

---

# Pseudostabile Mengen in Graphen

---

INAUGURAL-DISSERTATION  
ZUR  
ERLANGUNG DES DOKTORGRADES  
DER MATHEMATISCH-NATURWISSENSCHAFTLICHEN FAKULTÄT  
DER UNIVERSITÄT ZU KÖLN

vorgelegt von  
Jens Dörpinghaus  
aus Lindlar

Köln 2018

Berichterstatter:

PROF. DR. RAINER SCHRADER  
Institut für Informatik  
Universität zu Köln  
Deutschland

PROF. DR. HUBERT RANERAT  
Institut für Nachrichtentechnik (INT)  
TH Köln  
Deutschland

Tag der mündlichen Prüfung: 8.1.2018

## Zusammenfassung

Diese Arbeit beschäftigt sich mit der Überdeckung von einfachen Graphen mit pseudostabilen Mengen und der minimalen Zerlegung von Graphen in pseudostabile Mengen. Dabei ist die Wertigkeit einer minimalen Zerlegung des Graphen  $G = (V, E)$  durch  $\zeta(G)$  gegeben.

Pseudostabile Mengen stellen eine Verallgemeinerung von stabilen Mengen dar. Pseudostabile Mengen zerfallen in stabile Mengen, erlauben aber unter verschiedenen Nebenbedingungen bestimmte Pfade zwischen diesen. Zur Charakterisierung dieser Pfade wird der darstellende Graph  $G_P$  einer Zerlegung  $P$  des Graphen  $G = (V, E)$  in pseudostabile Mengen betrachtet. Dabei führen bestimmte Voraussetzungen zu verschiedenen Unterproblemen, die in dieser Arbeit definiert und untersucht werden. Alle Probleme sind im Allgemeinen  $\mathcal{NP}$ -vollständig, wie in der vorliegenden Arbeit gezeigt wird. Es werden allerdings auch Graphenklassen beschrieben, auf denen die einzelnen Probleme in polynomieller Zeit lösbar sind.

Pseudostabile Mengen erlauben stets nur einen Pfad zwischen stabilen Teilmengen, mehrfach pseudostabile Mengen mehrere Pfade. Die beiden in dieser Arbeit betrachteten Hauptprobleme sind minPS – eine minimale Zerlegung eines Graphen  $G$  in pseudostabile Mengen – und minMPS – eine Zerlegung eines Graphen  $G$  in mehrfach pseudostabile Mengen.

Eine Zerlegung in pseudostabile Mengen erlaubt nur einen Pfad der Länge 3 zwischen zwei stabilen Mengen. Für den darstellenden Graphen  $G_P$  gilt, dass er kreisfrei ist und für alle  $v \in V(G_P)$  gilt, dass  $\delta(v) \in \{0, 1, 2\}$ . Diese Zerlegung entspricht der optimalen Lösung verschiedener Rangierprobleme auf Güterbahnhöfen. Dies wird in der vorliegenden Arbeit vertieft und es werden von dieser neuen Formulierung aus weitere Lösungsheuristiken und Schranken hergeleitet.

Mehrfach pseudostabile Mengen erlauben mehrere Pfade der Länge 3 zwischen stabilen Mengen. Gibt es keine Einschränkungen auf dem darstellenden Graphen  $G_P$  so ist dies die allgemeinste Fassung des Problems. In dieser Arbeit wird gezeigt, dass dieses Problem eine graphentheoretische Umformulierung des *soft Document Clustering* ist. Es ist bereits bekannt, dass *hard Document Clustering* einer Zerlegung eines Dokumentengraphen in stabile Mengen bzw. Cliquen entspricht. Hier wird die Verallgemeinerung auf das weiter gefasste *soft Document Clustering* diskutiert und es werden Lösungsheuristiken diskutiert.

Neben diesen beiden Unterproblemen und den dazugehörigen Anwendungsproblemen werden auch weitere mögliche Verallgemeinerungen, zum Bei-

spiel Pseudocliquen, definiert. Eine minimale Zerlegung eines einfachen Graphen  $G = (V, E)$  in Pseudocliquen hat dabei die Wertigkeit  $\bar{\zeta}(G)$ . Diese betten sich nahtlos in die in der vorliegenden Arbeit diskutierten Grundlagen für pseudostabile Mengen ein, da gezeigt wird, dass für den komplementären Graphen  $\bar{G}$   $\bar{\zeta}(\bar{G}) = \zeta(G)$  gilt.

## Abstract

This thesis introduces the covering of simple graphs with pseudostable sets and the minimal partition of a graph in pseudostable sets. The weight of a minimal partition of a graph  $G = (V, E)$  is denoted by  $\zeta(G)$ .

Pseudostable sets are a generalization of stable sets. They consist of one or two stable sets, but allow to add distinct paths between them. To characterize them and to clarify the conditions, the graph of a set covering  $G_P$  to a graph partition  $P$  of the graph  $G = (V, E)$  in pseudostable sets is introduced. Different conditions lead to different subproblems. This thesis shows that all problems are in general  $\mathcal{NP}$ -complete. But it is possible to show some graph classes for which some problems are computable in polynomial time.

Pseudostable sets allow only one path between stable subsets, multiple pseudostable sets allow more than one path. This thesis discusses two subproblems in detail: minPS – a minimal partition of a graph  $G$  in pseudostable sets – and minMPS – a minimal partition of a graph  $G$  in multiple pseudostable sets.

The problem minPS searches for a minimal set covering  $P$  of the graph  $G = (V, E)$  with a subset  $B \subset G$  of blue nodes and edges with pseudostable tuples  $T$  where  $G_P$  is acyclic and  $\delta(v) \in \{0, 1, 2\}$  for all  $v \in V(G_P)$ . This problem is equivalent to the Train Marshalling Problem covering the rearrangement of cars of an incoming train in a hump yard. This thesis will discuss several novel insights, bounds and heuristics for this problem.

The problem minMPS searches for a minimal set covering  $P$  of the graph  $G = (V, E)$  with a subset  $B \subset G$  of blue nodes and edges with multiple pseudostable tuples  $M$  where  $G_P$  is acyclic and  $\delta(v) \in \{0, 1, 2\}$  for all  $v \in V(G_P)$ . If there are no restrictions on the graph  $G_P$ , this is the most general formulation. It is a graph theoretical reformulation of soft document clustering. Hard document clustering is well known for being solved by a graph partition into stable sets or cliques. Thus this work will discuss the more general formulation with pseudostable sets.

Beside these two subproblems and their application this thesis also considers other related topics like pseudocliques. A minimal partition of a simple graph  $G = (V, E)$  in pseudocliques has the value  $\bar{\zeta}(G)$ . We will show that for the complementary graph  $\bar{G}$   $\bar{\zeta}(\bar{G}) = \zeta(G)$  holds.



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Zielsetzung . . . . .	1
1.2	Bekannte Resultate . . . . .	2
<b>2</b>	<b>Definitionen und Grundlagen</b>	<b>5</b>
2.1	Graphen . . . . .	5
2.2	Intervallgraphen . . . . .	7
2.3	Kreise in Graphen . . . . .	9
2.4	Komplexitätstheorie . . . . .	9
2.5	Lineare Programmierung . . . . .	11
2.6	Das Mengenüberdeckungsproblem . . . . .	12
<b>3</b>	<b>Strukturen und Eigenschaften Pseudostabiler Mengen</b>	<b>15</b>
3.1	Definition . . . . .	15
3.2	Strukturen von pseudostabilen Mengen . . . . .	29
3.3	Zusammenhänge zwischen pseudostabilen und stabilen Mengen . . . . .	37
3.4	Zusammenfassung . . . . .	41
<b>4</b>	<b>Pseudostabile Mengen: Komplexität, Schranken und Algorithmen</b>	<b>43</b>
4.1	minPS . . . . .	43
4.2	minPS ist $\mathcal{NP}$ -vollständig . . . . .	47
4.3	Abschätzungen für minPS . . . . .	53
4.4	minPS auf disjunkten großen Diamanten . . . . .	58
4.5	minPS mit disjunkten Listen von Pfaden . . . . .	61
4.6	Lineare Programmierung . . . . .	62
4.7	minPS in Verbindung mit dem komplementären Graphen . . . . .	64
4.8	Zusammenfassung . . . . .	66
<b>5</b>	<b>Sortier- und Rangierprobleme in Güterbahnhöfen</b>	<b>67</b>
5.1	SRSP . . . . .	69
5.2	TMP . . . . .	74
5.3	Allgemeine Definitionen und Notationen . . . . .	79
5.4	TMP und minPS . . . . .	83
5.5	Der Zusammenhang zwischen minPS und TMP . . . . .	84

5.6	Bekannte Abschätzungen und Algorithmen . . . . .	92
5.7	Neue Resultate . . . . .	100
5.8	Auswertung . . . . .	102
5.9	Zusammenfassung . . . . .	108
<b>6</b>	<b>Mehrfach pseudostabile Mengen: Schranken und Algorithmen</b>	<b>109</b>
6.1	minMPS . . . . .	109
6.2	Abschätzungen für minMPS . . . . .	111
6.3	Lineare Programmierung . . . . .	122
6.4	minMPS in Verbindung mit dem komplementären Graphen	122
6.5	Zusammenfassung . . . . .	123
<b>7</b>	<b>Document Clustering</b>	<b>125</b>
7.1	Definition und Grundlagen . . . . .	126
7.2	Dokumentenähnlichkeit . . . . .	130
7.3	Ein graphentheoretischer Ansatz des Clusterings . . . . .	133
7.4	Neue Ansätze . . . . .	136
7.5	Document Clustering auf MEDLINE . . . . .	142
7.6	Implementierung und Auswertung . . . . .	142
7.7	Zusammenfassung . . . . .	150
<b>8</b>	<b>Fazit und Ausblick</b>	<b>153</b>
	<b>Abbildungsverzeichnis</b>	<b>158</b>
	<b>Glossar</b>	<b>160</b>
	<b>Literaturverzeichnis</b>	<b>161</b>



# Einleitung

Die vorliegende Arbeit beschäftigt sich mit der Überdeckung bzw. der Zerlegung von Graphen in pseudostabile Mengen und untersucht diese im Hinblick auf Komplexität und mögliche Schranken sowie Heuristiken. Zunächst werden in Kapitel 2 verschiedene Aspekte der Graphen- und der Komplexitätstheorie beschrieben. Danach folgt ein weiteres Kapitel, das die Struktur und Eigenschaften von pseudostabilen Mengen betrachtet. Hier werden grundlegende Notationen und die Definition mittels Graphenüberdeckung und Graphenzerlegung eingeführt.

Es werden zwei spezielle Probleme, die Minimierungsprobleme  $\text{minPS}$  in Kapitel 5.6.2 und  $\text{minMPS}$  in Kapitel 6, theoretisch dargestellt und analysiert. In einem folgenden Kapitel werden wiederum angewandte Probleme dargestellt, die mit diesem Ansatz gelöst werden können. Damit ist dieser Teil der Arbeit der Anwendung der im theoretischen Teil gewonnenen Resultate auf dieses Problem gewidmet.

So folgt in Kapitel 5 dann eine allgemeine Definition von Klassen der Zugbildungs- und Rangierprobleme nach [41], danach die konkrete Problembeschreibung des TMP und ihrer Grundlagen nach [16] und [8]. In Kapitel 7 wiederum wird das Document Clustering näher beschrieben.

## 1.1 Zielsetzung

Neben der Darstellung der Minimierungsprobleme  $\text{minPS}$  und  $\text{minMPS}$  sollen bekannte Abschätzungen und Heuristiken betrachtet werden. Diese sollen danach auch anhand der angewandten Probleme analysiert und diskutiert werden. Somit ist ein Ziel der vorliegenden Arbeit die Verbesserung bekannter und die Entwicklung neuer Ansätze.

Rangierbahnhöfe haben infolge des auf der Schiene abgewickelten Güterverkehrs eine enorme Bedeutung. Dort werden Züge sortiert und neu zusammengestellt, damit sie mit wenig Aufwand möglichst schnell ihr Ziel erreichen. Die dort auftretenden Sortierprobleme sind in aller Regel algorithmisch schwer zu behandeln. Für das bekannte TMP werden mithilfe der Ergebnisse dieser Arbeit neue und effiziente Heuristiken hergeleitet.

*Document Clustering* versucht die Sortierung bzw. Kategorisierung von Textdokumenten. Es wird gezeigt, dass ein Teilaspekt dieses Problems als graphentheoretisches Problem dargestellt werden und mit Hilfe der neuen Ergebnisse dieser Arbeit gelöst werden kann. Es sollen verschiedene Ansätze untersucht werden.

Ziel dieser Arbeit ist also zum einen die exakte Definition der pseudostabilen Mengen und ihre gründliche theoretische Untersuchung und zum anderen die Verbesserung der Ergebnisse zu verschiedenen Rangierproblemen, insbesondere zu [8] und des *Document Clustering*.

## 1.2 Bekannte Resultate

Pseudostabile Mengen sind als graphentheoretische Konstruktion noch nicht näher untersucht worden. Eine Teilveröffentlichung dieser Arbeit mit dem Fokus auf *Document Clustering* findet sich in [21].

Eine allgemeine Definition von Klassen der Zugbildungs- und Rangierprobleme wird in [41] gegeben, Ergebnisse zum TMP in den Arbeiten von Dahlhaus u. a. und Beygang ([16] und [8]).

Das TMP ist ein Spezialfall des allgemeinen Zugbildungsproblems. Dabei werden Stumpfgleise zum Rangieren benutzt, um einen Zug zu erhalten, in dem die Wagen nach bestimmten Kriterien sortiert sind. Die Wagen dürfen innerhalb ihrer Kriterien aber eine beliebige Abfolge haben.

Im Jahr 2000 wurde von Dahlhaus u. a. in [16] bewiesen, dass das TMP  $\mathcal{NP}$ -vollständig ist. Sie leiteten außerdem einige Schranken her. Bereits 1983 gab es von Zhu u. Zhu, die das Problem unter verschiedenen Nebenbedingungen betrachteten, erste Resultate und polynomielle Algorithmen [87]. In einer weiteren Arbeit von Dahlhaus u. a. [17] wurden unter anderem verwandte Probleme betrachtet. Weitere Schranken und Algorithmen finden sich in der Arbeit von Beygang [8] bzw. Beygang u. a. [9]. Das Problem wurde ebenfalls in der Arbeit von Hansmann [41] und Brueggemann u. a. [14] bearbeitet.

Diese Arbeit baut im Wesentlichen auf die Ergebnisse von Dahlhaus, Horak, Miller und Ryan und Beygang auf und entwickelt daraus eine äquivalente graphentheoretische Formulierung des TMP. Ziel dieser Arbeit ist es, aus dieser Formulierung neue polynomielle Algorithmen und Abschätzungen für das TMP herzuleiten.

Document Clustering wurde bislang vor allem in seiner Anwendung betrachtet, der Fokus lag auf Methoden und Heuristiken. Die Autoren in [55] versuchen zum Beispiel Dokumente aus der MEDLINE-Datenbank mit evolutionären Algorithmen zu untersuchen, [65] hingegen nutzen Verfahren aus dem Bereich des maschinellen Lernens. Nur wenige Autoren wie beispielsweise [76] benutzen Verfahren, die sich der Methoden aus der Graphentheorie bedienen. Oft werden auch nur spezielle Varianten untersucht. Die Autoren in [45] etwa untersuchen das Problem im Kontext der Durchführung von Suchen. Andere Autoren wie [61] untersuchen das breitere Feld des *Hierarchical Clustering*.

Grundsätzlich ist die eigentliche Fragestellung aber schon wesentlich älter, worauf auch [62] hinweist. Hier soll das Problem des *soft Document Clustering* neu als graphentheoretisches Problem formuliert werden und es soll der Frage nachgegangen werden, ob sich dadurch neue Heuristiken und Abschätzungen finden.



# Definitionen und Grundlagen

## 2.1 Graphen

Ein (*ungerichteter*) *Graph*  $G$  besteht aus einer endlichen Menge  $V$  von Knoten und einer endlichen Menge  $E \subset V \times V$ , deren Elemente  $e = (u, v) = (v, u) \in E$  als Kanten bezeichnet werden. Eine Kante  $e = (u, v) \in E$  verbindet die Knoten  $u$  und  $v$ , die bei einem ungerichteten Graphen beide als Endknoten von  $e$  bezeichnet werden.  $u$  und  $v$  sind dann benachbart bzw. adjazent.

In einem *gerichteten Graphen*  $G$  hat jede Kante  $(u, v) \in E$  eine Richtung, also einen Anfangsknoten  $u$  und einen Endknoten  $v$ . Ein *Hypergraph*  $H = (V, E)$  ist ein Graph, bei dem ein Element der Kantenmenge  $E$  eine nicht leere Teilmenge beliebiger Größe von  $V$  ist.

Als  $Adj(v)$  wird bei gerichteten und ungerichteten Graphen die Menge aller zu  $v$  adjazenten Knoten bezeichnet, also  $(v, w) \in E \Leftrightarrow w \in Adj(v)$ .

Für einen gerichteten Graphen bezeichnet  $\delta^+(v) := \{(a, b) \in E : a = v\}$  die Menge aller von einem Knoten  $v$  ausgehenden Kanten und  $\delta^-(v) := \{(a, b) \in E : b = v\}$  alle in  $v$  mündenden Kanten. Ein *vollständiger Graph*  $K_n$  ist ein ungerichteter Graph mit  $n$  Knoten, die alle paarweise verbunden sind.

Ein *Pfad* von einem Knoten  $v_0 \in V$  zu einem Knoten  $v_l \in V$  ist eine Folge von Knoten  $[v_0, v_1, \dots, v_l]$ , wobei immer  $(v_{i-1}, v_i) \in E$  sein muss für  $i \in (1, \dots, l)$ . Die Länge des Pfades ist  $l$ . Sind nicht alle Knoten eines Pfades unterschiedlich, spricht man stattdessen von einem *Zyklus*.

Ein gerichteter Graph, der keine Zyklen enthält, heißt *azyklisch*. Ein *Kreis* der Länge  $l$  ist ein Pfad der Länge  $l$  mit  $v_0 = v_l$ . Eine Kante  $e \in E$  wird als *Sehne* bezeichnet, falls für einen Kreis  $[v_0, v_1, \dots, v_{l-1}, v_0]$  gilt  $e = (v_i, v_j)$ , wobei  $|l - j| > 1$ . Ein *Dreieck* ist ein Kreis  $C_3$  der Länge drei bzw. ein vollständiger Untergraph  $K_3$  mit drei Knoten.

Das Komplement eines Graphen  $G$  ist definiert als Graph  $\bar{G} = (V, \bar{E})$  mit  $\bar{E} = \{(x, y) \in V \times V \mid x \neq y \text{ und } (x, y) \notin E\}$ . Ein *Wald* ist ein Graph, der keinen Kreis enthält. Die *Arborizität*  $a(G)$  eines Graphen  $G$  ist die kleinste Anzahl von Wäldern, in die sich ein Graph zerlegen lässt. Ein *planarer Graph* (auch: plättbarer Graph) ist ein Graph, der sich kreuzungsfrei in einer Ebene zeichnen lässt. Formal sind diese isomorph zu einem ebenen Graphen. Ein ebener Graph  $G = (V, E)$  mit  $V \subset \mathbb{R}^2$  ist ein Graph, dessen Kanten Polygonzügen zwischen Knoten entsprechen, alle verschiedene Endpunkte haben und deren Inneres nie Knoten oder einen Punkt anderer Kanten enthält. Vergleiche hierzu auch Diestel [S. 94 f. 20].

Eine *Knotenfärbung* des Graphen  $G$  ist eine Abbildung  $f : V \rightarrow \mathbb{N}$  mit  $f(u) \neq f(v) \quad \forall (u, v) \in E$ . Die natürliche Zahl, auf die Knoten abgebildet werden, interpretiert man als Farbklasse; zwei adjazente Knoten müssen also immer in einer unterschiedlichen Farbklasse sein.

Grundsätzlich sucht man eine minimale Färbung eines Graphen. Ein Graph  $G$  heißt *k-färbbar*, wenn eine Färbung mit maximal  $k$  Farben existiert, also  $f(u) \leq k \quad \forall k \in V$ . Die *chromatische Zahl*  $\chi(G)$  eines Graphen  $G$  ist definiert als das kleinste  $k$  für das eine  $k$ -Färbung von  $G$  existiert. Die Berechnung der chromatischen Zahl ist im Allgemeinen ein  $\mathcal{NP}$ -vollständiges Problem (vgl. Krumke und Noltemeier [60]). Für jede zulässige Färbung eines Graphen  $G$  bilden die Farbklassen eine stabile Menge von  $G$ . Für eine *stabile Menge*  $S$  in  $G$  gilt  $(u, v) \notin E \quad \forall u, v \in S$ . Für jede Kante  $e \in E$  darf also maximal einer der Endknoten von  $e$  in  $S$  liegen.

Eine *Clique*  $U$  von  $G$  ist eine Teilmenge von  $G$ , so dass für alle Knoten  $u, v \in U$  gilt  $(u, v) \in E$ . Es müssen also alle Knoten in  $U$  paarweise miteinander verbunden sein. Eine *maximale Clique* ist eine Clique, die keine echte Teilmenge einer Clique mit größerer Kardinalität ist. Man bezeichnet mit der *Cliquenzahl*  $\omega(G)$  die Kardinalität der größten Clique in  $G$ . Da alle Knoten in einer Clique adjazent sind und bei einer Färbung einer anderen Farbklasse zugeordnet werden müssen, gilt immer  $\chi(G) \geq \omega(G)$ .

Mit *Cliquenüberdeckung* der Größe  $k$  wird eine Zerlegung eines Graphen  $G$  in  $k$  paarweise disjunkte Cliques bezeichnet.

Mit der *Stabilitätszahl*  $\alpha(G)$  bezeichnet man die Kardinalität einer größten stabilen Menge in  $G$  und mit der *Cliquenüberdeckungszahl*  $\kappa(G)$  die kleinste Anzahl Cliques, die benötigt werden, um die Knoten von  $G$  zu überdecken. Da die Knoten in einer stabilen Menge nicht adjazent sein dürfen, muss jeder Knoten durch eine andere Clique überdeckt werden. Es gilt also immer  $\kappa(G) \geq \alpha(G)$ .

Ein Graph  $G$  heißt  $\chi$ -perfekt, wenn für jeden induzierten Teilgraphen  $G_A$  gilt  $\chi(G_A) = \omega(G_A)$ .

Ein Graph  $G$  heißt  $\alpha$ -perfekt, wenn für jeden induzierten Teilgraphen  $G_A$  gilt  $\alpha(G_A) = \kappa(G_A)$ .

Ein Graph  $G$  ist genau dann  $\alpha$ -perfekt, wenn  $G$   $\chi$ -perfekt ist. *Perfekte Graphen* sind Graphen, die  $\chi$ -perfekt oder  $\alpha$ -perfekt sind.

**Korollar 2.1.** (Vgl. Golumbic [S. 57–35]) *Ein Graph  $G$  ist genau dann perfekt, wenn sein Komplement  $\bar{G}$  perfekt ist.*

## 2.2 Intervallgraphen

Für eine Familie  $F$  von nichtleeren Mengen wird der zugehörige *Schnittgraph* konstruiert, indem man jeder Menge einen Knoten zuweist und zwei Knoten genau dann durch eine Kante verbunden sind, wenn die Schnittmenge der zugehörigen Mengen nicht leer ist. Wenn man den Schnittgraph von Intervallen auf einer total geordneten Menge (also z. B. den reellen Zahlen) bestimmt, nennt man den Graphen *Intervallgraph*.

Ein Intervallgraph vererbt diese Eigenschaft an seine Teilgraphen wie in [S. 14–35] gezeigt worden ist:

**Theorem 2.2.** *Ein induzierter Teilgraph eines Intervallgraphen ist ein Intervallgraph.*

Ein Graph  $G$  ist ein *chordaler Graph*, wenn jeder Kreis der Länge  $l > 3$  eine Sehne besitzt.

**Theorem 2.3.** [S. 172–35] *Jeder Intervallgraph  $G$  ist ein chordaler Graph.*

Die größte Clique in einem vollständigen Graphen  $K_n$  ist der Graph selbst und  $\omega(K_n) = n$ . Vollständige Graphen sind perfekt.

Sind  $G_1$  und  $G_2$  Untergraphen von  $G$  und es gilt  $G = G_1 \cup G_2$  und  $S = G_1 \cap G_2$ , so entsteht  $G$  durch *Zusammenkleben entlang  $S$*  von  $G_1$  und  $G_2$ .

**Proposition 2.4.** (Vgl. Diestel [S. 138–20]) *Ein Graph ist genau dann chordal, wenn er rekursiv durch Zusammenkleben entlang vollständiger Untergraphen konstruiert werden kann, ausgehend von vollständigen Graphen.*

**Theorem 2.5.** [S. 95–35][S. 139–20] *Jeder chordale Graph ist perfekt.*

Auf einem Intervallgraphen können viele Probleme in polynomieller Zeit gelöst werden. So kann in polynomieller Zeit eine minimale Färbung gefunden werden [35][60], aber auch eine minimale Cliquesüberdeckung, wie von Gupta, Lee und Leung in [38] gezeigt worden ist. Ebenso können auch alle maximalen Cliques in polynomieller Zeit gefunden werden [S. 65 60].

Für Intervallgraphen kann in linearer Zeit die chromatische Zahl und die Kardinalität einer maximalen Clique bestimmt werden, wenn ein perfektes Eliminationsschema gegeben ist [S. 65 60].

Ein *perfektes Eliminationsschema* ist eine Folge  $[v_1, v_2, \dots, v_n]$  von Knoten mit der Eigenschaft, dass jeder Knoten  $v_i$  ein simplizialer Knoten des induzierten Teilgraphen  $G_{\{v_i, \dots, v_n\}}$  ist. Das heißt,

$$X_i = \{v_j \in \text{Adj}(v_i) \mid j > i\}$$

ist eine Clique.

**Theorem 2.6.** [S. 83 35] *Jeder chordale Graph besitzt ein perfektes Eliminationsschema.*

Ein perfektes Eliminationsschema lässt sich für chordale Graphen in linearer Zeit bestimmen.

Für Intervallgraphen findet sich also ein perfektes Eliminationsschema, indem alle Intervalle nach ihrem linken Intervallende sortiert werden. Mit Hilfe eines perfekten Eliminationsschemas  $[v_1, v_2, \dots, v_n]$  können Intervallgraphen nun also in linearer Zeit gefärbt werden.

Ein einfach zu implementierender Algorithmus für das Färbungsproblem mit einer Laufzeit von  $O(n \log n)$ , der die oben genannten Ergänzungen für Intervallgraphen nutzt, findet sich in einer Arbeit von Gupta, Lee und Leung [37].

Dieser Algorithmus betrachtet alle aufsteigend sortierten Intervallenden der Reihe nach. Er weist jedes Intervall der kleinsten freien Farbklasse zu und schließt diese dann. Wird das entsprechende rechte Intervallende betrachtet, wird diese Farbklasse wieder frei gegeben.

Durch ein perfektes Eliminationsschema sind alle Cliques eines Intervallgraphen fortlaufend angeordnet, können also auch mit der selben Laufzeit bestimmt werden.



## 2.3 Kreise in Graphen

Gegeben sei ein (ungerichteter) Graph  $G = (V, E)$ . Gesucht sind alle elementaren Zyklen, also Zyklen in denen lediglich Anfangs- und Endknoten gleich sind. Grundsätzlich können alle Algorithmen für ungerichtete und gerichtete Graphen gleichsam benutzt werden [S. 91–64].

Ein Algorithmus, der per unbeschränktem Backtracking alle Kanten des Graphen, ausgehend von einem Startknoten, untersucht und versucht Zyklen zu finden, wird in der Arbeit von Tiernan [82] vorgestellt. Tarjan konnte in [80] die Laufzeit dieses Algorithmus für einen gerichteten Graphen  $G$  auf  $O(V(G) \cdot E(G))(C + 1)$  reduzieren. Dabei ist  $C$  die Anzahl von elementaren Zyklen im Graphen  $G$ . In einer weiteren Arbeit wurde von Johnson in [53] wiederum eine Verbesserung auf eine Laufzeit von  $O(V(G) + E(G))(C + 1)$  erreicht.

## 2.4 Komplexitätstheorie

Die Komplexitätstheorie beschäftigt sich mit der Frage, wie effektiv man ein Problem lösen kann. Zunächst braucht man einen Ansatz, Berechnungen zu formalisieren. Dies geht beispielsweise mit einer (deterministischen) Turingmaschine (DTM). Andere Ansätze haben sich stets als äquivalent zu Turingmaschinen herausgestellt. Jede Turingmaschine besitzt eine Recheneinheit, ein zweiseitig unbeschränktes Band und einen Lese- und Schreibkopf. Eine deterministische Turingmaschine befindet sich stets in einem von endlich vielen Zuständen, wobei sie im Startzustand beginnt und hält, wenn sie einen Endzustand erreicht. Eine nichtdeterministische Turingmaschine kann zu jedem Zustand mehrere Folgezustände besitzen. Sie akzeptiert eine Eingabe, falls mindestens eine Rechnung zu einem akzeptierenden Endzustand führt. Weiterführende Informationen zu Turingmaschinen und den Grundlagen der Theoretischen Informatik finden sich beispielsweise in den Werken von Hopcroft, Motwani und Ullmann [46], Blum [11], Reischuk [69] und Garey u. Johnson [29].

Eine Funktion heißt *(DTM-)berechenbar* wenn es eine DTM  $M$  gibt mit  $f_M = f$ , d. h. ist  $f(x)$  definiert, so schreibt  $M$   $f(x)$  auf das Ausgabeband, sonst hält  $M$  nicht (vgl. auch [S. 12–29]).

Spezielle Entscheidungsprobleme können mit „ja“ oder „nein“ beantwortet werden. Formal ist also ein  $x \in \{0, 1\}^*$  gegeben und es ist zu entscheiden,

ob  $x \in f^{-1}(1)$  ist.  $L = f^{-1}(1)$  sind Wörter einer Sprache, es soll also das *Spracherkennungsproblem* mittels einer DTM gelöst werden:

gegeben  $L \subset \{0, 1\}^*$  und  $x \in \{0, 1\}^*$ , ist  $x \in L$ ?

Eine Sprache  $L$  heißt *entscheidbar*, falls die charakteristische Funktion von  $L$

$$\chi_L : \{0, 1\}^* \rightarrow \{0, 1\}, \chi_L(w) = \begin{cases} 1 & w \in L \\ 0 & w \notin L \end{cases}$$

berechenbar ist (vgl. [S. 355 83]).  $M$  akzeptiert die Eingabe also wenn  $x \in L$  ist und verwirft sie, wenn  $x \notin L$ .

Eine Sprache  $L$  heißt *semi-entscheidbar*, falls die „halbe“ charakteristische Funktion von  $L$

$$\chi_L : \{0, 1\}^* \rightarrow \{0, 1\}, \chi_L(w) = \begin{cases} 1 & w \in L \\ \text{undef} & w \notin L \end{cases}$$

berechenbar ist (vgl. [S. 355 83]).  $M$  akzeptiert die Eingabe also wenn  $x \in L$  ist und verwirft sie, oder gerät in eine Endlosschleife, wenn  $x \notin L$ .

**Lemma 2.7.** *Eine Sprache ist genau dann semi-entscheidbar, wenn sie von einer DTM akzeptiert wird.*

$\mathcal{P}$  ist die Menge aller Sprachen, die von einer deterministischen Turingmaschine entschieden werden, deren Laufzeit durch ein Polynom in der Länge des Inputs beschränkt ist.

$\mathcal{NP}$  ist die Menge aller Sprachen, die von einer nichtdeterministischen Turingmaschine akzeptiert werden, deren Laufzeit durch ein Polynom in der Länge des Inputs beschränkt ist.

Für Probleme in  $\mathcal{NP}$  existiert zu einer Lösung ein polynomieller Nachweis, der die Lösung überprüft.

Es gilt  $\mathcal{P} \subset \mathcal{NP}$ . Ob allerdings  $\mathcal{P} = \mathcal{NP}$  gilt oder nicht, ist derzeit noch unklar.

Sind  $L$  und  $L'$  zwei Sprachen, so heißt  $L'$  *polynomiell reduzierbar* auf  $L$  ( $L' \preceq_p L$ ), falls eine in polynomieller Zeit berechenbare Funktion  $f$  existiert, so dass  $x \in L' \Leftrightarrow f(x) \in L$  ist.

**Lemma 2.8.** [S. 395 83] *Sei  $L' \preceq_p L$ . Ist  $L' \in \mathcal{P}$  so folgt, dass auch  $L \in \mathcal{P}$ . Ist  $L' \in \mathcal{NP}$  so folgt, dass auch  $L \in \mathcal{NP}$ .*

Eine Sprache  $L$  heißt  $\mathcal{NP}$ -schwer, falls für alle  $L' \in \mathcal{NP}$  gilt, dass  $L' \preceq_p L$ .

Eine Sprache  $L$  heißt  $\mathcal{NP}$ -vollständig, falls  $L$   $\mathcal{NP}$ -schwer ist und  $L \in \mathcal{NP}$ .

**Lemma 2.9.** [S. 395 83] Sei  $L$   $\mathcal{NP}$ -vollständig. Dann ist auch  $L'$   $\mathcal{NP}$ -vollständig, wenn  $L' \in \mathcal{NP}$  und  $L \preceq_p L'$ .

## 2.5 Lineare Programmierung

Informationen zu mathematischen Optimierungsproblemen und (ganzzahligen) linearen Programmen finden sich unter anderem in den Werken von Chvátal [15], Schrijver [73], Faigle, Kern und Still [24] und Zions [88].

Ein *lineares Programm* ( $LP$ ) ist eine Instanz eines linearen Optimierungsproblems. Gesucht ist dabei ein Maximum (oder Minimum) einer linearen Funktion unter Beachtung einer endlichen Anzahl von linearen Nebenbedingungen für das Definitionsgebiet dieser Funktion. Eine Darstellung für  $i$  Nebenbedingungen und  $j$  Variablen ist beispielsweise

$$\begin{aligned} \max \quad & \sum_{j=1}^n c_j x_j \\ \text{s.d.} \quad & \sum_{j=1}^n a_{ij} x_j \leq b_i \quad (i = 1, 2, \dots, m) \\ & x_j \geq 0 \quad (j = 1, 2, \dots, n) \end{aligned}$$

Oder in Matrixschreibweise mit  $x, c \in \mathbb{R}^n$ ,  $b \in \mathbb{R}^m$ ,  $A \in \mathbb{R}^{m \times n}$ :

$$\begin{aligned} \max \quad & c^T x \\ \text{s.d.} \quad & Ax \leq b \end{aligned}$$

Ist  $x \in \mathbb{Z}^n$ , so handelt es sich um ein *ganzzahliges lineares Programm* ( $IP$ ).

Erweitert man den Definitionsbereich der Lösung eines IP auf reelle Werte, entsteht eine *LP-Relaxierung* zu diesem IP. Diese liefert eine obere Schranke des IP [S. 230 73].

Eine Teilmenge  $P \subset \mathbb{K}^n$  heißt *Polyeder*, falls es ein  $m \in \mathbb{Z}^+$ , eine Matrix  $A \in \mathbb{K}^{m \times n}$  und einen Vektor  $b \in \mathbb{K}^m$  gibt mit

$$P = P(A, b) = \{x \in \mathbb{K}^n \mid Ax \leq b\}$$

Für die *Menge der zulässigen Lösungen*  $P(A, b)$  gilt immer eine der drei folgenden Eigenschaften:

1.  $P(A, b) = \emptyset$ , das LP hat also keine Lösung.
2. Die Zielfunktion  $c^T x$  ist nach oben unbeschränkt auf  $P(A, b)$ .
3. Die Zielfunktion ist nach oben beschränkt, eine zulässige Lösung mit maximalem Zielfunktionswert heißt dann *optimale Lösung*.

Um ganzzahlige lineare Programme zu lösen, gibt es verschiedene Ansätze. Das Schnittebenenverfahren wird beispielsweise in [24] behandelt, Branch-and-Bound in [88]. Weder diese beiden Verfahren noch andere bekannte Lösungsansätze haben jedoch polynomielle Laufzeit, denn es handelt sich um ein  $\mathcal{NP}$ -vollständiges Problem [73].

Lineare Programme sind allerdings in polynomieller Zeit lösbar, als Beispiel für einen Algorithmus mit polynomieller Laufzeit sei das Innere-Punkt-Verfahren genannt [24].

In der Praxis benutzt man oft das Simplexverfahren. Dieses hat exponentielle Laufzeit, ist dennoch oft sehr effizient [S. 129–73]. Eine lineare Ungleichung ist ein Halbraum im  $\mathbb{R}^n$ , also ist  $P(A, b)$  der Durchschnitt von endlich vielen Halbräumen. Die lineare Zielfunktion ist eine Hyperebene des  $\mathbb{R}^{n+1}$ , ihr Maximum muss also in einer Ecke des Polyeders angenommen werden. Das Simplexverfahren betrachtet nun, ausgehend von einer Startecke, die Ecken des Polyeders und ermittelt so eine Lösung des LP. Eine Darstellung dieses Verfahrens findet sich unter anderem in [73] oder [15].

## 2.6 Das Mengenüberdeckungsproblem

Viele Optimierungsprobleme, insbesondere auf Graphen, können durch das Mengenüberdeckungsproblem dargestellt werden.

**Definition 2.10** (Mengenüberdeckungsproblem MÜP). Gegeben sei eine endliche, nichtleere Menge  $S$  mit einer Familie  $F = \{S_1, \dots, S_n\}$  von Teilmengen von  $S$ , also  $S_j \subset S$  für  $i = 1, \dots, n$ . Eine Teilmenge  $G \subset F$  ist eine Überdeckung von  $S$  genau dann, wenn

$$\bigcup_{S_i \in G} S_i = S$$

ist.

**Optimierungsvariante:** Gesucht ist eine minimale Menge  $G \subset F$  die  $S$  überdeckt.

**Entscheidungsvariante:** Gegeben ein  $k > 0$ : Existiert eine Menge  $G \subset F$  die  $S$  überdeckt und nicht mehr als  $k$  Elemente beinhaltet?

Im Folgenden beschreibe MÜP immer das Optimierungsproblem, E-MÜP das Entscheidungsproblem. Diese Notation wird im folgenden auch auf weitere Probleme angewandt.

Das Optimierungsproblem lässt sich auch als ganzzahliges lineares Programm darstellen:

$$\begin{array}{ll}
 \text{[MÜP-IP]} & \min \quad \sum_{i=1}^n x_i \\
 & \text{s.d.} \quad \sum_{i: s \in S_i} x_i \geq 1 \quad \forall s \in S \quad (\text{MÜP-IP.1}) \\
 & \quad \quad \quad x_i \geq 1 \quad \forall i = 1, \dots, n \quad (\text{MÜP-IP.2}) \\
 & \quad \quad \quad x_i \in \mathbb{Z} \quad \forall i = 1, \dots, n
 \end{array}$$

Die Bedingung MÜP-IP.1 stellt dabei sicher, dass jedes Element mindestens einmal überdeckt wird.

Das Mengenüberdeckungsproblem ist in seiner Entscheidungsvariante  $\mathcal{NP}$ -vollständig. Ein Beweis findet sich im Werk von Garey und Johnson [30]; auch ist dieses Problem Bestandteil von Karp's Liste von 21  $\mathcal{NP}$ -vollständigen Problemen [S. 94-56].

Jeder Teilmenge von  $F$  kann man auch ein Gewicht zuordnen. Dies führt direkt zum gewichteten Mengenüberdeckungsproblem:

**Definition 2.11** (Gewichtetes Mengenüberdeckungsproblem GMÜP). Gegeben sei eine Menge  $S$  mit einer Familie  $F = \{S_1, \dots, S_n\}$  von Teilmengen von  $S$ , also  $S_j \subset S$  für  $i = 1, \dots, n$ . Jedem Element in  $F$  ist ein nicht negatives Gewicht  $w(S_i)$  zugeordnet, d. h.

$$w : F \rightarrow \mathbb{R}^+$$

**Optimierungsvariante:** Gesucht ist eine minimal gewichtete Menge  $G \subset F$  die  $S$  überdeckt.

**Entscheidungsvariante:** Gegeben ein  $k > 0$ : Existiert eine Menge  $G \subset F$  die  $S$  überdeckt und deren Gewicht nicht größer als  $k$  ist?

Auch dieses Problem lässt sich als ganzzahliges lineares Programm darstellen:

$$\begin{array}{ll}
 \text{[GMÜP-IP]} & \min \quad \sum_{i=1}^n x_i w(S_i) \\
 & \text{s.d.} \quad \sum_{i:s \in S_i} x_i \geq 1 \quad \forall s \in S \quad (\text{GMÜP-IP.1}) \\
 & \quad \quad \quad x_i \geq 1 \quad \forall i = 1, \dots, n \quad (\text{GMÜP-IP.2}) \\
 & \quad \quad \quad x_i \in \mathbb{Z} \quad \forall i = 1, \dots, n
 \end{array}$$

Die Bedingung GMÜP-IP.1 stellt dabei sicher, dass jedes Element mindestens einmal überdeckt wird.

Auch das gewichtete Mengenüberdeckungsproblem ist  $\mathcal{NP}$ -vollständig.

# Strukturen und Eigenschaften Pseudostabiler Mengen

Die Zerlegung eines Graphen in stabile Mengen erlaubt nur Kanten zwischen Knoten in verschiedenen stabilen Mengen. Will man einen Sachverhalt modellieren, der bestimmte Abhängigkeiten in Form von Pfaden zwischen diesen stabilen Mengen erlaubt, muss man einen anderen Ansatz wählen.

