



**Politechnika Gdańska**  
**WYDZIAŁ ELEKTRONIKI**  
**TELEKOMUNIKACJI I INFORMATYKI**



**Katedra: Katedra Algorytmów i Modelowania Systemów**

**Forma i poziom studiów: stacjonarne, jednolite magisterskie**

**Kierunek studiów: Informatyka**

**Krzysztof Turowski**  
**106732**

## **Praca dyplomowa magisterska**

**Temat pracy: Szkieletowe kolorowanie grafów**

**Kierujący pracą: prof. dr hab. inż. Marek Kubale**

### **Zakres pracy:**

Celem pracy jest szczegółowe omówienie zagadnienia szkieletowego kolorowania grafów: modelu kolorowania wierzchołkowego o dwóch ograniczeniach: różnych kolorach na końcach każdej krawędzi oraz różnicy kolorów na końcach wyróżnionego podgrafu szkieletowego równej co najmniej 2. W części przeglądowej przedstawiono szczegółowo dotychczas opublikowane wyniki w zakresie oszacowań szkieletowej liczby chromatycznej grafu w zależności od jego parametrów: liczby chromatycznej, stopnia grafu oraz struktury podgrafu szkieletowego. W części praktycznej zawarto opis czterech nowych algorytmów kolorowania szkieletowego, będących modyfikacją najszerzej znanych przybliżonych heurystyk kolorowania klasycznego. Stworzono również aplikację, obrazującą wynik działania poszczególnych algorytmów oraz wykonano testy praktycznej wydajności oraz czasu działania poszczególnych metod.

### **Abstract:**

The aim of this thesis is to describe the graph backbone coloring problem, a generalization of vertex coloring problem with additional conditions: not only the colors on the endpoints of an edge must be different, but also the difference between the endpoint of edges of given subgraph (also named the backbone of the graph) must be at least equal 2. The theoretical part includes the detailed survey of all published results in the area of the approximation of backbone chromatic number with respect to other graph parameters such as the chromatic number, graph degree or the structure of backbone graph. The practical part consists of the description of four new algorithms for backbone coloring problem, modifications of widely known vertex coloring heuristics. The application was used for visualizing the results of presented methods and was tested for practical performance ratio and running time.

Gdańsk, 2011 rok

# Spis treści

<b>1. Wprowadzenie</b>	<b>1</b>
<b>2. Kolorowanie grafów</b>	<b>3</b>
2.1. Klasyczne kolorowanie grafów . . . . .	3
2.2. Szkieletowe kolorowanie grafów . . . . .	4
<b>3. Ograniczenia liczby <math>BBC_\lambda(G, H)</math></b>	<b>9</b>
3.1. Doskonałe skojarzenie w szkielecie . . . . .	10
3.2. Gwiazdozbiór w szkielecie . . . . .	14
3.3. Ścieżka Hamiltona w szkielecie . . . . .	17
3.4. Drzewo spinające w szkielecie . . . . .	20
3.5. Szkielet d-zdegenerowany . . . . .	23
<b>4. Złożoność obliczeniowa kolorowania szkieletowego</b>	<b>26</b>
4.1. Grafy ogólne . . . . .	26
4.2. Split grafy . . . . .	29
<b>5. Projekt aplikacji</b>	<b>32</b>
5.1. Technologie i formaty . . . . .	32
5.2. Wybrane algorytmy kolorowania . . . . .	33
5.3. Przeprowadzone testy . . . . .	41
5.4. Wyniki testów . . . . .	42
<b>6. Wnioski</b>	<b>54</b>
<b>Bibliografia</b>	<b>57</b>

# Rozdział 1

---

## Wprowadzenie

Zagadnienia związane z problemem przydziału częstotliwości w telekomunikacji stanowią istotną klasę zagadnień modelowanych jako problemy kolorowania grafów. W ramach proponowanych modeli nakładane mogą być różnorakie ograniczenia, związane często z technicznymi aspektami systemu, przykładowo z mocami poszczególnych nadajników lub wymaganym pasmem odstępu między sąsiednimi stacjami w celu uniknięcia zakłóceń.

Niniejsza praca dotyczy szczególnego modelu kolorowania wierzchołkowego grafów, nazwanego kolorowaniem szkieletowym. Pierwsza, teoretyczna część pracy została ujęta w rozdziałach 2–4. Przedstawiony w nich został przegląd literatury dotyczącej kolorowania szkieletowego grafów wraz z omówieniem najważniejszych rezultatów, metod dowodzenia i przypadków granicznych.

Rozdział 2 zawiera wprowadzenie do kolorowania wierzchołkowego grafów. Podano definicję problemu kolorowania klasycznego, kolorowania szkieletowego oraz przyjętych w poszczególnych modelach oznaczeń. Przedstawiono również opis typowych klas grafów i szkieletów, będących obiektem badań.

W rozdziale 3 przedstawiono podstawowe ograniczenia na wymaganą liczbę kolorów w zależności od parametrów grafu dla typowych podgrafów szkieletowych: doskonałego skojarzenia, zbioru parami rozłącznych gwiazd, ścieżki Hamiltona oraz drzewa rozpinającego. W każdym przypadku omówiono ograniczenia na wartość szkieletowej liczby chromatycznej. Dodatkowo, zamieszczono opublikowane rezultaty oraz problemy otwarte dla specjalnych klas grafów, ze szczególnym uwzględnieniem split grafów oraz grafów planarnych.

Rozdział 4 dotyczy złożoności obliczeniowej zagadnienia kolorowania szkieletowego. Przedstawiono uzyskane wyniki w zakresie trudności rozwiązywania problemu w zależności od struktury grafu wejściowego, postaci szkieletu oraz wartości zadanych parametrów kolorowania.

Drugą częścią niniejszej pracy jest praktyczna realizacja programu, implementującego szereg przykładowych algorytmów kolorowania szkieletowego grafów. W rozdziale 5 zawarto specyfikację wymagań aplikacji oraz opisano algorytmy i obsługiwane formaty. Przedstawiono również opis zestawów testowych oraz wyniki otrzymane podczas ich realizacji.

Otrzymane wyniki zestawiono ze sobą, a także z rozwiązaniami optymalnymi. Sformułowano wnioski odnośnie przebiegu prac projektowych, ich wyników oraz perspektyw dalszych prac nad różnymi wariantami problemu.

# Rozdział 2

---

## Kolorowanie grafów

Zagadnienia związane z kolorowaniem grafów są jednym z najstarszych i najbardziej znanych problemów kombinatorycznych. Można wyróżnić kilkadziesiąt różnych modeli kolorowania grafów, a wiele z nich posiada znaczenie nie tylko teoretyczne, ale również praktyczne: od szeregowania zadań i organizacji produkcji do telefonii komórkowej i technologii cienkowarstwowej.

### 2.1. Klasyczne kolorowanie grafów

Problem kolorowania wierzchołkowego jest jednym z podstawowych problemów w teorii grafów. Pomimo prostej definicji problemu, jest to jednak problem bardzo trudny do rozwiązania – z drugiej strony, posiada on bardzo szerokie zastosowania praktyczne. To powoduje, że modele kolorowania grafów były i są obszarem intensywnych badań.

Formułując zagadnienie kolorowania grafów należy wprowadzić pojęcie pokolorowania grafu:

**Definicja 2.1.** *Pokolorowaniem grafu  $G = (V, E)$  nazywamy przyporządkowanie  $c : V \rightarrow \mathbb{N}_+$  przy warunku  $|c(u) - c(v)| > 0$  dla każdej krawędzi  $\{u, v\} \in E$ .*

**Definicja 2.2.**  *$k$ -pokolorowaniem grafu  $G = (V, E)$  dla pewnego  $k \in \mathbb{N}_+$  nazywamy pokolorowanie  $c$  grafu  $G$  przy pomocy  $k$  kolorów.*

Klasyczne zagadnienie kolorowania wierzchołkowego można wówczas sformułować w następującej postaci optymalizacyjnej i decyzyjnej:

**Problem 2.3.** Dany jest graf  $G = (V, E)$ . Znajdź pokolorowanie  $c$  grafu  $G$ , minimalizujące  $\max_{v \in V} c(v)$ .

**Problem 2.4.** Dany jest graf  $G = (V, E)$  i liczba  $k \in \mathbb{N}_+$ . Stwierdź, czy istnieje pokolorowanie  $c$  grafu  $G$ , spełniające  $\max_{v \in V} c(v) \leq k$ .

**Definicja 2.5.** Najmniejsza liczba  $k$ , dla której istnieje pokolorowanie grafu przy użyciu kolorów  $\{1, 2, \dots, k\}$ , nazywana jest *liczbą chromatyczną grafu  $G$*  i oznaczana jest przez  $\chi(G)$ .

W ogólnym przypadku problem znalezienia liczby chromatycznej jest NP-trudny, o ile nie zostałyby udowodniona słynna hipoteza  $P = NP$ . Dla wartości  $k \geq 3$  nie istnieje algorytm działający w czasie wielomianowym i sprawdzający, czy graf można pokolorować przy użyciu co najwyżej  $k$  kolorów [9].

Historycznie, duże znaczenie odegrał problem kolorowania grafów planarnych, znany również pod nazwą kolorowania map. Hipoteza, wysunięta w 1852 roku, głosiła, że każdą mapę – lub równoważnie każdy graf planarny – można pokolorować wykorzystując do tego celu jedynie cztery kolory. Pomimo wielkich wysiłków, twierdzenie to zostało wykazane dopiero po 120 latach przez Appela i Hakena [1], a jego dowód wymagał istotnego użycia obliczeń komputerowych w celu rozpatrzenia niemal 2000 przypadków szczególnych.

## 2.2. Szkieletowe kolorowanie grafów

Prosta definicja problemu kolorowania wierzchołkowego umożliwia bardzo łatwe wprowadzanie najróżniejszych ograniczeń. W modelach nieklasycznych oprócz wierzchołków mogą być również kolorowane pozostałe części grafu: krawędzie oraz ściany. Można także wprowadzić warunek kolorowania każdego wierzchołka kolorem z zadanego zbioru lub zmienić funkcję celu, aby minimalizowała np. sumę użytych kolorów.

W pracy [3] Broersma zdefiniował model kolorowania szkieletowego, w którym poszukiwane jest pokolorowanie grafu  $G$  przy pewnych ograniczeniach narzuconych na jego zadany podgraf  $H$ , nazywany *szkieletem grafu  $G$* . Taką parę grafów  $(G, H)$ , w której  $H$  jest ustalonym podgrafem grafu  $G$ , nazywamy *grafem  $G$  ze szkieletem  $H$* . Dodatkowe ograniczenie wymaga, aby kolory wierzchołków incydentnych do krawędzi w  $H$  nie tylko nie były identyczne, ale również niezbyt podobne do siebie. Tak postawiony problem jest uogólnieniem m.in.  $L(2, 1)$ -etykietowania lub kolorowania radiowego grafów.

**Definicja 2.6.** Pokolorowaniem szkieletowym grafu  $G = (V, E)$  ze szkieletem  $H =$

$(V', E')$  nazywamy przyporządkowanie  $c : V \rightarrow \mathbb{N}_+$  przy warunkach:

$$\begin{aligned} |c(u) - c(v)| &> 0 && \text{dla każdej krawędzi } \{u, v\} \in E, \\ |c(u) - c(v)| &\geq \lambda && \text{dla każdej krawędzi } \{u, v\} \in E'. \end{aligned}$$

**Definicja 2.7.**  $k$ -pokolorowaniem szkieletowym grafu  $G = (V, E)$  ze szkieletem  $H$  dla pewnego  $k \in \mathbb{N}_+$  nazywamy pokolorowanie szkieletowe  $c$  grafu  $G$ , w którym  $\max_{v \in V} c(v) = k$ .

Problem kolorowania wierzchołkowego można przedstawić zarówno w postaci optymalizacyjnej, jak i decyzyjnej:

**Problem 2.8.** Dany jest graf  $G = (V, E)$  ze szkieletem  $H$ . Znajdź pokolorowanie szkieletowe  $c$  grafu  $G$ , minimalizujące  $\max_{v \in V} c(v)$ .

**Problem 2.9.** Dany jest graf  $G = (V, E)$  ze szkieletem  $H$  oraz liczba  $k \in \mathbb{N}_+$ . Stwierdź, czy istnieje pokolorowanie szkieletowe  $c$  grafu  $G$  ze szkieletem  $H$ , spełniające  $\max_{v \in V} c(v) \leq k$ .

**Definicja 2.10.** Najmniejsza liczba  $k$ , dla której istnieje pokolorowanie grafu  $G$  ze szkieletem  $H$  tylko przy użyciu kolorów ze zbioru  $\{1, 2, \dots, k\}$ , nazywana jest  $\lambda$ -szkieletową liczbą chromatyczną grafu  $G$  ze szkieletem  $H$  i oznaczana jest przez  $BBC_\lambda(G, H)$ .

W literaturze najczęściej rozpatrywany jest szczególny przypadek kolorowania szkieletowego dla  $\lambda = 2$ , z tego powodu 2-szkieletowa liczba chromatyczna nazywana jest po prostu szkieletową liczbą chromatyczną  $BBC(G, H)$ . Ponadto, dla odróżnienia kolorowania szkieletowego od modelu klasycznego, w rozważaniach często niejawnie przyjmuje się, że parametr  $\lambda$  przyjmuje wartości większe od 1.

Głównym zastosowaniem praktycznym kolorowania szkieletowego jest przydział pasm częstotliwości np. stacjom bazowym telefonii komórkowej w taki sposób, aby istniała struktura szkieletowa gwarantująca poprawną komunikację i brak zakłóceń w przypadku, gdy nadajniki będą emitowały moc przekraczającą dopuszczalną.

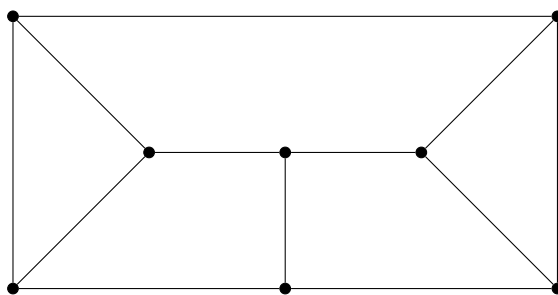
### 2.2.1. Szczególne klasy grafów

Tak jak w przypadku problemu kolorowania klasycznego wierzchołków, również dużo miejsca w literaturze dotyczącej kolorowania szkieletowego poświęcone jest grafom planarnym. Klasa ta ma przydatne własności: przede wszystkim z twierdzenia o czterech barwach wiadomo, że liczba chromatyczna dowolnego grafu planarnego nie może być większa od 4, poza tym np. wiadomo, że każdy graf planarny zawiera

wierzchołek stopnia co najwyżej 5. Co więcej, wiadomo, że dla grafu planarnego niezawierającego indukowanego podgrafu  $K_3$  zawsze istnieje 3-pokolorowanie.

Pewną szczególną podklasą grafów planarnych są tzw. grafy Halina.

**Definicja 2.11** (Halin [10]). *Graf Halina*  $G = (V, E)$  jest grafem powstałym z pewnego drzewa niezawierającego wierzchołków stopnia 2 poprzez połączenie wszystkich jego liści w cykl, zachowując planarność grafu.

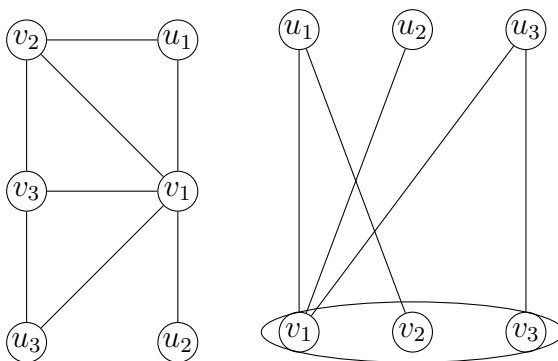


RYS. 2.1. Przykładowy graf Halina

Przykładowy graf Halina został przedstawiony na rys. 2.1. Ponieważ graf Halina jest planarny i nie jest dwudzielny,  $3 \leq \chi(G) \leq 4$ . Wykazano, że liczba chromatyczna grafu Halina jest równa 4 tylko wtedy, gdy graf jest izomorficzny z kołem nieparzystym  $W_{2k+1}$ .

Kolejną badaną klasą grafów, ze względu na swoje własności, są split grafy.

**Definicja 2.12** (Hammer, Földes [11]). *Split graf*  $G = (V, E)$  jest grafem, dla którego istnieje podział wierzchołków  $(C, I)$  taki, że zbiór  $C$  tworzy klikę, a zbiór  $I$  jest zbiorem niezależnym w  $G$ .



RYS. 2.2. Reprezentacje przykładowego split grafu



Split grafy są klasą grafów doskonałych, co oznacza, że liczba chromatyczna dowolnego ich podgrafu indukowanego jest równa rozmiarowi największej kliki w tym podgrafie, w szczególności dla całego grafu zachodzi  $\chi(G) = \omega(G)$ , gdzie  $\omega(G)$  jest liczbą klikową grafu. Dzięki temu wiele problemów NP-trudnych w ogólności, m.in. problem maksymalnej kliki oraz problem maksymalnego zbioru niezależnego, a także problem kolorowania grafów, mogą być rozwiązane w czasie wielomianowym.

Z drugiej strony, split grafy są przydatne przy wyznaczaniu górnych oszacowania  $\lambda$ -szkieletowej liczby chromatycznej dla dowolnego grafu  $G$ , gdyż można stworzyć jego nadgraf  $S(G)$ , będący split grafem, za pomocą poniższego algorytmu:

---

**Algorytm 1** Algorytm konstrukcji split grafu

---

Wybierz największy zbiór niezależny  $I$  w  $G$

$C \leftarrow V \setminus I$

**if** istnieje  $v \in I$  sąsiadujący z wszystkimi wierzchołkami z  $C$  **then**

$I \leftarrow I \setminus \{v\}$

$C \leftarrow C \cup \{v\}$

Połącz wierzchołki ze zbioru  $C$  w klikę

---

Zatem split grafów umożliwiają wyprowadzenie alternatywnych górnych oszacowań dla dowolnych grafów.

Z uwagi na wymienione właściwości, split grafy są bardzo intensywnie badaną klasą grafów. Jeżeli bowiem istnieje ograniczenie  $BBC_\lambda(S(G), H) \leq f(\lambda, k)$  dla split grafów  $S(G)$  o liczbie chromatycznej  $k$  oraz wybrany zbiór niezależny ma rozmiar  $\alpha(G) = \alpha(S(G))$ , to w rezultacie otrzymane oszacowanie wynosi

$$\begin{aligned} BBC_\lambda(G, H) &\leq BBC_\lambda(S(G), H) \leq f(\lambda, \chi(S(G))) \\ &= f(\lambda, \omega(S(G))) \leq f(\lambda, n - \alpha(S(G)) + 1) \\ &= f(\lambda, n - \alpha(G) + 1) \end{aligned}$$

W przypadku wielu klas grafów takie ograniczenie może być dokładniejsze niż wynikające z rozważań ogólnych.

### 2.2.2. Szczególne klasy szkieletów

Typowo badane klasy szkieletów są związane z praktycznym zastosowaniem kolorowania szkieletowego. Szkielet przykładowo powinien zapewniać lepsze warunki komunikacji z określonymi centrami lub też gwarantować istnienie dróg w szkielecie między ustalonymi wierzchołkami. Wśród rozpatrywanych klas szkieletów można wyróżnić:

- skojarzenie  $M$ ,
- *gwiazdozbiór*, czyli zbiór parami wierzchołkowo rozłącznych gwiazd  $S$ ,
- ścieżkę  $P$ ,
- drzewo  $T$ .

O grafach, których szkielety są indukowane przez wierzchołki skojarzeń, gwiazdozbiorów, ścieżek lub drzew będziemy mówili, że są grafami  $z$ , odpowiednio, skojarzeniem, gwiazdozbiorem, ścieżką lub drzewem w szkielecie. Należy zauważyć, że skojarzenie jest szczególnym przypadkiem zbioru gwiazd, a ścieżka – szczególnym przypadkiem drzewa.

