Knotenfärbung & Vier-Farben-Satz

Eric Kunze

29. Januar 2020

Inhaltsverzeichnis

Der Vortrag stützt sich zu großen Teilen auf die Ausführungen in [Bü10] und wurde um einige zusätzliche Resultate ergänzt.

1 Knotenfärbung

Definition 1.1. Eine **(Knoten)-Färbung** eines Graphen $G = (V, E)$ ist eine Abbildung $c\colon V \to S$ mit einer beliebigen Menge S, sodass

$$c(u) \neq c(v) \text{ für alle } u, v \in V \text{ mit } (u, v) \in E \tag{1.1}$$

Die Elemente von S heißen **Farben**.

Definition 1.2. Ein Graph ist k**-färbbar**, falls es eine Färbung $c\colon V \to \{1, \dots, k\}$ gibt. Das kleinste k, für das G noch k-färbbar ist, ist die **chromatische Zahl** $\chi(G)$ von G.

Für $G = (V, E)$ gilt

$$1 \leq \chi(G) \leq |V| \tag{1.2}$$

Nun stellt sich die Frage, wie man eine gültige Färbung eines Graphen erhält. Dies beantwortet der folgende Algorithmus.

Algorithmus 1.3 (Färbungsalgorithmus).

- Input: Graph $G = (V, E)$ mit $|V| = n$
- Output: Färbung $c\colon V \to \{1, 2, \ldots\}$

Schritt 1: Nummeriere die Knoten mit $v_1, \ldots, v_n$
Schritt 2: Färbe den Knoten v_1 mit der Farbe 1
Schritt 3: Betrachte der Reihenfolge nach alle Knoten und färbe diese mit der kleinstmöglichen Farbe

Diese Abschätzung in Gleichung (1.2) lässt sich unter Nutzung des Maximalgrades eines Graphen noch verschärfen. Wir bezeichnen mit $\Delta(G)$ den Maximalgrad von G, also $\Delta(G) := \max\{d(v) : v \in V\}$.

Satz 1.4. *Sei $G = (V, E)$ ein Graph. Dann gilt*

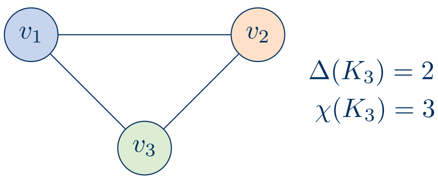
$$1 \leq \chi(G) \leq \Delta(G) + 1 \tag{1.3}$$

Beweis. Wir betrachten einen Knoten $v \in V$ mit $d(v) = \Delta(v)$. Algorithmus 1.3 färbt die $\Delta(v)$ vielen Nachbarknoten von v mit einer von v verschiedenen Farbe. Demnach brauchen wir also für den "schlechtesten" Knoten $\Delta(G) + 1$ viele verschiedene Farben. □

Die Abschätzung in Gleichung (1.3) in tatsächlich scharf, d.h. es gibt Graphen G, für die $\chi(G) = \Delta(G) + 1$ gilt:

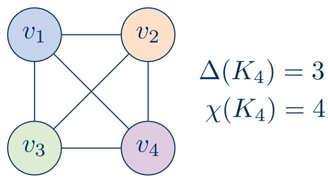
Kreise ungerader Länge

z.B. $G = K_3$



vollständige Graphen

z.B. $G = K_4$



Jedoch sind diese Klassen von Graphen auch schon die einzigen, die Gleichung (1.3) mit Gleichheit erfüllen. Somit kommt man in allen anderen Fällen in Algorithmus 1.3 auch mit $\Delta(G)$ vielen Farben aus. Die Idee dahinter ist eine clevere Anordnung der Knoten vor der Farbzuweisung. Kritisch sind nämlich nur die Fälle, in denen ein Knoten Maximalgrad hat, denn dann benötigt man bereits $\Delta(G)$ viele Farben für die Nachbarn und im Zweifelsfall eine weitere für den Knoten selbst. In vielen Fällen bleibt aber vor dem letzten Knoten eine Farbe übrig, sodass Algorithmus 1.3 keine zusätzliche beanspruchen muss.