Dieses Kapitel diskutiert zunächst die verschiedenen Modelle, da die Kandidaten für die Pfade zwischen stabilen Mengen definiert werden müssen. Dann folgt die Darstellung des Problems als Mengenüberdeckungsproblem. Dazu werden pseudostabile Tupel definiert.

Im Anschluss werden die Strukturen pseudostabiler Mengen untersucht und ein weiterer Abschnitt widmet sich dem Zusammenhang zwischen stabilen und pseudostabilen Mengen.

## 3.1 Definition

Im folgenden werden nun pseudostabile Mengen betrachtet. Diese erlauben zusätzlich Pfade zwischen stabilen Mengen. Ihnen wird ein Gewicht zugeordnet, so dass man auch hier eine minimale Zerlegung suchen kann.

Die Kandidaten für zusätzlich erlaubte Kanten und Knoten außerhalb von stabilen Mengen können entweder in einer separaten Liste gespeichert werden oder als Teilgraph definiert werden:

**Definition 3.1.** Ein Graph  $G = (V, E)$  mit einem Teilgraphen  $G' = B \subset G$  von blauen Knoten und Kanten wird *Blauer Graph* genannt und mit  $G^B$  bezeichnet.

Dieser blaue Teilgraph stellt die Kandidaten für die Pfade und Knoten außerhalb der stabilen Mengen. Das bedeutet, dass nur solche Pfade außerhalb der stabilen Mengen liegen dürfen, die in  $B$  liegen. Sowohl  $G'$  als auch  $B$  bezeichnen also den selben Teilgraphen. Die Notation soll stets so gewählt werden, dass sie der Übersichtlichkeit dient. Auch kann die Notation  $G^B$  zu  $G$  vereinfacht werden, wenn aus dem Zusammenhang klar ist, dass es sich um einen blauen Graphen handelt.

Es soll nun zunächst eine Formulierung pseudostabiler Mengen als Mengenüberdeckungsproblem gegeben werden, anschließend eine Formulierung durch die Zerlegung eines Graphen. Beide Formulierungen sind äquivalent, jedoch sind nicht alle Unterprobleme gleich intuitiv darstellbar.

### 3.1.1 Darstellung als Mengenüberdeckungsproblem

Wir definieren zunächst zwei verschiedene Teilmengen des Graphen  $G^B$  um daraus eine Überdeckung zu erstellen.

**Definition 3.2** (Pseudostabiles Tupel).  $T \subset G$  ist ein *pseudostabiles Tupel*, wenn es aus zwei stabilen Menge  $D_1$  und  $D_2$  sowie einem Pfad  $p$  besteht, so dass

$$T = D_1 \cup p \cup D_2$$

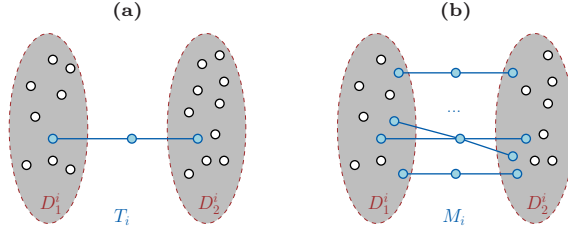
Dabei besteht die Schnittmenge zwischen  $D_1$  und  $p$  sowie  $p$  und  $D_2$  jeweils nur aus einem Knoten. Für  $p$  gilt, dass es nur aus drei blauen Knoten besteht, die durch blaue Kanten in Form eines Pfades verbunden sind. Das heißt, dass  $p \subset B(G)$ ,  $|V(p)| = 3$  und  $p$  ist zusammenhängend und kreisfrei. Alternativ kann  $T$  auch stabil sein, d. h.  $D_1$  ist stabil und  $p = D_2 = \emptyset$ . Dann ist die Wertigkeit  $\zeta(T) = 1$ , sonst gilt  $\zeta(T) = 2$ .

Man kann nun aber auch mehr als nur einen Pfad zwischen den stabilen Mengen  $D_1$  und  $D_2$  zulassen. Dazu definieren wir mehrfach pseudostabile Tupel, vgl. auch Abb. 3.1.

**Definition 3.3** (Mehrfach pseudostabiles Tupel).  $M \subset G$  ist ein *mehrfach pseudostabiles Tupel*, wenn es aus zwei stabilen Menge  $D_1$  und  $D_2$  sowie Pfaden  $p_1, \dots, p_i$  besteht, so dass

$$M = D_1 \cup p_1 \cup \dots \cup p_i \cup D_2$$





**Abbildung 3.1:** Darstellung eines pseudostabilen Tupel  $T_i$  in (a) und eines mehrfach pseudostabilen Tupels  $M_i$  in (b). Die beiden Mengen  $D_1$  und  $D_2$  sind jeweils stabil und es existieren einige Pfade von blauen Knoten und Kanten der Länge 3. Die Mengen  $\mathfrak{P}(T_i)$  und  $\mathfrak{P}(M_i)$  bestehen jeweils aus allen blauen Knoten, die weder in  $D_1$  noch in  $D_2$  liegen. Es ist nicht gefordert, dass die Pfade disjunkt sind.

Dabei besteht die Schnittmenge zwischen  $D_1$  und  $p_j$  sowie  $p_j$  und  $D_2$  für alle  $j \in \{1, \dots, i\}$  jeweils nur aus einem Knoten. Für  $p_j$  gilt, dass es nur aus drei blauen Knoten besteht, die durch blaue Kanten in Form eines Pfades verbunden sind. Das heißt, dass  $p_j \subset B(G)$ ,  $|V(p_j)| = 3$  und  $p_j$  ist zusammenhängend und kreisfrei. Alternativ kann  $M$  auch stabil sein, d. h.  $D_1$  ist stabil und  $D_2 = \emptyset$  sowie  $p_j = \emptyset$  für alle  $j \in \{1, \dots, i\}$ . Dann ist die Wertigkeit  $\zeta(T) = 1$ , sonst gilt  $\zeta(T) = 2$ .

Jedes pseudostabile Tupel ist also offensichtlich auch ein mehrfach pseudostabiles Tupel mit  $i = 1$  Pfaden.

Es sei  $Pf(T)$  oder  $Pf(M)$  die Menge aller Pfade zwischen den stabilen Mengen in  $T$  bzw.  $M$ .  $\widetilde{Pf}$  wiederum besteht nur aus der Menge der inneren Knoten, die nicht in einer stabilen Menge sind. Das heißt, dass  $\widetilde{Pf}(T) = Pf(T) \setminus (D_1 \cup D_2)$  bzw. genauso für  $M$ . Die Definition für Vereinigungen von Mengen  $T$  oder  $M$  erfolgt für  $Pf$  und  $\widetilde{Pf}$  analog:

$$Pf(T_1 \cup \dots \cup T_n) = Pf(T_1) \cup \dots \cup Pf(T_n)$$

Man sucht im Allgemeinen nach einer minimalen Knotenüberdeckung  $S = \{S_1, \dots, S_n\}$  des Graphen  $G$ , wobei die Elemente  $S_i$  entweder ausschließlich pseudostabile oder mehrfach pseudostabile Tupel sind. Das heißt, dass  $S = \{T_1, \dots, T_n\}$  oder  $S = \{M_1, \dots, M_n\}$ . Dabei sei das Gewicht  $w$  dieser Überdeckung definiert als:

$$w(S) = \sum_{i=1}^n \zeta(S_i) + \sum_{i=1}^n \sum_{j \in \{1, \dots, n\} \setminus \{i\}} w_{i,j} \quad (3.1)$$

wobei

$$w_{i,j} = \begin{cases} -1 & S_i \cap S_j = D_1^i = D_2^j \\ 0 & \text{sonst} \end{cases} \quad (3.2)$$

Die erste Bedingung stellt dabei sicher, dass zwei identische stabile Mengen in zwei verschiedenen Tupeln nicht doppelt gewichtet werden. Alle anderen Fälle sind irrelevant. Diese Gewichtsfunktion gilt für pseudostabile Tupel genauso, wie für mehrfach pseudostabile Tupel. Damit kann nun also eine Optimierungsvariante definiert werden, in der eine minimal gewichtete Überdeckung eines Graphen gesucht wird.

Zunächst soll die Optimierungsvariante für pseudostabile Tupel diskutiert werden. Zu einem gegebenen Graphen  $G = (V, E)$  mit einer Menge von blauen Knoten und Kanten  $B \subset G$  sei

$$\mathfrak{T} = \{T_1, \dots, T_n\}$$

die Menge aller pseudostabilen Teilmengen in  $G$  mit  $B$ . Mit

$$\mathfrak{P}(T_i) = T_i \setminus \{D_1^i \cup D_2^i\}$$

wiederum sei die Menge aller inneren Knoten von Pfaden innerhalb von  $T_i$  bezeichnet.  $\mathfrak{P}(S)$  mit  $S = \{S_1, \dots, S_n\}$  ist definiert als  $\mathfrak{P}(S) = \mathfrak{P}(S_1) \cup \dots \cup \mathfrak{P}(S_n)$ . Nun kann das Optimierungsproblem wie folgt definiert werden:

$$\begin{aligned} \min \quad & \sum_{i=1}^n t_i \zeta(T_i) + \sum_{i=1}^n t_i \sum_{j=1}^n t_j w_{i,j} \\ \text{s.d.} \quad & \sum_{T \in \mathfrak{T}: v \in Pf(T)} t_i = 1, \forall v \in V \\ & \sum_{T \in \mathfrak{T}: v \in T} t_i \geq 1, \forall v \in V \\ & t_i \in \{0, 1\} \end{aligned} \quad (\text{IP1})$$

Dabei zeigt die Variable  $t_i$  an, ob die Menge  $T_i$  Teil der gewählten Überdeckung ist. Die erste Zeile bezieht sich auf die Gewichtsfunktion aus Gleichung (3.1). Diese soll minimiert werden. Die nächste Zeile stellt sicher, dass jeder Knoten  $v \in V$  nur einmal in einem Pfad eines pseudostabilen Tupels sein darf. Die letzte Zeile stellt sicher, dass jeder Knoten  $v \in V$  von

mindestens einem Tupel überdeckt wird. Das heißt, dass jeder Knoten, sofern er nicht innerer Knoten eines Pfades ist, von beliebig vielen Mengen überdeckt werden darf.

Soll diese Bedingung vernachlässigt werden, also ein innerer Knoten eines Pfades  $p$  auch Teil eines anderen Pfades sein darf, kann das Optimierungsproblem wie folgt definiert werden:

$$\begin{aligned} \min \quad & \sum_{i=1}^n t_i \zeta(T_i) + \sum_{i=1}^n t_i \sum_{j=1}^n t_j w_{i,j} \\ \text{s.d.} \quad & \sum_{T \in \mathfrak{T}: v \in T} t_i \geq 1, \forall v \in V \\ & t_i \in \{0, 1\} \end{aligned} \tag{IP2}$$

Sowohl Formulierung IP1 als auch IP2 können sowohl für pseudostabile Tupel  $T$  als auch für mehrfach pseudostabile Tupel  $M$  angewendet werden. Dabei wird lediglich statt  $\mathfrak{T}$  die Menge  $\mathfrak{M} = \{M_1, \dots, M_n\}$  definiert und verwendet.

**Bemerkung 3.4.** In den Definitionen 3.2 und 3.3 wurde nicht gefordert, dass die Pfade innerhalb eines (mehrfach) pseudostabilen Tupels disjunkt sein müssen. In der Formulierung IP2 können also alle inneren Knoten von Pfaden mehrfach Teil von Pfaden zwischen verschiedenen stabilen Mengen sein.

Als weiteres wichtiges Werkzeug muss noch der *Graph der Überdeckung* definiert werden.

Eine Überdeckung eines Graphen  $G = (V, E)$  mit einem Teilgraphen  $G' = B \subset G$  von blauen Knoten und Kante mit pseudostabilen Tupeln  $T$  oder mehrfach pseudostabilen Tupeln  $M$  induziert den Graphen dieser Überdeckung. Dabei entspricht jede stabile Menge  $D$  einem Knoten in diesem Graphen, und alle Pfade  $p$  entsprechen Kanten.

**Definition 3.5** (Graph einer Überdeckung). Gegeben sei eine Überdeckung  $S = \{S_1, \dots, S_n\}$  eines Graphen  $G = (V, E)$  mit einem Teilgraphen  $G' = B \subset G$  von blauen Knoten und Kante mit pseudostabilen Tupeln  $T$  oder mehrfach pseudostabilen Tupeln  $M$ . Dann ist  $G_s = (V, E)$  der gerichtete Graph dieser Überdeckung mit

$$\begin{aligned} V &= \{D \subset S_1, \dots, S_n\} \\ E &= \{(D_1^i, D_2^i), \text{ für } i \in \{1, \dots, n\} \text{ wenn } D_2^i \neq \emptyset\} \end{aligned}$$

Nun können verschiedene relevante Optimierungsprobleme wie folgt definiert werden:

**Definition 3.6** (minPS). Gesucht ist eine minimale Überdeckung  $S$  des Graphen  $G = (V, E)$  mit einem Teilgraphen  $B \subset G$  von blauen Knoten und Kante mit pseudostabilen Tupeln  $T$  entsprechend der Gleichung IP1 wobei für den Graph  $G_S$  der Überdeckung gilt, dass er kreisfrei ist und für alle  $v \in V(G_S)$  gilt, dass  $\delta(v) \in \{0, 1, 2\}$ .

**Definition 3.7** (minMPS). Gesucht ist eine minimale Überdeckung  $S$  des Graphen  $G = (V, E)$  mit einem Teilgraphen  $B \subset G$  von blauen Knoten und Kante mit mehrfach pseudostabilen Tupeln  $M$  entsprechend der Gleichung IP1 wobei für den Graph  $G_S$  der Überdeckung gilt, dass er kreisfrei ist und für alle  $v \in V(G_S)$  gilt, dass  $\delta(v) \in \{0, 1, 2\}$ .

minPS' und minMPS' bezeichnen das jeweilige Minimierungsproblem entsprechend der Gleichung IP2. Hier dürfen also innere Knoten von Pfaden Teil mehrerer Pfade sein. minPS-a und minMPS-a wiederum bezeichnen das jeweilige Minimierungsproblem ohne Einschränkungen auf dem Graphen  $G_S$ . Das heißt

**Definition 3.8** (minPS-a). Gesucht ist eine minimale Überdeckung  $S$  des Graphen  $G = (V, E)$  mit einem Teilgraphen  $B \subset G$  von blauen Knoten und Kanten mit pseudostabilen Tupeln  $T$  entsprechend der Gleichung IP1.

**Definition 3.9** (minMPS-a). Gesucht ist eine minimale Überdeckung  $S$  des Graphen  $G = (V, E)$  mit einem Teilgraphen  $B \subset G$  von blauen Knoten und Kanten mit mehrfach pseudostabilen Tupeln  $M$  entsprechend der Gleichung IP1.

Siehe auch Tabelle 3.1 für eine übersichtliche Darstellung. Diese Definition als Mengenüberdeckungsproblem wird immer wieder aufgegriffen werden. Insbesondere, um zu zeigen, dass minPS und minMPS  $\mathcal{NP}$ -vollständig sind.

Im Folgenden soll das Problem durch die Zerlegung eines Graphen definiert werden.

### 3.1.2 Darstellung durch Zerlegung des Graphen

Gegeben sei ein Graph  $G = (V, E)$  mit einem Teilgraphen  $G' = B \subset G$  von blauen Knoten und Kante. Diese blauen Knoten und Kanten sollen nun im Folgenden die Kandidaten für die Pfade werden.

Variante	Tupel	Gleichung	$G_S$	Knoten in $\mathfrak{P}(S)$
minPS	$T$	IP1	nur kreisfreie Pfade	eindeutig
minMPS	$M$	IP1	nur kreisfreie Pfade	eindeutig
minPS-a	$T$	IP1	-	eindeutig
minMPS-a	$M$	IP1	-	eindeutig
minPS'	$T$	IP2	nur kreisfreie Pfade	-
minMPS'	$M$	IP2	nur kreisfreie Pfade	-
minPS'-a	$T$	IP2	-	-
minMPS'-a	$M$	IP2	-	-

**Tabelle 3.1:** Die verschiedenen Optimierungsvarianten des Überdeckungsproblem mit pseudostabilen Mengen.

Es sei  $P_i \subset G$  eine Teilmenge von  $G$  mit  $i \in \mathbb{N}^+$ . Der Index  $i$  wird zur eindeutigen Identifizierung der Mengen benötigt.

**Definition 3.10.**  $P_i$  ist eine *pseudostabile Menge*, wenn  $P_i = D_i$  selbst stabil ist oder wenn sie in eine Aneinanderreihung von stabilen Mengen  $D_j^i$  mit

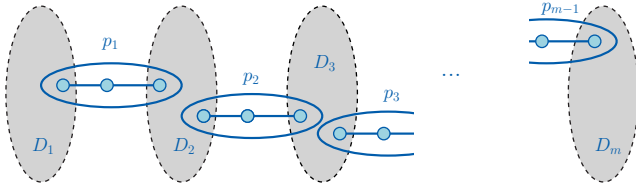
$$D_1^i, p_1^i, D_2^i, p_2^i, D_3^i, \dots, p_{m_i-1}^i, D_{m_i}^i$$

und  $m_i \geq 2$  zerfällt. Dabei besteht die Schnittmenge entweder zwischen stabilen Mengen  $D_j$  und Mengen  $p_{j+1}$  oder zwischen  $p_j$  und  $D_{j+1}$  jeweils nur aus einem Knoten. Für die paarweise disjunkten Mengen  $p_j$  gilt, dass sie nur aus drei blauen Knoten bestehen, die durch blaue Kanten in Form eines Pfades verbunden sind. Das heißt, dass  $p_j \subset B(G)$ ,  $|V(p_j)| = 3$  und  $p_j$  ist zusammenhängend und kreisfrei. Die Wertigkeit dieser Menge  $P_i$  ist  $m_i$ .

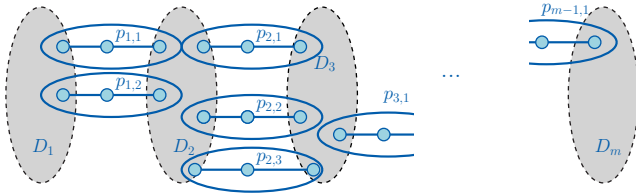
**Definition 3.11.**  $P_i$  ist eine *mehrfach pseudostabile Menge*, wenn  $P_i = D_i$  selbst stabil ist oder wenn sie in eine Aneinanderreihung von stabilen Mengen  $D_j^i$  mit

$$D_1^i, p_{1,1}^i, \dots, p_{1,n_1}^i, D_2^i, p_{2,1}^i, \dots, p_{2,n_2}^i, D_3^i, \dots, p_{m_i-1,1}^i, \dots, p_{m_i-1,n_{m_i-1}}^i, D_{m_i}^i$$

und  $m_i \geq 2$  zerfällt. Dabei besteht die Schnittmenge zwischen aufeinander folgenden stabilen Mengen  $D_j$  und Mengen  $p_{j+1,n}$  mit  $n \in \{1, \dots, n_{j+1}\}$  und  $p_{j,n}$  mit  $n \in \{1, \dots, n_j\}$  und  $D_{j+1}$  jeweils nur aus einem Knoten. Für die paarweise disjunkten Mengen  $p_j$  gilt, dass sie nur aus drei blauen Knoten



**Abbildung 3.2:** Skizze einer beispielhaften Zerlegung  $D_1, p_1, D_2, p_2, \dots, p_{m-1}, D_m$  in pseudostabile Mengen.



**Abbildung 3.3:** Skizze einer beispielhaften Zerlegung  $D_1, p_{1,1}, p_{1,2}, D_2, p_{2,1}, p_{2,2}, p_{2,3}, \dots, p_{m-1}, D_m$  in mehrfach pseudostabile Mengen.

bestehen, die durch blaue Kanten in Form eines Pfades verbunden sind. Das heißt, dass  $p_j \subset B(G)$ ,  $|V(p_j)| = 3$  und  $p_j$  ist zusammenhängend und kreisfrei. Die Wertigkeit dieser Menge  $P_i$  ist  $m_i$ .

Eine Darstellung der Aneinanderreihung von stabilen Mengen in pseudo-stabilen Mengen findet sich in Abb. 3.2, eine Darstellung der Aneinanderreihung von stabilen Mengen in einer mehrfach pseudostabilen Menge in Abb. 3.3.

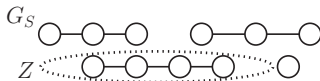
**Bemerkung 3.12.** Es ist ohne weiteres möglich, die Pfade  $p_i$  auch als Teilmenge von Knoten  $(u, v, w) \subset G$  anzugeben. Dadurch wird die Problemstellung nicht eingeschränkt.

Beide Definitionen sind äquivalent. Dass sich diese Äquivalenz immer nach dem selben Schema zeigen lässt, soll exemplarisch durch die folgenden zwei Fälle gezeigt werden:

**Lemma 3.13.** *Jede minimale Überdeckung  $S$  eines Graphen  $G = (V, E)$  mit einem Teilgraphen  $B \subset G$  von blauen Knoten und Kanten mit pseudo-stabilen Tupeln gemäß Definition 3.6 ist äquivalent zu einer minimalen Zerlegung des Graphen  $G$  in pseudostabile Mengen gemäß Definition 3.10.*

*Beweis.* „ $\Rightarrow$ “ Gegeben sei eine minimale Überdeckung  $S$  des Graphen  $G = (V, E)$  mit einem Teilgraphen  $B \subset G$  von blauen Knoten und Kanten mit pseudostabilen Tupeln  $T$  entsprechend der Gleichung IP1 wobei für den Graphen  $G_S$  der Überdeckung gilt, dass er kreisfrei ist und für alle  $v \in V(G_S)$  gilt, dass  $\delta(v) \in \{0, 1, 2\}$ .

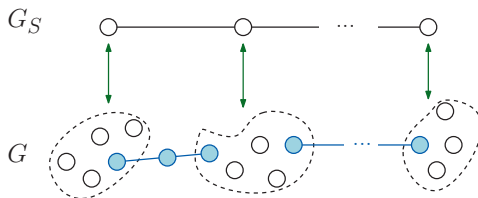
Da der Graph  $G_S$  kreisfrei ist kann jede Zusammenhangskomponente  $Z \subset G_S$  einzeln betrachtet werden, vgl. Abb. 3.4.



**Abbildung 3.4:** Beispielhafte Darstellung eines  $G_S$  und einer Zusammenhangskomponente  $Z$  (gestrichelt).

Eine Zusammenhangskomponente  $Z$  hat entweder nur einen Knoten, entspricht also einer stabilen Menge  $D^i$ , oder besteht aus mindestens zwei Knoten  $v_1$  und  $v_j$  mit  $\delta(v) = 1$ .

Im ersten Fall kann eine stabile Menge  $D^{i_1}$  erzeugt werden, im zweiten Fall können kontinuierlich alle stabilen Mengen der Reihe nach von  $v_1$  bis  $v_j$  betrachtet werden und analog stabile Mengen der Form  $(D_1^i, D_2^i)$   $i \in \{1, \dots, n\}$  erzeugt werden, wenn  $D_2^i \neq \emptyset$ . Das heißt es ergibt sich die Form  $D_1^1, D_2^1, D_2^2, \dots, D_1^j, D_2^j$ . Dabei gilt immer, dass  $D_2^i = D_1^{i+2}$ , da sonst die entsprechende Kante nicht in  $G_S$  vorhanden wäre. Alle Pfade werden entsprechend zugeordnet, vgl. Abbildung 3.5.



**Abbildung 3.5:** Darstellung von  $G$  und  $G_S$  und der Zuordnung der Mengen.

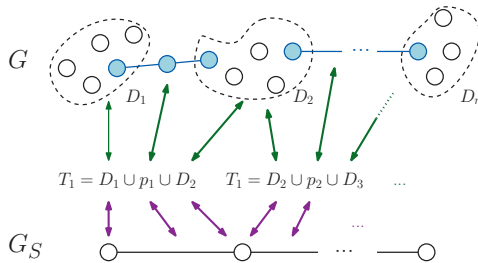
Es bleibt zu zeigen, dass alle Schnittmengen zwischen stabilen Mengen  $D^{i_1}$  und  $D^{j_1}$  immer auf diese Weise einer der beiden stabilen Mengen zugewiesen werden können, so dass Definition 3.10 erfüllt bleibt.

Dies bedeutet, dass jeder Knoten in  $G$  eindeutig entweder einer stabilen Menge oder einem Pfad zugeordnet werden muss. Gleichung IP1 stellt si-

cher, dass dies für jeden Knoten eines Pfades gilt. Für jeden anderen Knoten gilt, dass er einer beliebigen stabilen Menge die ihn überdeckt zugewiesen werden darf. Sollte sich dabei zeigen, dass eine stabile Menge dieser Überdeckung eliminiert werden kann, so war die Überdeckung nicht minimal. Das wiederum würde der Voraussetzung widersprechen.

„ $\Leftarrow$ “ Da jede Zerlegung eines Graphen auch immer eine Überdeckung des Graphen ist, ist lediglich zu zeigen, dass eine pseudostabile Menge gemäß Lemma 3.10 der Definition 3.6 genügt. Gilt dies, so entspricht auch eine minimalen Zerlegung des Graphen  $G = (V, E)$  mit einem Teilgraphen  $B \subset G$  von blauen Knoten und Kanten in pseudostabile Mengen einer minimalen Überdeckung  $S$  des Graphen  $G$  mit pseudostabilen Tupeln.

Es ist sofort ersichtlich, dass je zwei aufeinander folgende stabile Mengen innerhalb einer pseudostabilen Menge ein pseudostabiles Tupel bilden. Jede pseudostabile Menge bildet eine Zusammenhangskomponente des Graphen  $G_S$  für die Überdeckung mit pseudostabilen Tupeln. Vgl. hierzu Abb. 3.6.



**Abbildung 3.6:** Darstellung von  $G$  und  $G_S$  und der Zuordnung von pseudostabilen Mengen und Zusammenhangskomponenten.

Die Wertigkeit dieser Zusammenhangskomponente entspricht der Gleichung 3.1. Dies kann sukzessive für jede pseudostabile Menge in der Zerlegung gesehen, so dass die Zerlegung der Definition 3.6 entspricht.  $\square$

**Lemma 3.14.** *Jede Überdeckung  $S$  eines Graphen  $G = (V, E)$  mit einem Teilgraphen  $B \subset G$  von blauen Knoten und Kante mit mehrfach pseudostabilen Tupeln gemäß Definition 3.7 ist äquivalent zu einer Zerlegung des Graphen  $G$  in mehrfach pseudostabile Mengen gemäß Definition 3.11.*

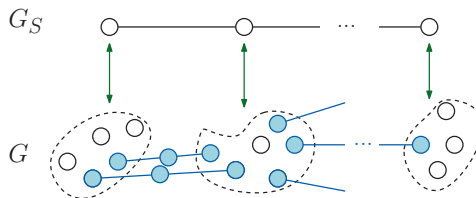
*Beweis.* „ $\Rightarrow$ “ Gegeben sei eine minimale Überdeckung  $S$  des Graphen  $G = (V, E)$  mit einem Teilgraphen  $B \subset G$  von blauen Knoten und Kante mit mehrfach pseudostabilen Tupeln  $T$  entsprechend der Gleichung IP1 wobei



für den Graph  $G_S$  der Überdeckung gilt, dass er kreisfrei ist und für alle  $v \in V(G_S)$  gilt, dass  $\delta(v) \in \{0, 1, 2\}$ .

Da der Graph  $G_S$  kreisfrei ist kann nun erneut jede Zusammenhangskomponente  $Z \subset G_S$  einzeln betrachtet werden. Eine Zusammenhangskomponente  $Z$  hat entweder nur einen Knoten, entspricht also einer stabilen Menge  $D^i$ , oder besteht aus mindestens zwei Knoten  $v_1$  und  $v_j$  mit  $\delta(v) = 1$ .

Im ersten Fall kann eine stabile Menge  $D^{i_1}$  erzeugt werden, im zweiten Fall können kontinuierlich alle stabilen Mengen der Reihe nach von  $v_1$  bis  $v_j$  betrachtet werden und analog stabile Mengen der Form  $(D_1^i, D_2^i)$   $i \in \{1, \dots, n\}$  erzeugt werden, wenn  $D_2^i \neq \emptyset$ . Das heißt  $D_1^1, D_2^1, D_2^2, \dots, D_1^j, D_2^j$ . Dabei gilt immer, dass  $D_2^i = D_1^{i+2}$ , da sonst die entsprechende Kante nicht in  $G_S$  vorhanden wäre. Alle Pfade werden entsprechend zugeordnet, vgl. Abbildung 3.7.



**Abbildung 3.7:** Darstellung von  $G$  und  $G_S$  und der Zuordnung der Mengen.

Es bleibt zu zeigen, dass alle Schnittmengen zwischen stabilen Mengen  $D^{i_1}$  und  $D^{j_1}$  immer einer der beiden stabilen Mengen zugewiesen werden können, so dass Definition 3.11 erfüllt bleibt.

Das heißt, jeder Knoten in  $G$  muss eindeutig entweder einer stabilen Menge oder einem Pfad zugeordnet werden können. Gleichung IP1 stellt sicher, dass dies für jeden Knoten eines Pfades gilt. Für jeden anderen Knoten gilt, dass er einer beliebigen stabilen Menge die ihn überdeckt zugewiesen werden darf. Sollte sich dabei zeigen, dass eine stabile Menge dieser Überdeckung eliminiert werden kann, so war die Überdeckung nicht minimal.

„ $\Leftarrow$ “ Da jede Zerlegung eines Graphen auch immer eine Überdeckung des Graphen ist, ist lediglich zu zeigen, dass eine pseudostabile Menge gemäß Lemma 3.11 die Definition 3.7 erfüllt.

Es ist sofort ersichtlich, dass je zwei aufeinander folgende stabile Mengen innerhalb einer pseudostabilen Menge ein pseudostabiles Tupel bilden. Jede pseudostabile Menge bildet eine Zusammenhangskomponente des Graphen  $G_S$  für die Überdeckung mit pseudostabilen Tupeln. Die Wertigkeit dieser

Zusammenhangskomponente entspricht der Gleichung 3.1. Dies kann sukzessive für jede pseudostabile Menge in der Zerlegung geschehen, so dass die Zerlegung der Definition 3.7 entspricht.  $\square$

Alle weiteren Beweise können ganz analog geführt werden. Die Benennung kann ebenfalls analog erfolgen. Eine pseudostabile Menge genügt entweder Definition 3.10 oder entspricht einer Zusammenhangskomponente des Graphen  $G_S$ , wie sie in der Formulierung als Mengenüberdeckung gegeben ist.

Damit ist gezeigt, dass sich das Problem entweder als Graphenüberdeckungsproblem oder als Graphenzerlegungsproblem darstellen lässt. Beide Definitionen sind äquivalent.

### 3.1.3 Notation pseudostabiler Mengen

Die Notation der Wertigkeit einer Zerlegung  $P$  des Graphen  $G$  mit  $B \subset G$  in pseudostabile Mengen wurde bereits mit  $\zeta(P)$  eingeführt. Die Wertigkeit einer minimalen Zerlegung eines Graphen  $G$  in pseudostabile Mengen wird mit  $\zeta(G)$  bezeichnet. Ist die Teilmenge  $B$  von Bedeutung, etwa weil zwei verschiedene Teilgraphen  $B$  und  $B' \subset G$  betrachtet werden, so kann dieser durch  $\zeta(G, B)$  explizit genannt werden. So kann  $\zeta$  zum einen auf Graphen als Wertigkeit der minimalen Zerlegung oder Überdeckung eines Graphen mit pseudostabilen Mengen verwendet werden, zum anderen aber auch nur die Wertigkeit einer möglichen Überdeckung oder Zerlegung angeben.

An dieser Stelle ist noch die notwendige Notation pseudostabiler Mengen einzuführen. Es sei wieder  $P_i \subset G$  eine Teilmenge von  $G$  mit  $i \in \mathbb{N}^+$ . Da grundsätzlich auch eine rein stabile Menge pseudostabil ist, kann mit dem letzten Abschnitt wiederholt werden:

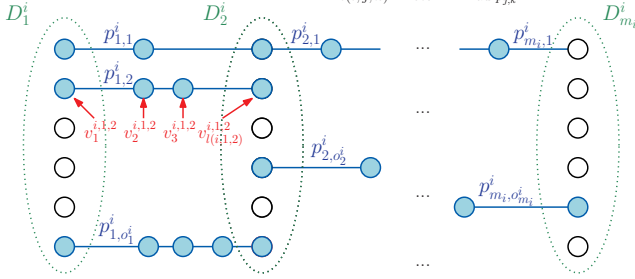
$P_i$  ist entweder stabil, d. h.  $P_i = D_1^i$  oder es existieren in  $P_i$  mehrere stabile Mengen. Das heißt es liegen  $m_i \geq 2$  stabile Mengen  $D_1^i, \dots, D_{m_i}^i \subset P_i$ .

Die Variable  $m_i$  gibt also die Anzahl der stabilen Mengen in  $P_i$  an. Ist aus dem Kontext ersichtlich, welche Menge  $P_i$  betrachtet wird, kann der Index  $i$  bei allen Variablen auch entfallen. Ist  $m_i > 1$  so ist durch die stabilen Mengen  $D_j^i$  alleine keine Überdeckung von  $P_i$  gegeben.

Es existieren auch nur Pfade aus blauen Knoten, die mit blauen Kanten verbunden sind und deren Anfangs- und Endknoten in verschiedenen stabilen Mengen liegen.

Eine pseudostabile Menge  $P_i$

$m_i$  stabile Teilmengen  $D_1^i, \dots, D_{m_i}^i$   
 $o_j^i$  Pfade von  $D_j^i$  nach  $D_{j+1}^i$   
 $l(i, j, k)$  Knoten in Pfad  $p_{j,k}^i$



**Abbildung 3.8:** Darstellung zur Notation pseudostabiler Mengen.

Dabei gibt die Variable  $i$  eines Pfades  $p_{j,k}^i$  die Menge  $P_i$  an, in der  $p_{j,k}^i$  liegt. Die Variable  $j$  wiederum bezeichnet die stabile Menge  $D_j^i$ , in der der Anfangsknoten des Pfades liegt, der Endknoten liegt immer in  $D_{j+1}^i$ . Die Pfade, die in einer stabilen Menge  $D_j^i$  beginnen, werden durchnummeriert. Dies geschieht mit der Variable  $k$ .  $p_{j,k}^i$  ist also in der Menge  $P_i$  der  $k$ -te Pfad, der die stabilen Mengen  $D_j^i$  und  $D_{j+1}^i$  verbindet. Die Anzahl der Pfade, die  $D_j^i$  und  $D_{j+1}^i$  verbinden, wird mit  $o_j^i$  bezeichnet.

Weiter sei  $l(i, j, k)$  bzw.  $l(p_{j,k}^i)$  die Länge des Pfades  $p_{j,k}^i$ . An dieser Stelle ist die Länge der Pfade  $p$  noch nicht auf 3 beschränkt, in Lemma 3.18 wird allerdings bewiesen, dass jede Zerlegung mit Pfaden, deren Länge größer als 3 ist, in eine gleichwertige Zerlegung mit Pfadlängen, die nicht größer als 3 sind, umgeformt werden kann. So ist die Definition 3.10 für eine gesuchte minimale Zerlegung relevant, die hier gegebene Notation aber zur Vollständigkeit nötig.

Zum besseren Verständnis der Notationen wird auf die Skizze in Abb. 3.8 verwiesen.

**Bemerkung 3.15.** In dem Fall, dass es zwischen zwei stabilen Mengen stets nur einen Pfad geben darf, kann die Zählvariable  $k$  auch entfallen.

### 3.1.4 Wertigkeit pseudostabiler Mengen

In den, in dem vorherigen Abschnitt 3.1 diskutierten, Definition 3.10 oder 3.10, wurde stets gefordert, dass die Länge von Pfaden zwischen stabilen

Mengen die Länge drei nicht überschreitet. Dies soll im Folgenden begründet werden. Doch zunächst soll die Definition der Wertigkeit pseudostabiler Mengen und mehrfach pseudostabiler Mengen allgemein erfolgen:

**Definition 3.16.** Zu einer (mehrfach) pseudostabilen Menge  $P_i$  ist

$$\zeta(P_i) = m_i + \left( \sum_{r=1}^{|Pf^i|} \sum_{s=1}^{o_r^i} |p_{r,s}^i| - 3 \right) \quad (3.3)$$

die Wertigkeit dieser Mengen.

Haben alle Pfade die Länge 3, so entfällt die hintere Summe. Es gilt dann

$$\zeta(P_i) = m_i$$

Für rein pseudostabile Mengen kann die oben genannte Formel noch vereinfacht werden, da die innere Summe entfällt:

**Lemma 3.17.** *Es sei  $P_i$  eine pseudostabile Teilmenge  $P_i \subset G$  eines Graphen  $G = (V, E)$  mit einer Teilmenge von blauen Knoten und Kanten  $B(G) \subset G$ . Dann ist die Wertigkeit dieser Menge*

$$\zeta(P_i) = \left( \sum_{r=1}^{m_i} |p_r^i| - 2 \right) + 1$$

*Beweis.* Da  $|Pf^i| = m_i - 1$  ist, gilt

$$\begin{aligned} \zeta(P_i) &= m_i + \left( \sum_{r=1}^{|Pf^i|} |p_r^i| - 3 \right) \\ &= m_i + \left( \sum_{r=1}^{m_i-1} |p_j^i| - 2 - 1 \right) \\ &= m_i - (m_i - 1) \cdot 1 + \left( \sum_{r=1}^{m_i-1} |p_j^i| - 2 \right) \\ &= m_i - m_i + 1 + \left( \sum_{r=1}^{m_i-1} |p_j^i| - 2 \right) \\ &= \left( \sum_{i=1}^{m_i-1} |p_j^i| - 2 \right) + 1 \end{aligned}$$

□

Dies entspricht, wenn alle Pfade die Länge 3 haben wieder der Anzahl der stabilen Mengen:

$$\zeta(P_i) = \left( \sum_{i=1}^{m_i-1} |p_j^i| - 2 \right) + 1 = m_i - 1 + 1 = m_i$$

Für eine Zerlegung  $P$  eines Graphen  $G$  in  $|P|$  (mehrfach) pseudostabile Mengen  $P_1, \dots, P_{|P|}$  ist die Wertigkeit dieser Zerlegung durch

$$\zeta(P) = \sum_{i=1}^{|P|} \zeta(P_i)$$

gegeben.

Weiter sei  $M_P = \sum_{i=1}^{|P|} m_i$  die Anzahl aller stabilen Mengen in  $P$  und  $Pf_P = Pf^1 \cup \dots \cup Pf^{|P|}$  die Menge aller Pfade in  $P$ . Wenn klar ist, welche Zerlegung  $P$  betrachtet wird, kann der Index  $P$  auch entfallen.

Die Definition der Wertigkeit in Formel 3.3 hat natürlich eine gewisse Beliebigkeit. Es steht frei, diese auch anders zu definieren. Sie wurde bereits in 3.3 verallgemeinert. Die vorliegende Formulierung ist für alle in dieser Arbeit vorgestellten Anwendungsprobleme gültig.

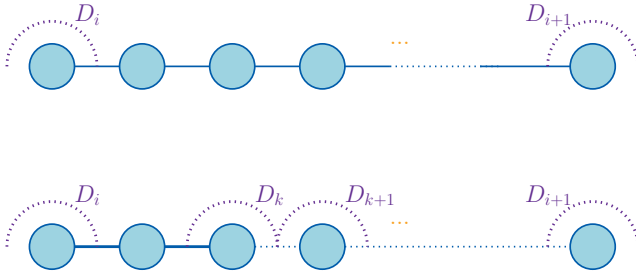
## 3.2 Strukturen von pseudostabilen Mengen

Um die Struktur pseudostabiler Mengen genauer zu untersuchen, sei zunächst die Länge der Pfade Gegenstand näherer Untersuchungen.

**Lemma 3.18.** *Gegeben sei ein Graph  $G = (V, E)$  mit einer Teilmenge von blauen Knoten und Kanten  $B(G) \subset G$  sowie eine minimale Zerlegung  $P$  dieses Graphen in (mehrfach) pseudostabile Mengen mit Wertigkeit  $\zeta(G)$  und Pfaden  $Pf$ , wobei für mindestens einen Pfad  $p_i \in Pf$  gilt  $l(p_i) \geq 4$ .*

*Dann kann aus dieser Zerlegung  $P$  eine Zerlegung  $P'$  mit  $\zeta(P) = \zeta(P')$  konstruiert werden, wobei für alle  $p'_i \in Pf'$  gilt  $l(p'_i) = 3$*

*Beweis. Voraussetzung:* Gegeben sei ein Graph  $G = (V, E)$  mit einer Teilmenge von blauen Knoten und Kanten  $B(G) \subset G$  sowie eine minimale Zerlegung  $P$  dieses Graphen in (mehrfach) pseudostabile Mengen mit Wertigkeit  $\zeta(G)$  und einem Pfad  $p_i = [v_1, \dots, v_{l(p_i)}] \in Pf$  mit  $l(p_i) \geq 4$ .



**Abbildung 3.9:** Skizze zum Beweis von Lemma 3.18 und der Anordnung der stabilen Mengen.

**Z.z.:** Es kann eine Zerlegung  $P'$  mit der selben Wertigkeit  $\zeta(P)$  konstruiert werden, die den Pfad  $p_i$  nicht beinhaltet.

Die Kosten für den Pfad  $p_i$  betragen  $l(p_i) - 3$ . Es sei nun  $v_3$  eine neue stabile Menge innerhalb der Zerlegung  $P'$  und  $[v_1, v_2, v_3]$  ein Pfad. Dies ist zulässig, da nach Voraussetzung alle Knoten in  $p_i$  und alle Kanten blau sind. Jeder weitere Knoten  $v_4, \dots, v_{l(p_i)}$  sei weiter eine stabile Menge in  $P'$ .