Szczególnym przypadkiem grafów szkieletowych są grafy zdegenerowane:

**Definicja 2.13** (Lick, White [15]). Graf  $G$  jest  $d$ -zdegenerowany, jeśli jego dowolny podgraf  $H$  ma wierzchołek stopnia co najwyżej  $d$ , czyli  $\delta(H) \leq d$ .

Wyniki wyprowadzone dla tej klasy grafów można wprost zastosować do innych klas grafów, korzystając z faktu, że wiele z nich spełnia powyższą definicję dla pewnego ustalonego  $d$ . Przykładowo, grafy indukowane przez skojarzenia i drzewa są 1-zdegenerowane, natomiast grafy planarne 5-zdegenerowane.

### 2.2.3. Przyjęte założenia

W całej niniejszej pracy zdecydowano się przyjąć szereg założeń. Przede wszystkim, wszystkie rozpatrywane grafy są spójne, ponieważ analogiczne zagadnienie dla grafów niespójnych ogranicza się tylko do zidentyfikowania składowych spójności i rozwiązania problemu dla każdej składowej z osobna. Dodatkowo przyjęto, że liczba wierzchołków w grafie jest równa co najmniej 3, eliminując przypadki trywialne grafów spójnych  $K_1$  i  $K_2$ . Oba założenia prowadzą do wniosku, że tematem pracy są grafy spójne o liczbie chromatycznej  $\chi(G) \geq 2$ .

Kolejne założenie dotyczy struktury szkieletu: przyjęto, że szkielet  $H = (V', E')$  jest podgrafem spinającym cały graf  $G = (V, E)$ , czyli  $V' = V$  oraz  $E' \subseteq E$ . Pewnym uzupełnieniem tego założenia jest przyjęcie, że każdy wierzchołek grafu jest incydentny do krawędzi szkieletowej, a zatem szkielety nie posiadają wierzchołków izolowanych. Prowadzi to m.in. do wniosku, że gdy krawędzie szkieletu są skojarzeniem, to liczba wierzchołków w grafie musi być parzysta.

Przyjęto, że dla grafu  $G = (V, E)$  ze szkieletem  $H = (V', E')$  obowiązywać będą następujące oznaczenia:

- $n = |V|$  – rząd grafu,
- $m = |E|$  – rozmiar grafu,
- $m' = |E'|$  – rozmiar szkieletu.

# Rozdział 3

---

## Ograniczenia liczby $BBC_\lambda(G, H)$

Najważniejszym problemem, badanym w pracach dotyczących kolorowania szkieletowego grafów, jest zależność  $\lambda$ -szkieletowej liczby chromatycznej  $BBC_\lambda(G, H)$  od liczby chromatycznej  $\chi(G)$ . Trywialne są zależności:

$$\chi(G) \leq BBC_\lambda(G, H) \leq (\chi(G) - 1)\lambda + 1 \quad (3.1)$$

Łatwo sprawdzić, że przyporządkowując każdemu wierzchołkowi zamiast koloru  $i$  kolor  $(i - 1)\lambda + 1$  można otrzymać pokolorowanie szkieletowe grafu  $G$  z dowolnym szkieletem  $H$ . Równie łatwo można wykazać, że równość  $BBC_\lambda(G, H) = \lambda + 1$ , przy przyjętych założeniach z punktu 2.2.3 zachodzi wtedy i tylko wtedy, gdy  $G$  jest dwudzielny.

Wprowadzono następujące oznaczenia górnych ograniczeń wartości  $\lambda$ -szkieletowej liczby chromatycznej w zależności od  $\lambda$  i  $\chi(G)$ :

**Definicja 3.1** (Broersma i in. [6]).

$$\begin{aligned} \mathcal{M}(\lambda, k) &= \max\{BBC_\lambda(G, M) : M \text{ jest skojarzeniem w } G, \chi(G) = k\} \\ \mathcal{S}(\lambda, k) &= \max\{BBC_\lambda(G, S) : S \text{ jest gwiazdozbiorem w } G, \chi(G) = k\} \\ \mathcal{P}(\lambda, k) &= \max\{BBC_\lambda(G, P) : P \text{ jest ścieżką w } G, \chi(G) = k\} \\ \mathcal{T}(\lambda, k) &= \max\{BBC_\lambda(G, T) : T \text{ jest drzewem w } G, \chi(G) = k\} \end{aligned}$$

Z definicji powyższych parametrów oraz m.in. z faktów, że skojarzenie jest szczególnym przypadkiem gwiazdozbioru, zaś ścieżka to szczególny przypadek drzewa, można wyprowadzić zależności:

$$\mathcal{M}(\lambda, k) \leq \mathcal{S}(\lambda, k) \quad (3.2)$$

$$\mathcal{P}(\lambda, k) \leq \mathcal{T}(\lambda, k) \quad (3.3)$$

Zależność między liczbami chromatycznymi  $BBC_\lambda(G, H)$  a  $\chi(G)$  była również intensywnie badana dla szczególnych klas grafów, przede wszystkim dla grafów planarnych oraz split grafów. W niniejszej pracy przyjęto oznaczenia analogiczne do definicji 3.1, przykładowo dla grafów planarnych ze szkieletem w postaci skojarzenia odpowiednia funkcja jest oznaczana jako  $\mathcal{M}_{planar}(\lambda, k)$ .

Drugą z przedstawionych metod jest ograniczanie wartości  $\lambda$ -szkieletowej liczby chromatycznej grafu  $G$  w zależności od jego maksymalnego stopnia  $\Delta(G)$  oraz od postaci szkieletu. Zaletą tego podejścia jest możliwość obliczenia ograniczenia w czasie wielomianowym względem rozmiaru danych wejściowych.

W każdym z wariantów przyjęto ograniczenia sformułowane w punkcie 2.2.3 niniejszej pracy, zatem rozpatrywane są jedynie szkielety będące podgrafami spinającymi grafu macierzystego.

### 3.1. Doskonałe skojarzenie w szkielecie

W pracach Broersmy wykazano, że funkcja ograniczająca od góry wartość 2-szkieletowej liczby chromatycznej grafów ze skojarzeniem w szkielecie przyjmuje wartości zbliżone do  $\frac{4}{3}$  ich liczby chromatycznej:

**Twierdzenie 3.1.** *Broersma i in. [6]*

$$\mathcal{M}(2, k) = \begin{cases} 6 & \text{dla } k = 4 \\ \frac{4k}{3} & \text{dla } k \bmod 3 = 0 \\ \frac{4k-1}{3} & \text{dla } k \bmod 3 = 1, k \neq 4 \\ \frac{4k+1}{3} & \text{dla } k \bmod 3 = 2 \end{cases} \quad (3.4)$$

W ogólnym przypadku, funkcja  $\mathcal{M}(\lambda, k)$  zachowuje się następująco:

**Twierdzenie 3.2** (Broersma i in. [5]).

$$\mathcal{M}(\lambda, k) = \begin{cases} \lambda + k - 1 & \text{dla } k \leq \lambda \\ 2k - 2 & \text{dla } \lambda + 1 \leq k \leq 2\lambda \\ 2k - 3 & \text{dla } k = 2\lambda + 1 \\ 2k - 2\frac{k}{\lambda+1} & \text{dla } k \geq 2\lambda + 2, k \bmod (\lambda + 1) = 0 \\ 2k - 2\lfloor \frac{k}{\lambda+1} \rfloor - 1 & \text{dla } k \geq 2\lambda + 2, 1 \leq k \bmod (\lambda + 1) < \frac{\lambda+3}{2} \\ 2k - 2\lfloor \frac{k}{\lambda+1} \rfloor - 2 & \text{dla } k \geq 2\lambda + 2, k \bmod (\lambda + 1) \geq \frac{\lambda+3}{2} \end{cases} \quad (3.5)$$

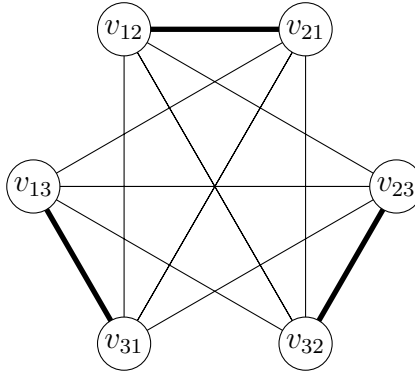
Stąd wniosek, że  $\lambda$ -szkieletowa liczba chromatyczna dla skojarzenia w szkielecie przyjmuje wartości zbliżone do  $(2 - \frac{2}{\lambda+1})$ -krotności liczby chromatycznej grafu.

Dowód w obu przypadkach – zarówno dla  $\lambda = 2$ , jak i dla zależności uogólnionej – opierał się na przyporządkowaniu poszczególnym zbiorom niezależnym  $V_i$ , odpowiadającym  $k$ -pokolorowaniu klasycznemu grafu  $G$ , jednego lub dwóch kolorów oraz wykazaniu, że uzyskane pokolorowanie jest szkieletowe.

Przykładowo, dla  $3 \leq k \leq \lambda$  wystarczy przyporządkować najpierw wszystkim wierzchołkom ze zbiorów  $V_i$  kolor  $i$  dla  $1 \leq i \leq k$ . Takie pokolorowanie niekoniecznie jest pokolorowaniem szkieletowym, dlatego należy dla każdej krawędzi skojarzenia wybrać wierzchołek oznaczony wyższym kolorem i zamienić jego kolor z  $i$  na  $\lambda + i - 1$ .

Rodzina grafów [5], dla których zachodzi  $BBC_\lambda(G, M) = \mathcal{M}(\lambda, k)$ , są grafy Turána  $T(k^2 - k, k - 1)$  – pełne grafy  $k$ -dzielne, złożone z  $k$  zbiorów niezależnych  $V_1, V_2, \dots, V_k$  o mocy  $k - 1$ :

- $V_i = \{v_{ij} : 1 \leq i, j \leq k, j \neq i\}$ ,
- $E = \{v_{ij}v_{pq} : 1 \leq i, j, p, q \leq k, j \neq i, q \neq p, p \neq i\}$ ,
- $E' = \{v_{ij}v_{ji} : 1 \leq i < j \leq k\}$ .



RYS. 3.1. Graf  $T(6, 2)$  ze skojarzeniem w szkielecie:  $BBC_\lambda(T(6, 2), M) = \mathcal{M}(\lambda, 3)$

Aby wykazać równość  $BBC_\lambda(T(k^2 - k, k - 1), M) = \mathcal{M}(\lambda, k)$  należy rozpatrzeć liczbę zbiorów  $V_i$ , do pokolorowania których konieczne jest użycie więcej niż jednego oraz dokładnie jednego koloru.

Badając zależność szkieletowej liczby chromatycznej od stopnia grafu, otrzymano w pracy [18] następujące ograniczenie:

**Twierdzenie 3.3** (Miskuf i in. [18]). *Dla dowolnego grafu  $G$  stopnia  $\Delta$  i o liczbie wierzchołków  $|V| \geq 3$  ze skojarzeniem  $M$  w szkielecie zachodzi*

$$BBC(G, M) \leq \Delta + 1 \quad (3.6)$$

*Szkic dowodu.* Punktem wyjścia jest obserwacja, że jedyny trudny przypadek zachodzi dla  $G$  będącego grafem regularnym, a  $M$  skojarzeniem doskonałym. W przeciwnym razie istnieje algorytm, kolorujący graf przy użyciu co najwyżej  $\Delta + 1$  kolorów.

Drugi krok polega na rozpatrzeniu zagadnienia najpierw dla grafów pełnych i cykli, a następnie osobno grafów spełniających i niespełniających warunki 2-spójności oraz wykazanie, że w każdym z tych przypadków istnieje uszeregowanie wierzchołków, dla którego pewni dwaj sąsiedzi ostatniego wierzchołka dostaną ten sam kolor.  $\square$

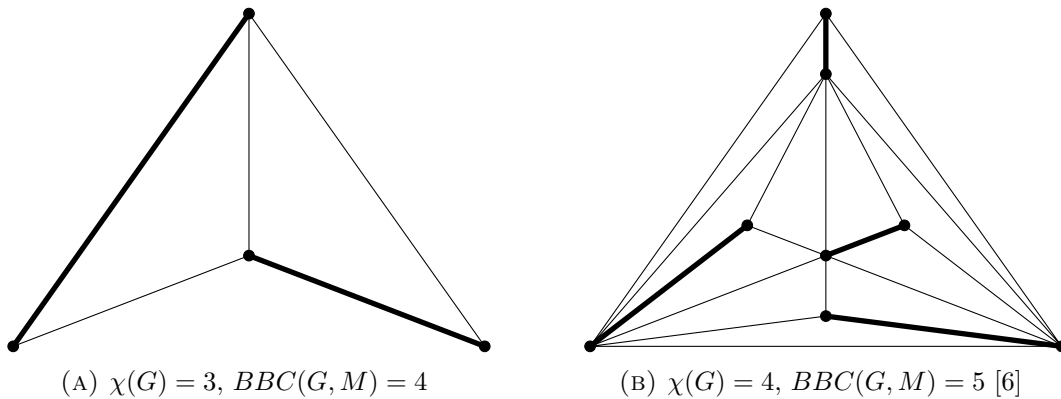
### 3.1.1. Grafy planarne

Zgodnie z twierdzeniem o czterech barwach, dla dowolnego grafu planarnego  $G$  zachodzi  $\chi(G) \leq 4$ . Zatem, z faktu, że  $\mathcal{M}_{planar}(2, k) \leq \mathcal{M}(2, k)$  oraz z twierdzenia 3.1 wynika zależność:

**Twierdzenie 3.4** (Broersma i in. [6]).

$$\mathcal{M}_{planar}(2, k) \leq \begin{cases} 3 & \text{dla } k = 2 \\ 4 & \text{dla } k = 3 \\ 6 & \text{dla } k = 4 \end{cases} \quad (3.7)$$

Łatwo wykazać, że niektóre z powyższych nierówności są równościami. W pierwszym przypadku dowolny graf dwudzielny wymaga kolorów 1 i 3. Z kolei jeśli graf zawiera  $K_3$ , w którym jedna krawędź należy do doskonałego skojarzenia  $M$  (jak na rys. 3.2a), to konieczne jest użycie co najmniej 4 kolorów. Pytaniem otwartym nadal pozostaje problem, czy dla  $\mathcal{M}_{planar}(2, 4) = 6$ . Wiadomo jedynie, że istnieje – pokazany na rys. 3.2b – graf planarny  $G$  ze skojarzeniem  $M$ , dla pokolorowania którego wymagane jest użycie pięciu kolorów, czyli  $5 \leq \mathcal{M}_{planar}(2, 4) \leq 6$ .



RYS. 3.2. Grafy planarne ze skojarzeniem w szkieletcie, ustanawiające dolne ograniczenie  $\mathcal{M}_{planar}(2, k)$  dla  $k = 3, 4$

Innym otwartym problemem jest pytanie o istnienie elementarnego dowodu zależności  $\mathcal{M}_{planar}(2, k) \leq 6$  bez wykorzystania twierdzenia o czterech barwach.

Stosując rozumowanie analogicznie do dowodu twierdzenia 3.4, można również wykazać odpowiednie zależności dla innych wartości parametru  $\lambda$ .

**Twierdzenie 3.5.** *Dla dowolnej wartości  $\lambda \geq 3$  zachodzi*

$$\mathcal{M}_{planar}(\lambda, k) = \lambda + k - 1 \quad (3.8)$$

*Dowód.* Dowód górnego ograniczenia można wyprowadzić wprost z twierdzenia 3.2 lub z faktu, że mając dane optymalne  $k$ -pokolorowanie grafu planarnego, wystarczy dla każdego wierzchołka zamienić jego kolor  $i$  na  $\lambda + i - 1$ , gdy jego sąsiad w skojarzeniu ma przypisany kolor  $j < i$ . Jest to pokolorowanie szkieletowe, ponieważ każdy wierzchołek w kolorze 1 zostanie bez zmian, a każdemu wierzchołkowi w kolorze  $k$  zostanie przypisany kolor  $\lambda + k - 1$ . W konsekwencji żadnym dwóm wierzchołkom w różnych kolorach nie można przypisać tej samej barwy.

Nietrudno wykazać, że odpowiednimi przypadkami grafów, ustanawiających dolne ograniczenie  $\mathcal{M}_{planar}(\lambda, k)$  są przykłady przedstawione na rys. 3.2a i rys. 3.2b.  $\square$

### 3.1.2. Split grafy

**Definicja 3.2.** Wierzchołek  $u$  jest nazywany *wierzchołkiem skojarzonym* (*matching neighbor*)  $mn(v)$  z wierzchołkiem  $v$ , gdy  $\{u, v\} \in E'$ .

**Definicja 3.3.** Zbiór niesąsiadów  $NN(v)$  wierzchołka  $v$  jest zbiorem wierzchołków, dla których  $\{u, v\} \notin E$ .

Zależności między  $\lambda$ -szkieletową liczbą chromatyczną a liczbą chromatyczną split grafów przedstawiają się następująco:

**Twierdzenie 3.6** (Broersma i in. [7]).

$$\mathcal{M}_{split}(\lambda, k) = \begin{cases} \lambda + 1 & \text{dla } k = 2 \\ k + 1 & \text{dla } k \geq 3 \text{ i } \lambda \leq \min\{\frac{k+1}{2}, \frac{k+5}{3}\} \\ k + 2 & \text{dla } k = 9 \text{ lub } k \geq 11 \text{ i } \frac{k+6}{3} \leq \lambda \leq \lceil \frac{k}{2} \rceil \\ \lceil \frac{k}{2} \rceil + \lambda & \text{dla } k = 3, 5, 7 \text{ i } \lambda \geq \lceil \frac{k}{2} \rceil \\ \lceil \frac{k}{2} \rceil + \lambda + 1 & \text{dla } k = 4, 6 \text{ lub } k \geq 8 \text{ i } \lambda \geq \lceil \frac{k}{2} \rceil + 1 \end{cases} \quad (3.9)$$

*Szkic dowodu.* Jeśli split graf jest dwudzielny, to do optymalnego pokolorowania szkieletowego potrzeba jedynie kolorów  $\{1, \lambda + 1\}$ . W pozostałych przypadkach należy wykonać podział zbioru wierzchołków jedną z dwóch metod, zależnie od wartości  $\lambda$  i  $k$ .

Prostsza metoda opisuje podział na 3 zbiory: zbiór niezależny  $I$ , klikę  $C_1 \subseteq C$ , zawierającą wierzchołki, których sąsiad z  $M$  także leży w  $C$  oraz klikę  $C_2 \subseteq C$ , zawierającą pozostałe wierzchołki z  $C$ .

W trudniejszych przypadkach dowód zakłada wykonanie podziału na 8 zbiorów: zbiór niezależny  $I$  dzielony jest na  $I_1$ , możliwie największy zbiór zawierający wierzchołki, których niesąsiedzi nie są połączeni krawędzią z  $M$  (o mocy nie mniejszej niż  $\lfloor \frac{|C|}{3} - \frac{|C|-|I|}{2} \rfloor$ ), oraz  $I_2$ , zawierający pozostałe wierzchołki. Podział kliki przeprowadzony jest następująco:

- $C_1 = \{v \in C : mn(v) \in C, NN(v) \cap I_1 \neq \emptyset\}$ ,
- $C_2 = \{v \in C : mn(v) \in I, NN(v) \cap I_1 \neq \emptyset\}$ ,
- $C_3 = \{v \in C : mn(v) \in C \setminus (C_1 \cup C_2)\}$ ,
- $C_4 = \{v \in C : mn(v) \in I_2\}$ ,
- $C_5 = \{v \in C : mn(v) \in I_1\}$ ,
- $C_6 = C \setminus (C_1 \cup C_2)$ .

Wyznaczone podzbiory są w ustalonej kolejności kolorowane zachłannie (czasem używając koloru 1 jako ostatniego), przy założeniu, że podczas kolorowania podzbioru pierwszeństwo mają wierzchołki o niższym kolorze sąsiada w skojarzeniu.

W pracy [19] podano dwie rodziny grafów, dla których zachodzi  $BBC_\lambda(G, H) = \mathcal{M}_{split}(\lambda, k)$ . Pierwszą zdefiniowano następująco:

- $C = \{v_i : 1 \leq i \leq k\}$ ,
- $I = \{u_i : 1 \leq i \leq k\}$ ,
- $E = \{v_i v_j : 1 \leq i < j \leq k\} \cup \{u_i v_j : 1 \leq i, j \leq k-1\} \cup \{u_k v_j : 2 \leq j \leq k\}$ ,
- $E' = \{u_i v_i : 1 \leq i \leq k\}$ .