Satz 1.5 (Brooks). *Sei G ein zusammenhängender Graph mit mehr als zwei Knoten.*

(i) Ist G vollständig oder ein Kreis ungerader Länge, dann gilt $\chi(G) = \Delta(G) + 1$.
(ii) Andernfalls gilt $\chi(G) \leq \Delta(G)$.

Beweis. Die Aussage (i) ist mit den obigen Beispielen klar. Aussage (ii) zeigen wir in mehreren Schritten und folgen dabei [CR14].

Fall 1 — Wir beschränken uns zunächst auf den Fall, dass G nicht regulär ist, d.h. nicht alle Knoten Maximalgrad $\Delta := \Delta(G)$ haben. Für $\Delta(G) \leq 2$ ist G ein Weg oder Kreis und die Aussage ist damit klar. Sei daher von nun an $\Delta(G) \geq 3$. Vollständige Induktion nach $|V|$ — Im Induktionsanfang betrachten wir $|V| = 3$, d.h. einen Graphen mit genau drei Knoten. Aufgrund der Annahme $\Delta(G) \geq 3$ ist hier nichts zu zeigen.

In der Induktionsvoraussetzung nehmen wir an, dass $\chi(H) \leq \Delta(H)$ für alle Graphen H mit $|H| < |G|$.

Im Induktionsschritt wählen wir einen Knoten $v \in V$ mit $d(v) < \Delta(G)$. Dieser existiert, weil gemäß unserer Annahme nicht alle Knoten Maximalgrad haben. Betrachten wir den Graphen $H := G - v$. Dieser besitzt weniger Knoten als G, also hat auch kein Knoten in H mehr Nachbarn als in G. Diese Einsicht liefert $\Delta(H) \leq \Delta(G)$ und mit der Induktionsvoraussetzung $\chi(H) \leq \Delta(H)$ existiert also insbesondere auch eine Δ-Färbung von H. Es verbleibt nun noch der Knoten v zu färben. Da v nur $d(v) < \Delta$ viele Nachbarn hat, bleibt mindestens eine der Δ vielen Farben für v übrig und wir erhalten damit eine Δ-Färbung von G.

Fall 2 — Ist G regulär, d.h. alle Knoten besitzen Maximalgrad Δ, dann existiert ein Knoten v wie oben nicht mehr. Ist v aber zumindest ein Artikulationspunkt, d.h. $H := G - v$ zerfällt in Komponenten, dann lässt sich das obige Verfahren noch retten. Sei dazu H' eine Komponente von H. Diese besitzt gemäß Induktionsvoraussetzung wie oben eine Δ-Färbung. Diese kann ebenso wie oben zu einer Δ-Färbung von $H' + v$ fortgesetzt werden, weil v in $H' + v$ nicht Maximalgrad hat (es fehlen die anderen Zusammenhangskomponenten, die "an v hängen"). Setzt man dieses Verfahren für alle Komponenten von H ein, erhält v unter Umständen jedes Mal eine andere Farbe. Dies behebt man, indem die Farben so zyklisch vertauscht (permutiert), dass v immer die gleiche Farbe bekommt. Dies stellt am Ende eine Δ-Färbung von G dar.

Fall 3 — Der verbleibende Fall beschäftigt sich nun mit Δ-regulären Graphen, die zweifach zusammenhängend sind, also weder einen Artikulationspunkt noch einen Knoten v mit $d(v) < \Delta(v)$ besitzen. In diesem Fall kann man zeigen, dass es einen induzierten Pfad[1] uvw in G gibt, sodass $H := G - \{u, w\}$ zusammenhängend ist. Färbt man nun die Knoten u und w gleich und verfährt mit H wie im ersten Fall,

[1] Ein induzierter Pfad ist ein Pfad, der gleichzeitig ein induzierter Teilgraph von G ist. Er ist also eine Folge von Knoten, sodass zwei im Pfad benachbarte Knoten mit einer Kante in G verbunden sind und zwei nicht-benachbarte Knoten im Pfad keine Kante in G besitzen. Ein induzierter Pfad wird auch Schlange genannt.

dann bleibt erneut eine der Δ Farben für v übrig, da mit u und w zwei Nachbarn gleiche Farben haben. $\square$