Für den ersten Pfad beträgt die Wertigkeit 0, für die neuen stabilen Mengen  $l(p_i) - 3$ , die Wertigkeit verändert sich also nicht gegenüber  $P$ . Vgl. hierzu Abb. 3.9.

Dieses Verfahren kann sukzessive für alle Pfade wiederholt werden, deren Länge 3 übersteigt. Die Wertigkeit der so konstruierten Zerlegung  $P'$  ist gleich der von  $P$ .  $\square$

**Bemerkung 3.19.** Damit ist die Bedingung, dass für alle Pfade  $p_i$   $l(p_i) \leq 3$  gilt, in den Definitionen 3.10 und 3.11 gerechtfertigt.

Weiter kann nun untersucht werden, welche Konstruktionen man im Graphen  $G' = B(G)$  ausschließen kann.

**Lemma 3.20.** Gegeben sei ein Graph  $G = (V, E)$  mit blauen Knoten und blauen Kanten  $B \subset G$ . Es sei nun

$$B' = B(G) \setminus \{v \in B(G) \text{ mit } \delta(v) = 0 \text{ in } B(G)\}$$

Dann gilt für die zwei minimalen Zerlegungen von  $G$  in pseudostabile Mengen bezüglich  $B$  und  $B'$   $\zeta(P(B)) = \zeta(P(B'))$ .

*Beweis.* Angenommen, es gibt eine minimale Zerlegung  $\zeta(P(B))$ , wobei  $v \in Pf$  mit  $\delta(v) = 0$  in  $G'$  ein Pfad in  $Pf$  ist. Dies widerspricht der Definition 3.10 oder 3.11, es handelt sich also nicht um eine gültige minimale Zerlegung von  $G$  in pseudostabile Mengen.

Da es also keine Zerlegung des Graphen  $G$  in pseudostabile Mengen gibt, bei denen Knoten  $v \in B(G)$  mit  $\delta(v) = 0$  in einem Element von  $Pf$  vorhanden sind, gilt  $\zeta(P(B)) = \zeta(P(B'))$ , wenn  $B' = B(G) \setminus \{v \in B(G) \text{ mit } \delta(v) = 0 \text{ in } B(G)\}$ .  $\square$

Isolierte Knoten in  $B$  können also auch ignoriert oder entfernt werden.

**Lemma 3.21.** *Gegeben sei ein Graph  $G = (V, E)$  mit blauen Knoten und blauen Kanten  $B(G) \subset G$ . Es sei nun*

$$B' = B(G) \setminus \{v, w \in B(G) \text{ mit } \delta(v) = 1 \text{ und } \delta(w) = 1 \text{ in } B \text{ und } (v, w) \in B(G)\}$$

*Dann gilt für die zwei minimalen Zerlegungen von  $G$  in pseudostabile Mengen bezüglich  $B$  und  $B'$   $\zeta(P(B)) = \zeta(P(B'))$ .*

*Beweis.* Angenommen, es gibt eine minimale Zerlegung  $\zeta(P(B))$  wobei  $[v, w]$  mit  $\delta(v) = 1$  und  $\delta(w) = 1$  in  $G'$  ein Pfad in  $Pf$  ist. Dieser Pfad verbindet also zwei stabile Mengen  $D_j^i$  und  $D_{j+1}^i$  miteinander. Es ist aber möglich, eine Zerlegung in pseudostabile Mengen mit der selben Wertigkeit ohne den Pfad  $[v, w]$  zu konstruieren.

Wenn  $[v, w] \in P^i$  ist, konstruiert man dazu zwei verschiedene pseudostabile Mengen  $P''^i$  und  $P'''^i \in P$ . Dazu sei  $D_1^i, \dots, D_j^i \in P''^i$ , ebenso seien die Pfade  $p_1^i, \dots, p_{j-1}^i$  in  $P''^i$ . Der Pfad  $p_j^i = [v, w]$  entfällt. Weiter sei  $D_{j+1}^i, \dots, D_{m_i}^i \in P'''^i$  und ebenso auch  $p_{j+2}^i, \dots, p_{m_i-1}^i$  in  $P'''^i$ . Dann gilt  $\zeta(P^i) = \zeta(P''^i) + \zeta(P'''^i)$ .

Es gibt also für jede minimale Zerlegung eines Graphen  $G$  mit blauen Knoten und blauen Kanten  $B(G) \subset G$  in pseudostabile Mengen, wobei

$$|\{v, w \in B \text{ mit } \delta(v) = 1 \text{ und } \delta(w) = 1 \text{ in } G' \text{ und } (u, v) \in B\}| \geq 1$$

gilt, eine minimale Zerlegung mit der gleichen Wertigkeit  $\zeta(P(B)) = \zeta(P(B'))$ , wenn

$$B' = B(G) \setminus \{v, w \in B(G) \text{ mit } \delta(v) = 1 \text{ und } \delta(w) = 1 \text{ in } B \text{ und } (u, v) \in B(G)\}$$

$\square$

Isolierte Pfade der Länge 2 in  $B$  können also auch entfernt werden. Man kann also von folgender Annahme ausgehen:

**Annahme 3.22.**  $B(G)$  beinhaltet keine isolierten Knoten und nur Pfade, die mindestens die Länge 3 haben.

**Lemma 3.23.** Gegeben sei ein Graph  $G = (V, E)$  mit einer Teilmenge von blauen Knoten und blauen Kanten  $B(G) \subset G$  sowie eine minimale Zerlegung  $P$  dieses Graphen in (mehrfach) pseudostabile Mengen mit Wertigkeit  $\zeta(G)$ .

Dann kann diese Zerlegung in eine Zerlegung  $P'$  gleicher Wertigkeit umgeformt werden, so dass für alle Pfade  $p_i = [v_1, v_2, v_3]$  mit  $v_i \in Pf$  gilt, dass  $v_2$  in einem Kreis ungerader Länge liegt<sup>1</sup> umgeformt werden.

*Beweis. Voraussetzung:* Gegeben sei ein Graph  $G = (V, E)$  mit einer Teilmenge von blauen Knoten und blauen Kanten  $B(G) \subset G$  sowie eine minimale Zerlegung  $P$  dieses Graphen in (mehrfach) pseudostabile Mengen mit Wertigkeit  $\zeta(G)$ . Es sei ein Pfad  $p_i = [v_1, v_2, v_3] \in Pf$ , wobei  $v_2$  nicht Teil eines ungeraden Kreises in  $G$  ist. Weiter sei  $D_1$  die stabile Menge in der sich  $v_1$  befindet und  $D_2$  die, in der sich  $v_3$  befindet.

**Z.z.:** Jeder Zerlegung  $P$  in (mehrfach stabile) Mengen, die einen solchen Pfad  $p_i$  beinhaltet, kann in eine gleichwertige zulässige Zerlegung  $P'$  umgeformt werden, in der  $p_i$  kein Pfad in  $Pf$  ist.

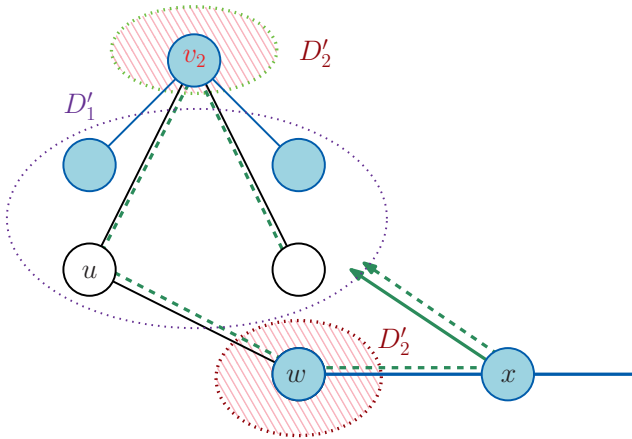
Der Graph  $D_1 \cup D_2 \subset G$  ist nun offensichtlich bipartit und enthält somit keine Kreise ungerader Länge. Nun ist aber nach Voraussetzung in  $D_1 \cup D_2 \cup \{v_2\} \subset G$  auch kein Kreis ungerader Länge vorhanden, dieser Graph also auch bipartit und so kann der Graph  $D_1 \cup D_2 \cup \{v_2\} \subset G$  auch in zwei stabile Mengen  $D'_1$  und  $D'_2$  zerlegt werden.

Ist  $P$  eine rein pseudostabile Menge, so waren sowohl  $D_1$  als auch  $D_2$  End- und Startort von maximal zwei Pfaden. Da nun einer entfällt, nämlich  $p_i$  können die verbliebenen maximal zwei Pfade an die neuen stabilen Mengen, unabhängig von ihrer neuen Sortierung, angeknüpft werden. Gelangen beide an  $D'_1$  oder  $D'_2$ , so ist dies eine gültige Zerlegung in pseudostabile Mengen, ebenso wenn ein Pfad seinen Endknoten in der einen, der andere Pfad seinen Anfangsknoten in der anderen, findet.

Ist  $P$  eine mehrfach pseudostabile Menge, wobei  $D_1$  und  $D_2$  jeweils mit mehr als einem Pfad verbunden sind, so müssen die neuen stabilen Mengen  $D'_1$  und  $D'_2$  auf folgende Art konstruiert werden: Es sei  $D'_2$  die stabile

<sup>1</sup>d. h.  $v_2 \subset C_n \subset G$  mit  $n = 2i + 1, i \in \mathbb{N}^0$





**Abbildung 3.10:** Skizze zum Beweis von Lemma 3.23.

Menge, der  $v_2$  zugeordnet wird. Alle Knoten aus  $D_1$  und  $D_2$ , die mit  $v_2$  verbunden sind, werden  $D'_1$  hinzugefügt. Dies ist möglich, da keine Kanten zwischen zwei Knoten, die jeweils mit  $v_2$  verbunden sind, existieren dürfen. Dies würde einen ungeraden Pfad der Länge Drei mit  $v_2$  implizieren, was der Voraussetzung widerspricht. Ebenso werden alle Knoten aus  $D_1$  und  $D_2$   $D'_1$  hinzugefügt, die weder Kanten in die jeweils anderen Menge  $D_1$  oder  $D_2$  noch zu  $v_2$  besitzen. Alle anderen Knoten sollen zu  $D'_2$  hinzugefügt werden.

Sind alle Pfade, die von  $D_1$  und  $D_2$  ausgehen, nun mit  $D'_1$  verbunden, ist dies wieder eine zulässige Zerlegung in mehrfach pseudostabile Mengen. Gibt es aber einen Knoten  $w$  in  $D'_2$ , der Teil eines Pfades  $[w, x, y]$  oder  $[y, x, w]$  ist, so würde dieser der Definition pseudostabiler Mengen widersprechen, wenn es auch einen Pfad von  $D'_1$  in die stabile Menge des Knoten  $y$  gibt. Ist dies nicht der Fall, wurde eine zulässige Zerlegung in mehrfach pseudostabile Mengen konstruiert. Sonst kann der Knoten  $x$  aber der stabilen Menge  $D'_1$  hinzugefügt werden. Dies geht, denn würde es eine Kante  $(x, q)$  mit  $q \in D'_1$  geben, so wäre  $[v_2, q, x, w, u]$  ein Kreis in  $G$ , wobei  $u$  ein beliebiger Knoten in  $D'_1$  ist, der mit  $w$  verbunden ist. Dieser Knoten  $u$  muss existieren, denn sonst könnte  $w$   $D'_1$  hinzugefügt werden. Dieser Kreis hat ungerade Länge und beinhaltet  $v_2$ , was ein Widerspruch zur Voraussetzung ist. Also können alle bestehenden Pfade entweder  $D'_1$  angeschlossen werden oder ihr mittlerer Knoten dieser stabilen Menge hinzugefügt werden. Dies impliziert eine Zerlegung in mehrfach pseudostabile Mengen mit der selben

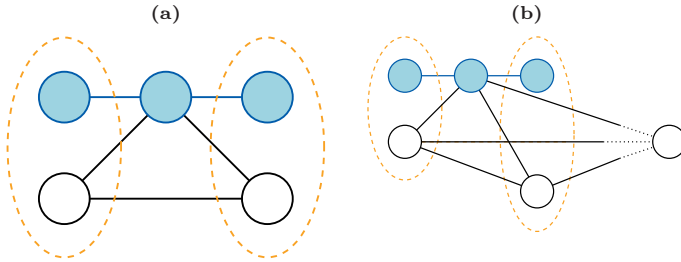


Abbildung 3.11: Skizzen zu den Beispielen 3.1 und 3.2.

Wertigkeit. Vgl. hierzu auch Abb. 3.10. □

Der Kreis ungerader Länge, in dem ein mittlerer Knoten eines Pfades  $p$  zwischen zwei stabilen Mengen liegt, muss nicht automatisch Teil dieses Pfades sein. Das kann durch folgendes Beispiel verdeutlicht werden:

**Beispiel 3.1.** Es sei  $G$  ein Graph mit einem  $K_3$  und einem Pfad  $p = [v_1, v_2, v_3]$  von mit blauen Kanten verbundenen blauen Knoten. Nur  $v_2$  sei Teil des  $K_3$ . Eine illustration findet sich in Abb. 3.11(a). Dort ist auch eine minimale Zerlegung in pseudostabile Mengen skizziert. Es fällt auf, dass der Pfad mit  $v_2$  ermöglicht, die Wertigkeit dieser Zerlegung auf zwei zu reduzieren, obwohl der Pfad  $p$  nicht in  $K_3$  liegt. △

**Beispiel 3.2.** Obwohl es für einen Pfad  $p = [v_1, v_2, v_3]$  zwischen zwei stabilen Mengen  $D_1$  und  $D_2$  in einer Zerlegung in pseudostabile Mengen immer einen Kreis ungerader Länge mit  $v_2$  in  $D_1$  und  $D_2$  geben muss – sonst wäre  $D_1 \cup D_2 \cup v_2$  ja bipartit – kann es durchaus auch Kreise ungerader Länge geben, die zwar die selbe Länge haben wie der Kreis in  $D_1$  und  $D_2$  aber nicht in  $D_1$  und  $D_2$  liegen.

Es sei eine ähnliche Situation wie in Beispiel 3.1 gegeben, zusätzlich gibt es einen weiteren  $K_3$ , von dem ein Knoten  $v_2$  ist und ein anderer im schon vorhandenen  $K_3$  liegt. Mit der in Abb. 3.11(b) skizzierten Situation kann der zusätzliche Knoten keiner der beiden stabilen Mengen  $D_1$  und  $D_2$  hinzugefügt werden, er kann aber auch kein Teil eines Pfades sein, muss also in einer weiteren stabilen Menge liegen. △

Auch hier kann wieder eine Annahme getroffen werden, die insbesondere für die spätere algorithmische Betrachtung von Vorteil ist.

**Annahme 3.24.** *Es soll im Folgenden für alle (mehrfach) pseudostabilen Mengen angenommen werden, dass in jedem Pfad  $p_i = [v_1, v_2, v_3]$  der Knoten  $v_2$  Teil eines Kreises ungerader Länge im Ursprungsgraphen ist.*

Damit folgt direkt:

**Lemma 3.25.** *Wenn ein Graph  $G$  keine Kreise ungerader Länge enthält, also bipartit ist, so ist die Zerlegung in (mehrfach) pseudostabile äquivalent zu einer Zerlegung in stabile Mengen.*

Nun sind also alle Pfade, die zwischen stabilen Mengen in einer (mehrfach) pseudostabilen Menge auftreten dürfen, wie folgt charakterisiert: Sie bestehen aus drei blauen Knoten  $v_1, v_2$  und  $v_3$ , sind durch blaue Kanten  $(v_1, v_2)$  und  $(v_2, v_3)$  verbunden und  $v_2$  ist Teil eines Kreises ungerader Länge im Ursprungsgraphen  $G$ . Für perfekte Graphen kann man sogar noch eine weitere Einschränkung machen:

**Lemma 3.26.** *Es sei  $G = (V, E)$  ein perfekter Graph mit blauen Knoten und blauen Kanten  $B(G) \subset G$ . Dann sind alle Knoten  $v_2$  der Pfade  $p_i = [v_1, v_2, v_3] \subset Pf$  einer minimalen Zerlegung  $P$  des Graphen  $G$  in (mehrfach) pseudostabile Mengen Teil eines  $K_3 \subset G$ .*

*Beweis.* Nach Annahme 3.24 müssen alle Pfade Teil eines Kreises ungerader Länge sein. Ist  $G$  ein perfekter Graph, so ist er ein Berge Graph.  $G$  enthält damit keinen ungeraden Kreis der Länge größer drei. Also müssen alle Pfade in einer minimalen Zerlegung  $P$  in (mehrfach) pseudostabile Mengen Teil eines ungeraden Kreises der Länge drei, also einem  $K_3$  sein.  $\square$

Als letzter Punkt kann noch auf den Zusammenhang zwischen Cliques und pseudostabilen Mengen eingegangen werden. Die Wertigkeit einer minimalen Überdeckung von einer größten Clique  $C$  in einem Graphen  $G$  ist kleiner oder gleich der minimalen Überdeckung des ganzen Graphen  $G$ .

**Lemma 3.27.** *Es sei  $G = (V, E)$  ein Graph mit blauen Knoten und Kanten  $B(G) \subset G$ . Weiter sei  $C$  eine größte Clique bzw. die größte Clique im Graphen  $G$ . Dann gilt für eine minimale Zerlegung  $P$  des Graphen  $G$  in (mehrfach) pseudostabile Mengen mit Wertigkeit  $\zeta(G)$  und eine minimale Zerlegung  $P'$  von  $C$  in (mehrfach) pseudostabile Mengen mit Wertigkeit  $\zeta(C)$*

$$\zeta(C) \leq \zeta(G)$$

*Beweis.* Sind sowohl in  $P$  als auch in  $P'$  alle Mengen stabil, d. h. es existieren keine Pfade in  $Pf$ , so gilt offensichtlich  $\zeta(C) \leq \zeta(G)$ .

Existieren in mindestens einer Zerlegung eine wirklich pseudostabile Menge, also Mengen  $P_i$  mit  $\zeta(P_i) > 1$ , so müssen folgende Fälle unterschieden werden:

- Ist  $P$  eine Zerlegung mit ausschließlich stabilen Mengen, in der Zerlegung  $P'$  existiert hingegen mindestens eine pseudostabile Menge, d. h. eine Menge  $P'_i$  mit  $\zeta(P'_i) > 1$ , dann gilt entweder
  1. Die Knoten von  $G \setminus C$  lassen sich den stabilen Teilmengen der pseudostabilen Mengen der Zerlegung  $P'$  zuweisen. Dann gilt  $\zeta(C) = \zeta(G)$ .
  2. Sonst werden für  $\zeta(G)$  mehr (pseudo-)stabile Mengen benötigt und es gilt  $\zeta(C) < \zeta(G)$ .
- Gibt es in der Zerlegung  $P$  mindestens eine pseudostabile Menge, d. h. eine Menge  $P_i$  mit  $\zeta(P_i) > 1$ , dann gilt, da  $P'$  minimal ist

$$\zeta(G) = \max \{ \zeta(G \setminus C), \zeta(C) \} \geq \zeta(C)$$

□

Es sollen noch zwei elementare Lemmata für die minimale Zerlegung vollständiger Graphen in mehrfach und rein pseudostabile Mengen diskutiert werden. Dieses Thema wird in den folgenden Abschnitten weiter vertieft.

**Lemma 3.28.** *Für die Zerlegung  $P$  eines vollständigen Graphen  $K_i$  mit  $i \geq 2$  in mehrfach pseudostabile Mengen gilt*

$$\zeta(K_i) \geq 2 \tag{3.4}$$

*Beweis.* Es sei  $G^B = K_i$  ein vollständiger blauer Graph mit  $i > 2$ . Ist  $i = 2$  so gilt  $\zeta(G) = \chi(G) = 2$ .

Man wähle nun zwei beliebige blaue Knoten  $v_1$  und  $v_2$  aus. Jeder weitere Knoten  $v_j \in G$  ist nun als blauer Knoten durch zwei blauen Kanten mit den Knoten  $v_1$  und  $v_2$  verbunden, kann also als Pfad zwischen diesen genutzt werden. So kann eine Zerlegung  $P$  mit zwei stabilen Mengen  $D_1 = \{v_1\}$  und  $D_2 = \{v_2\}$  sowie  $i - 2$  Pfaden zwischen diesen konstruiert werden. Die Wertigkeit dieser Zerlegung ist also zwei.

Für jeden Knoten, der nicht blau ist, oder jede Kante, die nicht blau ist, muss nun also eine weitere stabile Menge genutzt werden. Die Formel 3.4 gilt also immer.  $\square$

**Lemma 3.29.** *Für die Zerlegung  $P$  eines vollständigen Graphen  $K_i$  mit  $i \geq 2$  in pseudostabile Mengen gilt*

$$\zeta(K_i) \geq \left\lceil \frac{i+1}{2} \right\rceil \quad (3.5)$$

*Beweis.* Es sei  $G^B = K_i$  ein vollständiger blauer Graph mit  $i > 2$ . Ist  $i = 2$  so gilt  $\zeta(G) = \chi(G) = 2$ .

Man wähle nun  $\left\lceil \frac{i+1}{2} \right\rceil$  beliebige blaue Knoten aus  $G$  und setze diese als stabile Mengen  $D_j$ . Alle übrigen Knoten  $v_k$  in  $G$  können nun, da alle Kanten und alle Knoten blau sind, als Pfad zwischen diesen stabilen Mengen genutzt werden.

Für jeden Knoten oder jede Kante, die nicht blau ist, muss nun also eine weitere stabile Menge genutzt werden. Die Ungleichung 3.5 gilt also immer.  $\square$

### 3.3 Zusammenhänge zwischen pseudostabilen und stabilen Mengen

Es ist ebenfalls wichtig, die Beziehung zwischen Graphenfärbung, also einer Zerlegung eines Graphen in stabile Mengen und der Zerlegung eines Graphen in pseudostabile Mengen zu betrachten und die Ergebnisse formal festzuhalten. Aufgrund der Definition von pseudostabilen Mengen ergibt sich das folgende Lemma:

**Lemma 3.30.** *Eine Zerlegung eines Graphen  $G = (V, E)$  ohne blaue Knoten und Kanten, also mit  $B(G) = \emptyset$  in pseudostabile Mengen entspricht einer Graphenfärbung von  $G$  mit der selben Wertigkeit.*

Weiter erfüllt jede Zerlegung eines Graphen in stabile Mengen die Bedingungen für eine Zerlegung in mehrfach pseudostabile Mengen.

**Lemma 3.31.** *Eine Zerlegung eines Graphen  $G = (V, E)$  in stabile Mengen ist auch immer eine (nicht notwendigerweise minimale) Zerlegung dieses Graphen mit einer beliebigen Menge von blauen Knoten und blauen Kanten in (mehrfach) pseudostabile Mengen.*

*Beweis.* Eine Zerlegung eines Graphen  $G$  in stabile Mengen genügt Definition 3.10 bzw. 3.11. Alle anderen Voraussetzungen werden mangels Knoten, die nicht in stabilen Mengen sind, auch erfüllt.  $\square$

Für eine minimale Zerlegung in mehrfach pseudostabile Mengen folgt nun also direkt, dass es eine natürliche obere Schranke gibt:

**Lemma 3.32.** *Gegeben sei ein Graph  $G = (V, E)$  mit einer Teilmenge von blauen Knoten und Kanten  $B(G) \subset G$ . Für eine minimale Zerlegung dieses Graphen in (vollständig kreuzungsfreie) (mehrfach) pseudostabile Mengen mit Wertigkeit  $\zeta(G)$  gilt*

$$\zeta(G) \leq \chi(G).$$

Eine minimale Zerlegung in pseudostabile Mengen impliziert aber nicht notwendigerweise eine Aussage über eine Zerlegung in stabile Mengen.

**Lemma 3.33.** *Gegeben sei ein Graph  $G = (V, E)$  mit einer Teilmenge von blauen Knoten und Kanten  $B(G) \subset G$  sowie eine minimale Zerlegung  $P$  dieses Graphen in (mehrfach) pseudostabile Mengen mit Wertigkeit  $\zeta(G)$ . Dann impliziert dies eine Zerlegung in  $M + l(\widetilde{P}f)$  stabile Mengen, wobei*

$$M = \sum_{r=1}^{|P|} m_r \quad \text{und} \quad l(\widetilde{P}f) = \sum_{r=1}^{|P|} (l(p_r) - 2)$$

*Beweis.* Jede stabile Menge  $D_j^i \subset P^i \subset P$  der Zerlegung  $P$  ist bereits per Definition eine stabile Menge, es existieren also bereits  $M$  stabile Mengen. Jeder Knoten eines Pfades  $p_j^i$  bildet entweder eine neue stabile Menge, oder ist bereits in einer stabilen Menge. Dies sind die Anfangs- und Endknoten. Dies entspricht einer Anzahl von

$$l(\widetilde{P}f) = \sum_{r=1}^{|P|} (l(p_r) - 2)$$

Knoten. So kann eine Zerlegung des Graphen  $G$  in  $M + l(\widetilde{P}f)$  stabile Mengen konstruiert werden.  $\square$

Diese Zerlegung ist nicht notwendigerweise minimal. Andererseits bringt jedoch jede Zerlegung eines Graphen  $G$  in (mehrfach) pseudostabile Mengen auch eine Zerlegung in stabile Mengen  $D_j^i$  mit sich.

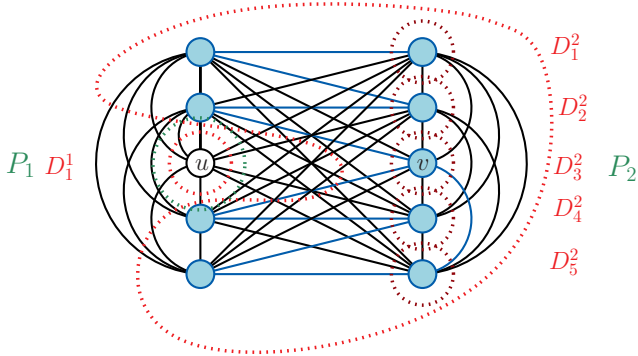


Abbildung 3.12: Skizze zu Beispiel 3.3.

**Lemma 3.34.** *Gegeben sei ein Graph  $G = (V, E)$  mit einer Teilmenge von blauen Knoten und Kanten  $B(G) \subset G$  sowie eine minimale Zerlegung  $P$  dieses Graphen in (mehrfach) pseudostabile Mengen mit Wertigkeit  $\zeta(G)$ . Dann impliziert diese Zerlegung eine Zerlegung des Graphen  $G \setminus \widetilde{Pf}$  in stabile Mengen.*

*Beweis.* Die Behauptung folgt direkt, da alle Knoten von  $G \setminus \widetilde{Pf}$  in einer der stabilen Mengen  $D_j^i \subset P$  liegen müssen.  $\square$

Diese Zerlegung ist jedoch erneut nicht notwendigerweise minimal. So auch in folgendem Beispiel:

**Beispiel 3.3.** Es sei  $G = K_9 \cup \{u\}$  wobei der Knoten  $u$  mit allen anderen Knoten in  $G$  verbunden ist, außer einem bestimmten Knoten  $v$ . Es sei nun eine Zerlegung  $P$  dieses Graphen in pseudostabile Mengen gemäß Abb. 3.12 gegeben, d. h. es existiert eine pseudostabile Menge  $P_i = D_1^i = \{u\}$  und eine weitere, in der es fünf stabile Mengen  $D_1^2, \dots, D_5^2$  mit jeweils einem Knoten gibt, wobei eine dieser Mengen aus dem Knoten  $v$  besteht. Diese sind durch Pfade der Länge drei verbunden. Es existiert also eine minimale Zerlegung in pseudostabile Mengen mit Wertigkeit 6. Man kann erkennen, dass die durch diese Zerlegung implizierte Zerlegung von  $G \setminus \widetilde{Pf}$  in stabile Mengen nicht minimal ist, da  $D_1^1$  und  $D_3^2$  vereinigt werden könnten.  $\triangle$

Unter bestimmten Bedingungen kann diese Zerlegung aber auch minimal sein.

**Lemma 3.35.** *Gegeben sei ein Graph  $G = (V, E)$  mit einer Teilmenge von blauen Knoten und Kanten  $B(G) \subset G$  sowie eine minimale Zerlegung  $P$  dieses Graphen in pseudostabile Mengen mit Wertigkeit  $\zeta(G)$  und  $m_i \leq 2$  für alle  $i \in \{1, \dots, |p|\}$ . Dann impliziert diese Zerlegung eine minimale Zerlegung des Graphen  $G \setminus \widetilde{P}f$  in stabile Mengen.*

*Beweis.* Angenommen, die Zerlegung  $P$  impliziert auf  $G \setminus \widetilde{P}f$  keine minimale Zerlegung in stabile Mengen. Da  $G \setminus \widetilde{P}f$  nur aus den stabilen Mengen  $D_j^i$  besteht, gibt es also mindestens eine stabile Menge, deren Knoten alle in andere stabile Mengen verteilt werden können. Man verteile also alle Knoten  $D_j^i$ , die nicht in der Menge der Pfade der Zerlegung  $P$ , also  $\widetilde{P}f$  liegen, auf die passenden stabilen Mengen.

- Ist  $j = 1 = m_i$ , so gibt es in der Zerlegung  $P$  eine Menge  $P^i = D_j^i$ . Die Zerlegung  $P$  war also nicht minimal, was ein Widerspruch zur Voraussetzung ist.
- Ist  $j = 1 < m_i$  so gibt es also einen von  $D_j^i$  ausgehenden Pfad. Da der verbliebene Knoten einer anderen stabilen Menge  $D_y^x$  hinzugefügt werden kann und diese per Definition ebenfalls nur ein- oder ausgehende Pfade hat (da  $m_x \leq 2$ ) könnten die beiden pseudostabilen Mengen  $P_i$  und  $P_x$  an dieser Stelle zusammengefügt werden. Die Zerlegung  $P$  war also nicht minimal, was ein Widerspruch zur Voraussetzung ist.
- Ist  $j = 2 = m_i$  so gibt es also in  $D_j^i$  einen eingehenden Pfad. Mit der selben Argumentation wie im vorherigen Fall können wieder zwei pseudostabile Mengen zusammengefügt werden. Die Zerlegung  $P$  war also nicht minimal, was ein Widerspruch zur Voraussetzung ist.
- Weitere Fälle können nicht auftreten, da  $m_i \leq 2$ .

Die Zerlegung  $P$  impliziert auf  $G \setminus \widetilde{P}f$  also unter den oben genannten Voraussetzungen immer eine minimale Zerlegung in stabile Mengen.  $\square$

Für mehrfach pseudostabile Mengen wird der Zusammenhang allerdings deutlich einfacher:

**Lemma 3.36.** *Gegeben sei ein Graph  $G = (V, E)$  mit einer Teilmenge von blauen Knoten und Kanten  $B(G) \subset G$  sowie eine minimale Zerlegung  $P$  dieses Graphen in mehrfach pseudostabile Mengen mit Wertigkeit  $\zeta(G)$ . Dann gilt für eine minimale Färbung von  $G \setminus \widetilde{P}f$  mit  $\chi(G \setminus \widetilde{P}f)$  Farben:*

$$\zeta(G) = \chi(G \setminus \widetilde{P}f)$$



*Beweis. Voraussetzung:* Zu einem Graphen  $G = (V, E)$  mit einer Teilmenge von blauen Knoten und Kanten  $B(G) \subset G$  sei  $P$  eine minimale Zerlegung des Graphen in mehrfach pseudostabile Mengen mit Wertigkeit  $\zeta(G)$  und  $F$  eine minimale Färbung des Graphen  $G \setminus Pf$  mit  $\chi(G \setminus Pf)$  Farben.

Angenommen  $\zeta(G) < \chi(G \setminus \widetilde{Pf})$ . Die Zerlegung  $P$  impliziert eine Zerlegung des Graphen  $G \setminus \widetilde{Pf}$  der Wertigkeit  $\zeta(G)$ . Ist  $\zeta(G) < \chi(G \setminus \widetilde{Pf})$ , so ist die Färbung aus der Voraussetzung nicht minimal, dies führt also zu einem Widerspruch.

Angenommen  $\zeta(G) > \chi(G \setminus \widetilde{Pf})$ . Dann gibt es entweder Pfade mit mehr als drei Knoten, was nach Annahme 3.19 nicht gestattet ist, oder es gibt isolierte Knoten, die nicht Teil eines Pfades sind, was der Definition von  $\widetilde{Pf}$  widerspricht. Auch dieser Fall ist also nicht möglich.  $\square$

Dies liefert nun eine untere Schranke.

**Lemma 3.37.** *Gegeben sei ein Graph  $G = (V, E)$  mit einer Teilmenge von blauen Knoten und Kanten  $B(G) \subset G$ . Für eine minimale Zerlegung  $P$  dieses Graphen in (mehrfach) pseudostabile Mengen mit Wertigkeit  $\zeta(G)$  gilt nun*

$$\chi(G \setminus B(G)) \leq \zeta(G)$$

*Beweis.* Mit Lemma 3.30 gilt  $\chi(G \setminus B(G)) = \zeta(G \setminus B(G)) \leq \zeta(G)$ .  $\square$

Diese Schranken müssen nicht scharf sein, sie können im Gegenteil sogar beliebig schlecht werden, etwa für Graphen  $G^B$ .

### 3.4 Zusammenfassung

Zunächst wurden (mehrfach) pseudostabile Tupel und der blaue Graph definiert, im Anschluss das Mengenüberdeckungsproblem. Danach wurde das Problem als Graphenzerlegungsproblem definiert. Es konnte gezeigt werden, dass beide Formulierungen für das Optimierungsproblem äquivalent sind.

Nachdem die notwendigen Notationen eingeführt wurden, wurde die Struktur mehrfach pseudostabiler bzw. pseudostabiler Mengen im Hinblick auf

eine minimale Zerlegung eines Graphen untersucht. Es konnte noch einmal formal gezeigt werden, dass nur Pfade der Länge drei relevant sind. Außerdem muss in einem solchen Pfad  $[v_1, v_2, v_3]$  der mittlere Knoten  $v_2$  Teil eines Kreises ungerader Länge im Ursprungsgraphen  $G$  sein. Im Anschluss wurden bereits einige elementare obere und untere Schranken für bestimmte Graphen angegeben.

Im letzten Abschnitt wurden verschiedene Zusammenhänge zwischen pseudo-stabilen und stabilen Mengen untersucht, insbesondere im Zusammenhang mit einer minimalen Zerlegung. Diese sind nicht trivial, allerdings konnten zwei wichtige Abschätzungen bewiesen werden. Zum einen gilt für einen Graph  $G$  immer

$$\zeta(G) \leq \chi(G)$$

und zum anderen gilt, wenn  $B \subset G$  die Menge von blauen Kanten und Knoten ist

$$\chi(G \setminus B(G)) \leq \zeta(G).$$

# Pseudostabile Mengen: Komplexität, Schranken und Algorithmen

In diesem Kapitel wird das Problem minPS untersucht, dass eine minimale Zerlegung eines Graphen  $G$  in pseudostabile Mengen untersucht. Dazu wird zunächst die Definition des Entscheidungs- und Optimierungsproblems gemäß der Herleitung im vorherigen Kapitel gegeben und allgemeine Grundlagen notiert. In einem folgenden Abschnitt wird die Komplexität des Problems untersucht.

minPS hat zwar eine große Ähnlichkeit mit minMPS, der minimalen Zerlegung eines Graphen  $G$  in mehrfach pseudostabile Mengen, allerdings sind einige Resultate gänzlich verschieden. Auch ist das Vorgehen oft nicht gleich, so dass es übersichtlicher ist, beide Probleme in einem eigenständigen Kapitel zu behandeln. Um Doppelungen zu vermeiden, wird aber so oft wie möglich auch auf Kapitel 6 bzw. dort auf dieses Kapitel verwiesen.

Weiter wird erneut versucht Abschätzungen für minPS zu finden und eine Darstellung als lineares Programm untersucht. Aus diesen Ansätzen werden dann Algorithmen bzw. Heuristiken hergeleitet.

In dem darauf folgenden Kapitel soll ein Anwendungsbeispiel diskutiert werden, das Train Marshalling Problem.

## 4.1 minPS

Die Definition für das Problem minPS erfolgt analog zu Kapitel 3:

**Definition 4.1** (minPS). Gegeben sei ein Graph  $G = (V, E)$  mit blauen Knoten und Kanten  $B \subset G$ . Gesucht ist eine minimale Zerlegung  $P(B)$  von  $G$  in pseudostabile Mengen.

Das entsprechende Entscheidungsproblem zu diesem Optimierungsproblem kann wie folgt definiert werden:

**Definition 4.2** (E-minPS). Gegeben sei ein Graph  $G = (V, E)$  mit blauen Knoten und Kanten  $B \subset G$ . Gegeben ein Wert  $k > 0$ : Gibt es eine Zerlegung  $P(B)$  von  $G$  in pseudostabile Mengen mit der Wertigkeit  $k$ ?

**Bemerkung 4.3.** Für eine minimale Zerlegung  $P = \{P_1, \dots, P_n\}$  eines Graphen  $G$  mit blauen Knoten und Kanten  $B \subset G$  in pseudostabile Mengen schreibt man statt  $\zeta(P)$  auch kurz  $\zeta(G)$  bzw.  $\zeta_P(G)$ . Wenn unklar ist, welche Menge  $B$  betrachtet wird, schreibt man statt  $P$  auch  $P(B)$ .

Zu einer Zerlegung  $P$  eines Graphen  $G = (V, E)$  mit blauen Knoten und Kanten  $B \subset G$  kann nun der Graph  $G_P$  dieser Zerlegung gebildet werden. Dies entspricht dem Graphen der Überdeckung aus Definition 3.5. Die Definition erfolgt ähnlich zu der Definition von Beygang [8, S. 31] im Kontext des TMP. Für eine minimale Zerlegung gilt natürlich  $G_S = G_P$ , da eine minimale Zerlegung immer einer minimalen Überdeckung entspricht.

**Definition 4.4** (Graph einer zulässigen Zerlegung in pseudostabile Mengen). Zu einer Zerlegung  $P$  eines Graphen  $G$  in pseudostabile Mengen sei  $G_P = (V, E)$  der ungerichtete Graph dieser Zerlegung mit

- $V = \{1, \dots, M\}$  wobei  $M$  die Anzahl aller stabilen Teilmengen  $D_j^i$  aller pseudostabilen Mengen  $P_i \subset P$  ist, d. h.  $M = \sum_{r=1}^{|P|} m_r$  und
- $(r, r+1) \in E$  wenn es einen Pfad  $p_r^i$  gibt, der  $D_r^i$  und  $D_{r+1}^i$  verbindet.

**Beispiel 4.1.** Man betrachte einen Graphen  $G = (V, E) = K_3$  mit  $V = \{u, v, w\}$ , also einen Graphen, der aus einem Dreieck besteht. Es sei  $B(G) = G$ . Alle Knoten des Graphen und alle Kanten sind blau. Eine Zerlegung dieses Graphen in stabile Mengen benötigt offensichtlich drei stabile Mengen. Er lässt sich aber auch in eine Menge gemäß der Definition 3.10 zerlegen: Jeweils drei Knoten des Dreiecks bilden einen Pfad, z. B.  $p = [u, v, w]$ . Nun sei  $D_1 = \{u\}$ ,  $D_2 = \{w\}$  und  $Pf = \{p\}$ . Der zugehörige Graph  $G_P$  besteht aus zwei Knoten,  $D_1$  und  $D_2$  und einer Kante zwischen diesen für den Pfad  $p$ . △

**Lemma 4.5.**  $P$  ist eine zulässige Zerlegung eines Graphen  $G$  in pseudostabile Mengen genau dann, wenn  $G_P$  azyklisch ist.

*Beweis.* Ist  $P$  eine zulässige Zerlegung in pseudostabile Mengen, so gilt für jede pseudostabile Teilmenge  $P_i \subset P$  mit  $m_i$  stabilen Mengen  $D_j^i$ , dass es genau  $m_i - 1$  Pfade gibt. Da es zwischen zwei stabilen Mengen

nur einen Pfad geben kann und dementsprechend auch nur zwischen zwei verschiedenen Knoten in  $G_P$ , kann kein Zyklus in  $G_P$  existieren.

Umgekehrt, existiert in  $G_P$  ein Zyklus  $z = [a_1, a_2, \dots, a_m, a_1]$  so ist dieser Teil einer pseudostabilen Teilmenge  $P_i \subset P$ . Diese Menge muss mindestens  $|z|$  stabile Teilmengen  $D_j^z$  beinhalten und damit auch Knoten in  $G_P$ . Allerdings gibt es nur  $|z| - 1$  Knoten, die im Zyklus verbunden werden. Dies widerspricht der Voraussetzung. Aber auch eine zusätzliche stabile Menge würde der Voraussetzung widersprechen.  $P$  ist also keine zulässige Zerlegung des Graphen  $G$  in pseudostabile Mengen.  $\square$

**Definition 4.6** (Abkürzungsfreier Pfad). Ein Pfad heißt abkürzungsfrei, wenn keine zwei Knoten dieses Pfades durch eine Kante verbunden sind, die nicht schon Teil des Pfades ist.

**Lemma 4.7.** *Ist  $P$  eine zulässige Zerlegung eines Graphen  $G$  in pseudostabile Mengen, dann beinhaltet  $G_P$  nur abkürzungsfreie Pfade.*

*Beweis.* Es sei  $P$  eine zulässige Zerlegung eines Graphen  $G$  in pseudostabile Mengen und  $G_P$  der zugehörige Graph dieser Zerlegung.