W drugim przypadku można podzielić graf na  $\lceil \frac{k}{3} \rceil$  rozłącznych kopii  $K_{3,3}$  i  $K_{2,2}$ :

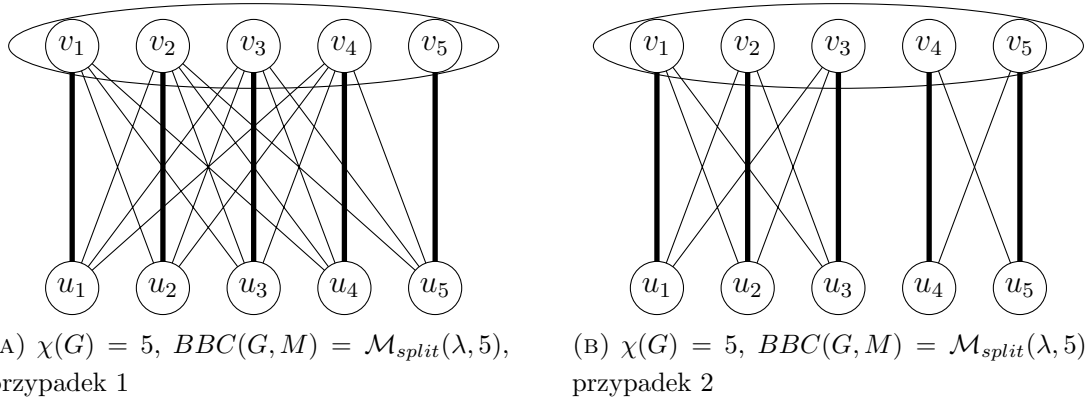
- $C = \{v_i : 1 \leq i \leq k\}$ ,
- $I = \{u_i : 1 \leq i \leq k\}$ ,
- $E = \{v_i v_j : 1 \leq i < j \leq k\} \cup \{u_i v_j : 1 \leq i, j \leq k, \lceil \frac{i}{3} \rceil = \lceil \frac{j}{3} \rceil\}$  gdy  $k \bmod 3 < 2$ ,
- $E = \{v_i v_j : 1 \leq i < j \leq k\} \cup \{u_i v_j : 1 \leq i, j \leq k, \lfloor \frac{i}{3} \rfloor = \lfloor \frac{j}{3} \rfloor\}$  gdy  $k \bmod 3 = 2$ ,
- $E' = \{u_i v_i : 1 \leq i \leq k\}$ .

Przykłady grafów z obu rodzin zostały przedstawione na rys. 3.3a i rys. 3.3b.  $\square$

## 3.2. Gwiazdozbiór w szkielecie

Naturalnym uogólnieniem skojarzenia jest gwiazdozbiór, czyli zbiór parami wierzchołkowo rozłącznych gwiazd. Zaobserwowano, że stosunek  $\mathcal{S}(\lambda, k)$  do liczby chromatycznej grafu  $k$  w granicy dla  $k \rightarrow \infty$  dąży do  $2 - \frac{1}{\lambda}$ .





Rys. 3.3. Split grafy ze skojarzeniem w szkielecie, ustanawiające dolne ograniczenie  $\mathcal{M}_{split}(\lambda, 5)$

**Twierdzenie 3.7** (Broersma i in. [5]).

$$\mathcal{S}(\lambda, k) = \begin{cases} \lambda + 1 & \text{dla } k = 2 \\ \lfloor \frac{3k}{2} \rfloor + \lambda - 2 & \text{dla } 3 \leq k \leq 2\lambda - 3 \\ k + 2\lambda - 2 & \text{dla } 2\lambda - 2 \leq k \leq 2\lambda - 1, \lambda \geq 3 \\ k + 2\lambda - 2 & \text{dla } 2\lambda - 1 \leq k \leq 2\lambda, \lambda = 2 \\ 2k - 1 & \text{dla } k = 2\lambda, \lambda \geq 3 \\ 2k - \lfloor \frac{k}{\lambda} \rfloor & \text{dla } k \geq 2\lambda + 1 \end{cases} \quad (3.10)$$

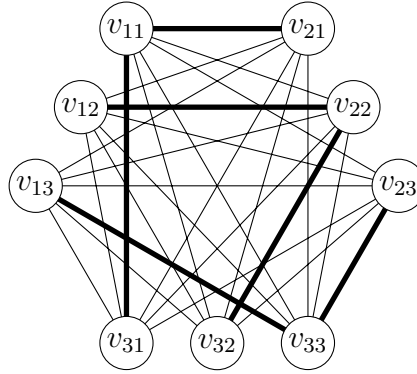
*Szkic dowodu.* Dowód opierał się na przyporządkowaniu poszczególnym zbiorom niezależnym  $V_i$ , odpowiadającym  $k$ -pokolorowaniu grafu  $G$ , jednego lub dwóch kolorów, którymi mogły być kolorowane wierzchołki z  $V_i$ .

Przykładowo, dla  $3 \leq k \leq 2\lambda - 3$  należy przydzielić  $V_i$  kolory  $\{i, k + \lambda - i - 1\}$  dla  $i \leq \frac{k}{2}$  oraz kolory  $\{2k + \lambda - i - 1, i\}$  w przeciwnym przypadku. Do pokolorowania danego wierzchołka z  $V_i$  używany jest wtedy pierwszy kolor ze zbioru, jeśli ten wierzchołek jest zarazem centrum jednej z gwiazd  $S$ , bądź też drugi – jeśli wierzchołek jest liściem gwiazdy z  $S$ .

Przykładem rodziny grafów, dla których  $\lambda$ -szkieletowa liczba chromatyczna jest równa wartości funkcji  $\mathcal{S}(\lambda, k)$  są grafy Turána  $T(k^2, k)$ , czyli pełne grafy  $k$ -dzielne, złożone z  $k$  zbiorów niezależnych  $V_1, V_2, \dots, V_k$  mocy  $k$ :

- $V_i = \{v_{ij} : 1 \leq i, j \leq k\}$ ,
- $E = \{v_{ij}v_{pq} : 1 \leq i, j, p, q \leq k, i \neq p\}$ ,
- $E' = \{v_{ii}v_{ji} : 1 \leq i, j \leq k, j \neq i\}$ .

Aby wykazać, że do pokolorowania grafów  $T(k^2, k)$  konieczne jest użycie maksymalnej możliwej liczby barw, rozpatrzono liczbę zbiorów  $V_i$ , do pokolorowania



Rys. 3.4. Graf  $T(9, 3)$  z gwiazdozbiorem w szkielecie:  $BBC_\lambda(T(9, 3), S) = \mathcal{S}(\lambda, 3)$

których konieczne jest użycie więcej niż jednego oraz dokładnie jednego koloru. Następnie wykazano, że kolory przydzielone centrom gwiazd muszą pochodzić z określonego zbioru i zbadano poszczególne przypadki dla różnych wartości  $\lambda$ .  $\square$

### 3.2.1. Split grafy

**Twierdzenie 3.8** (Broersma i in. [8]).

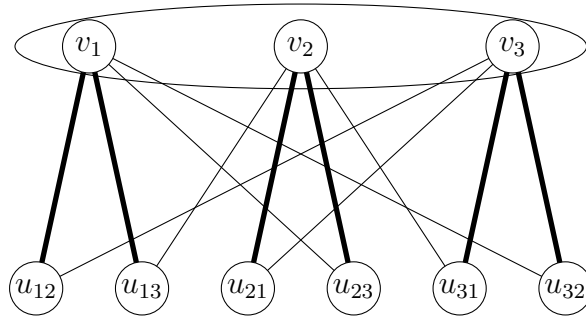
$$\mathcal{S}_{split}(\lambda, k) = \begin{cases} k + \lambda & \text{dla } k = 3 \text{ lub dla } k \geq 4 \text{ i } \lambda = 2 \\ k + \lambda - 1 & \text{w pozostałych przypadkach} \end{cases} \quad (3.11)$$

*Szkic dowodu.* Przedstawiony przez A. Salmana [19] dowód wymaga rozpatrzenia 10 podprzypadków, zależnych od ustawienia i liczby centrów w klicie i zbiorze niezależnym. Standardowa procedura dowodzenia zakłada następnie przydzielanie kolejno kolorów: centrom w klicie, liściom w klicie mającym centra w zbiorze niezależnym, potem liściom w klicie mającym centra w klicie. Na samym końcu kolorowany jest zbiór niezależny.

Przykładowa rodzina split grafów o  $k^2$  wierzchołkach, dla których  $BBC_\lambda(G, S) = \mathcal{S}_{split}(\lambda, k)$ , jest określona następująco:

- $C = \{v_i : 1 \leq i \leq k\}$ ,
- $I = \{u_{ij} : 1 \leq i, j \leq k, j \neq i\}$ ,
- $E = \{v_i v_j : 1 \leq i < j \leq k\} \cup \{u_{ij} v_p : 1 \leq i, j, p \leq k, p \neq j\}$ ,
- $E' = \{u_{ij} v_i : 1 \leq i, j \leq k, j \neq i\}$ .

$\square$



RYS. 3.5. Split graf z gwiazdobiorem w szkielecie, ustanawiający dolne ograniczenie  $S_{split}(\lambda, 3)$

### 3.3. Ścieżka Hamiltona w szkielecie

W pracy [4] wykazano, że szkieletowa liczba chromatyczna dla grafów ze szkieletem w postaci ścieżki jest ograniczona od góry zależnością od wartości liczby chromatycznej grafu  $G$ , a stosunek obu wartości nie przekracza około  $\frac{3}{2}$  dla dużych wartości  $\chi(G)$ :

**Twierdzenie 3.9** (Broersma i in. [4]).

$$\mathcal{P}(2, k) = \begin{cases} 2k - 1 & \text{dla } k \leq 4 \\ 2k - 2 & \text{dla } 5 \leq k \leq 6 \\ \frac{3k}{2} & \text{dla } k \geq 7, k \bmod 2 = 0 \\ \frac{3k+1}{2} & \text{dla } k \geq 7, k \bmod 4 = 1 \\ \frac{3k-1}{2} & \text{dla } k \geq 7, k \bmod 4 = 3 \end{cases} \quad (3.12)$$

*Szkic dowodu.* Dowód opierał się na przyporządkowaniu poszczególnym zbiorom niezależnym  $V_i$ , odpowiadającym  $k$ -pokolorowaniu grafu  $G$ , jednego lub trzech kolorów, którymi kolorowane były wierzchołki z  $V_i$ .

Dolne ograniczenie wartości funkcji można udowodnić poprzez pokazanie rodziny grafów Turána  $T(2k\Pi(k), k)$ , gdzie  $\Pi(k)$  jest liczbą takich permutacji ciągu  $(1, 1, 2, 2, \dots, k, k)$ , że dwa sąsiednie wyrazy permutacji są różne. Grafy te są grafami pełnymi  $k$ -dzielnymi, w których każdy zbiór niezależny  $V_1, V_2, \dots, V_k$  ma dokładnie  $2\Pi(k)$  wierzchołków.

Szkieletem grafu jest ścieżka Hamiltona, składająca się z  $\Pi(k)$  segmentów długości  $2k$ , odpowiadających kolejnym permutacjom ciągu  $(1, 1, 2, 2, \dots, k, k)$  bez sąsiednich powtórzeń. Każda taka permutacja wyznacza zarazem kolejność odwiedzania wierzchołków z poszczególnych zbiorów  $V_1, V_2, \dots, V_k$ , ponieważ zabronione jest przejście między dwoma wierzchołkami z tego samego zbioru  $V_i$ . Następnie, segmenty zostają połączone końcami w dowolny sposób tak, aby utworzyć ścieżkę Hamiltona.

Dowód ścisłych ograniczeń opiera się na wykazaniu ograniczeń liczności zbiorów niezależnych pokolorowanych jednym, dwoma lub co najmniej trzema kolorami oraz rozważeniu odpowiednio małych przypadków (dla  $k \leq 6$ ) osobno.  $\square$

### 3.3.1. Grafy planarne

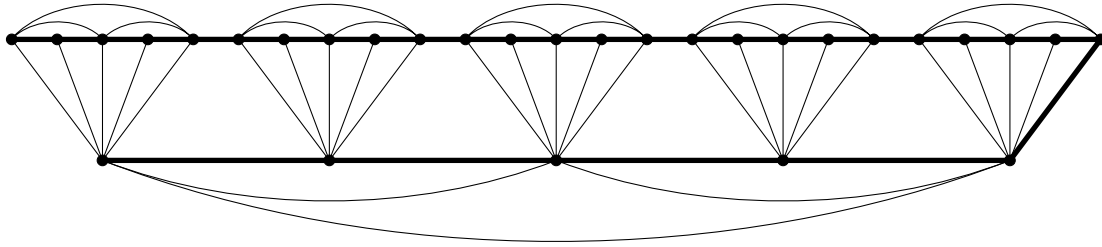
Kolejnym zagadnieniem jest górne ograniczenie szkieletowej liczby chromatycznej grafu planarnego ze ścieżką w szkielecie. Z twierdzenia 3.9 i twierdzenia o czterech barwach wynika wprost, że:

**Twierdzenie 3.10** (Broersma i in. [6]). *Dla grafów planarnych zachodzi*

$$\mathcal{P}_{\text{planar}}(2, k) \leq 2k - 1 \quad (3.13)$$

przy czym dla  $k \leq 3$  nierówność można zastąpić przez znak równości.

Dla przypadku  $\chi(G) = 4$  wykazano jedynie, że istnieje graf planarny  $G$  – przedstawiony na rys. 3.6 – taki, że dla pewnego szkieletu  $P$  zachodzi  $BBC(G, P) = 6$ . Zatem otwarty nadal jest problem wartości  $6 \leq \mathcal{P}_{\text{planar}}(2, 4) \leq 7$ , ale i nierozstrzygnięte jest pytanie, czy możliwy jest dowód nierówności  $\mathcal{P}_{\text{planar}}(2, k) \leq 7$  bez pomocy twierdzenia o czterech barwach.



Rys. 3.6. Graf planarny ze ścieżką Hamiltona w szkielecie:  $\chi(G) = 4$ ,  $BBC(G, P) = 6$  [6]

Wystarczy zaobserwować, że graf z rys. 3.6 składa się z sześciu identycznych gadżetów o 5 wierzchołkach. Jeśli istniałoby pokolorowanie pokazanego grafu przy użyciu 5 kolorów, to jeden wierzchołek z każdego gadżetu musiałby mieć kolor 3: w przeciwnym razie wierzchołki skrajne oraz centralny ścieżki szkieletowej w gadżecie musiałby otrzymać wyłącznie kolory ze zbioru  $\{1, 2\}$  albo  $\{4, 5\}$ . Takie przypisanie nie spełnia warunków pokolorowania, gdyż istniałoby dwa wierzchołki o jednakowym kolorze połączone krawędzią.

Lecz wówczas żaden z kolorów niższego gadżetu nie mógłby zawierać wierzchołka w kolorze 3 – zachodzi sprzeczność. Zatem  $BBC(G, P) > 5$ . Znalezienie pokolorowania szkieletowego powyższego grafu przy użyciu 6 kolorów nie stanowi trudności.

### 3.3.2. Split grafy

**Twierdzenie 3.11** (Broersma i in. [4]).

$$\mathcal{P}_{split}(2, k) = \begin{cases} 5 & \text{dla } k = 3 \\ k + 1 & \text{dla } k \neq 3 \end{cases} \quad (3.14)$$

*Szkic dowodu.* Nietrudno wykazać, że szkieletowa liczba chromatyczna split grafów dwudzielnych nie może być większa od 3. Zależność  $\mathcal{P}_{split}(2, 3) \leq 5$  również wynika ze wzoru dla grafów ogólnych, odpowiednią rodzinę grafów uzasadniającą dolne ograniczenie dla tego przypadku podano w pracy [4].

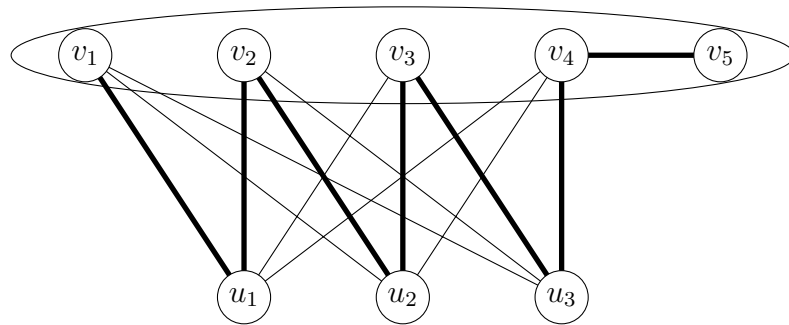
Gdy  $k \geq 4$  wystarczy rozpatrzyć dwa przypadki, w zależności od istnienia wierzchołka  $v \in C$ , niesąsiadującego w szkielecie z żadnym wierzchołkiem ze zbioru niezależnego  $I$ . Oba warianty należy następnie rozbić na pewną liczbę podprzypadków.

Przykładowo, gdy każdy wierzchołek z  $C$  ma sąsiadów w szkielecie tylko z  $I$  oraz jeden koniec  $P$  leży w klicie, to wystarczy przydzielić mu kolor  $k$ , a jego sąsiadowi w szkielecie kolor 1. Pozostałym wierzchołkom z  $I$  można przypisać kolor  $k + 1$ , a pozostałym wierzchołki z kliku kolory od 1 do  $k - 1$ , otrzymując pokolorowanie szkieletowe.

Istnieje rodzina split grafów o  $2k - 2$  wierzchołkach (dla  $k \geq 4$ ), dla których zachodzi  $BBC(G, P) = \mathcal{P}_{split}(2, k)$ :

- $C = \{v_i : 1 \leq i \leq k\}$ ,
- $I = \{u_i : 1 \leq i \leq k - 2\}$ ,
- $E = \{v_i v_j : 1 \leq i < j \leq k\} \cup \{u_i v_j : 1 \leq i \leq k - 2, 1 \leq j \leq k - 1\}$ ,
- $E' = \{v_i u_i, u_i v_{i+1} : 1 \leq i \leq k - 2\} \cup v_{k-1} v_k$ .

□



Rys. 3.7. Split graf ze ścieżką Hamiltona w szkielecie, ustanawiający dolne ograniczenie  $\mathcal{P}_{split}(2, 5)$

### 3.4. Drzewo spinające w szkielecie

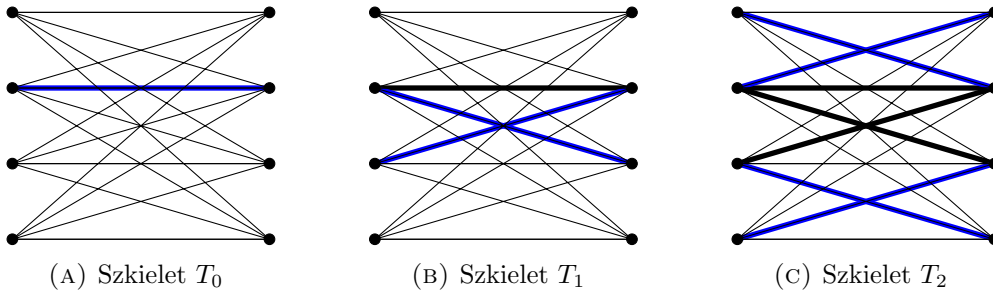
W ramach problemu kolorowania szkieletowego badano również przypadek, w którym szkieletem grafu jest drzewo spinające. Otrzymana zależność jest zaskakująco prosta:

**Twierdzenie 3.12** (Salman [19]).

$$\mathcal{T}(2, k) = 2k - 1 \quad (3.15)$$

*Szkic dowodu.* Łatwo można zauważyć, że dla dowolnego grafu  $G$  z dowolnym szkieletem  $H$  (w tym również z drzewem  $T$ ) zachodzi  $BBC(G, H) \leq 2\chi(G) - 1$ , gdyż wystarczy znaleźć optymalne pokolorowanie grafu  $G$  i zamienić każdy kolor  $i$  na  $2i - 1$ , aby otrzymać pokolorowanie szkieletowe.