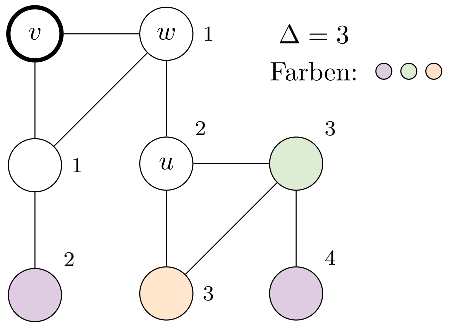


Abbildung 1: Variante von Algorithmus 1.3 zum Beweis von Satz 1.5. Die Reihenfolge der Färbung ist durch die Zahlen dargestellt. Der Knoten u hat Maximalgrad $d(u) = 3 = \Delta$. Da es auf dem Weg von v zu u aber den ungefärbten Nachbarn w von u gibt, bleibt mindestens eine Farbe übrig, diese erhält u.

Mithilfe von Algorithmus 1.3 sieht man auch leicht das folgende Resultat.

Satz 1.6 ([Bod20]). *Ein endlicher Graph G ist genau dann zweifärbbar, wenn er keine ungeraden Kreise enthält.*

Nun haben wir die chromatische Zahl $\chi(G)$ gut bezügliche der Anzahl an Knoten abgeschätzt. Im Folgenden wollen wir noch eine Abschätzung in Hinblick auf die Kantenzahl erzielen.

Satz 1.7 ([Die10]). *Sei $G = (V, E)$ ein Graph mit $|E| = m$. Dann gilt*

$$1 \leq \chi(G) \leq \frac{1}{2} + \sqrt{2m + \frac{1}{4}} \tag{1.4}$$

Beweis. Wir betrachten eine $k = \chi(G)$-Färbung c von G und entstehenden Farbklassen $c^{-1}(i)$. Betrachten wir zwei solche Farbklassen $c^{-1}(i)$, dann muss zwischen diesen beiden Klassen mindestens eine Kante existieren, da diese sonst gleich eingefärbt werden könnten und folglich zur gleichen Farbklasse gehörten. Jede der k Klassen hat also mindestens $k-1$ Kanten zu den benachbarten Klassen. Nun haben wir allerdings die Kanten doppelt gezählt, d.h. es gilt

$$m \geq \frac{1}{2}k(k-1) \quad \Rightarrow \quad k \leq \frac{1}{2} + \sqrt{2m + \frac{1}{4}}$$ $\square$

Wir können also mittlerweile einen Graphen G färben und die dazu benötigte Anzahl an Farben auch eingrenzen. Doch wie bestimmt man nun die chromatische Zahl $\chi(G)$ exakt? Dafür können wir Algorithmus 1.3 wie folgt nutzen.

(1) Angenommen wir kennen für einen Graphen G eine $\chi(G)$-Färbung.

(2) Sortiere nun die Knoten nach ihren Farben.

(3) Starte den Färbungsalgorithmus mit dieser neuen Sortierung

Man sieht leicht, dass der Algorithmus wieder eine $\chi(G)$-Färbung liefert.

Somit liegt die Idee nahe, “einfach” alle möglichen Anordnungen zu testen um am Ende die minimale Anzahl an Farben als chromatische Zahl zu setzen. Jedoch sind dabei im Allgemeinen $|V|!$ viele Fälle zu testen, was dieses Verfahren ineffizient macht.

Aus komplexitätstheoretischer Sicht ist die Bestimmung der chromatischen Zahl sogar ein NP-vollständiges Problem, d.h. es existiert vermutlich auch kein Algorithmus, der dieses Problem effizient lösen kann.

2 Vier-Farben-Problem

Wir betrachten eine Landkarte und wollen diese Einfärben unter Beachtung der Bedingung, dass aneinander grenzende Länder unterschiedlich gefärbt werden sollen. Wie viele Farben benötigen wir dafür?

Um das Problem zu lösen benötigen wir zuerst eine mathematische Modellierung, welche wir in dualen Graphen finden.

Definition 2.1. Sei $G = (V, E)$ ein planarer Graph. Der **duale Graph** $G^* = (V^*, E^*)$ zu einer planaren Darstellung von G hat in jeder Fläche von G genau einen Knoten. Außerdem gibt es für jede Kante $e \in E$ genau eine Kante $e^* \in E^*$, die die an e anliegenden Flächen verbindet.