Angenommen,  $G_P$  beinhaltet einen Pfad, der nicht abkürzungsfrei ist, so gibt es also eine Kante, die zwei Knoten  $j$  und  $k \in V(G_P)$  verbindet, wobei  $|j - k| > 1$ . Dies ist ein Widerspruch dazu, dass  $P$  eine zulässige Zerlegung von  $G$  in pseudostabile Mengen ist, denn diese Kante impliziert einen Pfad zwischen  $D_j$  und  $D_k$  der nach Voraussetzung nicht zulässig ist.  $\square$

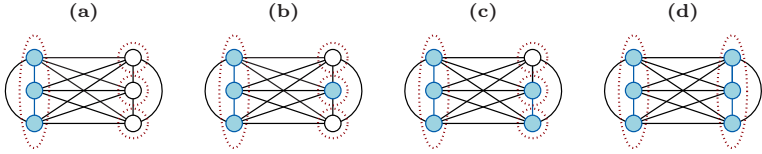
Lemma 4.7 folgt allerdings auch direkt aus Lemma 4.5.

**Bemerkung 4.8.** Es dürfen ganz offensichtlich keine Mehrfachkanten in einem Graphen  $G_P$  zu einer zulässigen Zerlegung  $P$  eines Graphen  $G$  in pseudostabile Mengen existieren.

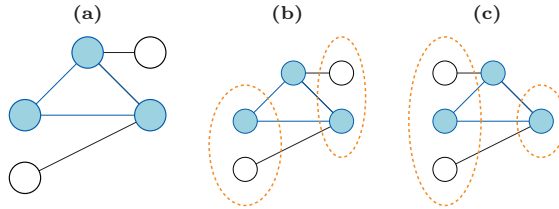
Es sei noch gemäß Kapitel 3 auf die entsprechende Version von minPS bzw. E-minPS hingewiesen, die die azyklische Eigenschaft von  $G_P$  vernachlässigen und mit minPS-a bzw. E-minPS-a bezeichnet werden.

**Beispiel 4.2.** Es sei  $G = K_6$  mit  $V(G) = \{u, v, w, x, y, z\}$ . Es seien  $u, v$  und  $w$  blaue Knoten die mit blauen Kanten verbunden sind, also  $V(B) = \{u, v, w\}$  und  $E(B) = \{(u, v), (v, w)\}$ . Dann gibt es eine minimale Zerlegung  $P$  von  $G$  in pseudostabile Mengen mit Wertigkeit  $\zeta(G) = 5$ , vergleiche Abb. 4.1(a).

Ist zusätzlich ein isolierter blauer Knoten  $y$  in  $B(G)$ , so ist immer noch  $\zeta(G) = 5$ , vergleiche Abb. 4.1(b).



**Abbildung 4.1:** Skizzen zu Beispiel 4.2.  $G = K_6$  mit  $V(G) = \{u, v, w, x, y, z\}$  und  $V(B) = \{u, v, w\}$  und  $E(B) = \{(u, v), (v, w)\}$ .



**Abbildung 4.2:** Darstellung der Graphen aus Beispiel 4.3.

Ist zusätzlich ein isolierter Pfad der Länge zwei vorhanden, es gibt also zwei Knoten  $y$  und  $z$  zusätzlich in  $B(G)$  und den Pfad  $(y, z)$  zusätzlich in  $B(G)$ , so ist immer noch  $\zeta(G) = 5$ . Diese minimale Zerlegung kann auf zwei verschiedene Weisen konstruiert werden. Vgl. dazu Abb. 4.1(c) und 4.1(d).  $\triangle$

Eine weitere Frage ist, wie die Knoten von  $G \setminus Pf'$  einer minimalen Zerlegung von  $G$  in pseudostabile Mengen in die stabilen Mengen  $D_1, \dots, D_n$  verteilt werden. Diese Verteilung ist nicht notwendigerweise eindeutig.

**Beispiel 4.3.** Man betrachte den Graphen in Abb. 4.2(a). Die chromatische Zahl dieses Graphen ist drei, man kann diese Graphen allerdings auch in pseudostabile Mengen zerlegen, bspw. mit Wertigkeit zwei, etwa in (b). Ebenso ist aber auch eine Zerlegung wie in (c) möglich. Eine minimale Zerlegung in pseudostabile Mengen impliziert also nicht notwendigerweise eine eindeutige Zuordnung der Knoten  $G \setminus Pf'$  in die stabilen Mengen  $D_1, \dots, D_n$ .  $\triangle$

## 4.2 minPS ist $\mathcal{NP}$ -vollständig

Um zu zeigen, dass minPS  $\mathcal{NP}$ -vollständig ist, wird im Folgenden zunächst das dynamisch gewichtete Mengenüberdeckungsproblem (DGMÜP) betrachtet und gezeigt, dass dieses  $\mathcal{NP}$ -vollständig ist.

Danach wird minPS als Mengenüberdeckungsproblem näher untersucht. Auch diese Formulierung ist  $\mathcal{NP}$ -vollständig, was sich genauso wie für DGMÜP zeigen lässt. Das minPS  $\mathcal{NP}$ -vollständig ist wird im folgenden Kapitel indirekt gezeigt, da gezeigt wird, dass minPS äquivalent zum TMP ist. 2000 wurde von Dahlhaus, Horak, Miller und Ryan in [16] bewiesen, dass das TMP  $\mathcal{NP}$ -vollständig ist. Hier soll aber der allgemeine Beweis geführt werden.

### 4.2.1 DGMÜP

In Abschnitt 2.6 wurde das gewichtete Mengenüberdeckungsproblem betrachtet. Dabei wurde jeder Menge  $S_i$  ein Gewicht  $w(S_i)$  zugeordnet. Man kann die Gewichtsfunktion allerdings auch abhängig von der gerade gewählten Überdeckung  $C$  setzen.

**Definition 4.9** (Dynamisch gewichtetes Mengenüberdeckungsproblem). Gegeben sei eine Menge  $S$  mit einer Familie  $F = \{S_1, \dots, S_n\}$  von Teilmengen von  $S$ , also  $S_j \subset S$  für  $i = 1, \dots, n$ . Es sei  $C$  eine Teilmenge von  $F$ . Weiter existiere eine Gewichtsfunktion  $w$ , die jedem Element in  $F$  ein nicht negatives Gewicht  $w(S_i, C)$  in Abhängigkeit von  $C$  zuordnet:

$$w : \{S_1, \dots, S_n\} \times \{0, 1\}^n \rightarrow \mathbb{N} \cup \infty$$

Diese ist eingeschränkt auf Linearkombinationen, d. h.

$$w(S_i, g_1, \dots, g_n) = a_{i,0} + g_1 a_{i,1} + \dots + g_n a_{i,n} \quad (4.1)$$

Dabei wird  $C$  durch einen Vektor  $g \in \{0, 1\}^n$  repräsentiert mit  $g_i = 1$  wenn  $S_i \in C$  und  $g_i = 0$  wenn  $S_i \notin C$ .

**Optimierungsvariante:** Gesucht ist eine minimal gewichtete Menge  $C \subset F$  die  $S$  überdeckt.

**Entscheidungsvariante:** Gegeben ein  $k > 0$ : Existiert eine Menge  $C \subset F$  die  $S$  überdeckt und deren Gewicht nicht größer als  $k$  ist?

**Beispiel 4.4.** Es sei  $S = \{1, 2, 3, 4\}$  und  $F = \{S_1, S_2, S_3, S_4, S_5, S_6\}$  mit

$$\begin{aligned} S_1 &= \{1\} \\ S_2 &= \{2\} \\ S_3 &= \{1, 2\} \\ S_4 &= \{1, 3\} \\ S_5 &= \{3, 4\} \\ S_6 &= \{4\} \end{aligned}$$

und entsprechenden Gewichten

$$\begin{aligned} w(S_1) &= w(S_2) = w(S_4) = w(S_6) = 1 \\ w(S_3) &= w(S_5) = 1 + 50g_4 \end{aligned}$$

Es gibt also beispielsweise zwei Überdeckungen von  $S$  mit Gewicht 3:  $S_2 \cup S_4 \cup S_6$  und  $S_1 \cup S_2 \cup S_5$ . Die Überdeckung mit dem kleinsten Gewicht ist  $S_3 \cup S_5$ . Alle anderen Überdeckungen haben ein wesentlich höheres Gewicht, etwa  $S_3 \cup S_4 \cup S_5$  mit Gewicht 103.  $\triangle$

**Bemerkung 4.10.** Die Gewichtsfunktion (4.1) zu einem (E-)DGMÜP kann in der Form

$$w(S_i, C) = \begin{pmatrix} a_{1,0} & a_{1,1} & \cdots & a_{1,n} \\ a_{2,0} & a_{2,1} & \cdots & a_{2,n} \\ \vdots & & & \vdots \\ a_{n,0} & a_{n,1} & \cdots & a_{n,n} \end{pmatrix} \begin{pmatrix} 1 \\ g_1 \\ \vdots \\ g_n \end{pmatrix} e_i$$

dargestellt werden, also in Matrixschreibweise  $w(S_i, C) = A g e_i$  mit  $A \in \mathbb{R}^{(n+1) \times (n)}$ ,  $g \in \{0, 1\}^{n+1}$  und  $e_i$  als den  $i$ -ten Einheitsvektor.

Um zu beweisen, dass auch DGMÜP  $\mathcal{NP}$ -vollständig ist, kann man zunächst die entsprechenden Entscheidungsvarianten E-GMÜP und E-DGMÜP betrachten. Diese beantworten die Frage, ob es eine Überdeckung  $C$  mit einem Gewicht  $k > 0$  gibt.

**Lemma 4.11.** *Die Entscheidungsvariante des dynamisch gewichteten Mengenüberdeckungsproblems ist in  $\mathcal{NP}$ .*

*Beweis.* Man rate eine Menge  $C \subset F$  und überprüfe, ob diese eine Überdeckung von  $S$  ist und ob das Gewicht dieser Überdeckung gleich  $t$  ist. Dies kann in polynomieller Zeit durchgeführt werden.  $\square$



**Lemma 4.12.** *Die Entscheidungsvariante des dynamisch gewichteten Mengenüberdeckungsproblems ist  $\mathcal{NP}$ -schwer.*

*Beweis. Zu zeigen:* E-GMÜP  $\preceq_p$  E-DGMÜP

Gegeben sei nun eine Instanz des E-GMÜP mit einer Menge  $S'$  mit einer Familie  $F' = \{S'_1, \dots, S'_n\}$  von Teilmengen von  $S'$  und einer Gewichtsfunktion  $w'(S_i)$ .

Daraus kann nun eine Instanz eines E-DGMÜP konstruiert werden.

Diese bestehe aus einer Menge  $S = S'$  mit einer Familie  $F = F' = \{S'_1, \dots, S'_n\}$  von Teilmengen von  $S$  und einer Gewichtsfunktion  $w(S_i, C) = \text{Age}_i$  mit

$$A = \begin{pmatrix} w'(S_1) & 0 & \cdots & 0 \\ w'(S_2) & 0 & \cdots & 0 \\ \vdots & & & \vdots \\ w'(S_n) & 0 & \cdots & 0 \end{pmatrix}$$

also  $w(S_i, C) = w'(S'_i)$ .

Es ist klar, dass diese Gewichtsfunktion in polynomieller Zeit aus  $w'(S_i)$  erzeugt werden kann.

Zu zeigen ist nun: Es existiert eine gewichtete Mengenüberdeckung mit Gewicht  $k$  genau dann, wenn es eine dynamisch gewichtete Mengenüberdeckung mit Gewicht  $k$  gibt.

„ $\Rightarrow$ “ Es sei  $G'$  eine Überdeckung der Menge  $S$  mit Gewicht  $k$ , also antwortet E-GMÜP mit „ja“. Dann existiert auch eine dynamisch gewichtete Überdeckung  $C'$  mit Gewicht  $k$ , also antwortet auch E-DGMÜP mit „ja“. Dies gilt, da die Einzelgewichte nicht von der gewählten Überdeckung  $C'$  abhängen.

„ $\Leftarrow$ “ Es sei  $C$  eine dynamisch gewichtete Überdeckung der Menge  $S'$  mit Gewicht  $t$ , also antwortet E-DGMÜP mit „ja“. Dann existiert mit  $C$  auch eine gewichtete Mengenüberdeckung mit Gewicht  $k$ , E-GMÜP antwortet also auch „ja“. Dies gilt, da  $w(S_i, C) = w'(S'_i)$ .  $\square$

Da DGMÜP das zu E-DGMÜP polynomiell äquivalente Optimierungsproblem ist, folgt daraus auch, dass DGMÜP  $\mathcal{NP}$ -vollständig ist.

**Lemma 4.13.** *DGMÜP ist polynomiell äquivalent zu E-DGMÜP*

*Beweis.* „ $\Leftarrow$ “ Gegeben sei eine Menge  $S$  mit einer Familie  $F = \{S_1, \dots, S_n\}$  von Teilmengen von  $S$ , also  $S_j \subset S$  für  $i = 1, \dots, n$ . Es sei  $C$  eine Teilmenge von  $F$ . Weiter existiere eine Gewichtsfunktion  $w$ , die jedem Element in  $F$  ein nicht negatives Gewicht  $w(S_i, C)$  zuordnet. Es sei

$$m = \sum_{i=1}^n w(S_i, \mathbb{I})$$

mit dem Einsvektor  $\mathbb{I}$ . Man bestimme für alle  $k \in \{1, \dots, m\}$  E-DGMÜP( $k$ ), und damit ein minimales  $k^*$ , für das E-DGMÜP( $k^*$ ) mit ja antwortet. Dies kann in linearer Zeit geschehen.

„ $\Rightarrow$ “ Gegeben sei eine Menge  $S$  mit einer Familie  $F = \{S_1, \dots, S_n\}$  von Teilmengen von  $S$ , also  $S_j \subset S$  für  $i = 1, \dots, n$ . Es sei  $C$  eine Teilmenge von  $F$ . Weiter existiere eine Gewichtsfunktion  $w$ , die jedem Element in  $F$  ein nicht negatives Gewicht  $w(S_i, C)$  zuordnet. Es sei  $C$  nun eine minimal gewichtete Überdeckung mit Wertigkeit  $k^*$ . Für jedes Element  $S_i \in C$  betrachte alle Elemente  $S_j \subset S_i$ . Für  $k$ , der Wertigkeit der Überdeckung, die  $S_j$  statt  $S_i$  verwendet, gibt es eine Überdeckung  $k \geq k^*$ . Für alle diese Werte antwortet E-DGMÜP mit ja. Dies kann in quadratischer Laufzeit geschehen.  $\square$

Damit folgt

**Theorem 4.14.** *DGMÜP ist  $\mathcal{NP}$ -vollständig.*

## 4.2.2 minPS

In Definition 3.6 wurde minPS als Mengenüberdeckungsproblem gegeben. Dies ist äquivalent zu der Formulierung als Graphenzerlegungsproblem.

Man betrachte nun zu einem Graphen  $G = (V, E)$  mit blauen Knoten und Kanten  $B \subset G$  die folgenden Mengen:

$$S = V(G)$$

$$\mathfrak{T} = \{T_1, \dots, T_n\}$$

die Menge aller pseudostabilen Teilmengen von  $G$  mit  $B$ . Es sei nun  $F = \mathfrak{T} = \{T_1, \dots, T_n\}$ . Es sei  $C \subset F$  mit einer Repräsentation  $g \in \{0, 1\}^n$  als Vektor, wobei  $g_i = 1$  wenn  $T_i \in C$  und  $g_i = 0$  wenn  $T_i \notin C$ .

Es sei nun

$$w(S_i, C) \rightarrow \mathbb{N}^0 \tag{4.2}$$

$$w(S_i, g_1, \dots, g_n) = \sum_{j=1}^n g_j \left( \zeta(T_j) + \sum_{k=1}^n t_k w_{j,k} \right)$$

wobei  $w_{i,j}$  gemäß 3.2 als

$$w_{i,j} = \begin{cases} -1 & S_i \cap S_j = D_1^i = D_2^j \\ 0 & \text{sonst} \end{cases} \quad (4.3)$$

gegeben ist.

**Optimierungsvariante:** Gesucht ist eine minimal gewichtete Menge  $C \subset F$  die  $S$  überdeckt.

**Entscheidungsvariante:** Gegeben ein  $k > 0$ : Existiert eine Menge  $C \subset F$  die  $S$  überdeckt und deren Gewicht nicht größer als  $k$  ist?

minPS-M ist offensichtlich eine Instanz von DGMÜP. Dies liefert jedoch keine Erkenntnisse zur Komplexität von minPS-M. Der später folgende Reduktionsbeweis ist jedoch an das Vorgehen im vorherigen Kapitel angelehnt.

**Bemerkung 4.15.** Erneut kann die Gewichtsfunktion in Matrixschreibweise dargestellt werden:

$$w(S_i, C) = \begin{pmatrix} a_{1,0} & a_{1,1} & \cdots & a_{1,n} \\ a_{2,0} & a_{2,1} & \cdots & a_{2,n} \\ \vdots & & & \vdots \\ a_{n,0} & a_{n,1} & \cdots & a_{n,n} \end{pmatrix} \begin{pmatrix} 1 \\ g_1 \\ \vdots \\ g_n \end{pmatrix} e_i$$

Dabei ist  $a_i, 0 = 0 \forall i \in \{1, \dots, n\}$  als Basisgewicht und

$$a_{i,j} = \zeta(T_i) + \sum_{k=1}^n t_k w_{j,k}$$

in allen anderen Fällen.

Genauso wie in Abschnitt 4.2.1 kann nun gezeigt werden, dass E-minPS  $\mathcal{NP}$ -vollständig ist. Klar ist zunächst

**Lemma 4.16.** *E-minPS(t) ist in  $\mathcal{NP}$ .*

*Beweis.* Man rate eine Menge  $C \subset F$  und überprüfe, ob diese eine Überdeckung von  $S$  ist und ob das Gewicht dieser Überdeckung gleich  $t$  ist. Dies kann in polynomieller Zeit durchgeführt werden.  $\square$

**Lemma 4.17.** *E-minPS( $t$ ) ist  $\mathcal{NP}$ -schwer.*

*Beweis.* Dieser Beweis funktioniert fast genauso wie der Beweis zu Lemma 4.12, lediglich das E-MÜP statt E-GMÜP betrachtet wird.

**Zu zeigen:** E-MÜP( $t$ )  $\preceq_p$  E-MPS( $t$ )

Gegeben sei nun eine Instanz des E-MÜP mit einer Menge  $S'$  und einer Familie  $F' = \{S'_1, \dots, S'_n\}$  von Teilmengen von  $S'$ . Eine Gewichtsfunktion  $w$  kann  $w'(S_i) \equiv 1$  gesetzt werden.

Daraus kann nun eine Instanz eines E-MPS-M konstruiert werden.

Diese bestehe aus einer Menge  $S = S'$  mit einer Familie  $F = F' = \{S'_1, \dots, S'_n\}$  von Teilmengen von  $S$  und einer Gewichtsfunktion  $w(S_i, G) = \text{Age}_i$  mit

$$A = \begin{pmatrix} 0 & w'(S_1) & \cdots & 0 & 0 \\ 0 & 0 & w'(S_2) & \cdots & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & \cdots & 0 & w'(S_n) \end{pmatrix} = \begin{pmatrix} 0 & 1 & \cdots & 0 & 0 \\ 0 & 0 & 1 \cdots & 0 & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \end{pmatrix}$$

also  $w(S_i, G) = w'(S'_i) \equiv 1$ .

Dies entspricht also einer Instanz von E-minPS auf einem vollständigen Graphen  $G = K_n$  mit  $B = \emptyset$ , also keinen blauen Knoten und Kanten. Dann ist  $\mathfrak{T}$  eine Menge, die nur aus stabilen Mengen besteht, bzw. aus pseudostabilen Tupeln  $T_i$  bei denen die zweite stabile Menge  $D_j^i = \emptyset$  ist.

Es ist klar, dass diese Gewichtsfunktion in polynomieller Zeit aus  $w'(S_i) \equiv 1$  erzeugt werden kann.

Zu zeigen ist nun: Es existiert eine MÜP mit Gewicht  $k$  genau dann, wenn es eine Lösung von minPS mit Gewicht  $k$  gibt.

„ $\Rightarrow$ “ Es sei  $B$  nun eine Überdeckung der Menge  $S$  mit Gewicht  $k$  (bzw.  $k$  Mengen), also eine Lösung des E-MÜP. Dann ist  $B$  auch eine Lösung des E-minPS mit Gewicht  $k$ , da dessen Einzelgewichte nicht von der gewählten Überdeckung  $B$  abhängen.

„ $\Leftarrow$ “ Es sei  $C$  nun eine Überdeckung der Menge  $S'$  mit Gewicht  $k$ , also eine Lösung des E-minPS. Dann ist  $C$  auch eine Lösung des E-MÜP mit Gewicht  $k$ , da  $w(S_i, G) = w'(S'_i) \equiv 0$  gilt.  $\square$

Genauso wie Lemma 4.13 kann das folgende Lemma bewiesen werden:

**Lemma 4.18.** *E-minPS und minPS sind polynomiell äquivalent.*

Damit ist minPS  $\mathcal{NP}$ -vollständig. Genauso kann auch der Beweis für alle anderen Varianten minMPS, minPS' etc. geführt werden.

### 4.3 Abschätzungen für minPS

Da minPS  $\mathcal{NP}$ -vollständig ist, kann ein Zugang zu diesem Problem eine Abschätzung sein. Es werden nun verschiedene Ansätze diskutiert.

#### 4.3.1 Abschätzung über Cliques

Zunächst werden vollständige Graphen  $K_n$  betrachtet. Ein erster Ansatz ist, die Anzahl der blauen Knoten in die Abschätzung mit einzubeziehen:

**Lemma 4.19.** *Für die Zerlegung  $P$  eines vollständigen Graphen  $K_n$  mit  $n^b$  blauen Knoten,  $n \geq n^b \geq 3$ ,  $n > 2$  in pseudostabile Mengen gilt stets*

$$\zeta(K_n) \geq n - \max\left\{0, \left\lceil \frac{n^b}{3} \right\rceil\right\} \quad (4.4)$$

*Beweis.* Es sei  $G = K_n$  ein vollständiger Graph mit  $n^b$  blauen Knoten und nur blauen Kanten,  $n \geq n^b \geq 0$ ,  $n > 2$ .

Ist kein Knoten blau, so gilt  $\zeta(G) = n$ .

Gibt es nur einen oder zwei blaue Knoten, gilt ebenso  $\zeta(G) = n$ , da kein Pfad gebildet werden kann.

Gibt es drei durch blaue Kanten verbundene blaue Knoten, so kann dies ein Pfad sein, die Wertigkeit verringert sich um eins. Für je zwei weitere blaue Knoten kann ein weiterer Pfad hinzugefügt werden.

Es gilt also immer die Gleichung 4.4. □

**Lemma 4.20.** *Für die Zerlegung  $P$  eines vollständigen Graphen  $K_n$  mit  $e^b$  blauen Kanten,  $n > 2$  in mehrfach pseudostabile Mengen gilt stets*

$$\zeta(K_n) \geq n - \max\left\{\left\lceil \frac{n+1}{2} \right\rceil, \left\lceil \frac{e^b}{2} \right\rceil\right\} \quad (4.5)$$

*Beweis.* Der linke Teil des Maximums folgt aus Lemma 3.29. Der rechte Teil folgt aus Lemma 4.19. Die Behauptung ergibt sich aus der Kombination beider Abschätzungen.  $\square$

Diese beiden Schranken können nun auch kombiniert werden, um ein besseres Resultat zu erhalten.

**Theorem 4.21.** *Für die Zerlegung  $P$  eines vollständigen Graphen  $K_n$  mit  $e^b$  blauen Kanten und mit  $n^b$  blauen Knoten,  $n \geq n^r \geq 0$ ,  $n > 2$  in mehrfach pseudostabile Mengen gilt stets*

$$\zeta(K_n) \geq n - \min \left\{ \max \left\{ \left\lceil \frac{n+1}{2} \right\rceil, \left\lfloor \frac{e^b}{2} \right\rfloor \right\}, \max\{0, n^b - 2\} \right\}$$

Nun kann eine untere Schranke konstruiert werden. Dazu seien zunächst alle maximalen Cliques eines Graphen  $G$  entsprechend ihrer Größe angeordnet, also  $C_1, \dots, C_m$  mit  $|C_i| \geq |C_{i+1}|$ . Es sei  $b_i$  die Anzahl von blauen Knoten in der Clique  $C_i$ . Dann sei

$$C'_i = C_i \setminus \bigcup_{j=1}^{i-1} C_j$$

womit die Abbildung  $p : \mathbb{N} \rightarrow \mathbb{N}^0$  definiert werden kann mit

$$p(i) = \max \left\{ 0, |C'_i| - \min \left\{ \max \left\{ \left\lceil \frac{|C'_i| + 1}{2} \right\rceil, \left\lfloor \frac{e^b(C'_i)}{2} \right\rfloor \right\}, \max \{0, n^b(C_i) - 2\} \right\} - \sum_{j=1}^{i-1} p(j) \right\} \quad (4.6)$$

und

$$p(1) = |C_1| - \min \left\{ \max \left\{ \left\lceil \frac{|C_1| + 1}{2} \right\rceil, \left\lfloor \frac{e^b(C_1)}{2} \right\rfloor \right\}, \max\{0, n^b(C_1) - 2\} \right\}$$

Dann sei weiter

$$l_G(G) = \sum_{i=1}^m p(i) \quad (4.7)$$

**Lemma 4.22.** Für einen Graphen  $G = (V, E)$  mit mehr als zwei Knoten und mehr als einer Kante sowie blauen Knoten und Kanten gilt für  $l_C(G)$  immer

$$\zeta(G) \geq l_C(G)$$

*Beweis.* Man wendet sukzessive Theorem 4.21 an. □

Die Schranke  $l_C(G)$  aus Lemma 4.22 kann auch genutzt werden, wenn nicht alle Summanden berechnet werden. Insbesondere gilt für einen Graphen  $G = (V, E)$  mit blauen Knoten und Kanten und einer beliebigen Clique  $C \subset G$

$$\zeta(G) \geq |C| - \max\left\{0, \left\lceil \frac{b_C}{3} \right\rceil\right\}$$

wobei  $b_C$  die Anzahl von roten Knoten in der Clique  $C$  ist.

### 4.3.2 Abschätzung über den Subgraphen $B$

Zunächst folgt im Rückblick auf Lemma 3.29, dass die Abschätzung

$$\zeta(K_i) \geq \left\lceil \frac{i+1}{2} \right\rceil$$

lieferte, das folgende Lemma:

**Lemma 4.23.** Für die Zerlegung  $P$  eines Graphen  $G$  mit  $\chi(G) \geq 2$  und  $|V(G)| \geq 2$  in pseudostabile Mengen gilt

$$\zeta(G) \geq \left\lceil \frac{\chi(G)+1}{2} \right\rceil = l_\chi(G) \tag{4.8}$$

*Beweis.* Es sei  $G^B = G' = (V(G), E(G))$  mit  $B = G$  ein vollständiger blauer Graph mit  $\chi(G) > 2$ . Es gilt offensichtlich, dass  $\zeta(G') \leq \zeta(G)$ .

Ist  $|E(G')| = 2$  so gilt die  $\zeta(G) = \chi(G) = 2$ .

Man wähle nun  $\left\lceil \frac{\chi(G)+1}{2} \right\rceil$  beliebige blaue Knoten aus  $G'$  und setze diese als stabile Mengen  $D_j$ . Alle übrigen Knoten  $v_k$  in  $G$  können nun, da alle Kanten und alle Knoten blau sind, entweder als Pfad zwischen diesen stabilen Mengen genutzt werden oder bilden weitere stabile Mengen. Im letzten Fall erhöht sich allerdings die Wertigkeit der Zerlegung. Für eine

untere Schranke soll deswegen angenommen werden, dass dieser Vorgang für alle Knoten möglich ist.

Für jeden Knoten in  $G$  der im Gegensatz zu dem selben Knoten in  $G'$  nicht blau ist oder jede Kante die nicht blau ist muss nun also eine weitere stabile Menge genutzt werden. Es gilt also immer

$$\zeta(G) \geq \left\lceil \frac{\chi(G) + 1}{2} \right\rceil$$

□

Es sei  $G$  ein Graph mit  $n$  Knoten, von denen  $n^b$  blau sind. Dann sei

$$\chi'(G) = \chi(G) - \left\lfloor \frac{1}{3}n^b \right\rfloor$$

Diese Funktion zieht also von einer minimalen Färbung für je drei blaue Knoten in  $G$  eine Farbe ab.

**Lemma 4.24.** *Es sei  $G$  ein Graph mit  $n > 1$  Knoten, von denen  $n^b$  blau sind und mindestens einer Kante. Es gilt für eine Zerlegung in pseudostabile Mengen immer*

$$l_R = \max \left\{ \left\lfloor \frac{\chi(G)}{2} \right\rfloor, \chi'(G) \right\} \leq \zeta(G) \quad (4.9)$$

*Beweis.* Zum rechten Teil des Maximums siehe Beweis zu Lemma 6.13. Hier soll der linke Teil des Maximums gezeigt werden.

Es sei  $P$  eine minimale Zerlegung des Graphen  $G$  in pseudostabile Mengen. Ist  $\zeta(G) < \chi(G)$  gibt es Pfade zwischen stabilen Mengen. Es muss jeder Pfad in zwei verschiedenen stabilen Mengen  $D_i$  und  $D_j$   $i \neq j$  enden, und da die Anzahl der stabilen Mengen in  $P$  sicher kleiner als  $\chi(G)$  ist, gilt

$$\chi(G) - \zeta(G) \leq \frac{\chi(G)}{2} \quad (4.10)$$

$$\chi(G) - \frac{\chi(G)}{2} \leq \zeta(G) \quad (4.11)$$

$$\frac{\chi(G)}{2} \leq \zeta(G) \quad (4.12)$$

Es gilt also stets Gleichung 4.9. □



Weiter gilt das folgende Lemma:

**Lemma 4.25.** *Es sei  $G$  ein Graph mit  $n > 1$  Knoten, von denen  $n^r$  rot sind und mindestens einer Kante. Es gilt für eine Zerlegung in mehrfach pseudostabile Mengen immer*

$$l_R = \max\{l_\chi(G), \chi'(G)\} \leq \zeta(G)$$

$$\text{mit } \chi'(G) = \chi(G) - \max\left\{0, \left\lceil \frac{n^r}{3} \right\rceil\right\}$$

*Beweis.* Der rechte Teil des Maximums wurde bereits im vorherigen Abschnitt gezeigt. Der linke Teil folgt direkt aus Lemma 4.23.  $\square$

Da die chromatische Zahl eines Graphen  $G$  nicht immer in polynomieller Zeit berechnet werden kann, ist es oft sinnvoll, diese in  $l_R$  durch eine untere Schranke zu ersetzen.

Eine weitere Verbesserung kann erfolgen, wenn man statt der Anzahl von blauen Knoten nur die Anzahl der blauen Knoten betrachtet, die Teil eines Pfades gemäß der Annahmen aus dem vorherigen Kapitel sind, d. h. der Knoten  $v$ , die Teil eines Pfades  $[a, b, c]$  von blauen Knoten, die mit blauen Kanten verbunden sind und deren mittlerer Knoten  $b$  Teil eines Pfades ungerader Länge in  $G$  ist, sind. Diese Schranken werden mit  $\tilde{l}_R$  bezeichnet.

Um dieses aufwändige Verfahren ein wenig zu beschleunigen, kann eine Abschätzung über blaue Kanten erfolgen:

**Lemma 4.26.** *Es sei  $G$  ein Graph mit  $n > 1$  Knoten und  $e > 1$  Kanten, von denen  $e^b$  blau sind. Es gilt für eine Zerlegung in pseudostabile Mengen immer*

$$l_B = \max\left\{l_\chi(G), \chi(G) - \left\lfloor \frac{e^b}{2} \right\rfloor\right\} \leq \zeta(G) \quad (4.13)$$

*Beweis.* Siehe Beweis zu Lemma 6.16 für die Ungleichung

$$\max\left\{2, \chi(G) - \left\lfloor \frac{e^b}{2} \right\rfloor\right\} \leq \zeta(G)$$

Mit Lemma 4.23 folgt dann

$$\max\left\{l_\chi(G), \chi(G) - \left\lfloor \frac{e^b}{2} \right\rfloor\right\} \leq \zeta(G)$$

$\square$

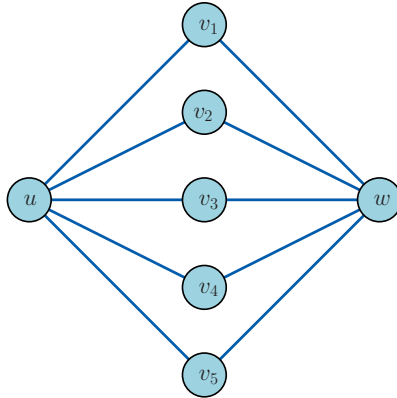


Abbildung 4.3: Skizze eines großen Diamanten.

**Definition 4.27.** Die unteren Schranken  $l_R$  und  $l_B$  aus Lemma 4.25 und 4.26 können nun zu  $l_{RB}(G)$  zusammengefasst werden:

$$l_{RB}(G) = \min \{l_B(G), l_R(G)\}$$

#### 4.4 minPS auf disjunkten großen Diamanten

Es sei  $G = (V, E)$  ein chordaler Graph. Der Graph  $B \subset G$  bestehe nur aus disjunkten, großen Diamanten. Vgl. hierzu auch Abb. 4.3.

**Definition 4.28** (Großer Diamant). Eine Knotenmenge  $u, v_1, \dots, v_n, w \subset B$  heißt *großer Diamant*, wenn es nur Kanten  $(u, v_i)$  und  $(v_i, w)$  mit  $i \in \{1, \dots, n\}$  in  $B$  gibt und für alle  $v_i$  gilt, dass  $v_i \subset K_3 \subset G$  und für  $u$  und  $w$ , dass  $u \not\subset K_3 \subset G$  und  $w \not\subset K_3 \subset G$ .

Auf einem solchen Graphen kann minPS in polynomieller Zeit gelöst werden. Dazu wird zunächst mit Hilfe des Algorithmus aus Abschnitt 2.2 ein perfektes Eliminationsschema auf dem Graphen  $G$  berechnet. Dies geht in linearer Zeit. Mit Hilfe eines perfekten Eliminationsschemas  $[v_1, v_2, \dots, v_n]$  können Graphen in linearer Zeit gefärbt werden, siehe Abschnitt 2.2. Dieser Algorithmus kann wie folgt angepasst werden:

- In jedem Schritt wird absteigend einem  $v_n$  aus dem perfekten Eliminationsschema eine minimale Farbe zugeordnet. Es kann überprüft

werden, ob an dieser Stelle eine neue, noch nicht benutzte Farbe verwendet wird. Dann gibt es zwei Möglichkeiten:

- $v_n$  ist ein mittlerer Knoten aus einem großen Diamanten. d. h. der Knoten kann Teil eines Pfades werden und die neue Farbe muss nicht genutzt werden. Die dazu gehörigen Pfade  $u$  und  $w$  müssen entweder eine andere Farbe haben oder ungefärbt sein, wobei ihnen dann eine minimale Farbe zugewiesen wird. Dazu muss überprüft werden, ob dies wirklich eine zulässige Zerlegung in pseudostabile Mengen ist. Dazu kann der Graph  $G_P$  der Zerlegung generiert werden. Alle Farben entsprechen stabilen Mengen  $D$  und die Pfade werden in einer Liste  $Pf$  gespeichert.
- Ist der Graph  $G_P$  azyklisch, so ist eine zulässige Zerlegung in pseudostabile Mengen gefunden und es wurde vermieden, eine weitere Farbe nutzen zu müssen.

Siehe Algorithmus 2 für Pseudocode.

**Lemma 4.29.** *Algorithmus 2 liefert eine gültige Zerlegung des chordalen Graphen  $G$  und  $B \subset G$  mit disjunkten großen Diamanten in pseudostabile Mengen.*

*Beweis.* Ist  $Pf = \emptyset$ , so hat Algorithmus 2 eine Zerlegung in stabile Mengen gefunden. Dies ist eine gültige Zerlegung in pseudostabile Mengen.

Ist  $Pf \neq \emptyset$ , so stellt Zeile 7 sicher, dass der Graph  $G_P$  azyklisch ist. Somit ist auch dies eine gültige Zerlegung in pseudostabile Mengen.  $\square$

**Lemma 4.30.** *Der Algorithmus 2 liefert eine Zerlegung  $P$  des chordalen Graphen  $G$  und  $B \subset G$  mit disjunkten großen Diamanten in pseudostabile Mengen. Es gibt keine Zerlegung  $P_2 \neq P$  mit  $\zeta(P_2) < \zeta(P)$ .*

*Beweis.* Angenommen, es existiert eine Zerlegung  $P_2 \neq P$  mit  $\zeta(P_2) < \zeta(P)$ .

Dann existiert mindestens ein Knoten  $n \in G$  für den entweder gilt  $f_2(n) < f(n)$  oder  $n \in Pf_2$ , aber  $n \notin Pf$ .

- Ist  $f_2(n) < f(n)$ , so muss

$$\max_{v \in N(n)} f_2(v) < \max_{v \in N(n)} f(v)$$

---

**Algorithmus 2** Coloring-PS

---

**Eingabe:** perfektes Eliminationsschema  $[v_1, v_2, \dots, v_n]$  des chordalen Graphen  $G$ ,  $B \subset G$  mit disjunkten großen Diamanten

**Ausgabe:** minPS  $f : V \rightarrow \{1, \dots, \zeta(G)\}$ ,  $Pf$

```
|1|  $f(v) = 0$  für alle Knoten.  
|2|  $\max f = 0$   
|3|  $G_P = \emptyset$ ,  $Pf = \emptyset$   
|4| for  $i = n, \dots, 1$  do  
|4|   Wähle für  $f(v_n)$  die kleinste freie Farbklassse.  
|5|   if  $(v_n) > \max f$  then  
|6|     if  $v_n$  ist ein mittlerer Knoten  $v$  in einem großen Diamanten in  $B$   
|6|     und ( $f(u) \neq f(w)$  oder  $f(u) = 0$ ) then  
|6|        $Pf \leftarrow (u, v_n, w)$   
|6|       Sind  $u$  oder  $w$  noch nicht gefärbt, wähle die kleinste freie Farb-  
|6|       klasse, so dass  $G_P$  azyklisch bleibt unter der Bedingung, dass  
|6|        $f(u)$  und  $f(w) < \max f$ .  
|6|       baue  $G_P$   
|7|       if  $G_P$  ist azyklisch then  
|7|          $f(v_n) = -1$   
|8|       else  
|8|         entferne  $(u, v_n, w)$  aus  $Pf$  und setze  $f(u), f(w)$  auf ihre Ur-  
|8|         sprungsfarbe zurück  
|9|       end if  
|10|     end if  
|11|   end if  
|11|   setze  $\max f$   
|12| end for  
|13| return  $f, Pf$ 
```

---

sein. Das heißt in der Nachbarschaft von  $n$  existiert wieder ein Knoten  $v \in N(n)$  mit  $f_2(v) < f(v)$ . Ist kein  $v \in N(n)$  in  $Pf_2$ , so betrachte sukzessive wieder alle  $v \in N(n)$ , denn es muss mindestens ein Knoten  $v$  geben für den gilt  $f_2(v) < f(v)$  oder  $v \in Pf_2$ .

- Es sei  $n$  also ein Knoten, für den ein  $w \in N(n)$  existiert mit  $w \in Pf_2$ , aber nicht  $w \in Pf$ . Der Knoten  $n$  ist nur zu  $\delta(n)$  Knoten benachbart und kann also maximal mit Farbe  $\delta(n)$  gefärbt werden. Wird diese Zahl in einer anderen Zerlegung kleiner, so muss der Knoten aus der Nachbarschaft als Farbe entfernt werden. Somit muss er Teil eines Pfades werden.

- Das heißt, es existiert auf jeden Fall mindestens ein Pfad mit einem Knoten  $v$ , der in  $Pf_2$  enthalten ist, aber nicht in  $Pf$ . Damit bleibt dieser Fall zu zeigen.

Es sei also  $n \in Pf_2$ , aber  $n \notin Pf$ . Damit gilt, dass  $n$  mittlerer Knoten  $v$  eines großen Diamanten in  $B$  ist und  $n \subset K_3 \subset G$ . Es sei  $[u, n, w] \in Pf_2$ .

Damit  $\zeta(P_2) < \zeta(P)$  gilt, muss in  $P_2$  entweder die Farbe von  $u$ ,  $n$ , oder  $w$  reduziert worden sein. Für  $u$  und  $w$  gilt, dass sie nicht Teil eines Kreises ungerader Länge in  $G$  sind, somit ist  $(u, w) \notin E(G)$ . Damit wurden sie vom Algorithmus 2 noch nicht gefärbt, da sie im perfekten Eliminationsschema nach dem Knoten  $n$  betrachtet werden. Somit ist die Bedingung in Zeile 6 erfüllt.

Ist  $P_2$  eine gültige Zerlegung in pseudostabile Mengen, so muss auch die Bedingung in Zeile 7 für  $P$  erfüllt sein. Damit muss  $n \in Pf$  sein. Jeder Pfad, der zur Minimierung von  $P_2$  beigetragen hat, muss auch in  $P$  als Pfad gewählt worden sein.  $\square$

Es folgt aus den beiden Lemmata also sofort:

**Theorem 4.31.** *minPS ist auf einem chordalen Graphen  $G$  und einem nur aus disjunkten, großen Diamanten bestehenden Graphen  $B \subset G$  in polynomialer Zeit lösbar.*

Wie gut die Ergebnisse des Algorithmus 2 auf beliebigen Graphen  $B \subset G$  sind, wird in einem späteren Abschnitt betrachtet.