Z drugiej strony, istnieje rodzina grafów  $G$ , do pokolorowania której należy użyć co najmniej  $2\chi(G) - 1$  kolorów: są to pełne grafy  $k$ -dzielne, zwane grafami Turána  $T(k^{k+1}, k)$ . Ich szkielet zdefiniowany jest rekurencyjnie: Począwszy od losowo wybranego wierzchołka grafu, tworzącego drzewo początkowe, w każdej spośród  $k$  iteracji do każdego wierzchołka  $v \in T \cap V_i$  przyłączanych jest  $k - 1$  jego sąsiadów w  $G$ , po jednym z każdego zbioru  $V_1, V_2, \dots, V_{i-1}, V_{i+1}, \dots, V_k$ . Przykładowy graf z rodziny został przedstawiony na rys. 3.8 wraz z kolejnymi etapami konstrukcji szkieletu.  $\square$



RYS. 3.8. Graf  $T(8, 2)$  z drzewem spinającym w szkielecie:  $BBC_\lambda(T(8, 2), T) = \mathcal{T}(2, 2)$

Postawiona w pracy [4] hipoteza, czy dla każdego grafu beztrójkąowego ( $K_3$ -free) z dowolnie wybranym drzewem spinającym zachodzi  $BBC(G, T) \leq \chi(G) + c$  dla pewnego  $c > 0$  została rozwiązana negatywnie. Konstrukcja przedstawiona przez Miskufa i in. w [17] pokazuje istnienie rodziny grafów beztrójkąowych  $R_n$  takich, że  $BBC(R_n, T_n) = 2\chi(R_n) - 1 = 2n - 1$ .

Inne badania dotyczyły zależności między  $\lambda$ -szkieletową liczbą chromatyczną a talią grafu (ang. *girth*), czyli długością najkrótszego cyklu w grafie

**Twierdzenie 3.13** (Wang i in. [22]). *Dla każdego grafu  $G$  o talii  $g(G) \geq 10$  istnieje drzewo spinające  $T$  takie, że*

$$BBC(G, T) \leq 4 \quad (3.16)$$

Problemem otwartym pozostaje pytanie o minimalną wartość talii grafu  $G$ , zapewniającą istnienie drzewa spinającego, dla którego  $BBC(G, T) \leq 4$ . Wiadomo jedynie, że parametr ten ma wartość z zakresu od 4 do 10.

### 3.4.1. Grafy planarne

Zależność dla grafów planarnych z drzewem spinającym w szkielecie wynika wprost z twierdzenia 3.12 i przedstawia się następująco:

**Twierdzenie 3.14** (Salman [19]).

$$\mathcal{T}_{planar}(2, k) \leq 2k - 1 \quad (3.17)$$

Odpowiednie rozważania są identyczne, jak przedstawione w punkcie 3.3.1 i dla  $k \leq 3$  również można udowodnić, że zachodzi równość. Podobnie jak w przypadku ścieżki, nierozstrzygnięty jest problem wartości  $6 \leq \mathcal{T}_{planar}(2, 4) \leq 7$ , jak i pytanie o istnienie dowodu nierówności  $\mathcal{T}_{planar}(2, k) \leq 7$  bez wykorzystania twierdzenia o czterech barwach.

### 3.4.2. Grafy Halina

Wykazano, że dla grafów Halina z wewnętrznym drzewem spinającym zachodzi:

**Twierdzenie 3.15** (Wang i in. [22]).

$$\mathcal{T}_{Halin}(2, k) = \begin{cases} 5 & \text{dla nieparzystej liczby liści w szkielecie oraz liści tylko} \\ & \text{na parzystych lub tylko na nieparzystych poziomach drzewa} \\ 4 & \text{w pozostałych przypadkach} \end{cases} \quad (3.18)$$

W tej samej pracy osiągnięto wynik znacznie silniejszy: dla dowolnego grafu dwudzielnego  $B$  o co najmniej 3 liściach, łącząc jego liście w cykl w dowolnym porządku otrzymujemy  $BBC(B \cup C, B) = 5$  tylko wtedy, gdy liczba liści w  $B$  jest nieparzysta, a odległości między tymi liśćmi w  $B$  są parzyste.

### 3.4.3. Split grafy

**Twierdzenie 3.16** (Broersma i in. [4]).

$$\mathcal{T}(2, k) = \begin{cases} 3 & \text{dla } k = 2 \\ k + 2 & \text{dla } k \geq 3 \end{cases} \quad (3.19)$$

Powyższą zależność można uogólnić na dowolne wartości parametru  $\lambda$ :

**Twierdzenie 3.17** (Salman [19]).

$$\mathcal{T}_{split}(\lambda, k) = \begin{cases} 1 + \lambda & \text{dla } k = 2 \\ k + \lambda & \text{dla } k \geq 3 \end{cases} \quad (3.20)$$

*Szkic dowodu.* Dowód opiera się na rozpatrzeniu ograniczenia szkieletu do kliki. Jeśli to ograniczenie tworzy gwiazdę, to wówczas można przyporządkować centrum gwiazdy w klice kolor  $k + \lambda - 1$ , natomiast pozostałe wierzchołki z  $C$  pokolorować przy użyciu  $\{1, 2, \dots, k - 1\}$ . Wierzchołkom ze zbioru niezależnego można przyporządkować wtedy albo kolor  $k + \lambda$ , albo odpowiedni kolor niesąsiada w  $G$ .

W drugim przypadku istnieją dwa wierzchołki  $x, y \in C$  niepołączone krawędzią w  $T$ , zatem można je pokolorować z odstępem mniejszym niż  $\lambda$ . Wierzchołki ze zbioru niezależnego są następnie kolorowane przy użyciu kolorów 1 i  $k + \lambda$ .

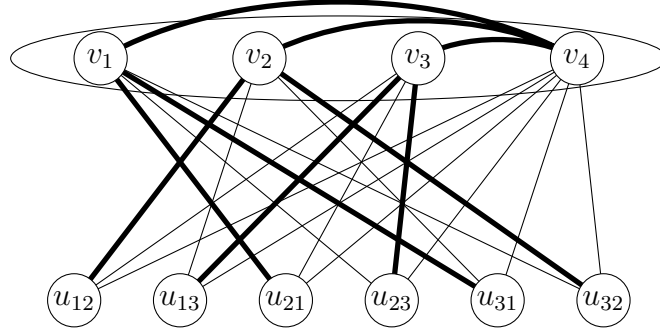
Przykładowa rodzina split grafów o  $k^2$  wierzchołkach i możliwie dużej  $\lambda$ -szkieletowej liczbie chromatycznej może być zdefiniowana następująco:

- $C = \{v_i : 1 \leq i \leq k\}$ ,
- $I = \{u_{ij} : 1 \leq i, j \leq k - 1, j \neq i\}$ ,
- $E = \{v_i v_j : 1 \leq i < j \leq k\} \cup \{u_{ij} v_l : 1 \leq i, j \leq k - 1, 1 \leq l \leq k, j \neq i\}$ ,
- $E' = \{v_i v_k : 1 \leq i \leq k - 1\} \cup \{u_{ij} v_j : 1 \leq i, j \leq k - 1, j \neq i\}$ .

□

Gdyby istniało pokolorowanie powyższego grafu przy użyciu co najwyżej  $k + \lambda - 1$  kolorów, to wierzchołki klikki  $C$  musiałyby być pokolorowane przy użyciu zestawu kolorów  $\{1, 2, \dots, k - 1, \lambda + k - 1\}$  (lub, symetrycznie,  $\{1, \lambda + 1, \lambda + 2, \dots, \lambda + k - 1\}$ ). Zatem, istniałyby wierzchołki  $v_i$  i  $v_j$  o barwach odpowiednio:  $k - 2$  i  $k - 1$ . Jak łatwo sprawdzić, wówczas wierzchołkowi  $u_{ij}$  nie można przyporządkować żadnego koloru mniejszego niż  $k + \lambda$ .





Rys. 3.9. Split graf z drzewem spinającym w szkielecie, ustanawiający dolne ograniczenie  $\mathcal{T}_{split}(\lambda, 5)$

### 3.5. Szkielet d-zdegenerowany

**Twierdzenie 3.18** (Miskuf i in. [18]). *Dla grafu  $G$  o maksymalnym stopniu  $\Delta$  z  $d$ -zdegenerowanym szkieletem  $H$  zachodzi*

$$BBC(G, H) \leq \Delta + d + 1 \quad (3.21)$$

Można również sformułować to twierdzenie w postaci ogólniejszej:

**Twierdzenie 3.19.** *Dla grafu  $G$  o maksymalnym stopniu  $\Delta$  z  $d$ -zdegenerowanym szkieletem  $H$  zachodzi*

$$BBC_\lambda(G, H) \leq (\lambda - 1)\Delta + d + 1 \quad (3.22)$$

W dowodzie twierdzenia wykorzystany zostanie, przedstawiony poniżej, prosty algorytm kolorowania grafów z  $d$ -zdegenerowanym szkieletem.

---

**Algorytm 2** Algorytm szkieletowego kolorowania grafu

---

Uporządkuj wierzchołki grafu  $G$  tak, że  $v_i$  ma co najwyżej  $d$  sąsiadów w szkielecie wśród poprzedników

**for**  $i \in \{1, 2, \dots, (\lambda - 1)\Delta + d + 1\}$  **do**

**for**  $j \in \{1, 2, \dots, n\}$  **do**

**if**  $v_j$  nie jest pokolorowany **then**

**if**  $i \notin \{c(v_j) : v_j \in N_G(v_i), j < i\}$  **then**

**if**  $i \notin \{c(v_j) - \lambda + 1, \dots, c(v_j) - 1 : v_j \in N_H(v_i), j < i\}$  **then**

$c(v_j) \leftarrow i$

---

*Dowód twierdzenia 3.19.* Wystarczy wykazać, że Algorytm 2 zwraca szkieletowe pokolorowanie grafu przy użyciu  $(\lambda - 1)\Delta + d + 1$  kolorów. Jest tak, gdyż algorytm nigdy

nie przydzielili wierzchołkowi koloru naruszającego warunki pokolorowania szkieletowego.

Z drugiej strony, maksymalna liczba kolorów zabronionych dla dowolnego wierzchołka  $v_i$  nie przekracza  $(\lambda - 1)\Delta + d$ , a zatem gdy można skorzystać z  $(\lambda - 1)\Delta + d + 1$  kolorów, zawsze przynajmniej jeden nie jest zabroniony.  $\square$

Ponieważ drzewa są grafami 1-zdegenerowanymi, w rezultacie otrzymywane jest ograniczenie

$$BBC_\lambda(G, T) \leq (\lambda - 1)\Delta + 2 \quad (3.23)$$

w szczególności dla  $\lambda = 2$

$$BBC(G, T) \leq \Delta + 2 \quad (3.24)$$

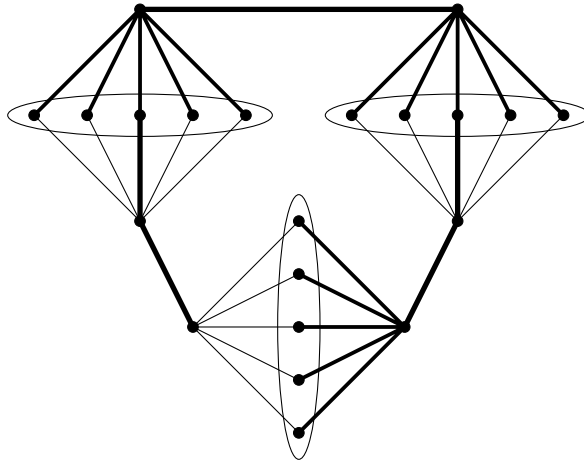
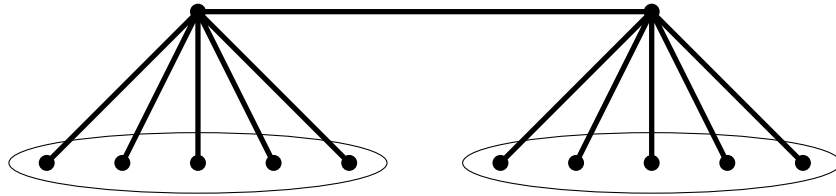
W pracy [18] wykazano również, że istnieje rodzina grafów z drzewami w szkielecie, spełniająca równość  $BBC(G, T) = \Delta + 2$ . Są to grafy o  $3t + 6$  wierzchołkach i stopniu  $\Delta = t + 1$ . Oznaczając wierzchołki grafu przez  $v_1, v_2, \dots, v_{3t+6}$ , można zdefiniować zbiory krawędzi następująco:

$$\begin{aligned} E = & \{v_i v_j : i, j \in \{1, 2, \dots, t+1\}\} \cup \{v_i v_j : i, j \in \{t+2, t+3, \dots, 2t+2\}\} \\ & \cup \{v_i v_j : i, j \in \{t+3, t+4, \dots, 2t+3\}\} \\ & \cup \{v_i v_j : i, j \in \{2t+4, 2t+5, \dots, 3t+4\}\} \\ & \cup \{v_i v_j : i, j \in \{2t+5, 2t+6, \dots, 3t+5\}\} \\ & \cup \{v_i v_j : i, j \in \{3t+6, 1, 2, \dots, t\}\} \\ & \cup \{v_{t+1} v_{t+2}, v_{2t+3} v_{2t+4}, v_{3t+5} v_{3t+6}\} \end{aligned}$$

$$\begin{aligned} E' = & \{v_i v_{t+1} : i \in \{1, 2, \dots, t\}\} \cup \{v_i v_{t+2} : i \in \{t+3, t+4, \dots, 2t+2\}\} \\ & \cup \{v_i v_{2t+4} : i \in \{2t+5, 2t+6, \dots, 3t+4\}\} \\ & \cup \{v_{t+1} v_{t+2}, v_{2t+3} v_{2t+4}, v_{3t+5} v_{3t+6}, v_{3t+6} v_1, v_{2t+2} v_{2t+3}\} \end{aligned}$$

Ponieważ w grafie istnieje kilka o rozmiarze  $\Delta$  z gwiazdą rozpinającą, jasne jest, że  $BBC(G, T) \geq \Delta + 1$ . Dowód opiera się na obserwacji, że każde kolorowanie z użyciem  $\Delta + 1$  kolorów gadżetu z rys. 3.11 nadaje centrom podwójnej gwiazdy kolory 1 oraz  $\Delta + 1$ . Jeśli przyjmiemy, że  $v_{t+1}$  otrzyma kolor 1 (rozważania w drugim przypadku wyglądają analogicznie), to  $v_{3t+6}$  musi otrzymać kolor 1 lub 2.

Z drugiej strony,  $v_{t+2}$  otrzyma kolor  $\Delta + 1$ , a  $v_{2t+3}$  wobec tego kolor  $\Delta + 1$  lub  $\Delta$ . Ponieważ wierzchołki  $\{v_{2t+4}, v_{2t+5}, \dots, v_{3t+4}\}$  indukują klikę a  $v_{2t+3} v_{2t+4} \in E'$ , to  $v_{2t+4}$  musi otrzymać kolor 1. To jednak prowadzi do wniosku, że  $v_{3t+5}$  ma kolor 1 lub 2 – sprzeczność.

Rys. 3.10. Przykład grafu o  $\Delta = 6$  i  $BBC(G, T) = 8$  [18]Rys. 3.11. Gadżet dla rodziny grafów o  $BBC(G, T) = \Delta + 2$  [18]

Wobec tego,  $BBC(G, T) > \Delta + 1$  – a pokazanie  $(\Delta + 2)$ -pokolorowania szkieletowego nie przedstawia większych trudności.

Istnieje również przykład rodziny grafów spełniających  $BBC(G, T) = \Delta + 2$  zarazem takich, że stopień grafu  $G$  jest większy od stopnia szkieletu  $T$ . Ich konstrukcja opiera się również na przedstawionym powyżej gadżecie i wynikającej z niego obserwacji.

# Rozdział 4

---

## Złożoność obliczeniowa kolorowania szkieletowego

### 4.1. Grafy ogólne

Oprócz wyznaczania wartości  $\lambda$ -szkieletowej liczby chromatycznej grafów można postawić również pytanie o złożoność obliczeniową problemu kolorowania szkieletowego:

**Problem 4.1.** Dla jakich wartości  $l$  można w czasie wielomianowym rozstrzygnąć, czy  $BBC_\lambda(G, H) \leq l$ ?

W ogólności problem wyznaczania  $\lambda$ -szkieletowej liczby chromatycznej grafu jest NP-trudny, zatem uzasadnione jest poszukiwanie szczególnych przypadków, dla których istnieje rozwiązanie wielomianowe oraz ustalenie granicy trudności problemu. Podstawowe twierdzenie dla kolorowania szkieletowego grafów brzmi następująco:

**Twierdzenie 4.1** (Broersma i in. [5]). *Dla danego grafu  $G$  ze szkieletem  $H$  można rozstrzygnąć w czasie wielomianowym, czy  $BBC_\lambda(G, H) \leq l$  dla każdego  $l \leq \lambda + 1$ .*

*Dowód.* Ponieważ  $H$  jest niepusty, dla każdego grafu  $G$  zachodzi  $BBC_\lambda(G, H) > \lambda$ , czyli dla dowolnego  $l \leq \lambda$  problem jest trywialny – takie  $l$ -pokolorowanie nie istnieje. Z kolei dla  $l = \lambda + 1$  należy zauważyć, że problem jest równoważny problemowi kolorowania grafu  $G$  dwoma kolorami. Jeśli graf  $G$  jest dwudzielny, to oczywiście zamieniając kolor 2 na  $\lambda + 1$  można otrzymać pokolorowanie szkieletowe.

Jeżeli natomiast istnieje pewne kolorowanie przy użyciu  $\lambda + 1$  kolorów, spełniające również warunki szkieletowe, to użyte kolory muszą pochodzić ze zbioru  $\{1, \lambda + 1\}$ , gdyż, zgodnie z założeniami przyjętymi w punkcie 2.2.3 pracy, każdy wierzchołek

ma sąsiada w szkielecie. Jednak skoro możliwe jest kolorowanie przy użyciu tylko 2 kolorów, to graf w konsekwencji jest dwudzielny.

Ponieważ dwudzielność grafu można sprawdzić w czasie wielomianowym, więc również w czasie wielomianowym można rozstrzygnąć, czy  $BBC_\lambda(G, H) \leq \lambda + 1$ .  $\square$

Dla pewnych klas szkieletów  $H$  można wykazać również silniejsze twierdzenie:

**Twierdzenie 4.2** (Broersma i in. [5]). *Dla danego grafu  $G$  ze spójnym i dwudzielnym szkieletem  $H$  można rozstrzygnąć w czasie wielomianowym, czy  $BBC_\lambda(G, H) \leq l$  dla każdego  $l \leq \lambda + 2$ .*

*Dowód.* Dowód dla  $l \leq \lambda + 1$  wynika wprost z twierdzenia 4.1, zatem należy osobno rozpatrzyć tylko przypadek graniczny. Jeśli  $l = \lambda + 2$ , to użyte kolory muszą należeć do zbioru  $\{1, 2, \lambda + 1, \lambda + 2\}$ . W przeciwnym razie, zachodziłaby sprzeczność z warunkiem incydencji wierzchołka z co najmniej jedną krawędzią szkieletu.

Przyjmijmy, za A. Salmanem [19]: niech  $V = V_0 \cup V_1$  będzie podziałem wierzchołków na partycje zgodnie z dwudzielnością szkieletu  $H$ . Dla ustalenia uwagi można przyjąć, że w zbiorze  $V_0$  znajduje się pewien wierzchołek, oznaczony barwą 1 lub 2. Wówczas wszystkie wierzchołki z  $V_0$  muszą być pokolorowane kolorem 1 lub 2, a wszystkie wierzchołki z  $V_1$  – kolorem  $\lambda + 1$  lub  $\lambda + 2$ . Czyli odpowiednie podgrafy indukowane w  $G$  muszą być dwudzielne.

Ostatecznie,  $BBC_\lambda(G, H) \leq \lambda + 2$  wtedy i tylko wtedy, gdy spełnione są następujące 3 warunki:

- podgraf  $G$  indukowany przez  $V_0$  jest dwudzielny,
- podgraf  $G$  indukowany przez  $V_1$  jest dwudzielny,
- żadna krawędź z  $H$  nie łączy wierzchołków o kolorach 2 i  $\lambda + 1$ .