Bemerkung 2.2. Die Darstellung des dualen Graphen G^* hängt von der planaren Darstellung des Graphen G ab und ist somit im Allgmeinen nicht eindeutig. Durch in Flächen hineinragende Kanten von G können Schlingen in G^* entstehen.

Bemerkung 2.3. Planarität überträgt sich auf den Dualgraphen. Betrachte dazu das folgende Vorgehen:

(1) zeichne alle Knoten des Dualgraphen G^* ein

(2) halbiere alle Kanten in G

(3) verbinde nun die Knoten mit den Halbierungspunkten durch "Kurven"

Nun haben wir unsere Landkarte in ein mathematisches Modell übertragen. Der folgende Satz erklärt uns, dass wir den daraus entstandenen (planaren) Graphen mit vier Farben einfärben können.

Satz 2.4 (Vier-Farben-Satz). *Jeder planare Graph lässt sich mit vier Farben färben.*

Der Beweis des Vier-Farben-Satzes ist bis heute nur mit einem Computer erbracht worden. Daher steht eine "endgültige", für den Menschen verständliche Lösung des Problems noch aus.

Wir wollen dennoch eine zumindest abgeschwächte Variante zeigen, nämlich den Fünf-Farben-Satz. Dafür erinnern wir uns zuerst an ein bekanntes Resultat, welches wir für den nachfolgenden Satz verwenden wollen.

Lemma 2.5. *Jeder planare Graph hat mindestens einen Knoten mit höchstens fünf Nachbarn.*

Satz 2.6 (Fünf-Farben-Satz). *Jeder planare Graph ist mit fünf Farben färbbar.*

Beweis. vollständige Induktion nach $n = |V|$:

(IA) Für $n \leq 5$ ist die Aussage klar.

(IV) Jeder planare Graph mit weniger als n Knoten ist fünffärbbar.

(IS) Sei $G = (V, E)$ ein planarer Graph in planarer Darstellung. Nach Lemma 2.5 existiert ein Knoten $v \in V$ mit höchstens fünf Nachbarn (d.h. $d(v) \leq 5$)

Fall 1: $d(v) \leq 4$ — Lösche v aus G und betrachte $G' := G - v$. Dieser hat nur noch $n - 1$ Knoten und ist nach Induktionsvoraussetzung entsprechend mit fünf Farben färbbar. Insbesondere benötigen wir für die vier Nachbarn von v auch maximal vier Farben. Fügen wir also v wieder hinzu und färben diesen Knoten mit der übrigen Farbe, so erhalten wir eine gültige 5-Färbung von G.

Fall 2: $d(v) = 5$ — Betrachten wir wieder $G' := G - v$ und finde eine zulässige 5-Färbung nach Induktionsvoraussetzung. Werden für die fünf Nachbarn nur vier Farben benötigt, so können wir wie in Fall 1 verfahren und v mit der übrigen Farbe färben. Nehmen wir also an, dass für alle fünf Nachbarn mit paarweise verschiedenen Farben gefärbt worden sind – nummeriere die Nachbarn mit $v_1, \ldots, v_5$ im Uhrzeigersinn, wobei der Index gleichzeitig auf die Farbe repräsentieren soll, die der entsprechende Knoten nach Induktionsvoraussetzung bekommen hat. Das Ziel ist nun durch Umfärben der Knoten $v_1, \ldots, v_5$ eine Farbe einzusparen. Betrachte dazu $H_{1,3}$, den Teilgraphen von G', der von den Knoten v_1 und v_3 induziert wird.

a) Wir nehmen an, dass es in $H_{1,3}$ keine Weg von v_1 nach v_3 gibt. Dann können wir in der Zusammenhangskomponente, die v_1 enthält die Knoten umfärben durch

$$\text{Farbe } 1 \leftrightarrow \text{ Farbe } 3$$

Damit ist v_1 nun mit Farbe 3 gefärbt und wir können v nach Hinzufügen die Farbe 1 zuweisen.

b) Angenommen die Knoten v_1 und v_3 sind in $H_{1,3}$ durch einen Weg verbunden. Da wir durch Umfärben in $H_{1,3}$ nichts erreichen würden, betrachten wir $H_{2,4}$. Zwischen v_2 und v_4 kann nun kein Weg in $H_{2,4}$ existieren, da G planar dargestellt war; wie oben bringt also Umfärben

$$\text{Farbe } 2 \leftrightarrow \text{ Farbe } 4$$

in der Zusammenhangskomponente von v_2 eine neue Färbungsmöglichkeit für v mit Farbe 2.