## 4.5 minPS mit disjunkten Listen von Pfaden

Es wurde bereits bei der Definition von pseudostabilen Mengen in Kapitel 3 erwähnt, dass zur Konstruktion von pseudostabilen Mengen auch eine Liste von Pfaden statt eines blauen Teilgraphens  $B \subset G$  genutzt werden kann. Die entsprechenden Pfade werden sonst aus  $B$  generiert, beide Varianten sind also äquivalent. Um eine Heuristik herzuleiten, ist die Listendarstellung allerdings insofern günstiger, weil sie die Pfade eindeutig angibt. Ein Dreieck  $[a, b, c] \subset B$  beinhaltet drei Pfade. In einer Liste kann auch explizit nur einer der Pfade, beispielsweise  $(a, b, c)$  angegeben werden. Somit kann der Algorithmus aus dem vorhergehenden Abschnitt noch ergänzt werden.

Es sei  $G = (V, E)$  ein chordaler Graph. Die Liste  $Pf^* = \{(a, b, c) \subset G, b \subset K_3 \subset G\}$  von möglichen Pfaden sei paarweise disjunkt auf dem mittleren

Knoten. Dann kann der Algorithmus 2 aus dem vorherigen Abschnitt auch hier in polynomieller Zeit eine optimale Lösung für minPS auf  $G$  berechnen.

**Lemma 4.32.** *Algorithmus 2 liefert eine gültige Zerlegung des chordalen Graphen  $G$  mit einer Liste  $Pf^*$  von möglichen Pfaden, die auf dem mittleren Knoten paarweise disjunkt sei, in pseudostabile Mengen.*

*Beweis.* Ist  $Pf = \emptyset$ , so hat Algorithmus 2 eine Zerlegung in stabile Mengen gefunden. Dies ist eine gültige Zerlegung in pseudostabile Mengen.

Ist  $Pf \neq \emptyset$ , so stellt Zeile 7 sicher, dass der Graph  $G_P$  azyklisch ist. Somit ist auch dies eine gültige Zerlegung in pseudostabile Mengen.  $\square$

**Lemma 4.33.** *Der Algorithmus 2 liefert eine Zerlegung  $P$  des chordalen Graphen  $G$  mit einer Liste  $Pf^*$  von möglichen Pfaden in pseudostabile Mengen. Es gibt keine Zerlegung  $P_2 \neq P$  mit  $\zeta(P_2) < \zeta(P)$ .*

*Beweis.* Die Beweisführung erfolgt analog zum Beweis von Lemma 4.30.  $\square$

Es folgt aus den beiden Lemmata also sofort:

**Theorem 4.34.** *minPS ist auf einem chordalen Graphen  $G$  mit einer Liste  $Pf^*$  von möglichen Pfaden in polynomieller Zeit lösbar.*

Es kann an dieser Stelle beobachtet werden, dass das Problem grundsätzlich nur dann in polynomieller Zeit lösbar scheint, wenn die Pfade eindeutig zu bestimmen sind und es pro Knoten in  $G$  nicht mehr als einen Pfad gibt.

## 4.6 Lineare Programmierung

Grundsätzlich entsprechen IP1 und IP2 bereits einer Formulierung als lineares Programm. Da diese allerdings vom Rechenaufwand her kaum zu handhaben ist, soll hier folgende einfachere Formulierung genutzt werden:

$$\begin{aligned}
& [\text{minPS-IP}] && \min && k \\
\text{s.d.} &&& \sum_{i:v \in S'_j, j \in \{1, \dots, n\}} x_i + \sum_{i:v \in S_j, j \in \{n+1, \dots, m\}} x_i \geq 1 && \forall v \in G \\
&&&&&& (\text{MPS-IP.1}) \\
&&&&&& \sum_{i=1}^{n+m} x_i w(S_1, X) \leq k && (\text{MPS-IP.2}) \\
&&&&&& x_i \in \{0, 1\}
\end{aligned}$$

Dabei entspricht  $S$  der Menge aller stabilen Mengen  $S_i$  im Graphen  $G$ . Es sei weiter

$$w(S_i, x_1, \dots, x_{n+m}) = a_{i,0} + x_1 a_{i,1} + \dots + x_n a_{i,n} + x_{n+1} a_{i,n+1} + \dots + x_{n+m} a_{i,n+m}$$

Für  $i \in \{n+1, \dots, n+m\}$  sei

$$a_{i,j} = \begin{cases} -1 & v_1^j \in S_i \text{ und } v_{i_j}^j \notin S_i \text{ falls } j \leq n \\ -1 & v_1^j \notin S_i \text{ und } v_{i_j}^j \in S_i \text{ falls } j \leq n \\ 0 & \text{sonst} \end{cases} \quad (4.14)$$

Bedingung MPS-IP.1 stellt dabei sicher, dass jeder Knoten von mindestens einer stabilen Menge oder von einem Pfad überdeckt wird. Bedingung MPS-IP.2 hingegen stellt sicher, dass die Wertigkeit der Zerlegung nicht größer wird als  $k$ .

Die Funktion  $w$  kann minimal den Wert  $-1$  annehmen, es können also maximal zwei Pfade mit einer stabilen Menge verbunden sein.

Durch das lineare Programm kann eine obere und untere Schranke definiert werden. Ist  $l_p$  die Wertigkeit einer Lösung von [minPS-IP] und im zugehörigen Graphen  $G_P$  dieser Lösung gibt es  $c$  Zyklen, so kann eine obere Schranke durch

$$u_{lp}(G) = l_p(G) + c$$

definiert werden. Die Herleitung wird in Abschnitt 6.3 erläutert. Es gilt also

$$l_p(G) \leq \zeta(G) \leq u_{lp}(G)$$

## 4.7 minPS in Verbindung mit dem komplementären Graphen

Es sei  $\bar{G}$  der Komplementgraph von  $G$  mit  $B \subset G$ . Es gilt, dass  $B \not\subset \bar{G}$ . minPS auf  $G$  löst auf  $\bar{G}$  das Problem einer minimalen Zerlegung in Pseudocliquen.

**Definition 4.35.**  $Q_i$  ist eine *Pseudoclique*, wenn  $Q_i = C_i$  selber eine Clique ist oder wenn sie in eine Aneinanderreihung von Cliques  $C_j^i$  mit

$$C_{1,1}^i, p_{1,1}^i, C_{2,2}^i, p_{2,2}^i, C_{3,3}^i, \dots, p_{m_i-1, m_i-1}^i, C_{m_i}^i$$

und  $m_i \geq 2$  zerfällt. Dabei besteht die Schnittmenge zwischen aufeinander folgenden Cliques  $C_j$  und Mengen  $p_{j+1}$  und  $p_j$  und  $C_{j+1}$  jeweils nur ein Knoten. Für die paarweise disjunkten Mengen  $p_j$  gilt, dass sie nur aus drei blauen Knoten bestehen, die durch blaue Kanten in Form eines Pfades verbunden sind. Das heißt, dass  $p_j \subset B(G)$ ,  $|V(p_j)| = 3$  und dass  $p_j$  zusammenhängend und kreisfrei ist. Die Wertigkeit dieser Menge  $Q_i$  ist  $m_i$ .

**Lemma 4.36.** *Jede minimale Zerlegung des Graphen  $G$  mit  $B \subset G$  in pseudostabile Mengen mit Wertigkeit  $\zeta G$  entspricht einer minimalen Zerlegung des Graphen  $\bar{G}$  mit  $B \not\subset \bar{G}$  in Pseudocliquen mit Wertigkeit  $\bar{\zeta}(\bar{G}) = \zeta(G)$ .*

*Beweis.* „ $\Rightarrow$ “ Es sei  $P$  eine Zerlegung des Graphen  $G$  mit  $B \subset G$  in pseudostabile Mengen mit Wertigkeit  $\zeta G$ .

Es gilt  $\bar{\chi}(\bar{G}) = \chi(G)$  (vgl. [S. 59 60]). Ist  $\zeta(G) < \chi(G)$ , so muss es auch eine Zerlegung  $Q$  des Graphen  $\bar{G}$  in Pseudocliquen mit Wertigkeit  $\bar{\zeta}(\bar{G}) = \zeta(G)$  geben. Diese kann wie folgt konstruiert werden:

- Jede stabile Menge  $D$  in  $P$  im Graphen  $G$  ist eine Clique  $C$  in  $Q$  im Graphen  $\bar{G}$ . Damit ist  $|D| = |C|$ .
- Da  $B \not\subset \bar{G}$ , kann die Menge von Pfaden  $Pf$  in  $P$  als Menge der Pfade  $Pf'$  in  $Q$  übernommen werden.

Damit gilt  $\bar{\zeta}(\bar{G}) = \zeta(G)$ .

„ $\Leftarrow$ “ Es sei  $Q$  eine Zerlegung des Graphen  $\bar{G}$  in Pseudocliquen mit Wertigkeit  $\bar{\zeta}(\bar{G})$ .



Nun kann wie bei der Hinrichtung diese Zerlegung in einer Zerlegung  $P$  des Graphen  $G$  in pseudostabile Mengen mit Wertigkeit  $\zeta(G) = \bar{\zeta}(\bar{G})$  konstruiert werden.  $\square$

Da minPS mit dem Graphen  $G$  auf  $\bar{G}$  das Problem einer minimalen Zerlegung in Pseudocliquen (minPC) löst, und dies auch in die Gegenrichtung gilt, stellt sich die Frage, ob sich abseits von der Problemmodellierung weitere Folgerungen daraus ableiten können.

Es sei an dieser Stelle auf Bhasker und Samad verwiesen, die das Problem einer minimalen Zerlegung in Cliques betrachteten (vgl. [10]). Das Problem ist ebenfalls grundsätzlich  $\mathcal{NP}$ -vollständig. Sie leiteten aber folgende obere Schranke für  $\bar{\chi}(G)$  her:

$$\bar{\chi}(G) < \left\lfloor \frac{1 + \sqrt{4n^2 - 4n - 8e + 1}}{2} \right\rfloor$$

wobei  $n$  die Anzahl der Knoten und  $e$  die Anzahl der Kanten in  $G$  ist. Es gilt, dass die Anzahl der Knoten in  $G$  gleich der Anzahl der Knoten in  $\bar{G}$  ist. Die Anzahl der Kanten  $\bar{e}$  in  $\bar{G}$  ist  $n^2 - e$ . Somit gilt

$$\begin{aligned} \chi(G) = \bar{\chi}(\bar{G}) &< \left\lfloor \frac{1 + \sqrt{4n^2 - 4n - 8(n^2 - e) + 1}}{2} \right\rfloor \\ &= \left\lfloor \frac{1 + \sqrt{-4n^2 - 4n + 8e + 1}}{2} \right\rfloor \end{aligned}$$

Da  $\zeta(G) \leq \chi(G)$  ist damit auch eine obere Schranke für eine Zerlegung in pseudostabile Mengen gefunden worden. Diese ist allerdings nur relevant, wenn  $\chi(G)$  nicht in polynomieller Zeit berechnet werden kann.

In diesem Fall ist auch eine weitere Feststellung von Bhasker und Samad wichtig: „In particular, it appears that our algorithms perform increasingly better relative to the coloring algorithms for larger graphs.“ [S. 11 10] Neue Ansätze für Heuristiken und exakte Lösungen finden sich beispielsweise bei Feder und Motwani (vgl. [26]), Gramm u.A. (vgl. [36] und Benati u.A. (vgl. [7]). Wenn die Laufzeit optimiert werden muss, kann es ein wichtiger Ansatz sein, das entsprechend komplementäre Problem zu lösen.

## 4.8 Zusammenfassung

In diesem Kapitel wurde zunächst das Problem minPS definiert und gezeigt, dass es im Allgemeinen  $\mathcal{NP}$ -vollständig ist. Es wurden verschiedene elementare Abschätzungen hergeleitet oder als Ausblick auf minMPS beschrieben. Für zwei Spezialfälle konnte gezeigt werden, dass minPS in polynomieller Zeit lösbar ist. Des Weiteren wurde ein ganzzahliges lineares Programm hergeleitet. Abschließend wurde minPS in Verbindung mit dem komplementären Graphen untersucht und das komplementäre Problem minPC untersucht.

Es bleibt noch zu klären, ob es nicht noch weitere Spezialfälle gibt, in denen minPS kombinatorisch in polynomieller Zeit lösbar ist.

Damit sind die Grundlagen beschrieben, um ein Anwendungsproblem konkret zu behandeln. Im nächsten Abschnitt wird das Train Marshalling Problem in die allgemeine Klasse der Rangierprobleme eingebettet und eine Lösung mittels minPS definiert.

# Sortier- und Rangierprobleme in Güterbahnhöfen

In Rangierbahnhöfen werden die ankommenden Züge mit Güterwagen aufgelöst und für den weiteren Transport auf der Schiene zu neuen Güterzügen zusammengestellt. Ziel ist es, die Wagen so anzuordnen, dass bestimmte Wagen mit demselben Ziel oder ähnlichem Inhalt in einer passenden Aneinanderreihung in den neuen Zügen enthalten sind. Eine genaue Darstellung von Rangierbahnhöfen findet sich beispielsweise bei Hiller [44]. Dort finden sich auch weitere Beispiele zu realen Rangierbahnhöfen.

Ein Rangierbahnhof besteht meistens aus einem oder mehreren Einfahrgleisen in der Einfahrgruppe, auf denen die ankommenden Züge stehen, etlichen Richtungsgleisen in der Richtungsgruppe, die zur Sortierung der Züge benutzt werden und einem oder mehreren Ausfahr Gleisen in der Ausfahrgruppe. Oft sind Einfahrgleise und Richtungsgleise durch einen Ablaufberg getrennt, der dazu benutzt wird, die Wagen auf die verschiedenen Richtungsgleise zu rangieren. Gelegentlich liegt auch der ganze Bahnhof im Gefälle oder das Rangieren wird ganz von Rangierlokomotiven, durch Umsetzen oder Abstoßen, übernommen, falls ein Ablaufbetrieb nicht möglich

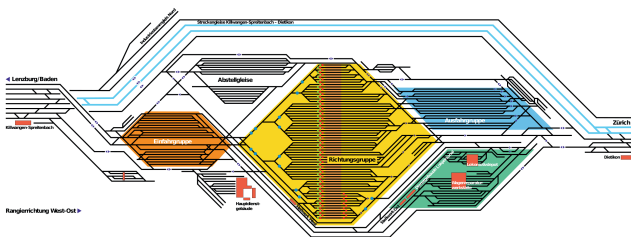
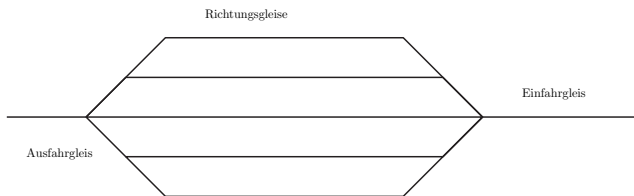


Abbildung 5.1: Gleisplanschema Rbf Zürich-Limmattal, entnommen aus [32].



**Abbildung 5.2:** Beispielhafte schematische Darstellung eines Rangierbahnhofes.

ist.

Reale Güterbahnhöfe können eine große Komplexität aufweisen. Als Beispiel sei auf den Rangierbahnhof Zürich-Limmattal in Abb. 5.1 verwiesen. Dieser besteht im Wesentlichen aus 18 Einfahrgleisen, 64 Richtungsgleisen mit einer Länge zwischen 650 und 850 Metern und 16 Ausfahrgleisen [S. 144 44][32].

Wir nehmen nun an, dass es ein Einfahrgleis und ein Ausfahrgleis gibt. Jeder ankommende Wagen fährt durch ein Einfahrgleis und wird über den Ablaufberg auf eines der Richtungsgleise geleitet. Auf dem Ausfahrgleis wird der Zug wieder zusammengestellt, indem sämtliche Wagen von den Richtungsgleisen an den Zug gehängt werden. Dabei soll am Ende ein sortierter Zug den Bahnhof verlassen können. Eine schematische Darstellung dieses vereinfachten Konzeptes findet sich in Abb. 5.2.

Die Rangieraufgabe ist es, die Züge zu trennen und die Wagen so auf die einzelnen Richtungsgleise zu rangieren, dass sie von dort idealerweise als Ganzes, also ohne weiteres Entkuppeln und Rangieren, auf das Ausfahrgleis rangiert werden können. Dadurch werden sie zu einem neuen Zug zusammen gestellt.

Es gibt im Hinblick auf die Zeit- und Kostenproblematik vier Möglichkeiten, den Betrieb zu optimieren:

- Die Anzahl der Richtungsgleise
- Die Länge der Richtungsgleise
- Die Anzahl der Rangieroperationen auf die Richtungsgleise
- Die Anzahl der Rangieroperationen auf das Ausfahrgleis

Dieses Problem ist selbstverständlich nicht nur auf das Zusammenstellen von Güterzügen beschränkt. Ähnliche Überlegungen sind zum Beispiel auch

nötig, um ankommende Straßen- oder U-Bahnen in ihrem Nachtdepot oder Personenzüge in einem Abstellbahnhof unterzubringen oder neu zusammen zu stellen.

Sowohl 2007 in der Arbeit von Di Stefano, Maue, Modelski, Navarra, Nun-kesser und van den Broek [19] als auch 2011 bei Hansmann [41] wurde ver-sucht, eine allgemeine Problemstellung zu formalisieren. Der Schwerpunkt bei [41] liegt allerdings eher auf theoretischen Aspekten. Eine weitere Dar-stellungen von Rangierproblemen finden sich auch 2009 in der Arbeit von Gatto, Maue, Mihalák und Widmayer [31].

Im Folgenden soll zunächst eine Beschreibung und Definition des allgemei-neren SRSP nach [41] und dann des spezielleren TMP gegeben werden.

## 5.1 SRSP

Eine allgemeine Formulierung der Zugbildungsprobleme ist die Klasse der *Sorting of rolling Stock Problems* (SRSP, SRS). Dabei müssen ankomen-de Züge auf beliebige Weise auf den Richtungsgleisen bewegt werden, damit sie den Bahnhof in einer strukturierten Form verlassen können. Der Fokus liegt hier auf der Minimierung der Richtungsgleise.

### 5.1.1 Ausgabesequenz

Die Wagen des ankommenden Zuges haben spezielle Eigenschaften, z. B. ihre Ladung, ihr Ziel, ihre Herkunft, eine Klasse oder einen bestimmten Wagentyp. Im Folgenden wird der Einfachheit halber von Zielen gespro-chen, Wagen mit demselben Ziel bilden dann eine Gruppe.

Wir markieren nun alle Positionen im fertigen Zug mit natürlichen Zahlen und fordern, dass Wagen mit derselben Markierung auch aus der selben Gruppe sein müssen. Also ist die Anzahl  $g$  von verschiedenen Gruppen dieselbe, wie die der verschiedenen Markierungen auf dem Ausfahrgeis.

Die Markierungen auf dem Ausfahrgeis können nun verschieden angeord-net werden. Bilden sie Blöcke, so sprechen wir von einer  $g$ -**blocks** Folge oder sonst, wenn sie ein anderes Schema bilden, von einer  $g$ -**pattern** Folge. Wenn es keinen Zusammenhang zwischen der Position im fertigen Zug und den Zielen gibt, so nennt man die Ausgabesequenz **free**. Sind die Wagen der Ausgabesequenz in einer vordefinierten Gruppe, nennt man die Aus-gabesequenz **ordered**. Wir gehen im Folgenden davon aus, dass wir einen

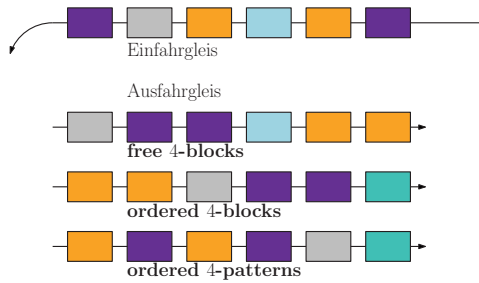


Abbildung 5.3: Abbildung zu Beispiel 5.1, Notation gemäß [41].

Zug auf einem einzigen Gleis zusammenstellen wollen. Die Struktur, in der man  $o$  Züge auf  $o$  Gleisen benötigt, wird als  $o$ -**ordered** bezeichnet.

Es gibt also vier mögliche Strukturen der Ausgabesequenz, **free  $g$ -pattern** und **ordered  $g$ -pattern** und als Spezialfälle **free  $g$ -blocks** und **ordered  $g$ -blocks**.

**Beispiel 5.1.** Die Zahl 1 symbolisiere einen Wagen der 1. Klasse, 2 einen der 2. Klasse, 3 einen Speisewagen und 4 einen Gepäckwagen. Es stehen nun auf dem Einfahrgleis zwei Wagen von Typ 1 und 2 und jeweils einer von Typ 3 und 4.

**free 4-blocks** würde zum Beispiel einen fertigen Zug der Form

3 2 2 4 1 1

liefern. Für **ordered 4-blocks** benötigen wir ein Ordnungskriterium, also z. B. erst 1. Klasse, Speisewagen, 2. Klasse und dann den Gepäckwagen was zu

1 1 3 2 2 4

führt. **Ordered 4-pattern** lägen vor, wenn z. B. gefordert würde jeweils 1.- und 2.-Klasse Wagen nacheinander und Gepäck- und Speisewagen an den Schluss zu rangieren:

1 2 1 2 3 4

Entsprechend wäre bei **free 4-pattern** die Wahl der Wagen völlig frei. Vergleiche hierzu auch Abb. 5.3  $\triangle$

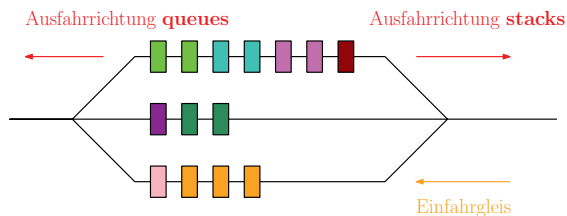


Abbildung 5.4: Darstellung der Gleisbelegung aus Beispiel 5.3.

### 5.1.2 Gleisstruktur

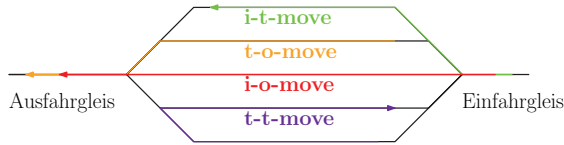
Die Gleisstruktur der Richtungsgleise (von denen im Folgenden nur als „Gleise“ gesprochen wird) ist ebenfalls variabel. Unbeschränkte, **unbounded** Gleise kommen in der Realität nicht vor, werden aber hier verwendet, da es oft einfacher ist, Algorithmen für unbeschränkte Gleise zu finden und diese auf die reale Beschränkung der Gleislänge umzurechnen [S. 941]. In der Länge beschränkte Gleise werden als **b-bounded** bezeichnet, wenn maximal  $b$  Wagen auf das Gleis passen.

**Beispiel 5.2.** Soll in einem Rangierbahnhof mit Richtungsgleisen, die maximal 30 Wagen fassen, ein Zug neu zusammengestellt werden, der für ein Chemiewerk 40 Wagen und für eine Möbelfabrik 50 Wagen enthält, so kann man diese beiden Ziele jeweils auf einem Gleis sammeln. Da die Gleise aber zu kurz sind, benötigt man vier Gleise.  $\triangle$

Man kann die Gleise nun als **stacks** realisieren, d. h. als Stumpfgleis, bei dem Einfahrt und Ausfahrt am selben Ende geschehen, oder als **queues**, bei denen Einfahrt und Ausfahrt am gegenüberliegenden Gleisende liegen. Ein Stack vertauscht die Reihenfolge der Wagen vom Rangiergleis zum Ausfahrgleis. Weiter bezeichnet **sido** (single in, double out) die Struktur, bei der nur an einem Gleisende eingefahren aber an beiden ausgefahren werden darf. Entsprechend **diso** (double in, single out), wenn an beiden Gleisenden eingefahren aber nur an einem ausgefahren werden darf und **dido** (double in, double out), wenn an beiden Gleisenden ein- und ausgefahren werden darf. Möchte man die Anzahl von Gleisen  $t$  im Rangierbahnhof mit angeben, schreibt man beispielsweise **t-stacks**.

**Beispiel 5.3.** Auf drei Richtungsgleisen seien folgende Wagen

Gleis 1 : 1 1 4 4 7 7 3  
 Gleis 2 : 5 6 6  
 Gleis 3 : 8 9 9 9



**Abbildung 5.5:** Darstellung möglicher Rangierbewegungen, Notation gemäß [41].

Im Vergleich zu **queues** ist es bei **stacks** nicht möglich, das Ziel 1 an den Anfang des Zuges zu rangieren. Vergleiche hierzu auch Abb. 5.4  $\triangle$

### 5.1.3 Sortiermethode

Grundsätzlich sind für jeden Wagen nur bestimmte Bewegungen möglich. Ein Wagen kann vom Einfahrgleis auf ein Richtungsgleis bewegt werden (**i-t-move**), er kann von einem Richtungsgleis zu einem anderen oder demselben rangiert werden (**t-t-move** oder schlicht **Rangierbewegung**), von einem Richtungsgleis zum Ausfahrgleis (**t-o-move**) oder direkt vom Einfahrgleis zum Ausfahrgleis (**i-o-move**). Eine Darstellung dieser Rangierbewegungen findet sich in Abbildung 5.5.

Wir können das Rangieren (d. h. alle **t-t-moves** und **i-o-moves**) ganz verbieten, diesen Fall bezeichnen wir als **no-shunting**. Werden maximal  $h$  solcher Bewegungen erlaubt, spricht man von  **$h$ -hump-shunting**.

Wir unterteilen die Gleisbewegungen in Ankunft (**i-o-move** or **i-t-move**) und Abfahrt (**t-o-move** und **i-o-move**). Weiter unterscheiden wir, ob die Ankunfts- und Abfahrtsbewegungen in den Fahrplänen getrennt oder gemischt ablaufen. Geschehen sie der Reihe nach (d. h. **t-o-moves** erst nach dem letzten **i-t-move**), spricht man von **sequential**. Wenn alle Gleisbewegungen gleichzeitig ablaufen, spricht man von **concurrent**. In beiden Fällen ist als Information die Reihenfolge der Wagen im ankommenden Zug vorhanden, nicht aber, wann diese Wagen ankommen. Sind die genauen Ankunfts- und Abfahrtszeiten der einzelnen Wagen bekannt, so spricht man von **time windows**.

Eine weitere zu klärende Frage ist, ob Wagen eines bestimmten Zieles auf verschiedene Richtungsgleise verteilt, also getrennt werden dürfen. Erlaubt man die beliebige Verteilung von Zielen auf die Gleise, so spricht man von **split**. Dürfen die Wagen eines Zieles auf maximal  $s + 1$  Gleise verteilt werden mit  $s \geq 0$ , spricht man von  **$s$ -split**. Dürfen die Wagen eines Zieles



<b>Gleisstruktur</b>						
<i>Art</i>						
	<b>stacks</b>	<b>queues</b>	<b>stacks/queues</b>	<b>sido</b>	<b>diso</b>	<b>dido</b>
<i>Länge</i>						
	<b>unbounded</b>	<b>b-bounded</b>				
<b>Sortiermethode</b>						
<i>Rangieren</i>						
	<b>no-shunting</b>	<b>h-hump-shunting</b>				
<i>Steuerung</i>						
	<b>sequential</b>	<b>concurrent</b>	<b>time-window</b>			
<i>Trennen</i>						
	<b>s-split</b>	<b>split</b>	<b>chain-split</b>			
<b>Ausgabesequenz</b>						
	<b>free</b>	<b>ordered</b>				
	<b>g-blocks</b>	<b>g-pattern</b>				

**Tabelle 5.1:** Parameter für SRSP, Notation gemäß [41].

nur in solcher Weise getrennt werden, dass die Richtungsgleise Gleis für Gleis aneinandergefügt den gewünschten Endzug ergeben, so spricht man von **chain-split**.

#### 5.1.4 Notation und Ziel

In [41] schlug Hansmann als Notation für SRSP ein Tripel  $\nu := \alpha|\beta|\gamma$  mit

$$\alpha \in \{\{\text{st,qu,sq,sd,ds,dd},\cdot\} \times \{\text{ub,b-bd},\cdot\}\}$$

$$\beta \in \{\{\text{nsh,h-hsh},\cdot\} \times \{\text{se,co,tw},\cdot\} \times \{\text{s-sp,sp,csp},\cdot\}\}$$

$$\gamma \in \{\{\text{fr,or},\cdot\} \times \{\text{g-bl,g-pa},\cdot\}\}$$

vor. Dabei gibt also  $\alpha$  die Gleisstruktur,  $\beta$  die Sortiermethode und  $\gamma$  die Art der Ausgabesequenz an. Abkürzungen können dabei der Tabelle 5.1 entnommen werden, wobei der Punkt alle Möglichkeiten abdeckt.

Ziel von SRSP kann die Minimierung verschiedener Parameter, etwa der Richtungsgleise, der Rangierbewegungen oder der Länge der Richtungsgleise sein. Im Folgenden wird nur die Minimierung der Anzahl der Richtungsgleise betrachtet.

## 5.2 TMP

Nun beschäftigen wir uns mit dem Zugbildungsproblem TMP („Train Marshalling Problem“). Dieses entspricht der SRSP Variante **st,ub|nsh,se,csp|fr,g-bl**.

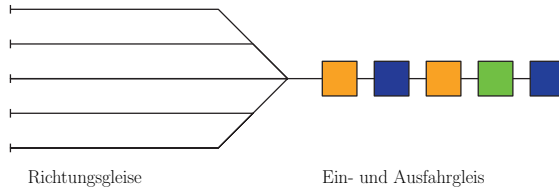
Wir nehmen an, die Richtungsgleise sind als Stack (**st**) realisiert, also ist die Einfahr- gleich der Ausfahrriechtung. Ihre Länge sei unendlich (**ub**). Weiter erlauben wir eine roll-in-Operation für jeden Wagen und eine pull-out-Operation für jedes Gleis, also kein weiteres Rangieren. Ein Wagen darf nur vom Einfahr- auf ein Richtungsgleis bewegt werden und die Wagen vom Richtungsgleis auf das Ausfahrgeleis (**nsh**).

In dem neu zusammengestellten Zug auf dem Ausfahrgeleis sollen die Wagen mit demselben Ziel hintereinander folgen, also einen Block (**g-bl**) bilden. Dabei ist es unerheblich, wie die Blöcke sortiert sind (**fr**). Der Ablauf der Zugzusammenstellung erfolgt dabei der Reihe nach, also sequenziell (**se**), man weiß also schon zu Beginn in welcher Reihenfolge die Wagen ankommen.

Da wir sequenziell vorgehen und keine Rangierbewegungen erlauben, können wir das Trennen von verschiedenen Zielen auf verschiedene Gleise zu lassen, solange die zulässige Blocksortierung im Zug erhalten bleibt, wenn die Inhalte der Richtungsgleise Stück für Stück an den Zug rangiert werden. Sie können also nur in Ketten getrennt werden (**csp**).

Im Jahr 2000 wurde von Dahlhaus, Horak, Miller und Ryan in [16] bewiesen, dass das TMP  $\mathcal{NP}$ -vollständig ist. Sie leiteten außerdem einige Schranken her, die zum Teil später noch diskutiert werden. Bereits 1983 gab es von Zhu und Zhu, die das Problem unter verschiedenen Nebenbedingungen betrachteten, erste Resultate und polynomielle Algorithmen [87]. In einer weiteren Arbeit 2000 von Dahlhaus, Manne, Miller und Ryan [17] wurden unter anderem verwandte Probleme betrachtet. Weitere Schranken und Algorithmen finden sich 2011 in der Arbeit von Beygang [8] bzw. 2010 bei Beygang, Krumke und Zeck [9]. Das Problem wurde 2011 ebenfalls in der Arbeit von Hansmann [41] und 2012 von Brueggemann, Fellows, Fleischer, Lackner, Komusiewicz, Koutis, Pfandler und Rosamond [14] bearbeitet.

Weitere Arbeiten im Umfeld des Themas sind rar, finden sich aber zum Beispiel 2004 bei Stefano u. Koči [77], die ebenfalls einen graphentheoretischen Ansatz wählten, um die Anordnung von Zügen in Nachtdepots zu untersuchen. Neuere Arbeiten sind 2015 von Adlbrecht u. a. [1], die das TMP vor allem in der Praxis untersuchten und es in ein gemischtes ganzzahliges



**Abbildung 5.6:** Graphische Darstellung zu Beispiel 5.4.

lineares Programm umformulierten, und 2017 von Rinaldi u. Rizzi [70], die dynamische Programmierung als Ansatz nutzen, veröffentlicht worden. Anders Haahr und Lusby, die 2016 versuchen, das Problem in Unterprobleme zu teilen und ihre Abhängigkeiten zu bestimmen (vgl. [39]).

### 5.2.1 Definition

Jedem *Wagen*  $i$  auf dem Bahnhof wird eine Zahl  $\sigma_i \in \mathbb{N}^+$  zugeordnet. Diese Zahl kann z. B. als Zielort interpretiert werden; die Wagen werden nach diesen Zahlen sortiert. Ein *Zug*  $\sigma$  der Länge  $n$  ist entsprechend eine Folge von Wagen

$$\sigma = (\sigma_1, \dots, \sigma_n)$$

mit  $\sigma_i \in \{1, \dots, d\}$  für  $i \in \{1, \dots, n\}$ . Gesucht ist nun also ein ausfahrender Zug, in dem Wagen mit gleichem Ziel in einem Block sind. Dabei ist man an der minimalen Anzahl der Rangierbewegungen einer Rangierlokomotive interessiert. Da per Definition nur eine Rangierbewegung pro Gleis erlaubt ist (nämlich vom Richtungsgleis auf das Ausfahrgleis) ist dies äquivalent zu einer Minimierung der Richtungsgleise. Ihre Anzahl wird mit  $K(\sigma)$  bezeichnet.

**Beispiel 5.4.** Sei  $\sigma = (1, 2, 1, 3, 2)$ . Dann gibt es drei Ziele, wobei jeweils der 1. und 3. Wagen, der 2. und letzte sowie der 4. Wagen ein gemeinsames Ziel haben. Vergleiche dazu auch Abb. 5.6  $\triangle$

In der Arbeit von Dahlhaus, Horak, Miller und Ryan [16] wird eine andere Formulierung des Problems verwendet. Gegeben sei ein Zug der Länge  $n$  und die Menge seiner Wagen  $I_n = \{1, 2, \dots, n\}$ . Dann sind die verschiedenen Ziele als Menge von Teilmengen  $S = \{S(1), \dots, S(t)\}$  gegeben, wobei  $S(m)$  alle Wagen mit dem Ziel  $m$  beinhaltet. Zwei Wagen  $w_i$  und  $w_j$  haben also dasselbe Ziel, wenn sie zu derselben Menge  $S(m)$  gehören.

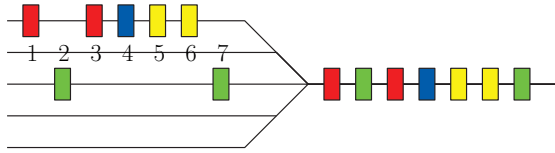


Abbildung 5.7: Graphische Darstellung zu Beispiel 5.5.

**Definition 5.1** (TMP). [S. 42 16] Finde die kleinste Zahl  $K(S)$ , so dass es eine Permutation  $\pi(1), \dots, \pi(t)$  von  $1, \dots, t$  gibt, damit die Folge von Zahlen

$$\underbrace{1, 2, \dots, n, 1, 2, \dots, n, 1, 2, \dots, n}_{K(S)\text{-mal}}$$

alle Elemente von  $S(\pi(1))$  beinhaltet, gefolgt von den Elementen von  $S(\pi(2))$  usw.

Jede Folge  $1, 2, \dots, n$  kann als ein Richtungsgleis interpretiert werden, entsprechend liefert die Permutation die Gleisbelegung der einzelnen Richtungsgleise. Man sieht also, dass beide Formulierungen äquivalent sind, insbesondere, dass  $K(S) = K(\sigma)$ .

Eine Instanz des TMP kann also durch Folgen oder Mengen dargestellt werden. Die Notationen werden im folgenden Abschnitt dargestellt.

**Beispiel 5.5.** Sei  $n = 7$ ,  $t = 4$  und  $S = \{S_1, S_2, S_3, S_4\}$  mit  $S_1 = \{1, 3\}$ ,  $S_2 = \{2, 7\}$ ,  $S_3 = \{4\}$  und  $S_4 = \{5, 6\}$ . Dann ist  $K(S) = 2$  mit  $\pi(1) = 1$ ,  $\pi(2) = 4$ ,  $\pi(3) = 2$ ,  $\pi(4) = 3$ :

$$\underbrace{1\ 2\ 3}_{S_1} \underbrace{4}_{S_3} \underbrace{5\ 6}_{S_4} 7\ 1 \underbrace{2\ 3\ 4\ 5\ 6\ 7}_{S_2}$$

Eine grafische Darstellung der Gleisbelegung und des Zuges findet sich in Abb. 5.7. △

Wenn die Eingabe als  $S$  bezeichnet wird, sprechen wir über eine Darstellung durch Mengen, sonst wird die Formulierung durch Folgen benutzt. Die Darstellung des Zuges erfolgt in Beispielen je nach Übersichtlichkeit entweder als Folge  $\sigma$  oder durch Mengen  $S$ .

Die folgende Darstellung der Grundlagen, Definitionen und Notationen folgt weitestgehend und wenn nicht anders angegeben der Arbeit von Beygang [8].

## 5.2.2 Grundlagen und Notation zum TMP

Mit  $\mathbb{S}^n$  wird die Menge aller Instanzen des TMP mit  $n$  Wagen bezeichnet. Für  $S \in \mathbb{S}^n$  ist  $d(S)$  die Anzahl der Ziele in dieser Instanz, wenn möglich abgekürzt durch  $d$ .

$S^{-1} \in \mathbb{S}^n$  ist der Zug, der durch die Umkehrung der Reihenfolge der Wagen im Zug  $S$  entsteht.

Eine *Gleisbelegung* (oder Neuzusammenstellung) für eine Instanz  $S \in \mathbb{S}^n$  kann durch eine Abbildung der Menge  $\{1, \dots, n\}$  nach  $\mathbb{N}$ , also

$$tr : \{1, \dots, n\} \rightarrow \mathbb{N}$$

dargestellt werden.

Die Bildmenge entspricht den Gleisen, denen eine bestimmte Menge an Wagen zugewiesen wurde.

Wenn die Gleise durch die im TMP erlaubten Rangierbewegungen zu einem Zug zusammengestellt werden können, so dass dort alle Ziele einen Block bilden, spricht man von einer *zulässigen Gleisbelegung* bzw. *zulässigen Neuzusammenstellung*.

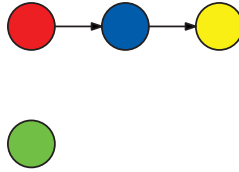
**Beispiel 5.6.** Gegeben sei der Zug und die Gleisbelegung aus Beispiel 5.5. Dann ist die Abbildung  $tr : \{1, \dots, 7\} \rightarrow \{1, 2\}$  gegeben durch  $tr(1) = tr(3) = tr(4) = tr(5) = tr(6) = 1$  und  $tr(2) = tr(7) = 2$ .  $\triangle$

In der folgenden Definition wird schon eine Ähnlichkeit zu Definition 3.5 sichtbar, die aber zu einem späteren Zeitpunkt genauer betrachtet werden soll. Die folgende Definition ist an die Arbeit von Beygang [8] angelehnt.

**Definition 5.2** (Gerichteter Graph einer zulässigen Gleisbelegung). Für eine zulässige Gleisbelegung  $tr$  einer Instanz  $S \in \mathbb{S}^n$  sei  $G_{S,tr} = (V, E)$  der gerichtete Graph der zulässigen Gleisbelegung  $tr$  mit

- $V = (1, \dots, d)$
- $(i, j) \in E$  wenn ein  $c$  mit  $1 \leq c < n$  existiert mit  $c \in S(i)$ ,  $(c + 1) \in S(j)$  und  $tr(c) = tr(c + 1)$ .

**Beispiel 5.7.** Gegeben der Zug und die Gleisbelegung aus Beispiel 5.5. Dann gilt lediglich für die Wagen 3 und 4 sowie 4 und 5, dass der kommende Wagen auf demselben Gleis einem anderen Ziel angehört. Dementsprechend liefern diese die einzigen beiden Kanten. Vgl. hierzu Abb. 5.8.  $\triangle$



**Abbildung 5.8:** Graphische Darstellung des gerichteten Graphen der zulässigen Gleisbelegung zu Beispiel 5.7.

Eine gerichtete Kante zwischen zwei Zielen kann es also nur geben, wenn das eine Ziel auf das andere auf einem Gleis folgt. Das heißt, es kann keine zulässige Gleisbelegung geben, deren gerichteter Graph Kreise enthält.

**Theorem 5.3.** [S. 31 8] Sei  $S \in \mathbb{S}^n$  und  $tr$  eine zulässige Gleisbelegung. Dann gilt für den gerichteten Graphen der zulässigen Gleisbelegung:

1.  $|\delta^+(v)| := |\{(v, w) \in E | w \in V\}| \leq 1$  für alle  $v \in V$ .
2.  $|\delta^-(v)| := |\{(w, v) \in E | w \in V\}| \leq 1$  für alle  $v \in V$ .
3.  $G_{S, tr}$  ist nicht notwendigerweise zusammenhängend.
4. Jede Zusammenhangskomponente von  $G_{S, tr}$  ist ein gerichteter Pfad der alle Knoten verbindet.
5.  $G_{S, tr}$  ist azyklisch.