Przekształcając warunki na problem 2-SAT: dla każdego  $v \in V$  wprowadzane są 2 zmienne,  $x_v$  i  $y_v$ . Wartości logiczne literałów  $x_v, \sim x_v$  odpowiadają przyporządkowaniu odpowiednio kolorów 1 i 2, analogicznie  $y_v, \sim y_v$  – kolorów  $\lambda + 1$  i  $\lambda + 2$ . Graf  $G[V_0] = (V_0, E_0)$  jest dwudzielny wtedy i tylko wtedy, gdy istnieje wartościowanie spełniające klauzule  $(x_u \vee x_v) \wedge (\sim x_u \vee \sim x_v)$  dla każdej krawędzi  $\{u, v\} \in E_0$ . Podobna zależność zachodzi również dla grafu  $G[V_1]$ . Wreszcie, każda krawędź szkieletowa  $\{u, v\} \in E'$  (zakładając, że  $u \in V_0$ ) spełnia warunek pokolorowania, jeśli równoważnie istnieje wartościowanie spełniające klauzulę  $x_u \wedge \sim y_v$ .

Ponieważ problem 2-SAT jest rozwiązywalny w czasie wielomianowym, zatem również w czasie wielomianowym można rozstrzygnąć, czy  $BBC_\lambda(G, H) \leq \lambda + 2$ .  $\square$

Z powyższego twierdzenia wynika także fakt, że problem znalezienia szkieletowego  $k$ -pokolorowania grafu jest również możliwy do rozwiązania dla szkieletów  $H$  o ustalonej liczbie składowych spójności, ograniczonej przez  $O(\log |V|)$ . Wystarczy

powiem wygenerować wszystkie możliwe podziały na odpowiednie zbiory niezależne  $V = V_0 \cup V_1$  i wyznaczyć dla nich odpowiednie formuły 2-SAT, a to można uczynić w czasie wielomianowym względem  $|V|$ .

Dla dowolnego grafu dwudzielnego niespójnego  $H$  istnieje aż do  $2^{\frac{|V|}{2}}$  sposobów wyznaczenia podziału na odpowiednie podzbiory  $V_0$  i  $V_1$ , a zatem pesymistycznym przypadku niemożliwe jest sprawdzenie wszystkich wariantów w czasie wielomianowym. Okazuje się, że dla skojarzeń w szkielecie, granica trudności problemu przebiega między  $l = \lambda + 1$  a  $l = \lambda + 2$ . W pierwszym przypadku istnieje algorytm deterministyczny, odpowiadający w czasie wielomianowym na pytanie o istnienie pokolorowania szkieletowego z użyciem co najwyżej  $l$  kolorów. Z kolei w drugim, problem jest trudny i wątpliwe jest, czy podobny algorytm w ogóle istnieje.

**Twierdzenie 4.3** (Broersma i in. [5]). *Problem rozstrzygnięcia, czy dla danego grafu  $G$  z doskonałym skojarzeniem  $M$  w szkielecie zachodzi  $BBC_\lambda(G, M) \leq l$  jest NP-zupełny dla każdego  $l \geq \lambda + 2$ .*

*Dowód.* Dowód polega na redukcji z problemu klasycznego kolorowania wierzchołkowego. Niech graf  $H$  o  $n$  wierzchołkach będzie instancją dla problemu  $k$ -kolorowania. Z grafu  $H$  można utworzyć graf  $G$  poprzez dodanie nowych wierzchołków  $u_1, \dots, u_n$  oraz połączenie krawędziami odpowiednich wierzchołków  $u_i$  z  $v_i$ . Wprowadzone krawędzie tworzą szkielet w  $G$ , będący oczywiście zarazem doskonałym skojarzeniem.

Następnie, rozpatrywane są dwa przypadki: w pierwszym należy wykazać, że jeśli  $\lambda \geq 3$  oraz  $l = \lambda + t$  dla pewnego  $t = 2, \dots, \lambda - 1$ , to problem  $l$ -kolorowania szkieletowego grafu  $G$  jest równoważny problemowi  $2t$ -kolorowania grafu  $H$ . Jeśli istnieje  $l$ -pokolorowanie szkieletowe, to – z uwagi na fakt, że każdy wierzchołek jest incydentny do pewnej krawędzi z  $M$  – kolory  $\{t+1, t+2, \dots, \lambda\}$  nie mogą być użyte. Zatem przekolorowując wierzchołki  $v_i$  o kolorze  $\lambda + i$  dla  $i = 1, 2, \dots, t$  na kolory  $t+i$  można otrzymać pokolorowanie wierzchołkowe dla  $2t$  kolorów. Stosując operację odwrotną nietrudno wyprowadzić szkieletowe pokolorowanie grafu  $G$  za pomocą  $l$  kolorów z danego  $2t$ -kolorowania grafu  $H$ .

Drugi przypadek zakłada  $\lambda \geq 2$  oraz  $l \geq 2\lambda$ . Wówczas problem  $l$ -kolorowania szkieletowego grafu  $G$  jest równoważny problemowi  $l$ -kolorowania grafu  $H$ . W jedną stronę dowód jest oczywisty:  $\chi(H) \leq \chi(G) \leq BBC_\lambda(G, M) = l$ . Z drugiej strony, mając dane  $l$ -pokolorowanie grafu  $H$  wystarczy przydzielać wierzchołkom  $u_i$  kolory 1 lub  $l$ , aby różnica kolorów była równa co najmniej  $\lambda$ . Ponieważ różnica  $l - 1$  jest odpowiednio duża, jeden z kolorów zawsze będzie dozwolony.  $\square$

**Twierdzenie 4.4** (Salman [19]). *Problem rozstrzygnięcia, czy dla danego grafu  $G$  ze ścieżką Hamiltona  $P$  w szkielecie zachodzi  $BBC_\lambda(G, P) \leq l$  jest NP-zupełny dla każdego  $l \geq \lambda + 3$ .*

*Dowód.* Dowód również polega na redukcji z problemu klasycznego kolorowania wierzchołkowego. W przypadku, gdy  $l = \lambda + t$  dla pewnego  $t = 3, \dots, \lambda$  z instancji  $H$  o  $n$  wierzchołkach dla problemu  $k$ -kolorowania konstruowany jest graf  $G$  poprzez dodanie  $n - 1$  wierzchołków oznaczonych  $a_1, a_2, \dots, a_{n-1}$ . Ścieżka Hamiltona  $P$  w szkieletcie jest skonstruowana z krawędzi  $v_1a_1, a_1v_2, \dots, a_{n-1}v_n$ . Okazuje się, że problem  $l$ -kolorowania szkieletowego takiego grafu  $G$  jest równoważny problemowi  $t$ -kolorowania grafu  $H$ .

Wymagane pokolorowanie może korzystać jedynie z kolorów  $\{1, 2, \dots, t\} \cup \{\lambda + 1, \dots, \lambda + t\}$ . Ponieważ szkielet  $P$  jest dwudzielny, to wystarczy zauważyć, że wierzchołki ścieżki muszą być na przemian pokolorowane barwami z pierwszego i drugiego podzbioru. Z drugiej strony, dla zadanego  $t$ -pokolorowania grafu  $H$  wystarczy przydzielić wierzchołkom  $a_i$  kolor  $\lambda + t$  aby otrzymać pokolorowanie szkieletowe grafu  $G$ .

Jeśli  $l \geq 2\lambda + 1$ , to z instancji  $H$  o  $n$  wierzchołkach dla problemu  $k$ -kolorowania konstruowany jest graf  $G$  poprzez dodanie  $3(n - 1)$  wierzchołków, oznaczonych  $a_i, b_i, c_i$  dla  $i = 1, 2, \dots, n - 1$ . Dodatkowe krawędzie  $v_ia_i, a_ib_i, b_ic_i, c_iv_{i+1}$ , stanowiące szkielet grafu, tworzą zarazem ścieżkę Hamiltona w  $G$ . Problem  $l$ -kolorowania szkieletowego tak skonstruowanego grafu  $G$  jest równoważny problemowi  $l$ -kolorowania grafu  $H$ . W jedną stronę dowód jest oczywisty, gdyż  $\chi(H) \leq \chi(G) \leq BBC_\lambda(G, P) = l$ . Z drugiej strony, mając dane  $l$ -pokolorowanie grafu  $H$  wystarczy przydzielić:

- wierzchołkom  $a_i$ : jeśli  $v_i$  ma kolor większy niż  $\lambda + 1$  – kolor 1, jeśli nie –  $l$ ,
- wierzchołkom  $b_i$  kolor  $\lambda + 1$ ,
- wierzchołkom  $c_i$ : jeśli  $v_{i+1}$  ma kolor większy niż  $\lambda + 1$  – kolor 1, jeśli nie –  $l$ .

□

## 4.2. Split grafy

Analogiczny problem dla innych klas grafów nie był dotychczas rozpatrywany. Dla split grafów można jednak wykazać dla przypadku  $\lambda = 2$  następujące twierdzenie:

**Twierdzenie 4.5.** *Dla danego split grafu  $G$  ze skojarzeniem  $M$  w szkieletcie dla każdego  $l$  można rozstrzygnąć w czasie wielomianowym, czy  $BBC(G, H) \leq l$ .*

*Dowód.* Dowód, zamieszczony w pracy [21], korzysta z obserwacji Broersmy [3], że dla każdego split grafu ze skojarzeniem w szkieletcie zachodzi  $\chi(G) \leq BBC(G, M) \leq \chi(G) + 1$ . Problem znalezienia liczby klikowej  $\omega(G)$  – a zarazem liczby chromatycznej  $\chi(G)$  – dla zadanego split grafu jest problemem wielomianowym, zatem

jedynym trudnym przypadkiem może być rozstrzygnięcie prawdziwości równości  $BBC(G, H) = \chi(G)$  dla danego grafu  $G$  ze szkieletem  $H$ .

Okazuje się, że jeśli  $BBC(G, H) = \chi(G)$ , to każdemu wierzchołkowi z kliki  $v$  musi odpowiadać pewien podzbiór  $I(v)$  (być może pusty) wierzchołków ze zbioru niezależnego, którym przydzielono ten sam kolor. Ponieważ dwa wierzchołki połączone krawędzią muszą mieć różne kolory, wiadomo, że  $I(v) \subseteq NN(v)$ . Problemem pozostaje zatem tylko podział wierzchołków z  $I$  na klasy kolorów tak, aby nie naruszyć warunków nakładanych przez krawędzie szkieletowe grafu.

Aby to rozstrzygnąć, dokonywana jest konstrukcja tzw. digrafu konfliktów. Przyjmując oznaczenia wierzchołków z kliki  $C$   $v_1, v_2, \dots$  oraz skojarzonych z nimi wierzchołków ze zbioru niezależnego  $I$  (nie wszystkie muszą istnieć)  $u_1, u_2, \dots$ , graf konfliktów powstaje poprzez wprowadzenie krawędzi z  $w_i$  do  $w_j$  wtedy, gdy nie istnieje analogiczna krawędź między  $v_i$  a  $u_j$ . Jeśli w tak otrzymanym grafie możliwy jest wybór pseudodrzewa spinającego  $F$ , aby móc pokolorować szkieletowo graf  $K_{\chi(G)}$  ze szkieletem  $F$  przy użyciu  $\chi(G)$  kolorów, to oryginalny graf  $G$  ze szkieletem  $M$  również ma szkieletowe  $\chi(G)$ -pokolorowanie.  $\square$

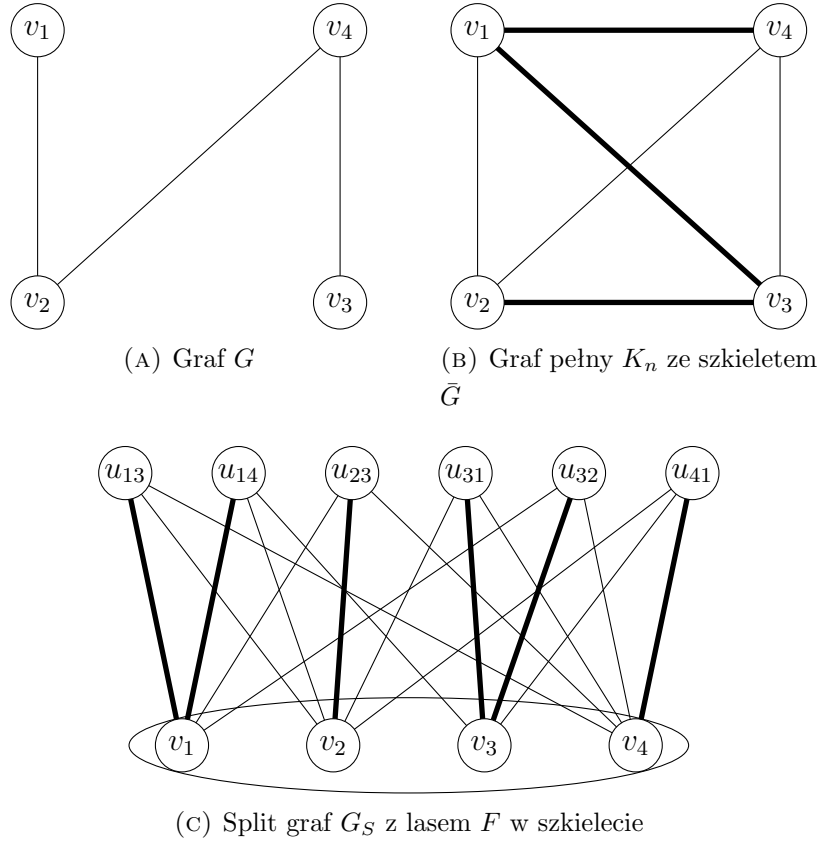
**Twierdzenie 4.6.** *Problem rozstrzygnięcia, czy dla danego split grafu  $G$  z lasem  $F$  w szkielecie zachodzi  $BBC(G, F) \leq l$  jest NP-zupełny dla  $l = \chi(G)$ .*

*Dowód.* Pytanie o istnienie w grafie  $G$  o  $n$  wierzchołkach ścieżki Hamiltona jest równoważne pytaniu o możliwość szkieletowego pokolorowania grafu  $K_n$  ze szkieletem  $\bar{G}$  przy użyciu  $n$  kolorów ( $\bar{G}$  oznacza dopełnienie grafu  $G$ ). Wystarczy zaobserwować, że wierzchołki mogą mieć sąsiednie kolory tylko wtedy, gdy krawędź między nimi nie należy do szkieletu. Zatem rozstrzygnięcie, czy  $BBC(K_n, G) \leq n$  jest NP-zupełne w przypadku ogólnym.

Kolejnym etapem redukcji jest rozwijanie wierzchołków. Na podstawie grafu  $\bar{G}$  można skonstruować split graf  $G_S$  ze szkieletem  $H_S$ , mający  $n + 2m$  wierzchołków, poprzez przydzielenie każdemu wierzchołkowi  $v_i$  dokładnie  $\delta_{\bar{G}}(v_i)$  nowych sąsiadów  $u_{ij}$  w szkielecie. Każdy wierzchołek  $u_{ij}$  jest oprócz tego połączony krawędzią z wszystkimi pozostałymi pierwotnymi wierzchołkami z wyjątkiem  $v_j$ . Ostatnim krokiem jest połączenie oryginalnych wierzchołków  $v_i$  w klikę. Przykład takiej transformacji jest widoczny na rys. 4.1.

W rezultacie otrzymywany jest split graf  $G_S = (C \cup I, E_S)$  o klice  $C = \{v_i : 1 \leq i \leq n\}$ , zbiorze niezależnym  $I = \{u_{ij} : v_i v_j \in E(G)\}$  oraz zbiorze krawędzi  $E_S = \{v_i v_j : 1 \leq i, j \leq n\} \cup \{v_i u_{jk} : 1 \leq i, j, k \leq n, u_{jk} \in I, i \neq k\}$ . Szkielet  $H_S = (C \cup I, E'_S)$  jest zbiorem gwiazd o centrach w  $C$  i liściach w  $I$ :  $E'_S = \{v_i u_{ij} : 1 \leq i, j \leq n, u_{ij} \in I\}$ . Wystarczy teraz zauważyć, że odpowiedź na pytanie  $BBC(G_S, H_S) \leq n$  jest równoważna  $BBC(K_n, G) \leq n$ , ponieważ każdy wierzchołek z  $I$  ma dokładnie





Rys. 4.1. Redukcja do kolorowania szkieletowego split grafu z lasem w szkielecie

jednego niesąsiada w  $C$  i wobec tego muszą one otrzymać ten sam kolor w każdym  $n$ -pokolorowaniu.  $\square$

Nieznacznie modyfikując tą konstrukcję poprzez wybranie dowolnego drzewa spinającego  $T$  z grafu oraz pominięcie go na etapie rozwijania wierzchołków otrzymujemy analogiczną redukcję również dla drzew oraz lasów niebędących gwiazdami.

**Twierdzenie 4.7.** *Problem rozstrzygnięcia, czy dla danego split grafu  $G$  z drzewem spinającym  $T$  w szkielecie zachodzi  $BBC(G, T) \leq l$  jest NP-zupełny dla  $l = \chi(G)$ .*

Choć problem kolorowania szkieletowego jest NP-zupełny, to istnieje dla niego algorytm 1-bezwzględnie przybliżony, działający w czasie  $O(n^2)$ . Możliwa jest również analogiczna konstrukcja algorytmu 1-bezwzględnie przybliżonego dla problemu kolorowania szkieletowego split grafów ze skojarzeniem w szkielecie w czasie liniowym względem liczby wierzchołków[20].

# Rozdział 5

---

## Projekt aplikacji

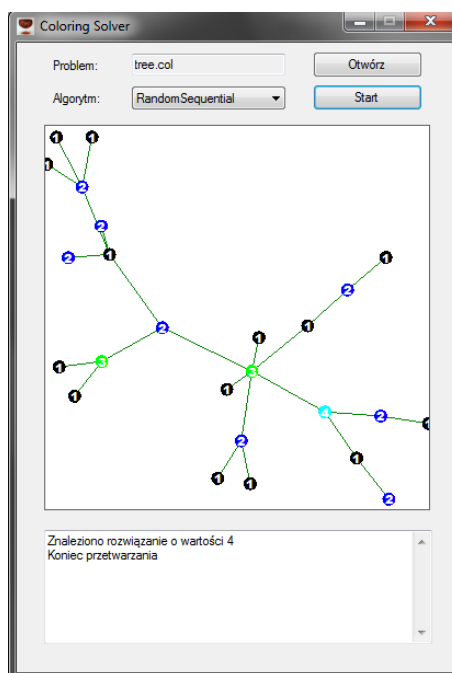
Ważnym elementem niniejszej pracy było wykonanie aplikacji, implementującej przykładowe algorytmy kolorowania szkieletowego grafów. Z założenia, aplikacja miała umożliwiać użytkownikowi obsługę przynajmniej jednego, powszechnie przyjętego formatu opisu grafów (np. DIMACS lub G6), automatycznie rysować wczytany graf na płaszczyźnie oraz udostępniać proste wywołania algorytmów z możliwie dużym doбором parametrów.

Dodatkowo, wykonano również prosty generator grafów, umożliwiający szybkie tworzenie i zapis losowych grafów oraz dobór pewnych parametrów tych grafów, takich jak liczba chromatyczna lub szkieletowa liczba chromatyczna.

### 5.1. Technologie i formaty

Do implementacji programu wybrano język programowania C# oraz środowisko uruchomieniowe .NET w wersji 4.0. Wybór ten został dokonany głównie ze względu na prostotę tworzenia programu, a w szczególności jego interfejsu graficznego – w przypadku tworzenia aplikacji o zastosowaniu wymagającym wysokiej wydajności należałoby rozważyć inny język programowania np. C lub C++. Okno główne programu zostało przedstawione na rys. 5.1.

Przyjęto, że podstawowym formatem obsługiwanym przez program będzie format tekstowy DIMACS. W tym formacie graf jest opisywany za pomocą listy krawędzi. Szczegółowy opis wybranego formatu można znaleźć m.in. na stronie projektu pod adresem <http://mat.gsia.cmu.edu/COLOR/general/ccformat.ps>. Ponieważ w modelu kolorowania szkieletowego występują dwa rodzaje krawędzi: zwykłe i szkieletowe, przyjęto dla nich odpowiednio oznaczenia  $e$  oraz  $b$ .



