\mathcal{O}(n^2)$.

3.4 Kolorowanie grafu zewnętrznie planarnego

Lemat 3.8. *Każdy graf zewnętrznie planarny G zawiera wierzchołek o stopniu co najwyżej 2, co znaczy, że $\delta(G) \leq 2$.*

Dowód. Bez straty ogólności można założyć że G jest grafem o maksymalnej liczbie krawędzi, co znaczy, że składa się wyłącznie z trójkątów. Skoro każdy z wierzchołków musi leżeć na zewnętrznej krawędzi oraz graf jest skończony, to oczywistym jest, że któryś ze wspomnianych trójkątów musi mieć dwie krawędzie, które leżą na zewnętrznej ścianie, a tym samym ich wspólny wierzchołek ma stopień równy 2. □

Twierdzenie 3.9. *Każdy graf zewnętrznie planarny jest 3-kolorowalny.*

Dowód. Wykażemy twierdzenie metodą indukcji matematycznej względem liczby wierzchołków grafu G. Jeśli $n \leq 3$, to teza jest oczywiście prawdziwa. Załóżmy zatem, że jest również prawdziwa dla każdej liczby od n mniejszej.

Na mocy lematu 3.8 w grafie istnieje wierzchołek v taki, że $\deg(v) \leq 2$. Z założenia indukcyjnego graf $G \setminus \{v\}$ jest 3-kolorowalny, a wierzchołek v możemy pokolorować na kolor, którego nie ma żaden z jego sąsiadów. □

W przeciwieństwie do twierdzenia o pięciu barwach (twierdzenie 3.5), z powyższego twierdzenia możemy z łatwością skonstruować algorytm 3-kolorowania grafu zewnętrznie planarnego, który będzie działał w złożoności $\mathcal{O}(n)$, ponieważ nie musimy sprawdzać sąsiadów wierzchołków (każdy z $n-3$ wierzchołków usuniemy i dodamy dokładnie raz; wystarczy tylko jedna kolejka do trzymania wierzchołków, których $\deg \leq 2$).

3.5 Liczba chromatyczna i przypadki szczególne

Twierdzenie 3.10. *Liczba chromatyczna grafu G spełnia nierówność $1 \leq \chi(G) \leq n$.*

Dowód. Pierwsza nierówność wynika wprost z definicji, druga jest prawdziwa, ponieważ nigdy nie musimy użyć więcej kolorów niż jest wierzchołków. □

Twierdzenie 3.11. *Liczba chromatyczna grafu G spełnia nierówność $\chi(G) \leq \frac{1}{2}+\sqrt{2m+\frac{1}{4}}$.*

Dowód. W optymalnym kolorowaniu dla każdej pary kolorów musi być chociaż jedna krawędź, która łączy dwa różnokolorowe wierzchołki, ponieważ inaczej możemy utożsamić te dwa kolory. Wobec tego otrzymujemy $m \geq \frac{\chi(G)(\chi(G)-1)}{2}$, co możemy przekształcić do $(\chi(G))^2 - \chi(G) \leq 2m$ oraz rozwiązać standardową nierówność kwadratową. □

Twierdzenie 3.12. *Liczba chromatyczna grafu G jest większa lub równa od liczności największej kliki w tym grafie, $\chi(G) \geq \omega(G)$.*

Dowód. Skoro w największej klice grafu G każdy wierzchołek jest połączony z innym krawędzią, to musimy użyć $\omega(G)$ kolorów to pokolorowania kliki i co najmniej tyle do pokolorowania zawierającego ją grafu G. □

Twierdzenie 3.13 (Brooks, 1941)**.** *Dany jest spójny graf G. Jeśli G nie jest grafem pełnym ani cyklem o nieparzystej długości, to $\chi(G) \leq \Delta(G)$. W przeciwnym wypadku $\chi(G) = \Delta(G) + 1$.*

Twierdzenie 3.14 (Grötzsch, 1959)**.** *Każdy graf planarny bez trójkątów jest 3-kolorowalny.*

Dowód twierdzenia 3.13 można znaleźć w [6, 7], a twierdzenia 3.14 w [8]. Ponadto istnieje algorytm 3-kolorowania grafu planarnego bez trójkątów, który działa w czasie liniowym [9].

4 Podsumowanie

Pokazaliśmy już, że kolorowalność grafu jest (w ogólności) problemem, na który nie potrafimy szybko znajdować odpowiedzi. Pozostaje jeszcze pytanie, dlaczego w ogóle powinniśmy się tym przejmować. Częściowej odpowiedzi udzieliliśmy już wcześniej — jest to problem NP-zupełny, więc dzięki wielomianowemu algorytmowi kolorowania grafu rozwiązalibyśmy wszystkie problemy w klasie NP (oraz wygrali 10^6 dolarów amerykańskich za potwierdzenie hipotezy $\mathsf{NP} = \mathsf{P}$).

Kolorowanie grafu samo w sobie jest jednak istotnym uogólnieniem wielu problemów. Oczywistym przykładem jest rozwiązywanie sudoku, ale bez trudu znajdziemy poważniejsze zastosowania. Algorytm, który korzysta z pewnych przybliżeń, jest wykorzystywany w kompilatorach: optymalizuje on czas dostępu do zmiennych, z których program korzysta. Innym przykładem będzie problem optymalnego przydziału zadań z pewnego zbioru do osób je wykonujących. Wariant problemu kolorowania grafu pojawi się również przy przydzielaniu częstotliwości radiowych do lokalnych stacji radiowych.

Literatura

[1] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, 1979.

[2] L. Banachowski, K. Diks, and W. Rytter, *Algorytmy i struktury danych*, pp. 250–255. PWN, 2018.

[3] P. Strzelecki, *Matematyka współczesna dla myślących laików*, pp. 145–146. Wydawnictwa UW, 2011.

[4] K. Appel and W. Haken, "Every planar map is four colorable," *Illinois Journal of Mathematics*, no. 3, 1977.

[5] N. Robertson, D. Sanders, P. Seymour, and R. Thomas, "A new proof of the four-colour theorem," *Electronic Research Announcements of the American Mathematical Society*, vol. 2, 1996.

[6] M. Zając, "A short proof of Brooks' theorem," 2018.

[7] R. Schwartz, "Notes on Brooks' theorem." `math.brown.edu/reschwar/M123/brooks.pdf`, 2016.

[8] Z. Dvořák, "A simplified discharging proof of Grötzsch theorem," 2013.

[9] Z. Dvořák, K. Kawarabayashi, and R. Thomas, "Three-coloring triangle-free planar graphs in linear time," in *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1176–1182, 2009.