**Definition 5.4** (Bereich eines Zyklus). Sei  $G_{S, tr}$  der gerichtete Graph einer zulässigen Gleisbelegung  $tr$  einer Instanz  $S \in \mathbb{S}^n$ . Dann geht der Bereich eines Zyklus  $[d_1, \dots, d_k]$  in  $G_{S, tr}$  über  $s$  Gleise, wenn  $|tr(N)| = s$ . Dabei ist  $N$  die Menge aller Wagen mit Zielen im Zyklus, d. h.  $N = \{i | i \in S(d_1) \cup \dots \cup S(d_k)\}$ .

Eine zulässige Gleisbelegung kann immer gefunden werden, wenn man jedes Ziel einem eigenen Gleis zuweist. Daraus folgt:

$$K(S) \leq d$$

Oder mit anderen Worten:  $|tr(S)| := |\{tr(i) | i \in S(1) \cup \dots \cup S(d)\}| \leq d$ . Ohne Beschränkung der Allgemeinheit nehmen wir nun an, dass  $tr(S) = \{1, \dots, |tr(S)|\}$ , also dass alle Gleise aufsteigend, beginnend mit Eins, nummeriert werden.

Der erste und der letzte Wagen in  $S$  mit dem Ziel  $k$  (mit  $1 \leq k \leq d$ ) wird mit  $first(S, k)$  bzw.  $last(S, k)$  bezeichnet.

Der erste und letzte Wagen auf dem Gleis  $i$  wird entsprechend mit  $first(i)$  bzw.  $last(i)$  bezeichnet.

Außerdem steht  $last(i, k)$  für den letzten Wagen auf Gleis  $i$ , bevor der  $k$ -te einem Rangiergleis zugewiesen wird.

$p(i, t)$  steht für die Position des Wagens  $i$  auf Gleis  $t$ . Ohne Beschränkung der Allgemeinheit gilt: Steht ein Wagen auf  $p(i, t)$ , so steht auf  $p(j, t)$  mit  $j < i$  ebenfalls ein Wagen, denn man kann davon ausgehen, dass alle Wagen den Richtungsgleisen in aufsteigender Reihenfolge zugewiesen werden.

Sei  $tr$  eine Gleisbelegung von  $S$ , dann wird ein Ziel als *getrennt* („split“) bezeichnet, wenn zwei Wagen mit demselben Ziel auf zwei verschiedene Gleise rangiert wurden, d. h. wenn  $i, j \in \{1, \dots, n\}$  existieren mit  $i, j \in S(k)$  (und  $1 \leq k \leq d$ ) und  $tr(i) \neq tr(j)$ . Sonst spricht man von *nicht getrennten Zielen* („unsplit destination“).

Weiter wird zwischen *vollendeten* und *unvollendeten Zielen* unterschieden. Angenommen, die ersten  $k$  Wagen von  $S$  wurden Schritt für Schritt den Richtungsgleisen zugewiesen, dann ist ein Ziel  $l$  *vollendet*, wenn alle Wagen mit Ziel  $l$  einem Gleis zugewiesen wurden. Das heißt, für alle  $1 \leq i \leq n$  mit  $i \in S(l)$  gilt  $i \leq k$ . Sonst ist das Ziel *unvollendet*.

Ist stattdessen eine Instanz  $\sigma$  mit Folgen gegeben, ist  $\Sigma^n$  die Menge aller Instanzen der Länge  $n$ . Analog wird für  $\sigma \in \Sigma^n$   $d(\sigma)$ ,  $first(\sigma, k)$ ,  $last(\sigma, k)$  etc. definiert.

## 5.3 Allgemeine Definitionen und Notationen

### 5.3.1 Struktur einer zulässigen Gleisbelegung

In der Arbeit von Beygang [8] wurde die Struktur einer zulässigen Gleisbelegung analysiert. Diese Strukturen sind wesentlich für die folgenden Kapitel.

Eine zulässige Lösung von  $S$  ist eine Abbildung  $tr : \{1, \dots, n\} \rightarrow \{1, \dots, d\}$ . Wenn erst  $l < n$  Wagen zugewiesen wurden, definieren wir eine *partielle Lösung*, indem wir das Definitionsgebiet einschränken auf  $\{1, \dots, l\}$ .

**Lemma 5.5.** *Für jede zulässige Gleisbelegung des TMP gibt es nicht mehr als zwei getrennte Ziele  $i$  und  $j$  auf jedem Gleis  $k$  mit  $first(k) = i$  und  $last(k) = j$ . Insbesondere müssen alle getrennten Ziele am Anfang oder am Ende jeden Gleises stehen.*

Getrennte Ziele können also nicht zwischen anderen Zielen auf einem Richtungsgleis auftreten. Außerdem können getrennte Ziele nicht auf mehr als zwei Richtungsgleise verteilt werden.

**Lemma 5.6.** *Es gibt immer eine optimale zulässige Lösung des TMP, die jedes Ziel maximal einmal trennt.*

Ebenso gilt, dass es zwei nicht getrennte Ziele geben muss.

**Theorem 5.7.** *Für jede zulässige Gleisbelegung gibt es mindestens ein Ziel, das nicht getrennt wird und am Anfang eines Richtungsgleises steht.*

Und mit derselben Argumentation:

**Theorem 5.8.** *Für jede zulässige Gleisbelegung gibt es mindestens ein Ziel, das nicht getrennt wird und am Ende eines Richtungsgleises steht.*

Es gibt also bei einer zulässigen Gleisbelegung nur zwei Möglichkeiten, Gleise einzuspären:

- Ein Gleis  $i$  ist zwar belegt, aber es gibt keine weiteren Wagen des Zieles  $last(i)$ . Das Gleis kann also wieder für ein neues Ziel verwendet werden.
- Wir haben die Möglichkeit, ein Ziel zu trennen. Dann benötigen wir statt drei Gleisen für drei Ziele nur zwei Gleise für drei Ziele.

**Definition 5.9** (Abgeschlossenes Ziel). Ein Ziel  $k \in S$  heißt abgeschlossen, wenn  $last(S, k)$  schon einem Richtungsgleis zugewiesen wurde.

**Definition 5.10** (Geschlossenes Gleis). Sei  $tr$  eine partielle Lösung der ersten  $l$  Wagen und  $i \in tr(L)$  mit  $L = \{1, \dots, l\}$ . Dann heißt das Gleis  $i$  geschlossen wenn

- der letzte Wagen auf  $i$  das Ziel  $k$  hat, d. h.  $last(i) \in S(k)$  und
- es ein anderes Gleis  $j \neq i$  gibt mit  $j \in tr(L)$  und  $first(j) \in S(k)$ .

Diese Definition betrifft nur die Gleise, deren letzter Wagen zu einem getrennten Ziel gehört. Geschlossene Gleise dürfen nicht weiter benutzt werden, da man das getrennte Ziel nicht wieder vereint bekommen würde.



### 5.3.2 Struktur der Eingabesequenzen

Ein wichtiger Punkt um die Eingabesequenz zu analysieren ist die Frage, inwieweit sich die Intervalle überschneiden, die durch das erste und letzte Auftreten eines Wagens eines bestimmten Zieles gegeben sind. Überschneidet sich kein einziges Intervall, würde dies bedeuten, dass alle Ziele in der Eingabe schon sortiert sind und man mit einem Richtungsgleis auskäme.

**Definition 5.11** ((Nicht-)Überlappende Instanzen). [S. 18 8] Gegeben sei eine Instanz  $S \in \mathbb{S}^n$ . Zwei Ziele  $1 \leq i, j \leq d$  *überlappen nicht* wenn  $last(S, i) < first(S, k)$  oder  $last(S, j) < first(S, i)$ . Sonst *überlappen* sie. Eine Instanz  $S$  des TMP wird *überlappend* genannt, wenn alle Ziele paarweise überlappen. Sonst wird sie *nicht überlappend* genannt.

Um zu überprüfen, ob alle Ziele überlappen, muss man also die größte Clique des zugehörigen Intervallgraphen  $G_S$  bestimmen. Wenn die Instanz überlappend ist, decken sich die Größe der maximalen Clique in  $G_S$  und  $d$ , die Anzahl der Ziele.

Die größte Clique eines Intervallgraphen kann in polynomieller Zeit gefunden werden, vgl. Kapitel 2.2.

**Definition 5.12.** Sei  $G_S$  der zu einer Instanz  $S \in \mathbb{S}^n$  gehörende Intervallgraph. Die *Überlappungszahl*  $\omega(S) \in \mathbb{N}$  wird durch die Größe einer maximalen Clique in  $G_S$  definiert.

Offensichtlich gilt also  $\omega(S) \leq d$ .

Um die Zusammenhänge zwischen den minimal benötigten Richtungsgleisen und der Eingabesequenz weiter zu analysieren sei zunächst

$$K_n := \max_{S \in \mathbb{S}^n} K(S).$$

Dann ist  $K_n$  die maximale Anzahl benötigter Gleise  $K(S)$ , wobei  $S$  alle möglichen Instanzen  $S \in \mathbb{S}^n$  abdeckt. Damit gilt immer  $K(S) \leq K_n \forall S \in \mathbb{S}^n$ .

**Proposition 5.13.** [S. 49 ff 16][S. 19 8] Sei  $S \in \mathbb{S}^n$  und  $S' = \{S(i_1), \dots, S(i_r)\}$  eine beliebige Teilmenge von  $S$  mit  $|S'(i_1) \cup \dots \cup S'(i_r)| = t$ , also eine Teilmenge mit  $t$  Wagen und  $r$  Zielen. Es gilt nun:

1.  $K(S) = K(S^{-1})$
2.  $K(S) \leq K(S') + K(S \setminus S') \leq K(S') + K_{n-t}$

3. Wenn  $S'$  so gewählt wurde, dass alle Wagen aus Zielen von  $S$ , die nur aus einzelnen Wagen bestehen, in eine neues Ziel zusammengefasst werden, dann gilt  $K(S) \leq K(S')$ .
4. Beinhalte  $S$  nun ein Ziel mit  $r$  Wagen oder  $r$  Ziele, die jeweils nur einen einzelnen Wagen haben. Dann  $K(S) \leq 1 + K_{n-r}$ .
5. Wenn  $|S(i)| + |S(j)| = r$  und die Zielmengen  $S(i)$  und  $S(j)$  überlappen nicht, so ist  $K(S) \leq 1 + K_{n-r}$ .
6.  $K(S') \leq K(S)$

Für die weitere Untersuchung des Problems ist es auch wichtig, den Begriff der Unterinstanz genauer zu definieren. Diese wurde von Beygang in [8] eingeführt und wird in Abschnitt 5.6.2 zur Definition einer Schranke verwendet.

**Definition 5.14** (Unterinstanzen). Sei  $S \in \mathbb{S}$  mit  $k \leq d(S)$ . Dann nennen wir  $S' = \{S'(i_1), \dots, S'(i_k)\}$  eine Unterinstanz von  $S$ , wenn  $i_s \neq i_t \ \forall s \neq t$  und  $\emptyset \neq S'(i_k) \subset S(k)$ .

**Definition 5.15** (Verschiedene Unterinstanzen). Sei  $S \in \mathbb{S}$ . Wir nennen zwei Unterinstanzen  $D_1$  und  $D_2$  von  $S$ , also

$$D_1 = \{S'(i_1), \dots, S'(i_{k_1})\}$$

$$D_2 = \{S'(j_1), \dots, S'(j_{k_2})\}$$

verschieden, wenn sie kein Ziel gemeinsam haben, also  $i_{l_1} \neq j_{l_2} \ \forall l_1 \in \{1, \dots, k_1\}$  und  $l_2 \in \{1, \dots, k_2\}$ .

**Definition 5.16** (Disjunkte Unterinstanzen). Sei  $S \in \mathbb{S}$ . Wir nennen zwei Unterinstanzen  $D_1$  und  $D_2$  von  $S$ , also

$$D_1 = \{S'(i_1), \dots, S'(i_{k_1})\}$$

$$D_2 = \{S'(j_1), \dots, S'(j_{k_2})\}$$

disjunkt, wenn der letzte Wagen der einen Instanz vor dem ersten Wagen des anderen Zieles ankommt. d. h.:

$n_1 \in S'(i_{l_1}), n_2 \in S'(j_{l_2}) \Rightarrow n_1 < n_2 \ \forall l_1 \in \{1, \dots, k_1\}$  und  $l_2 \in \{1, \dots, k_2\}$   
oder

$n_1 \in S'(i_{l_1}), n_2 \in S'(j_{l_2}) \Rightarrow n_1 > n_2 \ \forall l_1 \in \{1, \dots, k_1\}$  und  $l_2 \in \{1, \dots, k_2\}$ .

## 5.4 TMP und minPS

### 5.4.1 Trennbare Ziele

Es sei  $G_S$  oder kurz  $G$  der zu einer Instanz  $S \in \mathbb{S}^n$  des TMP gehörende Intervallgraph.

Ein Ziel  $d$  ist getrennt, wenn es in einer zulässigen Gleisbelegung auf zwei verschiedene Gleise verteilt ist. Es gibt mindestens zwei Ziele, die nicht getrennt werden können. Für jedes andere trennbare Ziel gilt, dass Wagen dieses Zieles dann am Ende und Anfang zweier verschiedener Richtungsgleise stehen müssen.

Trennbare Ziele sollen hier nur kurz vorgestellt und, wo nötig, ergänzt werden.

**Definition 5.17.** Ein Tripel  $(a, b, c) \in S \times S \times S$  von Zielen einer Instanz  $S \in \mathbb{S}^n$  nennt man angeordnetes Tripel, wenn gilt  $first(a) < first(c)$ .

**Theorem 5.18** (Trennbarkeit A). *Gegeben eine Instanz  $S \in \mathbb{S}^n$ . In einem angeordneten Tripel  $(a, b, c)$  mit paarweise überlappenden Zielen sei in dem einfahrenden Zug zwischen  $first(c)$  und  $last(a)$  kein Wagen von  $b$ . Dann ist  $b$  trennbar.*

Anhand des Intervallgraphens können die Bedingungen für 5.18 nicht überprüft werden. Allerdings kann ein Algorithmus angegeben werden, der diese in linearer Zeit anhand von  $\sigma$  bzw.  $I_n$  überprüft.

**Lemma 5.19** (Trennbarkeit B). *Gegeben  $S \in \mathbb{S}^n$ . Existiert ein angeordnetes Tripel  $(a, b, c)$  von Zielen, wobei nur  $a$  und  $b$  sowie  $b$  und  $c$  überlappen, so kann  $b$  getrennt werden.*

Zur Notation seien noch folgende Definitionen erwähnt:

**Definition 5.20.** Ein bezüglich zweier weiterer Ziele  $a$  und  $c$  trennbares Ziel  $b$  wird mit dem Tripel  $\tau_b = (a, b, c)$  bezeichnet.

Es sei  $T(P) \subset V \times V \times V$  die Menge aller trennbaren Ziele in der Instanz  $S \in \mathbb{S}^n$  des TMP, d. h.

$$T(S) = \{(a, b, c) \in G \mid b \text{ ist bezüglich } a, c \text{ trennbar}\}$$

Dann spannt dies mit Kanten zwischen den Knoten  $(a, b, c)$  einen Subgraph  $B \subseteq G$  auf:

$$V(B) = \{v \mid v \in T(S)\}$$

$$E(B) = \{(u, v) \mid u, v \in \tau_j \text{ mit } \tau_j \in T(S)\}$$

Damit besteht der blaue Teilgraph von  $G$  nur aus der Vereinigung von Pfaden der Länge 3, wobei der mittlere Knoten trennbar ist.

Nun können nicht mehr alle Elemente  $T(S)$  notwendigerweise eindeutig in  $B \subseteq G$  identifiziert werden.

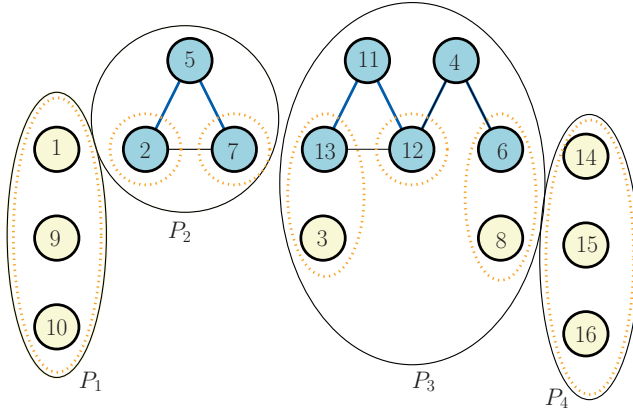
**Bemerkung 5.21.** Es sei der Pfad  $[a, b, c, d, e] \subset B$ . Ist  $b$  bezüglich  $a$  und  $c$  trennbar und  $d$  bezüglich  $c$  und  $e$ , so folgt daraus nicht zwangsläufig, dass  $c$  bezüglich  $b$  und  $d$  trennbar ist. Trotzdem ist der Pfad  $[b, c, d] \subset B$ . Es ist also zwangsläufig nötig, die Menge  $T(P)$  nicht nur zu berechnen, sondern auch zu speichern.

Es sei  $G_S$  oder kurz  $G$  der zu einer Instanz  $S \in \mathbb{S}^n$  des TMP gehörende Intervallgraph und  $P_i \subset G$  eine Teilmenge von  $G$  mit  $i \in \mathbb{N}^+$ . Gemäß Definition 3.10 kann nun  $P_i$  als pseudostabile Menge definiert werden. Zerlegt man den Graphen  $G$  in pseudostabile Mengen, so kann gezeigt werden, dass sich dadurch eine gültige Gleisbelegung für das TMP ergibt.

## 5.5 Der Zusammenhang zwischen minPS und TMP

Eine minimale Zerlegung des Intervallgraphen  $G$  zu einer Instanz  $S \in \mathbb{S}^n$  des TMP in stabile Mengen liefert eine zulässige Gleisbelegung, in der sich Ziele ein Gleis teilen, entsprechend der Knoten in den stabilen Mengen. Ziel dieses Abschnittes ist es zunächst zu zeigen, dass auch eine Zerlegung von  $G$  in pseudostabile Mengen eine zulässige Gleisbelegung liefert, die mindestens genauso gut ist wie eine Gleisbelegung, die durch stabile Mengen gegeben ist. Danach wird bewiesen, dass dieses Vorgehen sogar äquivalent zum TMP ist.

Gesucht ist dann eine Zerlegung  $P$  von  $G$  in pseudostabile Mengen einem Teilgraphen  $B$  mit trennbaren Zielen. Die pseudostabilen Mengen bestehen also anschaulich nur aus nicht adjazenten Mengen  $D_1, \dots, D_i$ . Erlaubt sind zusätzlich nur Kanten zwischen Knoten, die in einem  $\tau_i$  oder in verschiedenen stabilen Mengen liegen. Zu zeigen ist nun, dass diese pseudostabilen Mengen eine zulässige Gleisbelegung liefern. Eine Gleisbelegung aus diesen pseudostabilen Mengen erhält man, indem man alle Ziele einer stabilen Menge  $D_i$  auf ein Gleis rangiert und für jedes trennbare Ziel ein neues



**Abbildung 5.9:** Die pseudostabilen Mengen aus Beispiel 5.8. Die schwarzen Rahmen entsprechen pseudostabilen Mengen, die orangenen stabilen Mengen.

Gleis beginnt und das Ziel trennt. Diese Idee wird im Beweis von Lemma 5.22 formalisiert.

**Beispiel 5.8.** Gegeben sei eine Instanz  $S \in \mathbb{S}^{50}$  mit 16 Zielen und

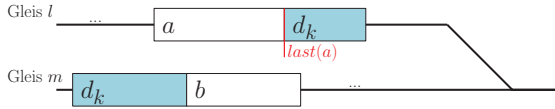
$$\begin{aligned} \sigma = & (1, 1, 2, 1, 2, 3, 3, 3, 4, 2, 2, 1, 5, 3, 3, 4, 2, 1, 1, 6, \\ & 6, 2, 5, 7, 8, 1, 9, 10, 8, 11, 12, 13, 2, 5, 8, 10, 14, 14, 15, 16, \\ & 16, 12, 7, 4, 10, 5, 7, 8, 13, 11) \end{aligned}$$

Eine gültige Zerlegung  $P$  des dazugehörigen Intervallgraphen  $G_S$  in pseudostabile Mengen findet sich durch

$$\begin{aligned} P_1 = \{2, 5, 7\} & \quad \text{mit } D_1^1 = \{2\}, D_2^1 = \{7\} \text{ und } \zeta(P_1) = 2 \\ P_2 = \{1, 9, 10\} & \quad \text{mit } D_1^2 = \{1, 9, 10\} \text{ und } \zeta(P_2) = 1 \\ P_3 = \{3, 4, 6, 8, 11, 12, 13\} & \quad \text{mit } D_1^3 = \{13, 3\}, D_2^3 = \{12\}, D_3^3 = \{6, 8\} \\ & \quad \text{und } \zeta(P_3) = 3 \\ P_4 = \{14, 15, 16\} & \quad \text{mit } D_1^4 = \{14, 15, 16\} \text{ und } \zeta(P_4) = 1 \end{aligned}$$

Die entsprechenden pseudostabilen Mengen sind in Abbildung 5.9 illustriert. Die schwarzen Rahmen entsprechen pseudostabilen Mengen, die orangenen stabilen Mengen. Es gilt  $\zeta(P) = 7$ .  $\triangle$

**Voraussetzung:** Gegeben sei eine pseudostabile Menge  $P \subseteq G$  eines Intervallgraphen  $G$  zu einer Instanz  $S \in \mathbb{S}^n$  mit stabilen Mengen  $D_1, \dots, D_j$  und blauen Knoten  $b_1, \dots, b_i \notin D_1 \cup \dots \cup D_j$ . Es sei  $|P| = n$ . Weiter sei die Bedingung erfüllt, dass



**Abbildung 5.10:** Die Gleisbelegung zum ersten Punkt im Beweis von Lemma 5.22.

- für alle Pfade  $(a, b, c)$  und alle Knoten  $v$  in der stabilen Menge von  $a$  gilt  $last(v) < first(a)$  und
- für alle Knoten  $v$  in der stabilen Menge von  $c$  gilt  $first(v) > last(b)$ .

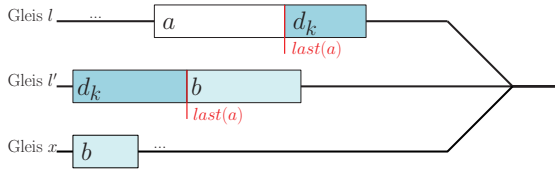
**Lemma 5.22.**  $P$  impliziert eine zulässige Gleisbelegung, die genau  $\zeta(P)$  Gleise benötigt.

*Beweis.* Es seien jetzt  $d_1, \dots, d_n$  alle Knoten der pseudostabilen Menge  $P$ . Sie seien aufsteigend so geordnet, dass für die mit ihnen verknüpften Ziele gilt  $first(d_1) < first(d_2) < \dots < first(d_n)$ .

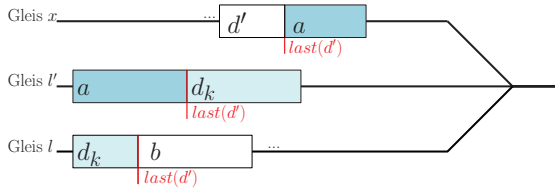
Betrachte jetzt für alle  $k \in \{1, \dots, n\}$  der Reihe nach jedes  $d_k$ .

- Ist  $d_k \in D_l$ , so kann das Ziel  $d_k$  komplett dem Gleis  $l$  zugewiesen werden. Dies liefert eine gültige Gleisbelegung, da die Menge  $D_l$  stabil ist und somit alle Ziele  $d_q \in D_l$  mit  $q < k$  schon abgeschlossen sind.
- Gilt jedoch  $d_k \notin D_1 \cup \dots \cup D_j$ , so muss ein eindeutiges Tripel  $(a, d_k, b)$  in  $T(P)$  existieren. Wir müssen jetzt vier Fälle unterscheiden:

1.  $a \in D_l, b \in D_m$ . So wird weder  $a$  noch  $b$  getrennt, beide gehören zu verschiedenen stabilen Mengen und werden verschiedenen Gleisen  $l$  und  $m$  zugewiesen.  $d_k$  wird nun so lange Gleis  $m$  zugewiesen, bis der letzte Wagen von  $a$  zugewiesen wird. Nach  $last(a)$  werden alle Wagen von  $d_k$  auf Gleis  $l$  zugewiesen. Dies funktioniert, da  $d_k$  bezüglich  $a$  und  $b$  trennbar ist und gemäß Definition auf Gleis  $l$  alle Ziele in  $D_l$  vor  $a$  zugewiesen werden und auf Gleis  $m$  alle Ziele in  $D_m$  nach  $b$  zugewiesen werden. Vergleiche hierzu Abb. 5.10.
2.  $a \in D_l, b \notin D_1 \cup \dots \cup D_j$ . Dem ersten Fall entsprechend wird das Ziel  $d_k$  getrennt, also zunächst einem Gleis  $l'$  zugewiesen, bis  $last(a)$  zugewiesen wurde, danach dem Gleis  $l$ ,  $a$  folgend. Auf Gleis  $l'$  können nun Wagen von  $b$  folgen, die bis dahin einem Gleis  $x$  zugewiesen wurden, vergleiche hierzu auch den dritten



**Abbildung 5.11:** Die Gleisbelegung zum zweiten Punkt im Beweis von Lemma 5.22.



**Abbildung 5.12:** Die Gleisbelegung zum dritten Punkt im Beweis von Lemma 5.22.

Punkt. Dies ist mit derselben Argumentation wie unter Punkt 1 eine gültige Gleisbelegung. Vergleiche hierzu Abb. 5.11.

3.  $a \notin D_1 \cup \dots \cup D_j$ ,  $b \in D_l$ . Das Ziel  $a$  wird nun, zu einem Zeitpunkt, nachdem  $first(d_k)$  zugewiesen wird, getrennt. Wurde es bis dahin einem Gleis  $l'$  zugewiesen und  $d_k$  Gleis  $l$ , kann es nun einem anderen Gleis zugewiesen werden. Dieser Vorgang wurde im zweiten Punkt genauer beschrieben. Gleis  $l'$  kann nun für Wagen des Zieles  $d_k$  benutzt werden, Gleis  $l$  für das Ziel  $b$ . Dies entspricht, mit derselben Argumentation wie in den vorhergehenden Punkten, einer zulässigen Gleisbelegung. Vergleiche hierzu auch Abb. 5.12.
4.  $a, b \notin D_1 \cup \dots \cup D_j$ . Werden sowohl  $a$  als auch  $b$  zusätzlich getrennt, lässt sich die Gleisbelegung genauso wie in Punkt Zwei und Drei finden.

Dies liefert eine gültige Gleisbelegung. Für jede stabile Menge  $D_1, \dots, D_j$  wird ein Gleis benötigt, also insgesamt  $M_P$  Gleise. Für jedes getrennte Ziel  $b$  in  $(a, b, c) \in T(P)$ , das in einem der Pfade  $p$  in  $T(P)$  enthalten ist, wird ein weiteres Gleis benötigt. Insgesamt ergibt sich also  $K(P) = \zeta(P)$ .  $\square$

Da die Gleisbelegung von Zielen in einer bestimmten pseudostabilen Men-

ge nur bestimmte Gleise betrifft, die von keiner anderen pseudostabilen Menge benutzt werden, ist es möglich, die Gleisbelegung auch für jede pseudostabile Menge in der Reihenfolge der Intervallenden ihrer Knoten zu bestimmen.

**Beispiel 5.9.** Es sei die Instanz  $S \in \mathbb{S}^{50}$  und die Zerlegung  $P$  des zugehörigen Intervallgraphen  $G_S$  in pseudostabile Mengen aus Beispiel 5.8 gegeben. Die dazugehörige Gleisbelegung ergibt sich nun also in der Reihenfolge der pseudostabilen Mengen:

- $P_1 = \{2, 5, 7\}$  mit  $D_1^1 = \{2\}$ ,  $D_2^1 = \{7\}$  und  $\zeta(P_1) = 2$  gibt eine Gleisbelegung für  $\zeta(P_1) = 2$  Gleise:

$$\begin{aligned} \text{Gleis 1} & : (2)[3, 33] (5)[34, 51] \\ \text{Gleis 2} & : (5)[1, 23] (7)[24, 47] \end{aligned}$$

- $P_2 = \{1, 9, 10\}$  mit  $D_1^2 = \{1, 9, 10\}$  und  $\zeta(P_2) = 1$  ist stabil und benötigt nur ein Gleis:

$$\text{Gleis 3} : (1)[1, 26] (9)[27, 27] (10)[28, 45]$$

- $P_3 = \{3, 4, 6, 8, 11, 12, 13\}$  mit  $D_1^3 = \{13, 3\}$ ,  $D_2^3 = \{12\}$ ,  $D_3^3 = \{6, 8\}$  und  $\zeta(P_3) = 3$  gibt eine Gleisbelegung für  $\zeta(P_3) = 3$  Gleise:

$$\begin{aligned} \text{Gleis 4} & : (3)[6, 15] (13)[32, 49] (11)[50, 51] \\ \text{Gleis 5} & : (11)[1, 30] (12)[31, 42] (4)[44, 51] \\ \text{Gleis 6} & : (4)[1, 16] (6)[20, 21] (8)[25, 48] \end{aligned}$$

- $P_4 = \{14, 15, 16\}$  mit  $D_1^4 = \{14, 15, 16\}$  und  $\zeta(P_4) = 1$  ist wieder stabil und benötigt nur ein Gleis:

$$\text{Gleis 7} : (14)[37, 38] (15)[39, 39] (16)[40, 41]$$

Es werden also genau  $\zeta(P) = 7$  Gleise benötigt.

△

Im Folgenden wird in Theorem 5.23 gezeigt, dass eine komplette Zerlegung eines Intervallgraphen in pseudostabile Mengen eine zulässige Gleisbelegung liefert, wobei genau so viele Gleise benötigt werden wie die Wertigkeit der Zerlegung. Mit Theorem 5.24 wird anschließend bewiesen, dass die beiden Probleme sogar äquivalent sind. Dazu wird aus einer gültigen Gleisbelegung eine Zerlegung des Intervallgraphen in pseudostabile Mengen konstruiert, deren Wertigkeit genau der Anzahl der benötigten Gleise entspricht.



**Theorem 5.23.** Eine Zerlegung  $P$  des Intervallgraphen  $G_S$  zu einer Instanz des TMP  $S \in \mathbb{S}^n$  in pseudostabile Mengen  $P_1, \dots, P_i$  liefert eine zulässige Gleisbelegung mit  $\zeta(P)$  Gleisen. Damit gilt

$$K(S) \leq \zeta(G)$$

*Beweis.* Sei  $k = 0$ . Betrachte jede pseudostabile Menge  $P_j \subset G_S$  der Zerlegung  $P$  mit  $1 \leq j \leq i$ :

- Ist  $P_j$  stabil, so findet sich die Gleisbelegung, indem alle Ziele in  $P_j$  einem Gleis zugewiesen werden. Es wird also genau ein Gleis benötigt, damit gilt  $k = k + 1 = k + \zeta(P_j)$ .
- Ist  $P_j$  nicht stabil, aber pseudostabil, gilt also  $\zeta(P_j) > 1$ , wende Lemma 5.22 an. Dies gibt eine zulässige Gleisbelegung mit  $\zeta(P_j)$  Gleisen. Damit gilt auch in diesem Falle  $k = k + \zeta(P_j)$ .

Also ist  $k = \sum_{i=1}^j \zeta(P_i) = \zeta(P)$ . □

**Theorem 5.24.** *minPS* ist äquivalent zu TMP.

*Beweis.* Sei  $S \in \mathbb{S}^n$  eine Instanz des TMP mit  $S = \{S(1), S(2), \dots, S(d)\}$ . Sei  $G = G_S$  der zugehörige Intervallgraph.

**Behauptung:**  $K(S) = a \Leftrightarrow \zeta(G) = a$

„ $\Rightarrow$ “ Es existiert eine zulässige Gleisbelegung, die  $K(S) = a$  Gleise benötigt. Das heißt, es existiert eine  $a$ -mal wiederholte Folge von Zahlen mit einer Permutation  $\pi$ , so dass die Folge zuerst die Elemente von  $S(\pi(1))$ , gefolgt von  $S(\pi(2))$  etc. beinhaltet:

$$\underbrace{1, 2, \dots, n, 1, \dots, n, \dots, 1, \dots, n}_{a\text{-mal}}$$

Sei  $v_{\pi(1)}$  der Knoten im Intervallgraphen  $G_S$ , der mit dem Ziel  $S(\pi(1))$  verbunden ist. Definiere nun alle pseudostabilen Mengen  $P_1, \dots, P_i$  wie folgt:

Es sei zunächst  $h := 0$  das aktuelle Gleis bzw. die aktuelle Runde und  $i, k := 1$  sowie  $j := 0$ , wobei  $i$  das aktuelle Ziel,  $j$  die aktuelle pseudostabile Menge und  $k$  die aktuelle stabile Menge  $D_k^j$  in  $P_j$  sei. Betrachte nun für jedes  $i = 1, \dots, d$  das Ziel  $\pi(i)$ :

1. Ist  $S(\pi(j))$  ausschließlich in der  $h + 1$ -ten Runde, so setze  $h := h + 1$ ,  $k := 1$  und  $j := j + 1$ . Es wird also eine neue pseudostabile Menge  $P_j$  und eine neue stabile Menge  $D_1^j$  erzeugt.  $v_{\pi(i)}$  kann nun  $D_1^j$  hinzugefügt werden. Dies ist offensichtlich kein Problem, weil dieser Knoten als einziger Knoten in dieser Menge stabil ist.
2. Ist  $S(\pi(j))$  ausschließlich in der  $h$ -ten Runde, so kann  $v_{\pi(i)}$  nun  $D_k^j$  hinzugefügt werden. Ein Ziel, das vorher in dieser Runde vorhanden ist, ist entweder schon abgeschlossen und hat damit keine Kante zu  $v_{\pi(i)}$ , womit die Menge  $D_k^j$  stabil bleibt, oder wurde getrennt und ist als solches, wie im dritten Punkt beschrieben wird, keine Teilmenge von  $D_k^j$ .
3. Ist  $S(\pi(j))$  in der  $h$ -ten und  $h + 1$ -ten Runde, so wird es getrennt. Es wird nun  $P_j$  hinzugefügt, ohne jedoch einer stabilen Menge hinzugefügt zu werden. Füge dazu noch  $(v_{\pi(i-1)}, v_{\pi(i)}, v_{\pi(i+1)}) \in B T(P_j)$  hinzu. Dieses Tripel ist offensichtlich in  $R(G)$ , da eine zulässige Gleisbelegung betrachtet wird. Für alle Ziele  $d'$  in  $D_k^j$  gilt nun, dass  $first(d') < first(S(\pi(i-1)))$  und für alle Ziele  $d'$  in  $D_{k+1}^j$   $first(d') > first(S(\pi(i+1)))$ , da die entsprechenden Ziele vor bzw. nach der Zuweisung der Ziele  $\pi(i-1)$  bzw.  $\pi(i+1)$  betrachtet wurden oder werden. Setze nun  $h := h + 1$  und  $k := k + 1$ , da wir sowohl ein neues Gleis als auch eine neue stabile Menge  $D_k^j$  betrachten.

Dies liefert eine zulässige Zerlegung von  $G$  in pseudostabile Mengen, denn durch die ersten beiden Punkte werden den Mengen  $D_k^j$  in  $P_j$  nur Knoten hinzugefügt, die untereinander keine Kanten haben. Sie sind zusammen auf einem Gleis und dürfen deswegen in einer zulässigen Gleisbelegung nicht überlappen. Durch den dritten Punkt können, wenn  $v_{\pi(j)}$  hinzugefügt wird, nur die folgenden Konstellationen mit dem vorhergehenden und folgenden Ziel auftreten:

- $v_{\pi(j)}$  überlappt mit  $v_{\pi(j-1)}$  und  $v_{\pi(j+1)}$  oder
- $v_{\pi(j)}$  überlappt mit  $v_{\pi(j-1)}$  und  $v_{\pi(j+1)}$ , die beide wieder selber überlappen, was beides erlaubte trennbare Konstellationen sind.

Überlappten  $v_{\pi(j-1)}$ ,  $v_{\pi(j)}$  und  $v_{\pi(j+1)}$  nicht, bliebe die Menge also (pseudo-)stabil. Dieser Fall kann nicht auftreten, da wir minimale Lösungen betrachten und das Trennen von nicht überlappenden Zielen nicht zu einer Minimierung der benötigten Gleise führt.

Alle anderen Kanten, etwa zwischen Knoten in verschiedenen stabilen Mengen  $D_k^j$  und  $D_{k+1}^j$ , werden nicht ausgeschlossen, widersprechen aber der Definition einer pseudostabilen Menge auch nicht.

Die Wertigkeit  $\zeta(P_i)$  der pseudostabilen Mengen  $P_i$  entspricht der Anzahl ihrer stabilen Mengen  $D_k^i$  und der trennbaren Ziele, die direkt auf ein trennbares Ziel folgen. Es gilt  $\zeta(G) = a$ , da sich bei jeder neuen Runde entweder die Anzahl der pseudostabilen Mengen oder die Wertigkeit einer dieser erhöht. Es folgt also, dass es zu jeder zulässigen Gleisbelegung einer Instanz  $S$  des TMP, die  $a$  Gleise benötigt, eine Zerlegung  $P$  des zugehörigen Intervallgraphen  $G_S$  gibt mit

$$\zeta(P) = a \text{ und } K(S) = \zeta(G)$$

„ $\Leftarrow$ “ Sei eine Zerlegung  $P$  von  $G_S$  in pseudostabile Mengen gegeben. Die Behauptung ergibt sich sofort aus Theorem 5.23.  $\square$

Es gibt also für jede optimale Lösung einer Instanz  $S \in \mathbb{S}^n$  des TMP, die  $K(S)$  Gleise benötigt, eine dazugehörige Zerlegung des Intervallgraphen  $G_S$  in pseudostabile Mengen.

**Beispiel 5.10.** Es sei folgende Instanz  $S \in \mathbb{S}^{50}$  mit  $d = 16$ ,

$$\begin{aligned} \sigma = & (1, 1, 2, 3, 2, 4, 4, 3, 2, 3, 1, 4, 5, 6, 2, 7, 8, 9, 8, 6, 4, 10, 10, 4, \\ & 11, 12, 11, 13, 11, 10, 6, 10, 6, 8, 10, 3, 14, 9, \\ & 15, 16, 10, 14, 12, 13, 10, 11, 15, 14, 3, 12) \end{aligned}$$

und der folgenden Gleisbelegung gegeben:

$$\begin{aligned} \text{Gleis 1 : } & (1)[1, 11] (5)[13, 13] (6)[14, 33] (16)[40, 40] (10)[41, 51] \\ \text{Gleis 2 : } & (10)[1, 35] (3)[36, 51] \\ \text{Gleis 3 : } & (3)[1, 10] (9)[18, 38] (15)[39, 47] \\ \text{Gleis 4 : } & (2)[3, 15] (7)[16, 16] (8)[17, 34] (14)[37, 48] \\ \text{Gleis 5 : } & (4)[6, 24] (11)[25, 46] \\ \text{Gleis 6 : } & (12)[1, 51] \\ \text{Gleis 7 : } & (13)[28, 44] \end{aligned}$$

Nun ist also

$$\begin{aligned} P_1 = \{1, 5, 6, 16, 10, 3, 9, 15\} \text{ mit } D_1^1 &= \{1, 5, 6, 16\}, D_2^1 = \{9, 15\} \text{ und } \zeta(P_1) = 3 \\ P_2 = \{2, 7, 8, 14\} \text{ mit } D_1^2 &= \{2, 7, 8, 14\} \text{ und } \zeta(P_2) = 1 \\ P_3 = \{4, 11\} \text{ mit } D_1^3 &= \{4, 11\} \text{ und } \zeta(P_3) = 1 \\ P_4 = \{12\} \text{ mit } D_1^4 &= \{12\} \text{ und } \zeta(P_4) = 1 \\ P_5 = \{13\} \text{ mit } D_1^5 &= \{13\} \text{ und } \zeta(P_5) = 1 \end{aligned}$$

$\triangle$

Da  $\text{minPS } \mathcal{NP}$ -vollständig ist, vgl. Abschnitt 4.2, gilt auch, dass  $\text{TMP } \mathcal{NP}$ -vollständig ist. Dieses Resultat findet sich allerdings schon in der Literatur, etwa bei Beygang (vgl. [8]). Der dort angebrachte Beweis soll hier nicht nochmal angeführt werden.

## 5.6 Bekannte Abschätzungen und Algorithmen

Es sollen nun zunächst bekannte Abschätzungen, Algorithmen und Heuristiken kurz dargestellt werden.

### 5.6.1 Einfache Abschätzungen

In der Literatur finden sich bereits eine große Anzahl von einfachen Abschätzungen. Eine erste obere Schranke in Abhängigkeit zur Wagenanzahl  $n$  findet sich in der Arbeit von Dahlhaus, Horak, Miller und Ryan:

**Theorem 5.25.** [S. 48 16] Sei  $S \in \mathbb{S}^n$ . Dann gilt

$$K(S) \leq \left\lceil \frac{n}{4} + \frac{1}{2} \right\rceil = u_D(S)$$

Eine weitere Schranke lässt sich in Abhängigkeit von der Zielanzahl  $d$  finden:

**Theorem 5.26.** [S. 36 8] Ist  $S \in \mathbb{S}^n$  eine überlappende Instanz so gilt

$$K(S) \geq \left\lceil \frac{d+1}{2} \right\rceil$$

In der Arbeit von Beygang [8] findet sich des weiteren die folgende Schranke:

**Theorem 5.27.** [S. 39 8] Ist  $S \in \mathbb{S}^n$  so gilt

$$K(S) \geq \left\lceil \frac{\omega(S)+1}{2} \right\rceil := l_{\text{greedy}}$$

In der Notation von Kapitel bedeuten die beiden Theoreme: Gegeben sei ein Intervallgraph  $G$  mit  $n$  Knoten und einem Teilgraphen  $B \subset G$ . Dann ist die Wertigkeit einer minimalen Zerlegung des Graphen  $G$  in pseudostabile Mengen durch  $\zeta(G) \geq \left\lceil \frac{n+1}{2} \right\rceil$  und  $\zeta(G) \geq \left\lceil \frac{\omega(G)+1}{2} \right\rceil$  beschränkt.