RYS. 5.1. Okno główne programu

## 5.2. Wybrane algorytmy kolorowania

Dla potrzeb niniejszej pracy zaimplementowano następujące algorytmy kolorowania szkieletowego:

- algorytm RS,
- algorytm LF,
- algorytm SL,
- algorytm SLF,
- algorytm GIS,
- algorytm *Branch and Bound*.

Większość wybranych metod działa w sposób sekwencyjny, ustalając kolejność wierzchołków, a następnie przydzielając im kolory w sposób zachłanny. Takie podejście jest uzasadnione między innymi poprzez poniższe twierdzenia:

**Twierdzenie 5.1.** *Dla każdego grafu  $G$  istnieje uszeregowanie wierzchołków takie, że algorytm zachłanny zwraca rozwiązanie optymalne dla problemu kolorowania klasycznego.*

*Dowód.* Mając dane pewne pokolorowanie optymalne  $c$  można uporządkować wierzchołki nierosnąco według koloru i otrzymać uszeregowanie  $v_1, v_2, \dots, v_n$ . Wystarczy wykazać, że pokolorowanie zachłanne  $c'$  wierzchołków w takiej kolejności jest również optymalne.

Niech  $G_i = (V_i, E_i)$  oznacza podgraf grafu  $G$ , indukowany przez wierzchołki  $v_1, v_2, \dots, v_i$ . Gdy  $v_i$  nie ma sąsiadów wśród wierzchołków  $v_1, v_2, \dots, v_{i-1}$ , to oczywiście jest, że  $c'(v_i) = 1 \leq c(v_i)$ . W szczególności jest to prawdziwe dla  $i = 1$ .

Przyjmijmy, że  $c'(v) \leq c(v)$  dla wszystkich wierzchołków  $v_1, v_2, \dots, v_{i-1}$ . Rozpatrując wierzchołek  $v_i$  i jego najpóźniejszego sąsiada  $v_j$  w sekwencji  $v_1, v_2, \dots, v_{i-1}$  widzimy, że zachodzą zależności:

$$c'(v_i) \leq \max_{u:uv_i \in E_i} c'(u) + 1 \leq \max_{u:uv_i \in E_i} c(u) + 1 = c(v_j) + 1 \leq c(v_i)$$

Zatem dla każdego wierzchołka  $v \in V$  zachodzi  $c'(v) \leq c(v)$ , stąd wniosek, że  $\max_v c'(v) \leq \max_v c(v)$ . Ostatecznie, wobec optymalności kolorowania  $c$ , również kolorowanie  $c'$  musi być optymalne.  $\square$

**Twierdzenie 5.2.** *Dla każdego grafu  $G$  ze szkieletem  $H$  istnieje uszeregowanie wierzchołków takie, że algorytm zachłanny zwraca rozwiązanie optymalne dla problemu kolorowania szkieletowego.*

*Dowód.* Analogicznie jak w dowodzie poprzedniego twierdzenia, niech dla pewnego optymalnego pokolorowania szkieletowego  $c$  uszeregowanie  $v_1, v_2, \dots, v_n$  będzie wyznaczone zgodnie z nierosnącymi kolorami w tym pokolorowaniu. Wystarczy wykazać, że zachłanne pokolorowanie szkieletowe  $c'$  wierzchołków w takiej kolejności jest również optymalne.

Niech  $G_i = (V_i, E_i)$  i  $H_i = (V_i, E'_i)$  oznaczają odpowiednio podgrafy grafów  $G$  i  $H$ , indukowane przez  $v_1, v_2, \dots, v_i$ . Oczywiście gdy  $v_i$  nie ma sąsiadów wśród wierzchołków  $v_1, v_2, \dots, v_{i-1}$ , to  $c'(v_i) = 1 \leq c(v_i)$ . W szczególności  $c(v_1) \geq c'(v_1)$ .

Przyjmijmy zatem, że  $c'(v) \leq c(v)$  dla wszystkich wierzchołków  $v_1, v_2, \dots, v_{i-1}$ . Dla wierzchołka  $v_i$  można wyróżnić jego najpóźniejszego sąsiada  $v_j$  oraz najpóźniejszego sąsiada szkieletowego  $v_l$  (o ile istnieje) w sekwencji  $v_1, v_2, \dots, v_{i-1}$ . Wystarczy tylko zaobserwować, że:

$$\begin{aligned} c'(v_i) &\leq \max\left\{ \max_{u:uv_i \in E_i} c'(u) + 1, \max_{u:uv_i \in E'_i} c'(u) + \lambda \right\} \\ &\leq \max\left\{ \max_{u:uv_i \in E_i} c(u) + 1, \max_{u:uv_i \in E'_i} c(u) + \lambda \right\} \\ &= \max\{c(v_j) + 1, c(v_l) + \lambda\} \leq c(v_i) \end{aligned}$$

Z powyższego wynika, że dla każdego wierzchołka  $v \in V$  zachodzi  $c'(v) \leq c(v)$ , czyli  $\max_v c'(v) \leq \max_v c(v)$ . Wobec optymalności kolorowania  $c$ , również kolorowanie  $c'$  musi być optymalne.  $\square$

Jakość algorytmu można ocenić poprzez funkcję dobroci [14], opisującą górne ograniczenie zależności między rozwiązaniem wyznaczonym przez algorytm a rozwiązaniem optymalnym dla instancji danego rzędu  $n$ . Im większe wartości przyjmuje funkcja dobroci, tym otrzymane rozwiązanie może być gorsze względem optymalnego. Niestety, świadczy to jedynie o pesymistycznym zachowaniu metody, dlatego przydatne jest również badanie zachowania algorytmu dla szczególnych klas grafów oraz dla małych przypadków.

Dodatkowo, analiza algorytmów kolorowania, tak jak w przypadku klasycznym, powinna rozpatrywać ich teoretyczną złożoność obliczeniową i rzeczywisty czas działania. Jest to kryterium zdecydowanie mniej ważne od poprzedniego, ale w przypadku dwóch metod o podobnej gwarancji aproksymacji i zbliżonej jakości w praktyce, z reguły, wybierana jest asymptotycznie szybsza.

Oprócz badania jakości i czasu działania można również badać strukturę najmniejszych grafów kolorowanych nieoptymalnie. Takie poszukiwania często wskazują słabe punkty metod, umożliwiając znalezienie usprawnień i nowych, lepszych algorytmów. W tym celu w pracy [12] wprowadzono pojęcie grafów trudnych do kolorowania, które w łatwy sposób można rozszerzyć na inne, nieklasyczne modele kolorowania. Wyczerpujący opis grafów dość trudnych oraz trudnych dla wszystkich ważniejszych algorytmów kolorowania sekwencyjnego można znaleźć w książce [14].

**Definicja 5.1.** Graf  $G$  nazywamy *trudnym do kolorowania* (ang. *hard-to-color*) dla algorytmu  $A$ , jeżeli każde wykonanie algorytmu zwraca pokolorowanie nieoptymalne.

**Definicja 5.2.** Graf  $G$  nazywamy *dość trudnym do kolorowania* (ang. *slightly hard-to-color*) dla algorytmu  $A$ , jeżeli możliwe jest wykonanie algorytmu zwracające pokolorowanie nieoptymalne.

Implementacje poszczególnych metod nie zawsze są optymalne z punktu widzenia złożoności obliczeniowej. Takie podejście było podyktowane zauważalnym narzutem wynikającym ze stałej proporcjonalności przy użyciu zaawansowanych struktur danych w języku C#, takich jak kolejka priorytetowa. Dlatego przyjęto, że kolejne wierzchołki będą znajdowane za pomocą zwykłego przeszukiwania liniowego na liście wierzchołków niepokolorowanych, co w praktyce prowadzi do zadowalających rezultatów czasowych. Również wykorzystano wbudowaną funkcję sortowania o złożoności obliczeniowej  $O(n \log n)$  zamiast sortowania w czasie liniowym, gdyż przy przyjętej wielkości instancji, różnice były pomijalnie małe.

### 5.2.1. Algorytm RS

Algorytm RS (ang. *Random Sequential*) jest najprostszym algorytmem kolorowania. Polega on na losowym wyznaczeniu kolejności kolorowania wierzchołków, a następnie przydzielania im barw metodą zachłanną: każdy kolejny wierzchołek otrzymuje najniższy kolor z możliwych.

---

**Algorytm 3** Algorytm *Random Sequential*


---

Wylosuj kolejność wierzchołków  $v_1, v_2, \dots, v_n$

**for**  $i = 1, 2, \dots, n$  **do**

    Pokoloruj wierzchołek  $v_i$  zachłannie

---

Zaletą tego algorytmu jest przede wszystkim jego prostota. Dodatkowo, zgodnie z twierdzeniem 5.2, algorytm zawsze ma szansę zwrócić wynik optymalny – wystarczy, że zostanie wylosowana poprawna kolejność kolorowania wierzchołków, zatem nie istnieje graf trudny dla tego algorytmu. Najmniejszą dość trudną instancją dla tego algorytmu jest graf  $C_3$  ze ścieżką  $P_3$  w szkielecie – wystarczy kolorować wierzchołki zgodnie z ich kolejnością na ścieżce, poczynawszy od jednego z jej końców.

Algorytm działa w całkowitym czasie  $O(n + m + m'\lambda)$ , ponieważ pokolorowanie zachłanne każdego wierzchołka wymaga  $O(\delta_H(v_i)\lambda + \delta_G(v_i))$  operacji.

### 5.2.2. Algorytm LF

Algorytm LF (ang. *Largest First*) został zaproponowany przez D. Welsha i M. Powella [23] w 1967 roku. Algorytm ten przydziela wierzchołkom kolory w sposób zachłanny, zgodnie z uszeregowaniem wierzchołków według ich nierosnących stopni.

Adaptacja tego algorytmu dla kolorowania szkieletowego wymaga zdefiniowania pojęcia stopnia wierzchołka w grafie. Naturalnym wyborem wydaje się dobranie wartości równej liczbie incydujących zwykłych krawędzi, powiększonej o liczbę krawędzi szkieletowych pomnożoną przez  $\lambda$ , czyli wymaganą różnicę barw w szkielecie. Innym wariantem jest wybór zamiast  $\lambda$  wagi  $2\lambda - 1$ , równej maksymalnej liczbie kolorów zabranianych przez wierzchołek pokolorowany swoim sąsiadom w szkielecie.

---

**Algorytm 4** Algorytm *Largest First*


---

Uporządkuj wierzchołki  $v_1, v_2, \dots, v_n$  nierosnąco wg stopni w  $G$

**for**  $i = 1, 2, \dots, n$  **do**

    Pokoloruj wierzchołek  $v_i$  zachłannie

---

Najmniejszą trudną (i jednocześnie dość trudną) instancją dla tej metody jest  $(P_4, M)$ , przedstawiona na rys. 5.2a. Do pokolorowania algorytm LF wymaga  $\lambda + 2$  barw, podczas gdy  $BBC_\lambda(P_4, M) = \lambda + 1$ .

Algorytm działa w czasie  $O(n+m+m'\lambda)$ : obliczenie stopni wierzchołków wymaga  $\Theta(m)$  operacji, sortowanie –  $\Theta(n)$  operacji, natomiast pokolorowanie zachłanne – tak jak w metodzie RS –  $O(n + m + m'\lambda)$ .

### 5.2.3. Algorytm SL

Algorytm SL (ang. *Smallest Last*), zaproponowany przez D. Matulę [16], również działa dwuetapowo, najpierw porządkując wierzchołki, a następnie przydzielając im kolory w sposób zachłanny. Porządkowanie polega na znajdowaniu wierzchołka o minimalnym stopniu i usuwaniu go z grafu, aż do usunięcia wszystkie wierzchołków. Następnie wierzchołki są kolorowane w kolejności odwrotnej do tej, w jakiej zostały usunięte.

W kolorowaniu klasycznym to podejście daje bardzo dobrą gwarancję w przypadku klas grafów o z góry ograniczonym minimalnym stopniu wierzchołka np. grafów planarnych lub grafów  $k$ -zdegenerowanych. Analogicznie jak dla algorytmu LF, konieczne jest odpowiednie nadanie wag krawędziom szkieletowym w celu przystosowania metody dla problemu kolorowania szkieletowego.

---

#### Algorytm 5 Algorytm *Smallest Last*

---

$S \leftarrow \emptyset$

**for**  $i = 1, 2, \dots, n$  **do**

Wybierz wierzchołek  $v_i$  o najmniejszym stopniu w grafie indukowanym przez wierzchołki  $V \setminus S$

Dodaj  $v_i$  na koniec listy  $S$

**for**  $i = n, n - 1, \dots, 1$  **do**

Pokoloruj wierzchołek  $v_i$  z listy  $S$  zachłannie

---

Tak jak dla algorytmu RS, najmniejszym dość trudnym grafem jest  $C_3$  ze ścieżką  $P_3$  w szkielecie, wystarczy wybierać wierzchołki zgodnie z kolejnością na ścieżce. Najmniejszy znaleziony graf trudny został przedstawiony na rys. 5.2b. Kolorując go algorytmem SL potrzeba co najmniej  $\lambda + 3$  barw, natomiast jego liczba szkieletowa jest równa  $\lambda + 2$ : algorytm uszereguje wierzchołki w taki sposób, że centrum jednej z gwiazd otrzyma kolor 2 lub  $\lambda + 1$ , co w konsekwencji musi doprowadzić do pokolorowania całego grafu przy użyciu więcej niż  $\lambda + 2$  kolorów.

Złożoność obliczeniowa algorytmu wynosi  $O(n+m+m'\lambda)$ , ponieważ szeregowanie SL można wykonać w czasie  $O(n+m)$ , a pokolorowanie zachłanne wierzchołków wymaga  $O(n+m+m'\lambda)$  operacji.

Z faktu, że każdy graf planarny jest 5-zdegenerowany można wywnioskować, że do pokolorowania szkieletowego grafów planarnych z dowolnym szkieletem algorytm SL wymaga nie więcej niż  $10\lambda-4$  barw. Jeśli szkielet zostanie ograniczony do skojarzenia wystarczy jedynie  $2\lambda+4$  kolorów.

#### 5.2.4. Algorytm SLF

Algorytm SLF (ang. *Saturated Largest First*) został po raz pierwszy przedstawiony przez D. Bréla za w pracy [2]. Bazuje on na spostrzeżeniu, że dobrym wyborem jest kolorowanie w pierwszej kolejności wierzchołków mających wielu różnobarwnych sąsiadów. W każdej iteracji algorytmu znajdowany, a następnie kolorowany zachłannie, jest wierzchołek o największej liczbie sąsiednich kolorów.

Równoważnym podejściem dla kolorowania szkieletowego jest wybór wierzchołków o największej liczbie kolorów już zabronionych. Analogicznie jak dla kolorowania klasycznego, w razie remisu pierwszeństwo ma wierzchołek o wyższym stopniu.

---

#### Algorytm 6 Algorytm *Saturated Largest First*

---

**for**  $i = 1, 2, \dots, n$  **do**

Wybierz niepokolorowany wierzchołek  $v_i$  o największej liczbie kolorów zabronionych

Pokoloruj wierzchołek  $v_i$  zachłannie

Aktualizuj listy kolorów zabronionych u sąsiadów  $v_i$

---

Tak, jak w przypadku algorytmu LF, najmniejszym grafem dość trudnym i trudnym jest ścieżka  $P_4$  ze skojarzeniem w szkielecie, przedstawiona na rys. 5.2a. Algorytm zawsze wybierze najpierw oba środkowe wierzchołki, przydzielając im kolory 1 i 2, zatem jeden z wierzchołków skrajnych otrzyma kolor  $\lambda+2$ , podczas gdy  $BBC_\lambda(P_4, M) = \lambda+1$ .

Można wykazać, że algorytm zawsze koloruje optymalnie grafy pełne dwudzielne ze spójnym szkieletem, natomiast ma problemy z grafami pełnymi ze skojarzeniem w szkielecie.

Efektywne asymptotycznie podejście wymaga użycia kolejki priorytetowej – struktury danych o logarytmicznym czasie wstawiania elementu oraz stałym wydobywania maksimum lub minimum. Na łączny czas działania algorytmu składają się operacje:



- dodanie wierzchołków do kolejki priorytetowej,
- usuwanie wierzchołków z kolejki priorytetowej,
- kolorowanie zachłanne wierzchołków,
- aktualizacja listy kolorów sąsiadów w grafie.

Każdy wierzchołek jest dodawany do kolejki co najwyżej  $\delta_G(v) + 1$  razy, zatem łączny czas operacji wstawiania to  $O((n + m) \log n)$ . Każdy wstawiony wierzchołek jest usuwany co najwyżej raz, a więc wymaga to  $O(n + m)$  operacji.

Aktualizację listy kolorów zabronionych pojedynczego sąsiada można wykonać w czasie  $O(\log n)$  w przypadku krawędzi zwykłej i  $O(\lambda \log n)$  dla krawędzi szkieletowej, co daje łączny czas aktualizacji  $O(m \log n + m' \lambda \log n)$ . Kolorowanie zachłanne ze swoją złożonością  $O(n + m + m' \lambda)$  jest asymptotycznie szybsze niż pozostała część algorytmu.

Ostatecznie, złożoność obliczeniowa algorytmu SLF to  $O((n + m + m' \lambda) \log n)$ .

### 5.2.5. Algorytm GIS

Algorytm GIS (ang. *Greedy Independent Set*), wprowadzony przez D. Johnsona [13], opiera się na fakcie, że każdy kolor w dopuszczalnym kolorowaniu grafu indukuje w grafie zbiór niezależny. Naturalnym podejściem wydaje się więc sekwencyjne wyznaczanie zbiorów niezależnych i kolorowanie ich kolejnymi barwami. Dodatkowym usprawnieniem tego podejścia jest wybór wierzchołków nie w sposób losowy, a o najmniejszym stopniu w grafie indukowanym przez wierzchołki niepokolorowane.

Implementacja algorytmu GIS dla kolorowania szkieletowego polega na wyznaczeniu dla każdego wierzchołka minimalnego dopuszczalnego koloru, a następnie sekwencyjnym znajdowaniu niepokolorowanych wierzchołków kolorowalnych możliwie niskim kolorem. W przypadku równości, decyduje kryterium niższego stopnia w grafie indukowanym przez niepokolorowane wierzchołki.

---

#### Algorytm 7 Algorytm *Greedy Independent Set*

---

```

for all  $v \in V(G)$  do
     $a(v) \leftarrow 1$ 
for  $i = 1, 2, \dots, n$  do
    Wybierz niepokolorowany wierzchołek  $v_i$  o najmniejszej wartości  $a(v_i)$ 
     $c(v_i) \leftarrow a(v_i)$ 
    Aktualizuj wartości  $a(v)$  u sąsiadów  $v_i$ 

```

---

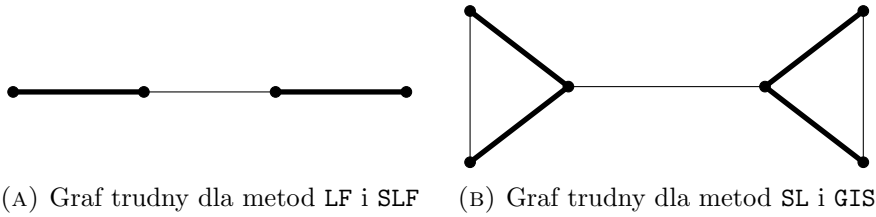
W ogólnym przypadku, szeregującym wierzchołki w sposób losowy, zgodnie z twierdzeniem 5.1 nie istnieje graf trudny. Dla wariantu zaimplementowanego w projekcie,

najmniejszym grafem dość trudnym jest cykl  $C_3$  ze ścieżką  $P_3$  w szkielecie, kolorowany w kolejności przechodzenia grafu ścieżką szkieletową. Najmniejszym znalezionym grafem trudnym jest natomiast graf przedstawiony na rys. 5.2b, optymalnie kolorowalny przy użyciu  $\lambda + 2$  barw.