Somit haben wir in jedem Fall eine gültige 5-Färbung für G gefunden. □

3 Färbung und SAT

Wir wollen im Folgenden noch kurz den komplexitätstheoretischen Zusammenhang zwischen der Graphenfärbung und dem Erfüllbarkeitsproblem der Aussagenlogik (SAT) anreißen. Letzteres besteht darin, für eine gegebene aussagenlogische Formel F eine Variablenbelegung zu finden, die die Formel auf wahr abbildet. Wir nutzen dazu die Konstruktion in [Meh17].

Satz 3.1 (Färbung $\preccurlyeq$ SAT). *Falls SAT effizient lösbar ist, dann ist auch Graphenfärbung effizient lösbar — Gegeben sei ein Graph G. Konstruiere nun eine Formel F mit*

- *G ist genau dann (k-) färbbar, wenn F erfüllbar ist.*
- *Die Konstruktion von F aus G ist in Polynomialzeit durchführbar.*

Beispiel 3.2 (3-Färbbarkeit). Sei $G = (V, E)$ ein 3-färbbarer Graph mit Farben $S = \{a, b, c\}$. Wir wollen nun daraus eine Formel wie in Satz 3.1 finden und definieren (aussagenlogische) Variablen $x_a = \top$ genau dann, wenn der Knoten v mit Farbe a gefärbt worden ist. Nun müssen wir zwei Eigenschaften modellieren:

- Jeder Knoten hat eine eindeutige Farbe — Dazu definieren wir uns ein Prädikat GE, das genau dann auf wahr abgebildet wird, wenn genau eines der Argumente wahr ist:
$$\mathrm{GE}(x, y, z) = (x \lor y \lor z) \land \neg((x \land y) \lor (x \land z) \lor (y \land z))$$
Damit lässt sich die Eigenschaft genau eine Farbe zu tragen mit
$$\bigwedge_{v \in V} \mathrm{GE}(v_a, v_b, v_c)$$
in die Sprache der Aussagenlogik übersetzen.
- Für alle $(u, v) \in E$ haben u und v verschiedene Farben (vgl. (1.1)) — Definieren wir ein Prädikat $\mathrm{VF}(u, v)$, das dann wahr ergibt, wenn u und v verschiedene Farben tragen:
$$\mathrm{VF}(u, v) = \neg((u_a \land v_a) \lor (u_b \land v_b) \lor (u_c \land v_c))$$
Die Eigenschaft (1.1) lässt sich also formulieren als
$$\bigwedge_{(u,v) \in E} \mathrm{VF}(u, v)$$

Schlussendlich ist die "zu G äquivalente" Formel gegeben durch

$$F = \bigwedge_{v \in V} \mathrm{GE}(v_a, v_b, v_c) \land \bigwedge_{(u,v) \in E} \mathrm{VF}(u, v)$$

Literatur

[Bod20] Bodirsky, Manuel: *Diskrete Strukturen.* Vorlesungsskript. `http://www.math.tu-dresden.de/~bodirsky/lehre/Diskrete-Strukturen.pdf`. Version: Januar 2020

[Bü10] Büsing, Christina: *Graphen- und Netzwerkoptimierung.* Heidelberg : Spektrum Akademischer Verlag, 2010. – ISBN 9783827424228

[CR14] Cranston, Daniel W. ; Rabern, Landon: Brooks' Theorem and Beyond. (2014). `http://dx.doi.org/10.48550/ARXIV.1403.0479`. – DOI 10.48550/ARXIV.1403.0479

[Die10] Diestel, Reinhard: *Graphentheorie.* 4. Auflage. Heidelberg [u.a.] : Springer, 2010. – ISBN 9783642149115

[Meh17] Mehlhorn, Kurt: *Welche Probleme können Rechner (effizient) lösen? Die P = NP Frage.* Vortragsfolien. `https://www.mpi-inf.mpg.de/fileadmin/inf/d1/teaching/winter16/ideen/folien/Komplexitaet1617.pdf`. Version: Januar 2017