### 5.6.2 CLIQUE BOUND

Gemäß der Definitionen in Abschnitt 5.3.2 leitete Beygang eine Schranke über die Cliquenzahl von Unterinstanzen, die sogenannte *Clique Bound* her:

**Theorem 5.28.** [S. 41 8] Sei  $S \in \mathbb{S}^n$  mit zwei verschiedenen und disjunkten Unterinstanzen  $D_1$  und  $D_2$  von  $S$

$$D_1 = \{S'(i_1), \dots, S'(i_{k_1})\}$$

$$D_2 = \{S'(j_1), \dots, S'(j_{k_2})\}$$

wobei der letzte Wagen von  $D_1$  vor dem ersten von  $D_2$  auf ein Richtungs-gleis rangiert wird. Dann gilt

$$K(S) \geq \left\lceil \frac{\omega(D_1) + \omega(D_2)}{2} \right\rceil$$

In [S. 43 ff. 8] wird ein Algorithmus vorgestellt, der in  $O(n^5)$  eine zulässige Lösung des TMP berechnet und die in 5.28 beschriebenen verschiedenen und disjunkten Unterinstanzen  $D_1$  und  $D_2$  findet, die die untere Schranke maximieren.

In Notation von Kapitel bedeutet dies: Gegeben sei ein Intervallgraph  $G$  mit einem Teilgraphen  $B \subset G$  sowie zwei disjunkte Teilgraphen  $G', G'' \subset G$  wobei  $G = G' \cup G''$ . Dann ist die Wertigkeit einer minimalen Zerlegung des Graphen  $G$  in pseudostabile Mengen durch  $\zeta(G) \geq \left\lceil \frac{\omega(G') + \omega(G'')}{2} \right\rceil$  beschränkt.

### 5.6.3 Untere Schranke durch die Cliquenzahl

Natürlich gilt auch die Abschätzung in Lemma 4.22, Für einen zu einer Instanz  $S$  des TMP gehörenden Intervallgraphen  $G_S$  gilt:

$$K(S) \geq l_C(G_S) \tag{5.1}$$

Eine ähnliche Schranke ist  $l_{ps}$  definiert, die allerdings eine andere Summenformel nutzt. Es sei wieder

$$C'_i = C_i \setminus \bigcup_{j=1}^{i-1} C_j$$

und damit definiere  $p : \mathbb{N} \rightarrow \{0, 1\}$

$$p(i) = \begin{cases} 1 & |C'_i| - \sum_{j=0}^i p(j) > 0 \\ 0 & \text{sonst} \end{cases}$$

mit  $p(0) = 0$ . Es sei weiter

$$\begin{aligned} l_{ps}(G) = & \left\lceil \frac{|C_1|+1}{2} \right\rceil + \max \left\{ 0, \left\lceil \frac{|C'_2|-p(1)+1}{2} \right\rceil \right\} \\ & + \dots + \max \left\{ 0, \left\lceil \frac{|C'_{m-1}|-\sum_{i=1}^{m-1} p(i)+1}{2} \right\rceil \right\} \\ & + \max \left\{ 0, \left\lceil \frac{|C'_m|-\sum_{i=1}^m p(i)+1}{2} \right\rceil \right\} \end{aligned}$$

Es gilt die folgende Schranke:

**Theorem 5.29.** *Es gilt für jede Instanz  $S \in \mathbb{S}^n$  des TMP mit dem zugehörigen Intervallgraphen  $G_S$*

$$K(S) \geq l_{ps}(G_S) \tag{5.2}$$

#### 5.6.4 GREEDY

GREEDY wurde von Beygang in [S. 50 ff. 8] eingeführt. Er berechnet eine zulässige Lösung des TMP, die mit der Färbung des zugehörigen Intervallgraphens zusammenhängt: GREEDY benutzt exakt  $\omega(G_S)$  Gleise.

Soll ein Wagen  $c$  auf die Gleise rangiert werden, so wird unterschieden:

- Existiert ein Gleis  $t$  mit  $S(c) = S(\text{last}(t))$ , so wird der Wagen  $c$  diesem Gleis zugewiesen.
- Sonst nehme ein Gleis  $t$ , wobei das Ziel  $S(\text{last}(t))$  schon abgeschlossen ist, sonst
- wird  $c$  einem neuen Gleis zugewiesen.

Eine Färbung des zu einer Instanz  $S \in \mathbb{S}^n$  gehörenden Intervallgraphen  $G_S$  weist allen adjazenten Knoten verschiedene Farben zu, also jeweils zwei überlappenden Zielen, die nicht einem Gleis zugewiesen werden dürfen. Also ist jede Lösung eines TMP der Form **st,ub|nsh,se,0-sp|fr,g-bl** (also ein TMP, in dem wir das Trennen von Zielen auf verschiedene Gleise verbieten) äquivalent zu einer Graphenfärbung im zugehörigen Intervallgraphen.

### 5.6.5 Ein weiterer Greedyansatz: Greedy-PS

In diesem Abschnitt wird ein Greedyansatz diskutiert. Es sei wieder die größte Clique  $C$  des Intervallgraphen  $G_S$  zu einer Instanz  $S \in \mathbb{S}^n$  des TMP und die Liste  $T = (t_1, \dots, t_{t_C})$  aller trennbaren Ziele in  $C$  gegeben, wobei letztere zunächst nach dem ersten, dann nach dem zweiten und dem dritten Knoten aufsteigend sortiert sei.

Entsprechend der Eingabe von Knoten gibt die Funktion *add* auch für trennbare Ziele  $(a, b, c)$  *true* oder *false* zurück, je nach dem, ob das Tripel der pseudostabilen Menge hinzugefügt werden konnte oder nicht. Idee des Greedyansatzes ist nun, lokal die beste Lösung zu erhalten. Dazu wird versucht, einer pseudostabilen Menge  $P_1$  jedes trennbare Tripel in  $T$  hinzuzufügen. Anschließend werden alle Elemente aus  $T$ , die in  $P_1$  vorhandene Knoten benutzen, entfernt und der Ansatz für weitere pseudostabile Mengen wiederholt, bis kein Element mehr in  $T$  vorhanden ist. Die restlichen Knoten von  $C$  müssen nun einer eigenen (pseudo-) stabilen Menge hinzugefügt werden.

Anschließend werden alle weiteren Knoten des Graphen in die vorhandenen pseudostabilen Mengen eingefügt bzw. neue pseudostabile Mengen erzeugt.

Der Algorithmus hat polynomielle Laufzeit. Da maximal  $|T| = t_C$  trennbare Ziele in der größten Clique  $C$  vorhanden sind, wird die erste while-Schleife in Zeile 3 nicht öfter als  $\omega(G) \cdot t_C$  ausgeführt und es gibt nicht mehr als  $t_C$  pseudostabile Mengen. Im ungünstigsten Fall wird sie  $n$  mal ausgeführt. Die Funktion in Zeile 5 muss im zugehörigen Graphen nach Kreisen suchen. Diese Laufzeit hängt von der Anzahl der Knoten und Kanten ab und beträgt nach Abschnitt 2.3  $O(V(G) \cdot E(G))(C + 1)$ . Gehen wir von einem Zyklus  $C$  aus und maximal  $\frac{n}{2}$  Knoten und Kanten im gerichteten Graphen ergibt sich eine Laufzeit von  $\frac{n}{2} \cdot \frac{n}{2} \cdot 2 = \frac{n}{2}$  und damit insgesamt  $\frac{1}{2}n^3$ . Die For-Schleife in Zeile 15 wird maximal  $n$  mal durchlaufen und die Funktion *add* in Zeile 24 hat eine Laufzeit von maximal  $n$ . Damit ergibt sich eine Laufzeit von  $f(n) = (\frac{1}{2}n^3 + n^2) = O(n^3)$ .

**Beispiel 5.11.** Gegeben sei wieder eine Instanz  $S \in \mathbb{S}^{50}$  mit  $d = 16$  und

$$\sigma = (1, 1, 2, 2, 2, 1, 2, 1, 3, 1, 4, 5, 2, 6, 4, 7, 2, 8, 3, 9, 5, 10, 9, 5, 11,$$

$$12, 13, 2, 7, 13, 10, 11, 9, 6, 1, 3, 14, 3, 15, 16, 11, 6, 15, 6, 15, 9, 6, 11, 4, 15)$$

---

**Algorithmus 3** GREEDY-PS

---

**Eingabe:** Graph  $G$  mit größter Clique  $C$  und Liste  $T = (t_1, \dots, t_{t_C})$  von trennbaren Zielen in  $C$ .

**Ausgabe:** Zerlegung  $P$  von  $G$  in pseudostabile Mengen

```
|1| besucht =  $\emptyset$ 
|2| zaehler = 0
|3| while  $|T| > 0$  do
|4|   zaehler ++
|5|    $P.add(\text{pseudostabile Menge } P_{\text{zaehler}})$ 
|6|   for jedes  $(a, b, c) = t_i \in T$  do
|7|     if  $P_{\text{zaehler}.add(a, b, c) = true$  then
|8|       besucht.add  $a, b, c$ 
|9|     end if
|10|  end for
|11|  for jedes  $v \in \textit{besucht}$  do
|12|    Lösche jedes  $t_i$  aus  $T$ , dass den Knoten  $v$  beinhaltet
|13|  end for
|14| end while
|15| for jeder Knoten  $v \in V(G)$  do
|16|   if  $v \notin \textit{besucht}$  then
|17|     for  $i = 1, \dots, \textit{zaehler}$  do
|18|       if  $P_i.add(v) = true$  then
|19|         besucht.add  $v$ 
|20|         exit
|21|       end if
|22|     end for
|23|     zaehler ++
|24|      $P.add(\text{pseudostabile Menge } P_{\text{zaehler}})$ 
|25|      $P_{\text{zaehler}.add(v)$ 
|26|   end if
|27| end for
|28| return  $P$ 
```

---



GREEDY-PS liefert nun folgende Gleisbelegung:

Gleis 1 : (1)[1, 35] (3)[36, 38]  
 Gleis 2 : (3)[9, 18] (9)[20, 46]  
 Gleis 3 : (2)[3, 28] (4)[49, 49]  
 Gleis 4 : (4)[11, 15] (7)[16, 29] (10)[31, 31]  
 Gleis 5 : (10)[22, 22] (11)[25, 48]  
 Gleis 6 : (5)[12, 24] (13)[27, 30] (14)[37, 37] (15)[39, 50]  
 Gleis 7 : (6)[14, 47]  
 Gleis 8 : (8)[18, 18] (12)[26, 26] (16)[40, 40]

Eine minimale, zulässige Lösung benötigt 5 Gleise, einfaches Färben benötigt 10 Gleise.  $\triangle$

### 5.6.6 Ganzzahliges lineares Programm

In der Arbeit von Beygang [8] wird ein ganzzahliges lineares Programm beschrieben, um eine obere und untere Schranke bzw. die exakte Lösung des TMP zu bestimmen.

Jede zulässige Gleisbelegung von  $S \in \mathbb{S}^n$  trennt ein Ziel  $k$  oder weist es eindeutig einem Gleis zu. Jede Gleisbelegung von  $S(k)$  kann also als Partition von  $S(k)$  in zwei Mengen  $S_1(k)$  und  $S_2(k)$  betrachtet werden, wobei  $S_1(k) \cup S_2(k) = S(k)$ ,  $S_1(k) \cap S_2(k) = \emptyset$  und eine der beiden Mengen leer sein kann, wenn  $k$  nicht getrennt wird.

Die verschiedenen Partitionen können als

$$S_{k,i} = \{S_1^i(k), S_2^i(k)\} \text{ mit } i \in \{1, \dots, |S(k)|\}$$

angegeben werden.

**Beispiel 5.12.** Für  $n = 8$  und  $S(k) = \{1, 3, 7, 8\}$  existieren die Partitionen  $b_{k,1} = \{\{1, 3, 7, 8\}, \emptyset\}$ ,  $b_{k,2} = \{\{1, 3, 7\}, \{8\}\}$ ,  $b_{k,3} = \{\{1, 3\}, \{7, 8\}\}$ , und  $b_{k,4} = \{\{1\}, \{3, 7, 8\}\}$ .  $\triangle$

In jeder Partition wird ein bestimmter Abschnitt des Gleises, nach der Definition von 5.1, einer Runde  $1, \dots, n$  benutzt. Da ein Ziel, wenn es getrennt wird, stets am Anfang eines Gleises und am Ende eines anderen Gleises steht, kann die Partition als Gleisblockung formuliert werden, also für  $b_{k,1}$  etwa  $b_{k,1} = [1, 8]$ ,  $b_{k,2} = [1, 7] \cup [8, n]$ ,  $b_{k,3} = [1, 3] \cup [7, n]$  und

$b_{k,4} = [1, 1] \cup [3, n]$ . Es existieren also für ein Ziel  $S(k) = \{k_1, \dots, k_{|S(k)|}\}$  genau  $|S(k)|$  Gleisblockungen mit

$$b_{k,1} = [k_1, k_{|S(k)|}]$$

und

$$b_{k,j} = [1, k_{j-1}] \cup [k_j, n] \quad \text{für } j = 2, \dots, |S(k)|$$

Um ein IP herzuleiten, wird in der Arbeit von Beygang [8] der gerichtete Graph der zulässigen Gleisbelegung betrachtet. Vergleiche hierzu Definition 5.2 und Theorem 5.3

Wird die azyklische Eigenschaft des Graphen relaxiert, kann das TMP durch das folgende IP dargestellt werden:

$$\begin{array}{ll}
 \text{[TMP-IP]} & \min \quad k \\
 & \text{s.d.} \quad \sum_{j=1, \dots, |S(i)|} x_{i,j} = 1 \quad \forall i = 1, \dots, d \quad \text{(TMP-IP.1)} \\
 & \quad \quad \sum_{(i,j): t \in b_{i,j}} x_{i,j} \leq k \quad \forall t = 1, \dots, d \quad \text{(TMP-IP.2)} \\
 & \quad \quad x_{i,j} \in \{0, 1\} \quad \forall i = 1, \dots, d, j = 1, \dots, |S(i)|
 \end{array}$$

Bedingung TMP-IP.1 stellt sicher, dass für jedes Ziel genau eine Gleisblockung gewählt wird. Bedingung TMP-IP.2 stellt sicher, dass nicht mehr als  $k$  Gleise benutzt werden, indem jedes  $i \in \{1, \dots, n\}$  in nicht mehr als  $k$  Mengen der gewählten Gleisblockungen vorkommt.

Die Lösung von TMP-IP mit  $S \in \mathbb{S}^n$  wird als *Pseudogleisbelegung* bezeichnet; jede Pseudogleisbelegung ist eine zulässige Gleisbelegung, wenn  $G_{S,tr}$  azyklisch ist.

**Beispiel 5.13.** Für die folgende Instanz  $S \in \mathbb{S}^{30}$  mit  $d = 8$  Zielen und

$$\sigma = (1, 1, 2, 2, 3, 4, 2, 5, 5, 2, 5, 3, 5, 3, 3, 4, 6, 1, 7, 5, 8, 8, 8, 7, 4, 1, 7, 6, 7, 6)$$

liefert das IP eine zulässige Gleisbelegung:

$$\begin{array}{ll}
 \text{Gleis 1 :} & (6)[1, 17] (1)[18, 31] \\
 \text{Gleis 2 :} & (1)[1, 2] (2)[3, 10] (7)[19, 29] \\
 \text{Gleis 3 :} & (4)[1, 6] (5)[8, 20] (8)[21, 23] (6)[28, 31] \\
 \text{Gleis 4 :} & (3)[5, 15] (4)[16, 31]
 \end{array}$$

Bei dieser Notation steht in den runden Klammern das jeweilige Ziel, wohingegen in den eckigen Klammern das jeweils benutzte (Teil-)Intervall steht.  $\triangle$

TMP-IP liefert eine untere Schranke für das TMP. Diese wird mit  $l_C(S)$  bezeichnet. Es gilt also für eine Instanz  $S \in \mathbb{S}^n$ :

$$l_C(S) \leq K(S)$$

**Lemma 5.30.** [S. 33 8] Sei  $S \in \mathbb{S}^n$ . Jede Pseudogleisbelegung kann in eine zulässige Gleisbelegung umgewandelt werden, indem für jeden Zyklus in  $G_{S,tr}$  ein weiteres Gleis hinzugefügt wird.

*Beweis.* Eine gerichtete Kante im gerichteten Graphen der Pseudogleisbelegung zwischen zwei Zielen kann es nur geben, wenn ein Ziel auf ein anderes auf einem Gleis folgt. Wenn man alle Wagen eines getrennten Zieltes im Zyklus nun einem neuen Gleis zuweist, ist dieser aufgelöst.  $\square$

Ist  $n_C$  die Anzahl der Zyklen im gerichteten Graphen der Gleisbelegung bzw. Pseudogleisbelegung, so kann aus  $l_C$  eine obere Schranke hergeleitet werden:

$$u_C(S) = l_C(S) + n_C(S)$$

Es gilt also für eine Instanz  $S \in \mathbb{S}^n$ :

$$K(S) \leq u_C(S)$$

Jede zulässige Lösung kann maximal einen Zyklus pro Gleis generieren. Also gilt  $n_C(S) \leq l_C(S)$ . Damit und durch die Definition folgt:

$$l_C(S) \leq K(S) \leq u_C(S) = l_C(S) + n_C(S) \leq l_C(S) + l_C(S) = 2l_C(S)$$

$$u_C(S) \geq K(S) \geq l_C(S) = u_C(S) - n_C(S) \geq u_C(S) - \frac{1}{2}u_C(S) = \frac{1}{2}u_C(S)$$

**Theorem 5.31.** [S. 33 8] Für  $S \in \mathbb{S}^n$  gilt

$$l_C(S) \leq K(S) \leq 2l_C(S)$$

und

$$\frac{1}{2}u_C(S) \leq K(S) \leq U_C(S)$$

**Beispiel 5.14.** Für die folgende Instanz  $S \in \mathbb{S}^{30}$  mit  $d = 8$  Zielen und

$$\sigma = (1, 1, 1, 2, 1, 1, 2, 2, 1, 3, 4, 2, 4, 3, 2, 2, 3, 4, 5, 3, 5, 5, 3, 2, 4, 2, 3, 5, 1, 2)$$

liefert das IP keine zulässige Gleisbelegung:

$$\begin{aligned} \text{Gleis 1 :} & \quad (3)[1, 23] \quad (2)[24, 31] \\ \text{Gleis 2 :} & \quad (2)[1, 16] \quad (5)[19, 28] \quad (1)[29, 31] \\ \text{Gleis 3 :} & \quad (1)[1, 9] \quad (4)[11, 25] \quad (3) [27, 31] \end{aligned}$$

Der Graph  $G_{S,tr}$  hat einen Zyklus. Eine zulässige Gleisbelegung kann mit einem zusätzlichen Gleis hergestellt werden.  $\triangle$

## 5.7 Neue Resultate

### 5.7.1 Laufzeitverbesserung für Greedy-PS

Der Greedyansatz aus Abschnitt 5.6.5 kann in seiner Laufzeit deutlich verbessert werden. Die Funktion `add` überprüft in jedem Schritt, ob es im Graph der Zerlegung in pseudostabile Mengen ein Zyklus existiert. Dieser Aufwand kann reduziert werden, wenn ein entsprechendes Objekt generiert wird.

Das Objekt `pseudostabile_menge` hat dabei die Funktion `add(n)`, um zu überprüfen, ob ein Knoten `n` einer der stabilen Mengen hinzugefügt werden kann. Die Funktion `add_tau(a,b,c)` wiederum fügt einen Pfad  $(a, b, c)$  hinzu. Dazu muss nur überprüft werden, ob dieses Tripe rechts oder links angefügt werden kann. Dies kann in folgenden Schritten geschehen:

- Es darf nur  $a$  in der letzten stabilen Menge  $D_m$ , oder  $c$  in der ersten stabilen Menge  $D_1$  enthalten sein. Sonst muss überprüft werden, ob sie einer dieser Mengen hinzugefügt werden kann. Wenn nicht, muss `false` zurückgegeben werden.
- Sonst kann eine neue stabile Menge  $D_0$  mit  $a$  oder  $D_{m+1}$  mit  $c$  generiert werden und der Pfad entsprechend in die Datenstruktur hinzugefügt werden.

Dies liefert in jedem Fall eine gültige pseudostabile Menge, da die Knoten immer entsprechend ihres Intervallanfangs sortiert sind. Die Laufzeit dieses Ansatzes ist linear in  $n$ , d. h. die Laufzeit der Heuristik 3 ist nun quadratisch,  $O(n^2)$ . Es bleibt zu untersuchen, wie die Laufzeit auf zufallsgenerierten Eingabedaten ist.

### 5.7.2 Coloring-PS

Eine Heuristik für allgemeine Intervallgraphen ergibt sich aus dem in Abschnitt 4.5 beschriebenen Algorithmus. Jeder Intervallgraph ist ein chordaler Graph, allerdings besteht der Graph  $B \subset G$  nicht notwendigerweise nur aus disjunkten, großen Diamanten. Die berechnete Lösung muss also nicht die Optimallösung sein. Wie sich der Algorithmus für beliebige Instanzen des TMP verhält, wird in einem späteren Abschnitt betrachtet.

Die Lösung kann aber – wie in Abschnitt 4.5 – nicht schlechter als eine normale Graphenfärbung werden. Er liefert also die obere Schranke  $u_{col}$ .

---

**Algorithmus 4** COLORING-PS

---

**Eingabe:** perfektes Eliminationsschema  $[v_1, v_2, \dots, v_n]$  des chordalen Graphen  $G$  und Liste  $T = (t_1, \dots, t_{i_C})$  von trennbaren Zielen in  $C$ .

**Ausgabe:** Zerlegung  $P$  von  $G$  in pseudostabile Mengen mit  $f : V \rightarrow \{1, \dots, \zeta(G)\}$ ,  $Pf$

```
[1]  $f(v) = 0$  für alle Knoten.  
[2]  $\max f = 0$   
[3]  $G_P = \emptyset$ ,  $Pf = \emptyset$   
[4] for  $i = n, \dots, 1$  do  
[4]   Wähle für  $f(v_n)$  die kleinste freie Farbklasse.  
[5]   if  $(v_n) > \max f$  then  
[6]     if  $v_n$  ist ein mittlerer Knoten  $v$  in  $T$  und ( $f(u) \neq f(w)$  oder  
[6]        $f(u) = 0$ ) then  
[6]          $Pf \leftarrow (u, v_n, w)$   
[6]         Sind  $u$  oder  $w$  noch nicht gefärbt, wähle die kleinste freie  
[6]         Farbklasse, so dass  $G_P$  azyklisch bleibt nur wenn  $f(u)$  und  
[6]          $f(w) < \max f$ .  
[6]         baue  $G_P$   
[7]         if  $G_P$  ist azyklisch then  
[7]            $f(v_n) = -1$   
[8]         else  
[8]           entferne  $(u, v_n, w)$  aus  $Pf$  und setze  $f(u), f(w)$  auf ihre Ur-  
[8]           sprungsfarbe zurück  
[9]         end if  
[10]       end if  
[11]     end if  
[11]     setze  $\max f$   
[12]   end for  
[13] return  $f, Pf$ 
```

---

Die For-Schleife in Zeile 4 wird maximal  $n$  mal durchlaufen. Für die Bedingung in Zeile 7 ergibt sich wie in Abschnitt 5.6.5 eine Laufzeit von maximal  $\frac{n^2}{2}$ . Damit beträgt die Laufzeit dieser Heuristik im Worst Case  $f(n) = n \cdot \frac{n^2}{2} = O(n^3)$ .

Man kann allerdings die selben Verbesserungen wie in Abschnitt 5.7.1 einführen und bekommt wieder eine Laufzeit von  $O(n^2)$ .

Anhand des Ablaufs kann man folgern, dass der Algorithmus 4 grundsätzlich eine kleinere Laufzeit haben müsste als Algorithmus 3. Man kann aber vermuten, dass für kleinere Instanzen – mit einer verhältnismäßig kleinen größten Clique im Graph  $G$  – der Greedy Algorithmus 3 schneller terminiert, da weniger häufig der Graph  $G_P$  gebaut (oder die Suche aus Abschnitt 5.7.1 ausgeführt) werden muss. Eine genaue Auswertung folgt im kommenden Abschnitt.

Die Heuristik 4 liefert also eine obere Schranke  $u_{col}$  für das TMP. Wird Zeile 7 entfernt, liefert die Heuristik eine untere Schranke  $l_{col}$ .

## 5.8 Auswertung

Für die Implementierung wurde die Sprache Python gewählt. Der objektorientierte Ansatz dieser Sprache vereinfacht die Programmierung komplexer Strukturen, wie sie insbesondere für die pseudostabilen Mengen nötig sind. Für die Programmierung einiger graphentheoretischer Aspekte wurde die freie Bibliothek NetworkX benutzt.

Gearbeitet wurde mit Python 3.4 auf einem Computer mit vier 2.4 GHz Prozessoren, 8 GB RAM und Linux Kernel 3.10. Das ganzzahlige lineare Programm wurde durch die GLPK (GNU Linear Programming Kit) in der Version 4.52 gelöst. Der folgende Abschnitt erklärt kurz die Generierung von Zufallsinstanzen. Darauf folgt eine detaillierte Darstellung der Resultate. Um vergleichbare Resultate zu erhalten, folgt die zufällige Generierung von Instanzen und ihre Darstellung im Wesentlichen dem Vorgehen aus [S. 60–8].

### 5.8.1 Instanzen

Um vergleichbare Resultate für perfekte Graphen zu erhalten, werden die Verfahren auf Intervallgraphen ausgeführt. Dies hat weiter den Vorteil, dass

diese direkt im folgenden Kapitel über Rangierprobleme verwendet werden können.

Um vergleichbare Resultate zu erhalten, folgt die zufällige Generierung von Instanzen und ihre Darstellung im Wesentlichen dem Vorgehen aus [S. 60 ff. 8].

Benutzt wird dazu die diskrete Gleichverteilung auf  $\Sigma^n$  mit festem  $n$ . Sei also  $(\Sigma^n, \mathcal{P}(\Sigma^n), P)$  der Wahrscheinlichkeitsraum in dem  $\mathcal{P}(\Sigma^n)$  alle messbaren Teilmengen von  $\Sigma^n$  sind und  $P$  ein Maß auf  $\mathcal{P}(\Sigma^n)$  ist mit

$$P(\Sigma^n) = 1 \quad \text{und} \quad P(E) = \frac{|E|}{|\Sigma^n|},$$

wobei  $E \subset \Sigma^n$  ein Ereignis ist.

Sei  $P(d, l)$  die Zahl der Instanzen, bei denen die ersten  $l$  Wagen  $d$  Ziele benutzen.

**Theorem 5.32.** [S. 60 ff. 8]  $P(d, l) = P(d + 1, l + 1) + d \cdot P(d, l + 1)$  für  $d \leq l$  und  $P(d, l) = 1$  für  $l \geq n$ .

Ein Algorithmus, der  $P(d, l)$  für  $d \leq l$  berechnet, findet sich in [S. 61 ff. 8] und wird hier als Algorithmus 5 wiedergegeben. Er hat eine Laufzeit von  $O(n^2)$ . Daraus lässt sich eine gleichverteilte Zufallsinstanz  $(\sigma_1, \dots, \sigma_n)$  mit  $\sigma_1 = 1$  erstellen. Diese wird in Algorithmus 6 erstellt.

---

#### Algorithmus 5 BERECHNUNG VON $P$

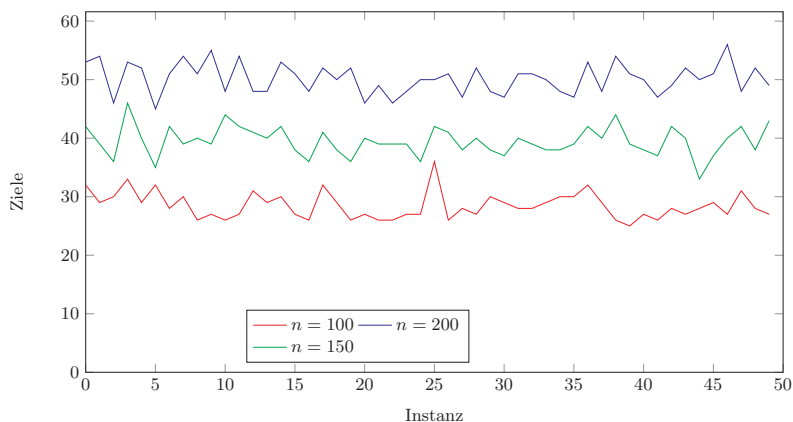
---

**Eingabe:**  $n \in \mathbb{N}$

**Ausgabe:**  $P(d, l) \forall d, l$

```
|1| for  $l = n, n - 1, \dots, 1$  do
|2|   for  $d = 1, \dots, l$  do
|3|     if  $l < n$  then
|3|        $P(d, l) = P(d + 1, l + 1) + d \cdot P(d, l + 1)$ 
|4|     else
|4|        $P(d, l) = 1$ 
|5|     end if
|6|   end for
|7| end for
|8| return  $P(d, l)$ 
```

---



**Abbildung 5.13:** Ziele in Zufallsinstanzen der Größe  $n = 200$ ,  $n = 150$  und  $n = 100$ .

---

### Algorithmus 6 INSTANZENGENERIERUNG

---

**Eingabe:**  $n \in \mathbb{N}$

**Ausgabe:**  $\sigma = (\sigma_1, \dots, \sigma_n)$

```

[1]  $d = 1$  // Anzahl der Ziele
[2]  $\sigma_1 = 1$ 
[3] for all  $i = 2, 3, \dots, n$  do
[4]   if  $\text{rand} < \frac{d \cdot P(d,i)}{P(d,i-1)}$  then
[4]      $\sigma_i = \text{uniform}(1, \dots, d)$  // diskrete Gleichverteilung über
        $\{1, \dots, d\}$ 
[5]   else
[5]      $d = d + 1$ 
[5]      $\sigma_i = d$ 
[6]   end if
[7] end for
[8] return  $\sigma$ 

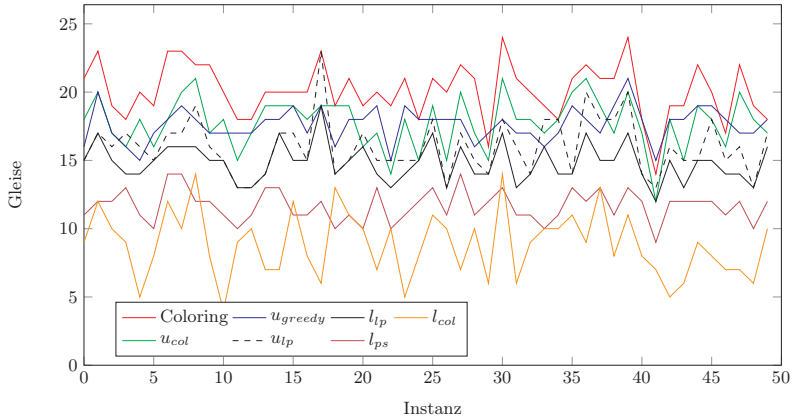
```

---

## 5.8.2 Auswertung

Der Algorithmus COLORING-PS 4 mit der oberen Schranke  $u_{col}$  und der unteren Schranke  $l_{col}$  hat wie der Algorithmus GREEDY-PS 3 mit der Schranke  $u_{greedy}$  eine Laufzeit von  $O(n^2)$ . Es ist nun zu untersuchen, welche Lauf-





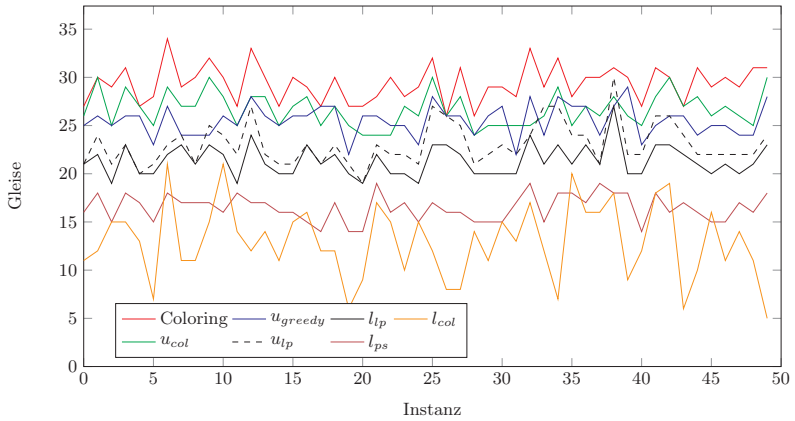
**Abbildung 5.14:** Ergebnisse verschiedener Heuristiken und Schranken für verschiedene Instanzen des TMP mit  $n = 100$ .

zeit sich auf den verschiedenen Instanzen ergibt und wie gut Lösungen sind, die von den Algorithmen auf Zufallsinstanzen des TMP geliefert werden.

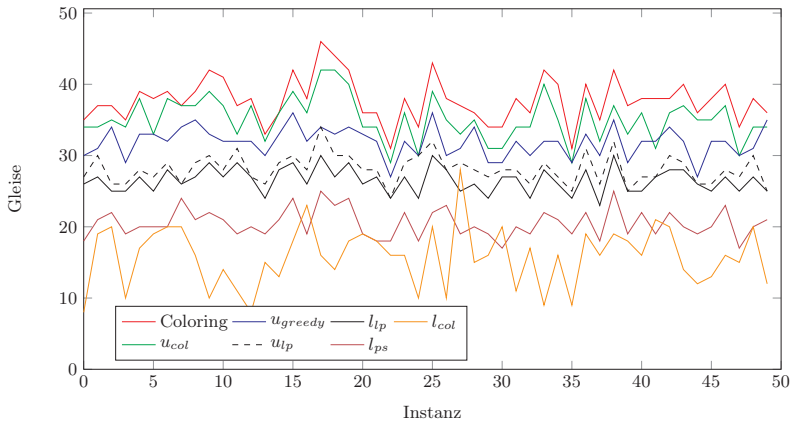
Die oberen und unteren Schranken durch das lineare Programm  $u_{lp}$  und  $l_{lp}$  liefern eine gute Abschätzung für die optimale Lösung, es kann hier den Ergebnissen aus [8] gefolgt werden. Die untere Schranke  $l_{ps}$  (vgl. auch Abschnitt 5.6.3) wurden ebenso wie das lineare Programm (vgl. auch Abschnitt 5.6.6) schon in der Literatur diskutiert.

Es stellt sich heraus, dass  $u_{col}$  niemals schlechter wird als eine normale Graphenfärbung. Das kann auch schon aus dem Entwurf des Algorithmus gefolgert werden. Für kleine Instanzen ist der Abstand zwischen  $u_{col}$  und  $u_{greedy}$  gering, oft sind die Lösungen von  $u_{col}$  sogar besser, vgl. Abb. 5.14. Je größer die Instanz wird, desto größer wird der Abstand, vgl. Abb. 5.15 und 5.16.  $u_{greedy}$  hat häufig einen kleinen Abstand zur Optimallösung, der jedoch mit steigender Instanzgröße auch immer größer wird. Interessant ist ein Hinweis auf Abbildung 5.17, in der die Lösungen für  $n = 300$  dargestellt werden. Deutlich heben sich hier alle Lösungen voneinander ab. Diese Wagenlänge ist für den regulären Güterzugverkehr unrealistisch, zeigt aber schön die natürlichen Grenzen der hier vorgestellten Heuristiken auf.

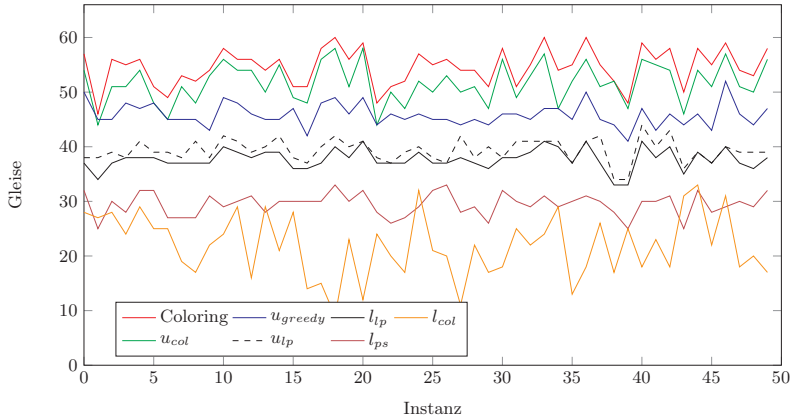
Eine Darstellung der verschiedenen Laufzeiten findet sich in Tabelle 5.2. Obwohl beide Algorithmen in  $O(n^2)$  sind, zeigen sich doch deutliche Unterschiede. Für kleine Instanzen bis  $n = 150$  zeigt sich, dass GREEDY-PS nicht nur eine kleinere durchschnittliche, sondern sogar eine kleinere maximale



**Abbildung 5.15:** Ergebnisse verschiedener Heuristiken und Schranken für verschiedene Instanzen des TMP mit  $n = 150$ .



**Abbildung 5.16:** Ergebnisse verschiedener Heuristiken und Schranken für verschiedene Instanzen des TMP mit  $n = 200$ .



**Abbildung 5.17:** Ergebnisse verschiedener Heuristiken und Schranken für verschiedene Instanzen des TMP mit  $n = 300$ .

	n=100			n=150		
	min	max	$\emptyset$	min	max	$\emptyset$
GREEDY-PS	0,0071	0,0314	0,0157	0,0317	0,1077	0,0571
COLORING-PS	0,0111	0,0502	0,0275	0,0297	0,1211	0,0781
	n=200			n=300		
	min	max	$\emptyset$	min	max	$\emptyset$
GREEDY-PS	0.0747	0.3095	0.1539	0.2767	0.8529	0.5896
COLORING-PS	0.0914	0.2388	0.1703	0.3428	0.6425	0.4840

**Tabelle 5.2:** Die Laufzeiten für die Algorithmen GREEDY-PS und COLORING-PS für verschiedene Instanzen in Sekunden.

Laufzeit hat als COLORING-PS. Erst bei Instanzen der Größe  $n = 200$  steigt die maximale Laufzeit von GREEDY-PS signifikant an, wobei die durchschnittliche Laufzeit immer noch geringer ist als bei COLORING-PS. Erst bei Instanzen der Größe  $n = 300$  zeigt sich deutlich, dass GREEDY-PS eine größere Laufzeit als COLORING-PS hat.

Laufzeitvorteile hat COLORING-PS also nur bei großen Instanzen. Allerdings berechnet es nur bei kleinen Instanzen eine günstigere oder zumindest häufig gleich günstige Lösung wie GREEDY-PS. Für Instanzen der Größe  $n = 100$  ist die durchschnittliche Laufzeit aber fast doppelt so lang wie bei GREEDY-PS. Da sie allerdings alle im kleineren Millisekundenbereich liegen, können beide Heuristiken anstelle des aufwendigen Linearen

Programms genutzt werden. Eine Kombination aus beiden liefert für kleine Instanzen von  $n = 100$  fast optimale Lösungen wie Abb. 5.14 entnommen werden kann.

## 5.9 Zusammenfassung

Zunächst wurden allgemeine Rangierprobleme auf Güterbahnhöfen und dann im speziellen das TMP beschrieben und bekannte Lösungsstrategien und Ansätze diskutiert. Es wurde gezeigt, dass eine Lösung des TMP einer Lösung von minPS auf dem zugehörigen Intervallgraphen entspricht. Dahingehend wurden die Ansätze aus [8] weiter untersucht und mit Hilfe der neuen Ergebnisse aus dem vorherigen Kapitel ergänzt.

So wurde eine neue Heuristik COLORING-PS vorgestellt, die unter bestimmten Umständen das TMP in polynomieller Zeit lösen würde. Für die hier verwendeten Zufallsinstanzen liefert es jedoch nur Näherungslösungen. Diese sind für kleine Instanzen teilweise deutlich besser als die Lösungen des bekannten GREEDY-PS. Für größere Instanzen bietet COLORING-PS keine besseren Lösungen als das bekannte GREEDY-PS, hat aber eine bessere Laufzeit.

Es bleibt zu untersuchen, ob es weitere Heuristiken gibt, die mit einer günstigen Laufzeit gute Lösungen für das TMP, insbesondere für große Instanzen, liefern.

## Mehrfach pseudostabile Mengen: Schranken und Algorithmen

In diesem Kapitel wird das Problem  $\text{minMPS}$  untersucht, das eine minimale Zerlegung eines Graphen  $G$  in mehrfach pseudostabile Mengen sucht. Dazu werden zunächst die Definition des Entscheidungs- und Optimierungsproblems gemäß der Herleitung in Kapitel 3 gegeben und allgemeine Grundlagen notiert. Die Komplexität des Problems  $\text{minPS}$  wurde schon in Abschnitt 4.2 untersucht, die Untersuchung für  $\text{minMPS}$  erfolgt analog:  $\text{minMPS}$  ist  $\mathcal{NP}$ -vollständig.