Algorytm działa w czasie  $\Theta(n^2)$ :  $i$ -ta iteracja algorytmu wymaga  $\Theta(n - i)$  czasu na znalezienie odpowiedniego wierzchołka  $v$ ,  $\Theta(1)$  na usunięcie go z listy wierzchołków niepokolorowanych i pokolorowanie oraz  $\Theta(\delta_G(v_i))$  na aktualizację dozwolonych kolorów oraz stopni u sąsiadów.

$$\sum_{i=1}^n (\Theta(n - i) + \Theta(1) + \Theta(\delta_G(v_i))) = \Theta(n^2) + \Theta(n) + \Theta(m) = \Theta(n^2)$$

Warto zwrócić uwagę, że czas działania nie zależy od wartości parametru  $\lambda$ , co należy uznać za atut algorytmu.



RYŚ. 5.2. Najmniejsze znane grafy trudne dla opisanych algorytmów

### 5.2.6. Algorytm *Branch and Bound*

Algorytm *Branch and Bound* jest jedynym zaimplementowanym w niniejszym projekcie dokładnym algorytmem kolorowania szkieletowego grafów. Przeszukuje on wyczerpująco drzewo możliwych kolorowań zaczynając od wierzchołka początkowego i przydzielając mu wszystkie możliwe kolory aż do górnego ograniczenia. W chwili znalezienia rozwiązania lepszego niż dotychczasowe, następuje aktualizacja górnego ograniczenia na liczbę kolorów. Przeszukiwanie kończy się, gdy wszystkie niesprawdzone jeszcze ścieżki poszukiwań nie będą mogły poprawić najlepszego znalezionego kolorowania.

Algorytm ma w pesymistycznym przypadku wykładniczy czas działania i z racji tego nadaje się jedynie do rozwiązywania małych instancji problemu, do kilkudziesięciu wierzchołków i kilkuset krawędzi. Jego czas działania jest silnie zależny od struktury grafu i wynikającego z niej drzewa przeszukiwań wszystkich pokolorowań. Został on zawarty w pracy jedynie jako punkt odniesienia dla wyżej podanych algorytmów.

## 5.3. Przeprowadzone testy

W ramach części projektowej niniejszej pracy wykonano szereg testów, mających na celu zbadanie zależności jakości przedstawionych w punkcie 5.2 niniejszej pracy algorytmów od następujących parametrów:

- rzędu grafu,
- gęstości grafu i szkieletu,
- struktury grafu i szkieletu,
- wartości parametru  $\lambda$ .

Przygotowano dwa zestawy testowe, składające się z grafów o  $n = 20$  i  $n = 2000$  wierzchołkach. W ramach każdego zestawu wyróżniono odpowiednie podzestawy:

- 100 grafów pełnych z losowym skojarzeniem w szkielecie,
- 100 grafów dwudzielnych o gęstościach  $\rho_G = 0.5$  i  $\rho_H = 0.1$ ,
- 100 grafów dwudzielnych o gęstościach  $\rho_G = 0.9$  i  $\rho_H = 0.5$ ,
- 100 grafów losowych o gęstościach  $\rho_G = 0.5$  i  $\rho_H = 0.1$ ,  $OPT = 5$ ,
- 100 grafów losowych o gęstościach  $\rho_G = 0.7$  i  $\rho_H = 0.3$ ,  $OPT = 5$ ,
- 100 grafów losowych o gęstościach  $\rho_G = 0.9$  i  $\rho_H = 0.5$ ,  $OPT = 5$ ,
- 100 grafów losowych o gęstościach  $\rho_G = 0.7$  i  $\rho_H = 0.1$ ,  $OPT = 5$ ,
- 100 grafów losowych o gęstościach  $\rho_G = 0.7$  i  $\rho_H = 0.5$ ,  $OPT = 5$ ,
- 100 grafów losowych o gęstościach  $\rho_G = 0.5$  i  $\rho_H = 0.1$ ,  $OPT = \frac{n}{4}$ ,
- 100 grafów losowych o gęstościach  $\rho_G = 0.9$  i  $\rho_H = 0.5$ ,  $OPT = \frac{n}{4}$ .

Grafy testowe zostały wygenerowane na podstawie zadanych prawdopodobieństw wystąpienia krawędzi oraz krawędzi szkieletowej, jednocześnie w celu zagwarantowania odpowiedniej liczby chromatycznej wymuszono istnienie klikki odpowiedniego rozmiaru w grafie  $G$ . W ramach projektu powstał prosty losowy generator grafów, przydzielający wierzchołki do zbiorów niezależnych, odpowiadających poszczególnym kolorom i losujący obecność każdej możliwej krawędzi między tymi zbiorami. Wygenerowane grafy wraz ze szkieletami zapisano do poszczególnych plików w formacie DIMACS.

W ramach oceny wpływu struktury grafu oraz szkieletu, zbadano jakość kolorowania grafów pełnych ze skojarzeniem w szkielecie oraz grafów dwudzielnych z dowolnym szkieletem. Wybrano te klasy grafów z uwagi na znaną wartość ich szkieletowej liczby chromatycznej – w pierwszym przypadku równą liczbie wierzchołków grafu dla  $\lambda \leq \frac{n}{2}$  (co wykazano w [18]), natomiast w drugim równą  $\lambda + 1$ .

Dla algorytmów szeregowania deterministycznego zbadano również wpływ metody obliczania stopnia wierzchołka na podstawie liczby krawędzi incydentnych

w grafie i szkielecie. W pierwszym wariancie waga krawędzi szkieletowej, wnoszona do stopnia wierzchołka, jest równa wartości współczynnika  $\lambda$ . Alternatywnie, krotkość krawędzi szkieletowej może być równa  $2\lambda - 1$  – maksymalnej liczbie kolorów, których dany wierzchołek zabrania swoim sąsiadom. Oba warianty dla algorytmu LF oznaczono w tabelach wyników odpowiednio indeksami <sup>1</sup> i <sup>2</sup>. Dla pozostałych algorytmów przyjęto pierwsze podejście.

Wszystkie testy zostały przeprowadzone dla podstawowej wartości  $\lambda = 2$ , jednak w celu porównania wpływu również tego parametru na zachowanie algorytmów, wykonano dodatkowe eksperymenty na zestawie grafów o 2000 wierzchołkach dla  $\lambda = 100$ .

## 5.4. Wyniki testów

Podstawowym parametrem oceny jakości jest średnia liczba użytych kolorów przez każdy z algorytmów. W sytuacjach, w których grafy mają znaną szkieletową liczbę chromatyczną, podano również liczbę przypadków, dla których algorytm ten znalazł optymalne rozwiązanie w zadanym limicie czasowym. Z uwagi na dużą liczbę instancji, przyjęto ograniczenie czasu działania pojedynczego testu do 20 s.

Dla zestawu grafów o 20 wierzchołkach dla porównania zamieszczono wyniki dokładnego algorytmu wykładniczego. Z kolei dla zestawu grafów o 2000 wierzchołkach porównano łączne czasy działania poszczególnych metod i – tam gdzie nie było możliwe wyznaczenie wartości optymalnej szkieletowej liczby chromatycznej – podano wartość liczby chromatycznej grafu.

Należy również nadmienić, że z powodu przyjętego formatu opisu grafu, dla większości testów czas wczytywania plików był znacząco dłuższy niż pomierzony i przedstawiony poniżej czas wykonania algorytmów.

TABELA 5.1. Grafy pełne ze skojarzeniem w szkielecie,  $N = 20$ ,  $\lambda = 2$ ,  $OPT = 20$

Algorytm	Łączny czas	Optymalne rozwiązanie	Średnia liczba użytych kolorów
BranchAndBound	2000.3	100%	20.0
RandomSequential	0.071	97%	20.0
LargestFirst <sup>1</sup>	0.054	93%	20.1
SmallestLast	0.047	100%	20.0
SaturatedLargestFirst	0.336	83%	20.2
GreedyIndependentSet	0.054	100%	20.0
LargestFirst <sup>2</sup>	0.047	90%	20.1

TABELA 5.2. Grafy dwudzielne,  $N = 20$ ,  $\rho_G = 0.5$ ,  $\rho_H = 0.1$ ,  $\lambda = 2$ ,  $OPT = 3$ 

Algorytm	Łączny czas	Optymalne rozwiązanie	Średnia liczba użytych kolorów
BranchAndBound	0.070	100%	3.0
RandomSequential	0.046	22%	4.4
LargestFirst <sup>1</sup>	0.030	78%	3.3
SmallestLast	0.029	52%	3.5
SaturatedLargestFirst	0.034	79%	3.2
GreedyIndependentSet	0.027	78%	3.5
LargestFirst <sup>2</sup>	0.034	78%	3.3

TABELA 5.3. Grafy dwudzielne,  $N = 20$ ,  $\rho_G = 0.9$ ,  $\rho_H = 0.5$ ,  $\lambda = 2$ ,  $OPT = 3$ 

Algorytm	Łączny czas	Optymalne rozwiązanie	Średnia liczba użytych kolorów
BranchAndBound	0.125	100%	3.0
RandomSequential	0.097	63%	4.1
LargestFirst <sup>1</sup>	0.037	100%	3.0
SmallestLast	0.090	100%	3.0
SaturatedLargestFirst	0.037	100%	3.0
GreedyIndependentSet	0.029	96%	3.2
LargestFirst <sup>2</sup>	0.040	100%	3.0

TABELA 5.4. Grafy losowe,  $N = 20$ ,  $\rho_G = 0.5$ ,  $\rho_H = 0.1$ ,  $\lambda = 2$ ,  $OPT = 5$ 

Algorytm	Łączny czas	Optymalne rozwiązanie	Średnia liczba użytych kolorów
BranchAndBound	91.13	100%	5.0
RandomSequential	0.049	5%	7.0
LargestFirst <sup>1</sup>	0.025	13%	6.3
SmallestLast	0.030	25%	6.1
SaturatedLargestFirst	0.037	37%	5.9
GreedyIndependentSet	0.030	2%	7.6
LargestFirst <sup>2</sup>	0.052	10%	6.6

TABELA 5.5. Grafy losowe,  $N = 20$ ,  $\rho_G = 0.7$ ,  $\rho_H = 0.3$ ,  $\lambda = 2$ ,  $OPT = 5$ 

Algorytm	Łączny czas	Optymalne rozwiązanie	Średnia liczba użytych kolorów
BranchAndBound	691.75	100%	5.0
RandomSequential	0.043	0%	8.9
LargestFirst <sup>1</sup>	0.032	4%	7.5
SmallestLast	0.038	10%	7.3
SaturatedLargestFirst	0.037	17%	6.8
GreedyIndependentSet	0.022	1%	8.8
LargestFirst <sup>2</sup>	0.048	6%	7.5

TABELA 5.6. Grafy losowe,  $N = 20$ ,  $\rho_G = 0.9$ ,  $\rho_H = 0.5$ ,  $\lambda = 2$ ,  $OPT = 5$ 

Algorytm	Łączny czas	Optymalne rozwiązanie	Średnia liczba użytych kolorów
BranchAndBound	1179.1	100%	5.0
RandomSequential	0.045	7%	8.8
LargestFirst <sup>1</sup>	0.034	11%	7.1
SmallestLast	0.028	22%	6.7
SaturatedLargestFirst	0.035	21%	6.6
GreedyIndependentSet	0.034	11%	7.3
LargestFirst <sup>2</sup>	0.033	12%	7.1

TABELA 5.7. Grafy losowe,  $N = 20$ ,  $\rho_G = 0.7$ ,  $\rho_H = 0.1$ ,  $\lambda = 2$ ,  $OPT = 5$ 

Algorytm	Łączny czas	Optymalne rozwiązanie	Średnia liczba użytych kolorów
BranchAndBound	713.10	100%	5.0
RandomSequential	0.035	1%	7.8
LargestFirst <sup>1</sup>	0.029	7%	6.8
SmallestLast	0.024	19%	6.4
SaturatedLargestFirst	0.041	22%	6.2
GreedyIndependentSet	0.026	0%	8.3
LargestFirst <sup>2</sup>	0.028	7%	6.8

TABELA 5.8. Grafy losowe,  $N = 20$ ,  $\rho_G = 0.7$ ,  $\rho_H = 0.5$ ,  $\lambda = 2$ ,  $OPT = 5$ 

Algorytm	Łączny czas	Optymalne rozwiązanie	Średnia liczba użytych kolorów
BranchAndBound	703.11	100%	5.0
RandomSequential	0.055	3%	9.5
LargestFirst <sup>1</sup>	0.023	16%	7.5
SmallestLast	0.034	24%	7.0
SaturatedLargestFirst	0.039	18%	6.8
GreedyIndependentSet	0.021	2%	8.6
LargestFirst <sup>2</sup>	0.021	14%	7.6

Algorytm dokładny zawsze znajdował optymalne rozwiązanie w zadanym limicie czasowym, ale w wielu przypadkach nie kończył swojego działania. Wyjątkiem były jedynie grafy pełne oraz dwudzielne, dla których liczba rozgałęzień i nawrotów w drzewie przeszukiwań była niewielka. Różnica w czasie działania algorytmu dokładnego i metod przybliżonych jest znaczna już dla niewielkich testów: dla porównania, wszystkie pozostałe algorytmy wykonywały się w znikomym czasie.

W przypadku algorytmu RS, wynik jest silnie uzależniony od początkowej permutacji wierzchołków – ponieważ nie istnieje graf trudny, zawsze możliwe jest wylosowanie kolejności optymalnej. Jednocześnie, zdecydowanie bardziej prawdopodobne jest wylosowanie kolejności skutkującej nieoptymalnym rozwiązaniem, tak jak to jest widoczne w przypadku grafów dwudzielnych. Jak łatwo zauważyć, pozostałe algorytmy rozwiązują problem zdecydowanie lepiej niż algorytm RS.

Testy na grafach losowych pokazują, że wyniki algorytmów RS oraz GIS są zauważalnie gorsze od pozostałych algorytmów, natomiast zdecydowanie lepiej radzą sobie metody SLF i SL. Algorytm SLF lepiej radzi sobie z grafami rzadkimi, natomiast używa nadmiarowej liczby barw przy kolorowaniu grafów pełnych ze skojarzeniem w szkielecie. Najlepsze algorytmy dla instancji losowych znajdują wynik optymalny tylko w 15 – 35% przypadków, natomiast bezbłędnie radzą sobie z gęstymi grafami dwudzielnymi, a także pełnymi ze skojarzeniem szkieletowym.

Przyglądając się bliżej wynikom eksperymentów można zaobserwować, że algorytmy przybliżone działają tym lepiej, im graf jest gęstszy lub jego szkielet rzadszy. Ten wniosek ma swoje podstawy teoretyczne: jeśli graf jest gęsty, to łatwiej podzielić poprawnie wierzchołki na zbiory niezależne, odpowiadające poszczególnym kolorom. Z drugiej strony, jeśli szkielet grafu jest rzadki, to rzadko zachodzi zabranianie kolorów na krawędziach szkieletowych, dzięki czemu algorytm wykonuje prawie identyczne kroki jak dla klasycznego kolorowania wierzchołkowego.

Różnice między dwoma wariantami algorytmów LF okazały się niewielkie, zatem można przypuszczać, że dalsze testy również nie wykażą znaczącej istotności wyboru metody obliczania stopnia wierzchołka.

TABELA 5.9. Grafy pełne ze skojarzeniem w szkielecie,  $N = 2000$ ,  $\lambda = 2$ ,  $OPT = 2000$

Algorytm	Łączny czas	Optymalne rozwiązanie	Średnia liczba użytych kolorów
RandomSequential	9.061	85%	2000.1
LargestFirst <sup>1</sup>	9.069	100%	2000.0
SmallestLast	20.94	100%	2000.0
SaturatedLargestFirst	282.1	100%	2000.0
GreedyIndependentSet	15.94	100%	2000.0
LargestFirst <sup>2</sup>	9.254	100%	2000.0

TABELA 5.10. Grafy dwudzielne,  $N = 2000$ ,  $\rho_G = 0.5$ ,  $\rho_H = 0.1$ ,  $\lambda = 2$ ,  $OPT = 3$

Algorytm	Łączny czas	Optymalne rozwiązanie	Średnia liczba użytych kolorów
RandomSequential	1.565	20%	18.2
LargestFirst <sup>1</sup>	1.885	100%	3.0
SmallestLast	8.282	100%	3.0
SaturatedLargestFirst	12.81	100%	3.0
GreedyIndependentSet	9.389	100%	3.0
LargestFirst <sup>2</sup>	2.561	100%	3.0

TABELA 5.11. Grafy dwudzielne,  $N = 2000$ ,  $\rho_G = 0.9$ ,  $\rho_H = 0.5$ ,  $\lambda = 2$ ,  $OPT = 3$

Algorytm	Łączny czas	Optymalne rozwiązanie	Średnia liczba użytych kolorów
RandomSequential	3.449	78%	3.7
LargestFirst <sup>1</sup>	2.667	100%	3.0
SmallestLast	10.94	100%	3.0
SaturatedLargestFirst	22.66	100%	3.0
GreedyIndependentSet	10.16	100%	3.0
LargestFirst <sup>2</sup>	2.630	100%	3.0



TABELA 5.12. Grafy losowe,  $N = 2000$ ,  $\rho_G = 0.5$ ,  $\rho_H = 0.1$ ,  $\lambda = 2$ ,  $OPT = 5$ 

<b>Algorytm</b>	<b>Łączny czas</b>	<b>Optymalne rozwiązanie</b>	<b>Średnia liczba użytych kolorów</b>
RandomSequential	4.691	0%	73.8
LargestFirst <sup>1</sup>	4.693	0%	39.4
SmallestLast	15.16	8%	19.3
SaturatedLargestFirst	25.00	13%	6.3
GreedyIndependentSet	15.16	0%	6.7
LargestFirst <sup>2</sup>	4.241	0%	39.4

TABELA 5.13. Grafy losowe,  $N = 2000$ ,  $\rho_G = 0.7$ ,  $\rho_H = 0.3$ ,  $\lambda = 2$ ,  $OPT = 5$ 

<b>Algorytm</b>	<b>Łączny czas</b>	<b>Optymalne rozwiązanie</b>	<b>Średnia liczba użytych kolorów</b>
RandomSequential	5.948	0%	38.7
LargestFirst <sup>1</sup>	5.789	0%	11.4
SmallestLast	16.41	9%	7.2
SaturatedLargestFirst	38.44	0%	6.8
GreedyIndependentSet	15.31	0%	6.5
LargestFirst <sup>2</sup>	6.131	0%	11.2

TABELA 5.14. Grafy losowe,  $N = 2000$ ,  $\rho_G = 0.9$ ,  $\rho_H = 0.5$ ,  $\lambda = 2$ ,  $OPT = 5$ 

<b>Algorytm</b>	<b>Łączny czas</b>	<b>Optymalne rozwiązanie</b>	<b>Średnia liczba użytych kolorów</b>
RandomSequential	7.812	0%	12.0
LargestFirst <sup>1</sup>	7.970	0%	7.1
SmallestLast	18.91	0%	6.3
SaturatedLargestFirst	49.23	0%	6.5
GreedyIndependentSet	17.81	0%	6.4
LargestFirst <sup>2</sup>	7.661	0%	7.1

TABELA 5.15. Grafy losowe,  $N = 2000$ ,  $\rho_G = 0.7$ ,  $\rho_H = 0.1$ ,  $\lambda = 2$ ,  $OPT = 5$ 

<b>Algorytm</b>	<b>Łączny czas</b>	<b>Optymalne rozwiązanie</b>	<b>Średnia liczba użytych kolorów</b>
RandomSequential	5.011	0%	40.1
LargestFirst <sup>1</sup>	5.626	0%	24.9
SmallestLast	15.63	10%	19.3
SaturatedLargestFirst	28.44	0%	6.3
GreedyIndependentSet	15.63	0%	7.0
LargestFirst <sup>2</sup>	5.787	0%	25.3

TABELA 5.16. Grafy losowe,  $N = 2000$ ,  $\rho_G = 0.7$ ,  $\rho_H = 0.5$ ,  $\lambda = 2$ ,  $OPT = 5$ 

<b>Algorytm</b>	<b>Łączny czas</b>	<b>Optymalne rozwiązanie</b>	<b>Średnia liczba użytych kolorów</b>
RandomSequential	6.412	0%	14.6
LargestFirst <sup>1</sup>	6.884	0%	8.3
SmallestLast	16.09	39%	5.5
SaturatedLargestFirst	36.41	0%	6.7
GreedyIndependentSet	15.01	0%	6.7
LargestFirst <sup>2</sup>	6.522	0%	8.1

TABELA 5.17. Grafy losowe,  $N = 2000$ ,  $\rho_G = 0.5$ ,  $\rho_H = 0.1$ ,  $\lambda = 2$ ,  $OPT = 500$ 

<b>Algorytm</b>	<b>Łączny czas</b>	<b>Optymalne rozwiązanie</b>	<b>Średnia liczba użytych kolorów</b>
RandomSequential	5.316	31%	501.0
LargestFirst <sup>1</sup>	5.780	69%	500.3
SmallestLast	14.69	91%	500.1
SaturatedLargestFirst	110.6	100%	500.0
GreedyIndependentSet	13.59	0%	568.4
LargestFirst <sup>2</sup>	5.281	62%	500.5

TABELA 5.18. Grafy losowe,  $N = 2000$ ,  $\rho_G = 0.9$ ,  $\rho_H = 0.5$ ,  $\lambda = 2$ ,  $OPT = 500$ 

Algorytm	Łączny czas	Optymalne rozwiązanie	Średnia liczba użytych kolorów
RandomSequential	10.94	0%	899.5
LargestFirst <sup>1</sup>	10.94	0%	876.8
SmallestLast	23.91	0%	882.0
SaturatedLargestFirst	233.6	0%	851.6
GreedyIndependentSet	17.66	0%	894.5
LargestFirst <sup>2</sup>	10.00	0%	880.2

Wraz ze wzrostem liczby wierzchołków, wszystkie algorytmy poza metodą **RS** radzą sobie optymalnie z grafami pełnymi ze skojarzeniem w szkielecie oraz grafami dwudzielnymi z dowolnym szkieletem. Algorytm **RS**, tak jak w poprzednim teście, znajduje na ogół najgorsze rozwiązania spośród wszystkich przedstawionych metod.

W przypadku grafów losowych o niewielkiej szkieletowej liczbie chromatycznej, algorytmy **SLF** i **GIS** przeważają zdecydowanie nad konkurencją, choć i one niestety rzadko potrafią znaleźć najlepsze istniejące rozwiązanie. Wraz ze wzrostem gęstości szkieletu można zaobserwować znaczącą poprawę jakości działania algorytmów **LF** i **SL**. Ten drugi dla szkieletów gęstych ( $\rho_H \leq 0.3$ ) osiąga wyniki zbliżone do wspomnianych, najlepszych metod **SLF** oraz **GIS**.

Przedstawione algorytmy znacznie lepiej rozwiązywały problem dla rzadkich grafów o rzadkich szkieletach i dużej wartości szkieletowej liczby chromatycznej. Jedy- nym wyjątkiem jest tu algorytm **GIS**, który nigdy nie potrafił znaleźć najlepszego rozwiązania – co udało się nawet algorytmowi **RS**. Dla grafów gęstszych wyniki osią- gane przez algorytmy przybliżone znacząco odbiegały od wartości optymalnej.

Dla dużych grafów można zaobserwować znaczący wzrost czasu działania algo- rytmu **SLF**, związany z koniecznością przechowywania i wyszukiwania kolorów za- bronionych dla każdego wierzchołka. Czas działania algorytmów **GIS** oraz **SL** jest 2 – 3-krotnie dłuższy od najszybszych metod: **RS** i **LF**, natomiast algorytm **SLF** po- trzebuje – w zależności od gęstości grafu – od kilku do nawet kilkudziesięciu razy więcej czasu niż pozostałe.

Również ten test potwierdza brak znaczących różnic w wynikach osiąganych przez algorytm **LF** w zależności od wyboru metody obliczania stopnia wierzchołka na podstawie liczby krawędzi incydentnych w grafie i w szkielecie. Można stąd wnio- skować, że krotność krawędzi wierzchołkowej ma znaczenie drugorzędne i jej wpływ na wynik algorytmu jest mało istotny, o ile będzie to wartość rzędu wielkości para- metru  $\lambda$ .

TABELA 5.19. Grafy pełne ze skojarzeniem w szkieletcie,  $N = 2000$ ,  $\lambda = 100$ ,  $OPT = 2000$ 

Algorytm	Łączny czas	Optymalne rozwiązanie	Średnia liczba użytych kolorów
RandomSequential	8.133	12%	2052.5
LargestFirst <sup>1</sup>	10.00	25%	2027.8
SmallestLast	19.69	100%	2000.0
SaturatedLargestFirst	311.1	0%	2078.5
GreedyIndependentSet	14.69	100%	2000.0
LargestFirst <sup>2</sup>	9.556	25%	2026.0

TABELA 5.20. Grafy dwudzielne,  $N = 2000$ ,  $\rho_G = 0.5$ ,  $\rho_H = 0.1$ ,  $\lambda = 100$ ,  $OPT = 101$ 

Algorytm	Łączny czas	Optymalne rozwiązanie	Średnia liczba użytych kolorów
RandomSequential	4.840	0%	801.5
LargestFirst <sup>1</sup>	3.912	90%	191.0
SmallestLast	9.388	100%	101.0
SaturatedLargestFirst	112.50	100%	101.0
GreedyIndependentSet	7.812	100%	101.0
LargestFirst <sup>2</sup>	4.463	88%	193.0

TABELA 5.21. Grafy dwudzielne,  $N = 2000$ ,  $\rho_G = 0.9$ ,  $\rho_H = 0.5$ ,  $\lambda = 100$ ,  $OPT = 101$ 

Algorytm	Łączny czas	Optymalne rozwiązanie	Średnia liczba użytych kolorów
RandomSequential	9.848	56%	181.0
LargestFirst <sup>1</sup>	6.564	100%	101.0
SmallestLast	15.31	100%	101.0
SaturatedLargestFirst	366.88	100%	101.0
GreedyIndependentSet	8.289	100%	101.0
LargestFirst <sup>2</sup>	7.163	100%	101.0

TABELA 5.22. Grafy losowe,  $N = 2000$ ,  $\rho_G = 0.5$ ,  $\rho_H = 0.1$ ,  $\lambda = 100$ ,  $\chi(G) = 5$ 

<b>Algorytm</b>	<b>Łączny czas</b>	<b>Średnia liczba użytych kolorów</b>
RandomSequential	14.38	3354.6
LargestFirst <sup>1</sup>	7.971	1243.5
SmallestLast	18.28	511.6
SaturatedLargestFirst	215.00	261.4
GreedyIndependentSet	13.75	224.2
LargestFirst <sup>2</sup>	8.240	1229.9

TABELA 5.23. Grafy losowe,  $N = 2000$ ,  $\rho_G = 0.7$ ,  $\rho_H = 0.3$ ,  $\lambda = 100$ ,  $\chi(G) = 5$ 

<b>Algorytm</b>	<b>Łączny czas</b>	<b>Średnia liczba użytych kolorów</b>
RandomSequential	18.28	2297.1
LargestFirst <sup>1</sup>	14.53	392.0
SmallestLast	25.00	201.8
SaturatedLargestFirst	509.69	281.2
GreedyIndependentSet	15.31	202.5
LargestFirst <sup>2</sup>	15.61	390.5

TABELA 5.24. Grafy losowe,  $N = 2000$ ,  $\rho_G = 0.9$ ,  $\rho_H = 0.5$ ,  $\lambda = 100$ ,  $\chi(G) = 5$ 

<b>Algorytm</b>	<b>Łączny czas</b>	<b>Średnia liczba użytych kolorów</b>
RandomSequential	23.59	703.1
LargestFirst <sup>1</sup>	20.63	361.4
SmallestLast	33.28	201.9
SaturatedLargestFirst	796.41	271.3
GreedyIndependentSet	18.28	202.2
LargestFirst <sup>2</sup>	19.58	362.9

TABELA 5.25. Grafy losowe,  $N = 2000$ ,  $\rho_G = 0.7$ ,  $\rho_H = 0.1$ ,  $\lambda = 100$ ,  $\chi(G) = 5$ 

<b>Algorytm</b>	<b>Łączny czas</b>	<b>Średnia liczba użytych kolorów</b>
RandomSequential	12.81	3195.7
LargestFirst <sup>1</sup>	10.00	1332.6
SmallestLast	18.75	472.6
SaturatedLargestFirst	225.31	281.3
GreedyIndependentSet	16.09	202.7
LargestFirst <sup>2</sup>	11.04	1324.3

TABELA 5.26. Grafy losowe,  $N = 2000$ ,  $\rho_G = 0.7$ ,  $\rho_H = 0.5$ ,  $\lambda = 100$ ,  $\chi(G) = 5$ 

<b>Algorytm</b>	<b>Łączny czas</b>	<b>Średnia liczba użytych kolorów</b>
RandomSequential	21.56	973.2
LargestFirst <sup>1</sup>	19.22	301.3
SmallestLast	30.63	201.6
SaturatedLargestFirst	764.84	261.4
GreedyIndependentSet	15.31	202.3
LargestFirst <sup>2</sup>	20.61	302.0

TABELA 5.27. Grafy losowe,  $N = 2000$ ,  $\rho_G = 0.5$ ,  $\rho_H = 0.1$ ,  $\lambda = 100$ ,  $\chi(G) = 500$ 

<b>Algorytm</b>	<b>Łączny czas</b>	<b>Średnia liczba użytych kolorów</b>
RandomSequential	15.16	7642.5
LargestFirst <sup>1</sup>	12.19	7281.8
SmallestLast	19.84	7418.5
SaturatedLargestFirst	589.22	7180.7
GreedyIndependentSet	10.00	5249.2
LargestFirst <sup>2</sup>	14.24	7234.9

TABELA 5.28. Grafy losowe,  $N = 2000$ ,  $\rho_G = 0.9$ ,  $\rho_H = 0.5$ ,  $\lambda = 100$ ,  $\chi(G) = 500$ 

Algorytm	Łączny czas	Średnia liczba użytych kolorów
RandomSequential	51.25	34354.5
LargestFirst <sup>1</sup>	49.38	33538.8
SmallestLast	60.94	34625.6
SaturatedLargestFirst	2003.9	32046.2
GreedyIndependentSet	13.44	22803.5
LargestFirst <sup>2</sup>	47.61	33601.9

Znaczne zwiększenie wartości parametru  $\lambda$  wydłużyło czasy działania dla metod kolorowania sekwencyjnego, co zgadza się z teoretycznymi analizami złożoności obliczeniowej poszczególnych algorytmów. Dodatkowo, można zaobserwować znaczne wydłużenie łącznego czasu działania algorytmu SLF, w przypadku szkielet jest gęsty. Jest to skutek zwiększenia czasu obliczeń stopnia nasycenia wierzchołka: im większą wartość ma parametr  $\lambda$ , tym większą liczbę barw zabronionych przez wierzchołki sąsiednie. Dzięki temu, pod względem czasu działania algorytm GIS jest kilkukrotnie szybszy od pozostałych algorytmów, polegających na szeregowaniu wierzchołków, a następnie ich zachłannym kolorowaniu.

Co ciekawe, algorytm GIS nie tylko gwarantuje jednakowy pesymistyczny czas działania niezależnie od wartości  $\lambda$ , ale zarazem niemal w każdym przypadku otrzymany wynik jest najlepszy spośród prezentowanych. W szczególności, dla grafów o dużej liczbie chromatycznej osiągnane wyniki są o 20 – 30% lepsze niż rezultaty pozostałych metod.

Algorytm SLF, pomimo swojego nieproporcjonalnie długiego czasu działania, osiąga wyniki gorsze tylko od metody GIS i – niekiedy – SL. Tak jak poprzednio, algorytmy LF i SL zdecydowanie lepiej kolorują grafy o gęstych szkieletach i niewielkiej liczbie chromatycznej, w takich przypadkach metoda SL dorównuje metodzie GIS, a nawet znajduje rozwiązania minimalnie lepsze.

Wszystkie algorytmy poza RS i LF rozwiązują poprawnie instancje z grafami dwudzielnymi, natomiast tylko metody SL oraz GIS poradziły sobie optymalnie z grafami pełnymi ze skojarzeniem w szkielecie. Tak jak w poprzednich testach, był to zestaw, dla którego algorytm SLF osiągał najgorsze wyniki względem pozostałych metod.

# Rozdział 6

---

## Wnioski

Problem kolorowania szkieletowego, będący rozwinięciem klasycznego kolorowania wierzchołkowego grafów, jest zagadnieniem stosunkowo nowym. Jednocześnie, problem ten jest silnie związany z takimi znanymi problemami jak poszukiwanie optymalnego  $T$ -kolorowania, kolorowanie radiowe lub  $L(2, 1)$ -etykietowanie. Niniejsza praca stanowi przegląd najważniejszych wyników w zakresie kolorowania szkieletowego grafów, w szczególności ograniczeń na szkieletową liczbę chromatyczną w ogólnym przypadku i dla różnych klas grafów.

Istnieje szereg hipotez oraz problemów otwartych dotyczących kolorowania szkieletowego. Przykładowo, poszukiwany jest dowód na maksymalną wartość szkieletowej liczby chromatycznej dla grafów planarnych ze skojarzeniem, ścieżką lub drzewem w szkielecie. Jednocześnie poszukiwane są ograniczenia dla różnych klas grafów, asymptotycznie lepsze od znanych ścisłych ograniczeń górnych.

Dotychczas nie badano algorytmów rozwiązujących zagadnienia kolorowania szkieletowego, dlatego w pracy przedstawiono szereg metod zaczerpniętych z kolorowania klasycznego. W rezultacie stwierdzono, że algorytm **SLF** zwraca najlepsze rozwiązania, gdy problem kolorowania szkieletowego jest zbliżony do klasycznego kolorowania wierzchołków. Niestety, wymaga on znacznie większych zasobów czasowo-pamięciowych niż pozostałe przedstawione metody. Ponadto wraz ze wzrostem wartości parametru  $\lambda$ , również pozostałe algorytmy szeregowania muszą ustąpić wobec ciekawej odmiany algorytmu zachłannego wyboru zbiorów niezależnych **GIS**.

Trzeba zaznaczyć, że znacząca większość ograniczeń i twierdzeń dla kolorowania szkieletowego grafów ogólnych bazuje na znajomości optymalnego rozwiązania odpowiedniego problemu kolorowania klasycznego. Ponieważ jest to problem NP-zupełny i, co więcej, bardzo trudno aproksymowalny, kolorowanie szkieletowe grafów można zakwalifikować do kategorii problemów jeszcze trudniejszych.

Sam fakt, że dla każdego przedstawionego algorytmu można podać bardzo małe



przykłady instancji dość trudnych i trudnych świadczy o trudności zagadnienia. Dodatkowym problemem jest możliwość zabrania przez wierzchołek innego koloru poza swoim. Wprowadza to relację sąsiedztwa kolorów – pojęcie nieobecne w zagadnieniu kolorowania klasycznego.

Opracowanie lepszych algorytmów dla tego modelu wymagałoby nie tylko dobrej metody kolorowania szkieletu, ale także implementacji łatwego przekolorowywania. Dotyczy to głównie przypadku, gdy zwykłe kolorowanie zachłanne powoduje zakazanie zbyt wielu kolorów oraz, w konsekwencji, powstanie lokalnie nieoptymalnego kolorowania propagowanego na resztę grafu.

Ciekawym w takiej sytuacji podejściem jest badanie szczególnych klas grafów, dla których problem kolorowania jest wielomianowy, i projektowanie wyspecjalizowanych algorytmów, wykorzystujących własności charakterystyczne dla tych klas. Oprócz poszukiwania wielomianowych algorytmów dokładnych, warto szukać algorytmów przybliżonych z dobrą gwarancją aproksymacji.



# Bibliografia

- [1] K. I. Appel, W. Haken. Every planar map is four colorable. Part I: Discharging. *Illinois Journal of Mathematics*, 21(3):429–490, 1977.
- [2] D. Brélaz. New methods to color the vertices of a graph. *Communications of the ACM*, 22(4):251–256, 1979.
- [3] H. J. Broersma. A general framework for coloring problems: old results, new results, and open problems. W: *Combinatorial Geometry and Graph Theory*, strony 65–79. Springer Berlin / Heidelberg, 2003.
- [4] H. J. Broersma, F. V. Fomin, P. A. Golovach, G. J. Woeginger. Backbone colorings for graphs: tree and path backbones. *Journal of Graph Theory*, 55(2):137–152, 2007.
- [5] H. J. Broersma, J. Fujisawa, B. Marchal, D. Paulusma, A. N. M. Salman, K. Yoshimoto.  $\lambda$ -backbone colorings along pairwise disjoint stars and matchings. *Discrete Mathematics*, 309(18):5596–5609, 2009.
- [6] H. J. Broersma, J. Fujisawa, K. Yoshimoto. Backbone colorings along perfect matchings. Memorandum, 2003.
- [7] H. J. Broersma, B. Marchal, D. Paulusma, A. N. M. Salman. Improved upper bounds for  $\lambda$ -backbone colorings along matchings and stars. W: J. van Leeuwen, G. F. Italiano, W. van der Hoek, C. Meinel, H. Sack, F. Plasil (red.), *SOFSEM 2007: Theory and Practice of Computer Science, 33rd Conference on Current Trends in Theory and Practice of Computer Science, Harrachov, Czech Republic, January 20-26, 2007, Proceedings*, Lecture Notes in Computer Science.
- [8] H. J. Broersma, B. Marchal, D. Paulusma, A. N. M. Salman. Backbone colorings along stars and matchings in split graphs: their span is close to the chromatic number. *Discussiones Mathematicae Graph Theory*, 29(1):143–162, 2009.
- [9] M. R. Garey, D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., 1979.

- 
- [10] R. Halin. Über simpliziale zerfällungen beliebiger (endlicher oder unendlicher) graphen. *Mathematische Annalen*, 156(3):216–225, 1964.
- [11] P. L. Hammer, S. Földes. Split graphs. *Congressum Numerantium*, 19:311–315, 1977.
- [12] P. Hansen, J. Kuplinsky. The smallest hard-to-color graph. *Discrete Mathematics*, 96(3):199–212, 1991.
- [13] D. S. Johnson. Approximation algorithms for combinatorial problems. W: A. V. Aho, A. Borodin, R. L. Constable, R. W. Floyd, M. A. Harrison, R. M. Karp, H. R. Strong (red.), *Proceedings of the 5th Annual ACM Symposium on Theory of Computing*, strony 38–49, 1973.
- [14] M. Kubale (red.). *Optymalizacja dyskretna. Modele i metody kolorowania grafów*. Wydawnictwa Naukowo-Techniczne, 2002.
- [15] D. R. Lick, A. T. White.  $k$ -degenerate graphs. *Canadian Journal of Mathematics*, 22:1082–1096, 1970.
- [16] D. W. Matula, L. L. Beck. Smallest-last ordering and clustering and graph coloring algorithms. *Journal of the ACM*, 30(3):417–427, 1983.
- [17] J. Miskuf, R. Skrekovski, M. Tancer. Backbone colorings and generalized Mycielski graphs. *SIAM Journal on Discrete Mathematics*, 23(2):1063–1070, 2009.
- [18] J. Miskuf, R. Skrekovski, M. Tancer. Backbone colorings of graphs with bounded degree. *Discrete Applied Mathematics*, 158(5):534–542, 2010.
- [19] A. N. M. Salman. *Contributions to Graph Theory*. PhD thesis, University of Twente, 2005.
- [20] K. Turowski. A note on fast approximate backbone coloring of split graphs with star-like backbones. *Zeszyty Naukowe Wydziału Elektroniki, Telekomunikacji i Informatyki Politechniki Gdańskiej*, 1(1):15–20, 2011.
- [21] K. Turowski. Optimal coloring of split graphs with matching backbone. *Przedłożone do druku*, 2011.
- [22] W. Wang, Y. Bu, M. Montassier, A. Raspaud. On backbone coloring of graphs. *Journal of Combinatorial Optimization*, strony 1–15, 2010.

- 
- [23] D. J. A. Welsh, M. B. Powell. An upper bound for the chromatic number of a graph and its application to timetabling problems. *The Computer Journal*, 10(1):85–86, 1967.