$\text{minMPS}$  hat zwar eine große Ähnlichkeit mit  $\text{minPS}$ , der minimalen Zerlegung eines Graphen  $G$  in mehrfach pseudostabile Mengen. Trotzdem können nicht alle Ergebnisse übertragen werden und ähnliche Ansätze liefern unterschiedliche Resultate. Um Doppelungen zu vermeiden, wird so oft wie möglich auch auf Kapitel 5.6.2 bzw. dort auf dieses Kapitel verwiesen.

Weiter wird erneut versucht, Abschätzungen für  $\text{minMPS}$  zu finden und eine Darstellung als lineares Programm untersucht.

### 6.1 $\text{minMPS}$

**Definition 6.1** ( $\text{minMPS}$ ). Gegeben sei ein Graph  $G = (V, E)$  mit blauen Knoten und Kanten  $B \subset G$ . Gesucht ist eine minimale Zerlegung  $P$  von  $G$  in mehrfach pseudostabile Mengen.

**Bemerkung 6.2.** Um zu zeigen, dass es sich um eine mehrfach pseudostabile Menge handelt, kann man die Notation  $\zeta^m(P)$  wählen. Für eine minimale Zerlegung  $P$  eines Graphen  $G$  in mehrfach pseudostabile Mengen schreibt man statt  $\zeta^m(P)$  auch kurz  $\zeta^m(G)$  bzw.  $\zeta_P^m(G)$ .



Abbildung 6.1: Graph zu Beispiel 6.1.

**Definition 6.3** (E-minMPS). Gegeben sei ein Graph  $G = (V, E)$  mit blauen Knoten und Kanten  $B \subset G$ . Gegeben sei ein Wert  $k > 0$ : Gibt es eine Zerlegung  $P$  von  $G$  in mehrfach pseudostabile Mengen mit der Wertigkeit  $k$ ?

Zu einer Zerlegung  $P$  eines Graphen  $G = (V, E)$  mit blauen Knoten und Kanten  $B \subset G$  kann nun der Graph  $G_P$  dieser Zerlegung gebildet werden. Dies entspricht dem Graphen der Überdeckung aus Definition 3.5. Die Definition erfolgt ähnlich zu der Definition 4.4.

**Definition 6.4** (Graph einer zulässigen Zerlegung in mehrfach pseudostabile Mengen). Zu einer Zerlegung  $P$  eines Graphen  $G$  in mehrfach pseudostabile Mengen sei  $G_P = (V, E)$  der gerichtete Graph dieser Zerlegung mit

- $V = \{1, \dots, M\}$  wobei  $M$  die Anzahl aller stabilen Teilmengen  $D_j^i$  aller pseudostabilen Mengen  $P_i \subset P$  ist, d. h.  $M = \sum_{r=1}^{|P|} m_r$  und
- $(r, r+1) \in E$  für jeden Pfad  $p_{r,s}^i$ , der  $D_r^i$  und  $D_{r+1}^j$  verbindet (mit  $s$  beliebig).

Im Gegensatz zum Graphen einer zulässigen Zerlegung in pseudostabile Mengen (vgl. Definition 4.4) ist der Graph gerichtet. Dies ist deswegen wichtig, weil ein ungerichteter Graph mit Mehrfachkanten Zyklen beinhaltet.

**Beispiel 6.1.** Man betrachte den Graphen  $G = (V, E)$ , der aus zwei disjunkten  $K_3$  besteht. Es sei  $B = G$ . Offensichtlich lässt sich der Graph  $G$  wieder in drei stabile Mengen zerlegen. Man benötigt nur eine Menge  $P$ , wenn man die Definition 3.11 betrachtet: In diesem Fall darf es mehrere Pfade zwischen zwei stabilen Mengen  $D_1$  und  $D_2$  geben. Der gerichtete Graph der zulässigen Zerlegung in pseudostabile Mengen ist also  $G_P = (\{u, v\}, \{(u, v), (v, u)\})$ . Siehe auch Abb. 6.1.  $\triangleleft$

Genauso wie Lemma 4.5 und 4.7 lassen sich die folgenden zwei Lemmata zeigen:

**Lemma 6.5.**  $P$  ist eine zulässige Zerlegung eines Graphen  $G$  in mehrfach pseudostabile Mengen genau dann, wenn  $G_P$  azyklisch ist.

**Lemma 6.6.** *Ist  $P$  eine zulässige Zerlegung eines Graphen  $G$  in mehrfach pseudostabile Mengen, so beinhaltet  $G_P$  nur abkürzungsfreie Pfade.*

Es darf nun offensichtlich Mehrfachkanten in dem Graphen  $G_P$  zu einer zulässigen Zerlegung  $P$  eines Graphen  $G$  in mehrfach pseudostabile Mengen geben.

Es sei noch gemäß Kapitel 3 auf die entsprechende Version von minMPS bzw. E-minMPS, die die azyklische Eigenschaft von  $G_P$  vernachlässigt hingewiesen. Sie wird mit minMPS-a bzw. E-minMPS-a bezeichnet.

Da jede pseudostabile Menge auch eine mehrfach pseudostabile Menge ist, gilt das folgende Lemma:

**Lemma 6.7.** *Für eine minimale Zerlegung eines Graphen  $G$  mit blauen Knoten und Kanten in stabile und mehrfach stabile Mengen gilt stets*

$$\zeta^m(G) \leq \zeta(G)$$

Dies bedeutet, dass alle oberen Schranken für eine minimale Zerlegung in stabile Mengen auch obere Schranken für eine minimale Zerlegung in mehrfach stabile Mengen sind.

## 6.2 Abschätzungen für minMPS

Ein vollständiger, nur aus blauen Knoten und Kanten bestehender Graph  $K_n^B$  kann nach Lemma 3.28 immer in eine mehrfach pseudostabile Menge mit Wertigkeit 2 zerlegt werden. Dies liefert eine erste untere Schranke:

**Lemma 6.8.** *Für einen Graphen  $G = (V, E)$  mit mehr als zwei Knoten und mehr als einer Kante gilt immer  $\zeta(G) \geq 2$ .*

*Beweis.* Dies folgt direkt aus Lemma 3.28. □

Diese Schranke ist selbstverständlich äußerst unscharf, kann aber, wie im Folgenden gezeigt wird, verbessert werden.

### 6.2.1 Abschätzung über Cliques

Zunächst werden vollständige Graphen  $K_n$  betrachtet. In einem ersten Ansatz wird die Anzahl der blauen Knoten betrachtet:

**Lemma 6.9.** *Für die Zerlegung  $P$  eines vollständigen Graphen  $K_n$  mit  $n^b$  blauen Knoten,  $n \geq n^b \geq 0$ ,  $n > 2$  in mehrfach pseudostabile Mengen gilt stets*

$$\zeta(K_n) \geq n - \max\{0, n^b - 2\} \quad (6.1)$$

*Beweis.* Es sei  $G = K_n$  ein vollständiger Graph mit  $n^b$  blauen Knoten und nur blauen Kanten,  $n \geq n^b \geq 0$ ,  $n > 2$ .

Ist kein Knoten blau, so gilt  $\zeta(G) = n$ .

Gibt es nur einen oder zwei blaue Knoten, gilt ebenso  $\zeta(G) = n$ , da kein Pfad gebildet werden kann.

Für jeden weiteren blauen Knoten kann dieser aber als Pfad zwischen zwei stabilen Mengen definiert werden, die durch zwei blaue Knoten gebildet werden.

Es gilt also immer Gleichung 6.1. □

**Beispiel 6.2.** Es sei  $G = K_6$  mit vier blauen Knoten  $v_1, \dots, v_4$  und zwei weiteren Knoten  $v_5, v_6$ . Es seien weiter alle Kanten blau.

Nun kann eine Zerlegung in mehrfach pseudostabile Mengen mit vier stabilen Mengen  $D_1 = \{v_1\}$ ,  $D_2 = \{v_2\}$ ,  $D_3 = \{v_5\}$  und  $D_4 = \{v_6\}$  und zwei Pfaden  $[v_1, v_3, v_2]$  und  $[v_1, v_4, v_2]$  konstruiert werden. Die Wertigkeit dieser Zerlegung ist vier.

Mit oberer Abschätzung gilt  $n - \max\{0, n^b - 2\} = 6 - 2 = 4 = \zeta(G)$ . △

Ebenso liefert aber auch die Anzahl der blauen Kanten eine untere Schranke:

**Lemma 6.10.** *Für die Zerlegung  $P$  eines vollständigen Graphen  $K_n$  mit  $e^b$  blauen Kanten,  $n > 2$  in mehrfach pseudostabile Mengen gilt stets*

$$\zeta(K_n) \geq n - \left\lfloor \frac{e^b}{2} \right\rfloor \quad (6.2)$$



*Beweis.* Es sei  $G = K_n$  ein vollständiger Graph mit  $e^b$  blauen Kanten und nur blauen Knoten mit  $n > 2$ .

Ist keine oder nur eine Kante blau, d. h.  $e^b = 0$  oder  $e^b = 1$ , so gilt  $\zeta(G) = n$ .

Für jeweils zwei weitere blaue Kanten kann maximal ein Pfad zwischen zwei stabilen Mengen gebildet werden.

Es gilt also immer Gleichung 6.2. □

Diese beiden Schranken können nun auch kombiniert werden, um ein besseres Resultat zu erhalten.

**Theorem 6.11.** *Für die Zerlegung  $P$  eines vollständigen Graphen  $K_n$  mit  $e^b$  blauen Kanten und mit  $n^b$  blauen Knoten,  $n \geq n^b \geq 0$ ,  $n > 2$  in mehrfach pseudostabile Mengen gilt stets*

$$\zeta(K_n) \geq n - \min \left\{ \left\lfloor \frac{e^b}{2} \right\rfloor, \max\{0, n^b - 2\} \right\}$$

Ähnlich wie in Abschnitt 4.3.1 kann nun eine neue Schranke konstruiert werden. Es seien nun zunächst alle maximalen Cliques eines Graphen  $G$  entsprechend ihrer Größe angeordnet, also  $C_1, \dots, C_m$  mit  $|C_i| \geq |C_{i+1}|$ . Es sei  $r_i$  die Anzahl von blauen Knoten und  $b_i$  die Anzahl von blauen Kanten in der Clique  $C_i$ . Dann sei

$$C'_i = C_i \setminus \bigcup_{j=1}^{i-1} C_j$$

und damit kann die Abbildung  $p : \mathbb{N} \rightarrow \mathbb{N}^0$  definiert werden mit

$$p(i) = \max \left\{ 0, |C'_i| - \min \left\{ \left\lfloor \frac{b_i}{2} \right\rfloor, \max\{0, r_i - 2\} \right\} - \sum_{j=1}^{i-1} p(j) \right\}$$

und

$$p(1) = |C_1| - \min \left\{ \left\lfloor \frac{b_1}{2} \right\rfloor, \max\{0, r_1 - 2\} \right\}$$

Dann sei weiter

$$l_C(G) = \sum_{i=1}^m p(i) \tag{6.3}$$

**Lemma 6.12.** *Für einen Graphen  $G = (V, E)$  mit mehr als zwei Knoten und mehr als einer Kante sowie blauen Knoten und blauen Kanten gilt immer*

$$\zeta(G) \geq l_C(G)$$

*Beweis.* Nach Theorem 6.11 gilt für jede Clique mit  $n$  Knoten,  $e^b$  blauen Kanten und  $n^b$  blauen Knoten, dass eine Zerlegung in mehrfach pseudostabile Mengen mindestens die Wertigkeit  $n - \min \left\{ \left\lfloor \frac{e^b}{2} \right\rfloor, \max\{0, n^b - 2\} \right\}$  hat.

Entsprechend gilt natürlich, da eine Clique, aus der  $o$  Knoten entfernt werden, immer noch eine Clique mit  $n - o$  Knoten ist: Falls noch  $n - o$  Knoten, von denen  $n^b - o^b$  blau sind, übrig bleiben, ist die Wertigkeit der Zerlegung einer solchen Clique in mehrfach pseudostabile Mengen immer noch  $n - o - \max\{0, n^b - o^b - 2\}$ . Falls kein Knoten übrig bleibt, ist die Wertigkeit 0.

Es sei nun  $C_s = \emptyset$  die Menge aller schon betrachteten Knoten. Man betrachte nun sukzessive alle Cliques  $C_j$  aus  $C_1, \dots, C_m$  mit  $|C_i| \geq |C_{i+1}|$ . Es sei  $r_i$  die Anzahl von blauen Knoten in der Clique  $C_i$

- Ist  $j = 1$ , so betrachtet man die größte Clique und benötigt nach Theorem 6.11 für eine Zerlegung in mehrfach pseudostabile Mengen mindestens die Wertigkeit  $|C_1| - \min \left\{ \left\lfloor \frac{b_1}{2} \right\rfloor, \max\{0, r_1 - 2\} \right\}$ . Man setze  $C_s = C_1$  und  $p(1) = |C_1| - \min \left\{ \left\lfloor \frac{b_1}{2} \right\rfloor, \max\{0, r_1 - 2\} \right\}$ .
- Ist  $j > 1$ , so werden zunächst alle bereits benutzten Knoten entfernt, man setzt also  $C'_j = C_j \setminus C_s$  und anschließend  $C_s = C_s \cup C_j$ . Es sind also nur noch  $|C'_j|$  Knoten vorhanden, von denen  $r'_i \leq r_i$  Knoten blau sind. Außerdem sind noch maximal  $b'_i \leq b_i$  blaue Kanten vorhanden. Wird dieser auch in mehrfach pseudostabile Mengen zerlegt, hat diese also mindestens die Wertigkeit

$$\begin{aligned} & |C'_i| - \min \left\{ \left\lfloor \frac{b'_i}{2} \right\rfloor, \max\{0, r'_i - 2\} \right\} \\ & \geq |C'_i| - \min \left\{ \left\lfloor \frac{b_i}{2} \right\rfloor, \max\{0, r_i - 2\} \right\} - \sum_{j=1}^{i-1} p(j) \end{aligned} \quad (6.4)$$

Es gibt aber auch schon  $\sum_{k=1}^{j-1} p(k)$  stabile Mengen, denen Knoten hinzugefügt werden können; diese müssen also berücksichtigt werden,

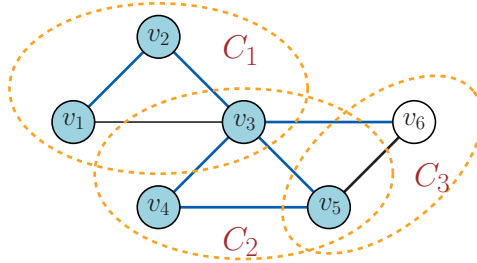


Abbildung 6.2: Graph zu Beispiel 6.3.

und da es keine negativen Wertigkeiten gibt, ist diese mindestens

$$\max \left\{ 0, |C'_i| - \min \left\{ \left\lfloor \frac{b_i}{2} \right\rfloor, \max\{0, r_i - 2\} \right\} - \sum_{j=1}^{i-1} p(j) \right\}$$

Die Summe ist genau  $l_C(G) = \sum_{i=1}^m p(i)$ , entspricht also Gleichung 6.3 mit den oben definierten Angaben. Dies liefert also eine untere Schranke. Diese kann verschärft werden, wenn in Gleichung 6.4 die tatsächlich verbliebene Anzahl von blauen Knoten und Kanten verwendet wird. Dies hat aber einen erhöhten Rechenaufwand zur Folge.  $\square$

**Beispiel 6.3.** Man betrachte den Graphen in Abb. 6.2. Man kann leicht sehen, dass eine minimale Zerlegung in mehrfach pseudostabile Mengen die Wertigkeit drei hat. Eine der größten Cliques ist  $C_1$ . Es ist nun

$$p(1) = 3 - \min \left\{ \left\lfloor \frac{2}{2} \right\rfloor, \max\{0, 3 - 2\} \right\} = 3 - 1 = 2$$

und, da  $C_s = \{v_1, v_2, v_3\}$  ist nun  $C'_2 = \{v_4, v_5\}$  und es ist

$$p(2) = 2 - \min \left\{ \left\lfloor \frac{1}{2} \right\rfloor, \max\{0, 2 - 2\} \right\} - 2 = 0$$

Weiter ist nun  $C_s = C_s \cup \{v_4, v_5\}$  und somit  $C'_3 = \{v_6\}$ . Dann ist

$$p(3) = \max \left\{ 0, 1 - \min \left\{ \left\lfloor \frac{0}{2} \right\rfloor, \max\{0, 0 - 2\} - 2 \right\} \right\} = 0$$

Dann ist  $l_C(G) = 2$ .  $\triangle$

Wichtig an dieser Schranke ist, dass sie auch eingesetzt werden kann, wenn man nicht alle Summanden berechnet. Sie wird dadurch allerdings natürlich schlechter. Man kann die Schranke  $l_C$ , in der nur  $i$  Summanden betrachtet werden mit

$$l_{C,i} = \sum_{j=1}^i p(j) \text{ mit } i \leq m$$

bezeichnen. Für die Schranke  $l_{C,1}$  gilt zusätzlich, dass eine beliebige Clique betrachtet werden kann, d. h. für eine beliebige Clique  $C$  mit  $b$  blauen Knoten im Graphen  $G$  gilt immer

$$\zeta(G) \geq |C| - \min \left\{ \left\lfloor \frac{b}{2} \right\rfloor, \max\{0, b-2\} \right\}$$

Dies folgt direkt aus Lemma 6.12, denn die Ungleichung gilt natürlich nicht nur für weniger Summanden sondern auch für einen beliebigen Summanden.

Wie sich diese Schranken genau verhalten, wird in den folgenden Abschnitten noch genauer betrachtet.

## 6.2.2 Abschätzung über den Teilgraphen $B$

Zunächst wird eine untere Schranke für eine Zerlegung eines Graphen in mehrfach pseudostabile Mengen mit disjunkten Pfaden betrachtet, um anschließend die Ergebnisse aus dem vorherigen Abschnitt auf den gesamten Graphen zu übertragen.

Es sei  $G$  ein Graph mit  $n$  Knoten, von denen  $n^b$  blau sind. Dann sei

$$\chi'(G) = \chi(G) - \left\lfloor \frac{1}{3}n^b \right\rfloor$$

Diese Funktion zieht also von einer minimalen Färbung für je drei blaue Knoten in  $G$  eine Farbe ab.

**Lemma 6.13.** *Es sei  $G$  ein Graph mit  $n > 1$  Knoten, von denen  $n^b$  blau sind und mindestens einer Kante. Es gilt für eine Zerlegung in mehrfach pseudostabile Mengen mit disjunkten Pfaden immer*

$$l_R(G) = \max \{2, \chi'(G)\} \leq \zeta(G)$$

*Beweis.* Es sei  $P$  eine minimale Zerlegung des Graphen  $G$  in mehrfach pseudostabile Mengen. Ist  $\zeta(G) < \chi(G)$  gibt es Pfade zwischen stabilen Mengen. Es muss mindestens drei blaue Knoten pro Pfad geben, da diese sich nicht schneiden dürfen, also

$$\chi(G) - \zeta(G) \leq \frac{1}{3} \cdot n^b$$

$$\chi(G) - \frac{1}{3} \cdot n^b \leq \zeta(G)$$

Da  $G$  mindestens eine Kante besitzt, gilt natürlich

$$\max \left\{ 2, \chi(G) - \frac{1}{3} \cdot n^b \right\} \leq \zeta(G)$$

und damit auch

$$\max \left\{ 2, \chi(G) - \left\lfloor \frac{1}{3} \cdot n^b \right\rfloor \right\} \leq \zeta(G)$$

□

Für mehrfach pseudostabile Mengen mit eindeutigen Pfaden kann die Schranke aus dem vorherigen Abschnitt übertragen werden:

**Lemma 6.14.** *Es sei  $G$  ein Graph mit  $n > 1$  Knoten, von denen  $n^b$  blau sind und mindestens einer Kante. Es gilt für eine Zerlegung in mehrfach pseudostabile Mengen immer*

$$l_R(G) = \max\{2, \chi'(G)\} \leq \zeta(G)$$

$$\text{mit } \chi'(G) = \chi(G) - \max\{0, n^b - 2\}$$

*Beweis. Voraussetzung:* Es sei  $G = (V, E)$  ein Graph mit  $n^b$  blauen Knoten und nur blauen Kanten,  $n \geq n^b \geq 0$ ,  $n > 1$ .

Ist  $n = 2$  so gilt, da es mindestens eine Kante geben muss  $2 \leq \zeta(G)$  sein.

Ist kein Knoten blau, so gilt  $\zeta(G) = \chi(G)$ .

Gibt es nur einen oder zwei blaue Knoten, gilt ebenso  $\zeta(G) = \chi(G)$ , da kein Pfad gebildet werden kann.

Für jeden weiteren blauen Knoten kann nun maximal ein Pfad zwischen zwei stabilen Mengen gebildet werden und die Wertigkeit der Zerlegung verringert sich pro Pfad maximal um eins.

Es gilt also immer  $\chi(G) - \max\{0, n^b - 2\} \leq \zeta(G)$  und natürlich  $2 \leq \zeta(G)$ .

□

**Bemerkung 6.15.** Lemma 6.9 folgt direkt auch aus Lemma 6.14, denn  $\chi(K_n) = n$ .

Da die chromatische Zahl eines Graphen  $G$  nicht immer in polynomieller Zeit berechnet werden kann, kann es sinnvoll sein, diese in  $l_R$  durch eine untere Schranke zu ersetzen.

Eine weitere Verbesserung kann erfolgen, wenn man statt der Anzahl von blauen Knoten nur die Anzahl derjenigen blauen Knoten betrachtet, die Teil eines Pfades gemäß der Annahmen aus dem vorherigen Kapitel sind, d. h. der Knoten  $v$ , die Teil eines Pfades  $[a, b, c]$  von blauen Knoten, die mit blauen Kanten verbunden sind und deren mittlerer Knoten  $b$  Teil eines Pfades ungerader Länge in  $G$  ist, sind. Diese Schranken werden mit  $\tilde{l}_R$  bezeichnet.

Um dieses aufwändige Verfahren ein wenig zu beschleunigen, kann – wie im vorherigen Abschnitt – auch wieder eine Abschätzung über blaue Kanten erfolgen:

**Lemma 6.16.** *Es sei  $G$  ein Graph mit  $n > 1$  Knoten und  $e > 1$  Kanten, von denen  $e^b$  blau sind. Es gilt für eine Zerlegung in mehrfach pseudostabile Mengen immer*

$$l_B(G) = \max \left\{ 2, \chi(G) - \left\lfloor \frac{e^b}{2} \right\rfloor \right\} \leq \zeta(G)$$

*Beweis. Voraussetzung:* Es sei  $G$  ein Graph mit  $n > 1$  Knoten und  $e > 1$  Kanten, von denen  $e^b$  blau sind,  $e \geq e^b \geq 0$ ,  $n > 1$ .

ist  $n = 2$  so gilt, da es mindestens eine Kante geben muss aber kein Pfad gebildet werden kann  $2 \leq \zeta(G)$ .

Ist kein oder nur eine Kante blau, so gilt offensichtlich  $\zeta(G) = \chi(G)$ .

Für jede weitere blaue Kante kann nun maximal ein Pfad zwischen zwei stabilen Mengen gebildet werden und die Wertigkeit der Zerlegung verringert sich pro Pfad maximal um eins.

Es gilt also immer

$$\max \left\{ 2, \chi(G) - \left\lfloor \frac{e^b}{2} \right\rfloor \right\} \leq \zeta(G)$$

□

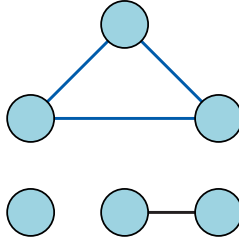


Abbildung 6.3: Graph zu Beispiel 6.4.

Die unteren Schranken  $l_R$  und  $l_B$  aus Lemma 6.14 und 6.16 können nun zusammengefasst werden:

**Lemma 6.17.** *Es sei  $G$  ein Graph mit  $n > 1$  Knoten, von denen  $n^b$  blau sind, und  $e > 1$  Kanten, von denen  $e^b$  blau sind. Es gilt für eine Zerlegung in mehrfach pseudostabile Mengen immer*

$$\begin{aligned} l_{RB}(G) &= \min \{l_B(G), l_R(G)\} \\ &= \max \{2, \chi(G) - \min \left\{ \left\lfloor \frac{e^b}{2} \right\rfloor, \max \{0, n^b - 2\} \right\} \} \end{aligned}$$

*Beweis.*

$$\begin{aligned} l_{RB}(G) &= \min \{l_B(G), l_R(G)\} \\ &= \min \left\{ \max \left\{ 2, \chi(G) - \left\lfloor \frac{e^b}{2} \right\rfloor \right\}, \max \left\{ 2, \chi(G) - \max \{0, n^b - 2\} \right\} \right\} \\ &= \max \left\{ 2, \chi(G) - \min \left\{ \left\lfloor \frac{e^b}{2} \right\rfloor, \max \{0, n^b - 2\} \right\} \right\} \end{aligned}$$

□

**Beispiel 6.4.** Man betrachte den Graphen in Abb. 6.3. Man sieht leicht, dass eine minimale Zerlegung in mehrfach pseudostabile Mengen die Wertigkeit zwei hat.

Es gilt nun

$$\chi(G) - \max \{2, 3 - \max \{0, 6 - 2\}\} = 3 - \max \{2, 4\} = -1$$

und damit  $l_R = 2$ . Weiter ist  $l_B = \chi(G) - \max \{2, 1\} = 3 - 2 = 1$  und damit  $l_B = l_R$ . In diesem Fall nehmen beide Schranken den selben Wert an.  $\triangle$

**Lemma 6.18.** *Es sei  $G = (V, E)$  ein Graph mit blauen Knoten und Kanten  $B(G) \subset G$ . Es seien  $D_1$  und  $D_2$  zwei Graphen für die gilt  $D_1 \cup D_2 = G$ ,  $D_1 \cap D_2 = \emptyset$ ,  $D_1 \neq \emptyset$  und  $D_2 \neq \emptyset$ . Dann gilt*

$$\zeta(G) \geq \left\lfloor \frac{l_{RB}(D_1) + l_{RB}(D_2)}{2} \right\rfloor = l_M$$

*Beweis.* Gegeben sei ein Graph  $G = (V, E)$  mit mit blauen Knoten und Kanten  $B(G) \subset G$ . Es seien  $D_1$  und  $D_2$  zwei Graphen für die gilt  $D_1 \cup D_2 = G$ ,  $D_1 \cap D_2 = \emptyset$ ,  $D_1 \neq \emptyset$  und  $D_2 \neq \emptyset$ .

Es sei  $P$  eine optimale Zerlegung von  $G$  in mehrfach pseudostabile Mengen mit der Wertigkeit  $\zeta(G)$ .  $P$  zerlegt  $D_1$  in pseudostabile Mengen mit der Wertigkeit  $a$ .

Es werden in  $D_1$  also mindestens  $a - l_{RB}(D_1)$  Knoten zu viel in Pfade gelegt.

Für  $D_2$  stehen zusätzlich  $\zeta(P) - a$  stabile Mengen in der Zerlegung  $P$  zur Verfügung. Es werden in  $D_2$  also  $(\zeta(P) - a) - l_{RB}(D_2)$  Knoten zu viel in Pfade gelegt.

Für jeden dieser Knoten, die nicht Teil einer zulässigen Zerlegung in pseudostabile Mengen sind, soll eine neue stabile Menge benutzt werden, wodurch eine zulässige Zerlegung konstruiert werden kann. So gilt nun

$$\zeta(G) + a - l_{RB}(D_1) + (\zeta(P) - a) - l_{RB}(D_2) \geq 0 \quad (6.5)$$

$$\zeta(G) + a - l_{RB}(D_1) + \zeta(P) - a - l_{RB}(D_2) \geq 0 \quad (6.6)$$

$$2\zeta(G) - l_{RB}(D_1) - l_{RB}(D_2) \geq 0 \quad (6.7)$$

$$(6.8)$$

$$2\zeta(G) \geq l_{RB}(D_1) + l_{RB}(D_2) \quad (6.9)$$

$$\zeta(G) \geq \frac{l_{RB}(D_1) + l_{RB}(D_2)}{2} \quad (6.10)$$

$$\zeta(G) \geq \left\lfloor \frac{l_{RB}(D_1) + l_{RB}(D_2)}{2} \right\rfloor \quad (6.11)$$

□

Die beiden Schranken  $l_{RB}$  und  $l_M$  sind zwar ähnlich, allerdings zeigt sich, dass im Allgemeinen  $l_M < l_{RB}$  ist, da man für die Schranke  $l_M$  die Teilmengen  $D_1$  und  $D_2$  geschickt suchen muss. Dazu zwei Beispiele:



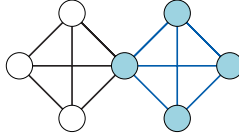


Abbildung 6.4: Graph zu Beispiel 6.5.

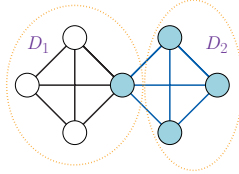


Abbildung 6.5: Graph zu Beispiel 6.5.

**Beispiel 6.5.** Es sei  $G$  ein Graph, der aus einem  $K_4$  besteht, wobei einer der Knoten blau ist Teil eines weiteren  $K_4$  der nur aus blauen Knoten und blauen Kanten besteht, ist. Vgl. dazu Abb. 6.4. Es gilt nun  $\chi(G) = 4$  und weiter  $l_{RB} = \max\{2, 4 - \min\{1, 3\}\} = 2$ .

Wählt man als Zerlegung  $D_1 = K_4$  und  $D_2$  als die Menge von blauen Knoten (siehe auch Abb. 6.5) so gilt  $l_{RB}(D_1) = 4$  und  $l_{RB}(D_2) = 2$ . Damit ist  $l_{RM} = \lfloor \frac{4+2}{2} \rfloor = 3$ . Die Wertigkeit einer zulässigen, minimalen Zerlegung in mehrfach pseudostabile Mengen beträgt  $\zeta(G) = \chi(G) = 4$ .  $\triangle$

**Lemma 6.19.** *Es gilt für einen Graphen  $G$  mit  $n^b \geq 2$  blauen Knoten stets*

$$l_{RM} \leq \max\{2, \chi(G \setminus B)\} \tag{6.12}$$

*Beweis.* Dies folgt unmittelbar aus der folgenden Beobachtung

$$\chi(G \setminus B) \geq \chi(G) - |V(R)|$$

Somit gilt für  $n^b \geq 2$

$$\chi(G) - (n^b - 2) \leq \chi(G) - n^b \leq \chi(G \setminus B)$$

und – weil die Anzahl der blauen Kanten nun nicht mehr ins Gewicht fällt – also Ungleichung 6.12.  $\square$

### 6.3 Lineare Programmierung

Grundsätzlich entsprechen die Formulierungen in IP1 und IP2 bereits einer Formulierung als lineares Programm. Da diese Formulierung vom Rechenaufwand her kaum zu handhaben ist, soll in Abschnitt 7.4.1 ein Lineares Programm für einen konkreten Anwendungsfall hergeleitet werden.

### 6.4 minMPS in Verbindung mit dem komplementären Graphen

Es sei  $\bar{G}$  der Komplementgraph von  $G$  mit  $B \subset G$ . Es gilt, dass  $B \not\subset \bar{G}$ . minMPS auf  $G$  löst auf  $\bar{G}$  das Problem einer minimalen Zerlegung in mehrfache Pseudocliquen.

**Definition 6.20.**  $Q_i$  ist eine *mehrfache Pseudoclique*, wenn  $Q_i = C_i$  selber eine Clique ist oder wenn sie in eine Aneinanderreihung von Cliques  $C_j^i$  mit

$$C_1^i, p_{1,1}^i, \dots, p_{1,n_1}^i, C_2^i, p_{2,1}^i, \dots, p_{2,n_2}^i, C_3^i, \dots, p_{m_i-1,1}^i, \dots, p_{m_i-1,n_{m_i-1}}^i, C_{m_i}^i$$

und  $m_i \geq 2$  zerfällt. Dabei besteht die Schnittmenge zwischen aufeinander folgenden Cliques  $C_j$  und Mengen  $p_{j+1}$  und  $p_j$  und  $C_{j+1}$  jeweils nur ein Knoten. Für die paarweise disjunkten Mengen  $p_j$  gilt, dass sie nur aus drei blauen Knoten bestehen, die durch blaue Kanten in Form eines Pfades verbunden sind. Das heißt, dass  $p_j \subset B(G)$ ,  $|V(p_j)| = 3$  und dass  $p_j$  zusammenhängend und kreisfrei ist. Die Wertigkeit dieser Menge  $Q_i$  ist  $m_i$ .

Erneut gilt wie in Abschnitt 4.7:

**Lemma 6.21.** *Jede minimale Zerlegung des Graphen  $G$  mit  $B \subset G$  in mehrfach pseudostabile Mengen mit Wertigkeit  $\zeta G$  entspricht einer minimalen Zerlegung des Graphen  $\bar{G}$  mit  $B \not\subset \bar{G}$  in mehrfache Pseudocliquen mit Wertigkeit  $\bar{\zeta}(\bar{G}) = \zeta(G)$ .*

*Beweis.* „ $\Rightarrow$ “ Es sei  $P$  eine Zerlegung des Graphen  $G$  mit  $B \subset G$  in mehrfach pseudostabile Mengen mit Wertigkeit  $\zeta G$ .

Es gilt  $\bar{\chi}(\bar{G}) = \chi(G)$  (vgl. [S. 59 60]). Ist  $\zeta(G) < \chi(G)$ , so muss es auch eine Zerlegung  $Q$  des Graphen  $\bar{G}$  in mehrfache Pseudocliquen mit Wertigkeit  $\bar{\zeta}(\bar{G}) = \zeta(G)$  geben. Diese kann wie folgt konstruiert werden:

- Jede stabile Menge  $D$  in  $P$  im Graphen  $G$  ist eine Clique  $C$  in  $Q$  im Graphen  $\overline{G}$ . Damit ist  $|D| = |C|$ .
- Da  $B \not\subset \overline{G}$ , kann die Menge von Pfaden  $Pf$  in  $P$  als Menge der Pfade  $Pf'$  in  $Q$  übernommen werden.

Damit gilt  $\overline{\zeta}(\overline{G}) = \zeta(G)$ .

„ $\Leftarrow$ “ Es sei  $Q$  eine Zerlegung des Graphen  $\overline{G}$  in Pseudocliquen mit Wertigkeit  $\overline{\zeta}(\overline{G})$ .

Nun kann wie bei der Hinrichtung diese Zerlegung in einer Zerlegung  $P$  des Graphen  $G$  in pseudostabile Mengen mit Wertigkeit  $\zeta(G) = \overline{\zeta}(\overline{G})$  konstruiert werden.  $\square$

minPS löst auf  $\overline{G}$  das Problem einer minimalen Zerlegung in mehrfache Pseudocliquen (minMPC). Wie in Abschnitt 4.7 kann nach der praktischen Relevanz gefragt werden. In den vorhergehenden Kapiteln wurden nur perfekte Graphen untersucht. Damit kann sowohl eine minimale Graphenfärbung als auch eine minimale Zerlegung des Graphen in Cliques in polynomialer Zeit gefunden werden.

Für allgemeine einfache Graphen ist die Feststellung von Bhasker und Samad besonders wichtig: „In particular, it appears that our algorithms perform increasingly better relative to the coloring algorithms for larger graphs.“ [S. 11 10] Es kann also effizienter sein auf großen Graphen statt einer Graphenfärbung auf dem Komplementgraphen direkt eine Zerlegung in Cliques zu berechnen. Dies ist von Relevanz, wenn ein Zugang zu minMPS oder minMPC über eine Graphenfärbung oder Zerlegung in Cliques gewählt wird.

## 6.5 Zusammenfassung

In diesem Kapitel wurde zunächst das Problem minMPS definiert. Dieses ist wie minPS im Allgemeinen  $\mathcal{NP}$ -vollständig.

Weiter wurden die unteren Schranken  $l_C(G)$  bzw.  $l_{C,i}(G)$ ,  $l_R$ ,  $l_B$  und damit  $L_{BR}$  sowie die Variationen  $l_M$  und  $l_{RM}$  hergeleitet.

Im folgenden Kapitel wird ein Praxisproblem beschrieben, das mit minMPS gelöst werden kann. Dazu werden verschiedene Heuristiken eingeführt, die von der Graphenfärbung inspiriert sind. Auch soll untersucht werden, inwiefern die unteren Schranken aus dem vorherigen Abschnitt verwendet werden können.



## Document Clustering

*Document Clustering* (oder auch: *Text Clustering*) ist ein spezielles Anwendungsgebiet des Textminings und ein Teilgebiet der Clusteranalysen. Aufgabe ist es, Ähnlichkeitsstrukturen in einer gegebenen Menge von unstrukturierten Dokumenten zu finden. Dabei kann es darum gehen, Dokumente schlicht in einer bestimmten Form automatisch zu kategorisieren, wobei die Kategorien vorher nicht bekannt sein müssen. Sind die Kategorien vorher gegeben, spricht man oft vom *Text Classification Problem*. Andere Anwendungsfelder verlangen aber auch ebenso eine Inhaltsanalyse, zum Beispiel in Form von Sachinformationen oder der Thematik der Dokumente. Vorhanden sind oft nur die Metadaten der Dokumente.

Hier soll nun ein graphentheoretischer Ansatz mittels einer Graphenzerlegung in mehrfach pseudostabile Mengen formuliert werden. Dabei werden die beiden Spezialfälle *soft* und *hard Document Clustering* abgedeckt. Für dieses  $\mathcal{NP}$ -vollständige Problem werden ein Greedy-Ansatz und eine Formulierung als ganzzahliges lineares Programm hergeleitet.

Dieser Ansatz durch eine Neuformulierung als graphentheoretisches Problem ist neu. Zur Herleitung sollen in einem ersten Abschnitt Definitionen und Grundlagen beschrieben werden. In einem weiteren Abschnitt soll kurz die Problematik des Ähnlichkeitsmaß zwischen verschiedenen Dokumenten beschrieben werden. Dann wird das Problem als PS-Document Clustering umformuliert und es werden verschiedene Ansätze zur Problemlösung beschrieben: Ein lineares Programm und verschiedene Greedy-Ansätze. Zum Schluss sollen diese Ansätze ausgewertet und diskutiert werden.

Die Forschung zum Thema Document Clustering betrachtete in den letzten Jahren vor allem Methoden und Heuristiken. In [55] versuchten Karaa u. a. beispielsweise Dokumente aus der MEDLINE Datenbank mittels evolutionären Algorithmen zu clustern. Mu u. a. nutzen wiederum maschinelles Lernen ([65], vgl. auch [3]). Nur wenige Autoren wie Stanchev nutzen überhaupt Graphen (vgl. [76]). Andere Autoren betrachteten verwandte Probleme. So betrachteten Hirsch u. Di Nuovo Document Clustering im

Kontext von Suchanfragen, Lee u. a. wiederum betrachteten hierarchisches Clustering (vgl. [61]).

Neben einer Fülle von Methoden gibt es auch eine wahre Flut von Anwendungsgebieten. Abseits der klassischen Anwendungen wie Patente, Bibliotheken sind hier auch Banken und der Marketingbereich, insbesondere aber das Stichwort „Social Media“ zu nennen. Auch hier ist der Vergleich von Texten wichtig, wie Sutanto u. Nayak beschreiben. Sie bedienen sich eines gerankten Document Clustering. Vgl. auch die Arbeiten von Aggarwal u. Reddy in [2] oder [84], [67] und [47]. Im Bereich der Bioinformatik und Medizininformatik geht es insbesondere um das automatische Erkennen und Klassifizieren anhand von medizinischen, chemischen oder biologischen Entitäten (vgl. z. B. [66], [49] oder [81]).

## 7.1 Definition und Grundlagen

Jain und Dubes führten schon 1988 aus, dass es eine große Anwendungsbandbreite für automatische Klassifizierung gebe. Grundsätzlich sind die Objekte in einer solchen Sicht nicht eingeschränkt, denn neben Texten können auch Bilder, Videos und alle anderen Formen von Daten kategorisiert werden. „Cluster analysis has found applications in such diverse disciplines as biology, psychology, archaeology, geology, marketing, information retrieval, and remote sensing.“ [S. 223 50] Dabei führt er auch aus, dass bereits seit den frühen 1960er Jahren in diesem Feld geforscht wird.

Zunächst wurde vor allem in der Kategorisierung und Verschlagwortung von Dokumenten geforscht. So wurde versucht, Texte automatisch mit Hilfe von einem vorgegebenen Vokabular von Schlagworten zu indizieren. Auch automatisches Sortieren und Filtern von Dokumenten gehörten zu den frühesten Anwendungsproblemen, so Feldman und Sanger (vgl. [S. 65ff 27]).

Durch die Digitalisierung und Vernetzung der verschiedenen Datenbanken und Archiven ist es nun allerdings auch möglich auf eine ständig wachsende Anzahl von Daten zuzugreifen und diese elektronisch zu verarbeiten. Nicht nur zu wissenschaftlichen Zwecken, sondern auch für Internetsuchmaschinen wird der Ansatz des Document Clustering verwendet (vgl. bspw. [S. 47 28]). Dazu werden auch Suchergebnisse aus ähnlichen Clustern angezeigt. Sucht der Nutzer etwa nach „Auto“, so finden sich auch Ergebnisse für „Kraftfahrzeug“ (vgl. [27, S. 82ff]). Ebenso sind Banken, der Marketingbereich und die Medizin an entsprechenden Resultaten interessiert. Zusätzlich sind neue technische Ansätze wie beispielsweise Deep Learning entwickelt worden.

Durch die wachsende Anzahl digital zugänglicher Fachveröffentlichungen sind sehr viele wissenschaftliche Quellen online zugänglich. Es stellt sich nur die Frage, wie ein Forscher einen Überblick über die für ihn relevanten Informationen bekommen kann.

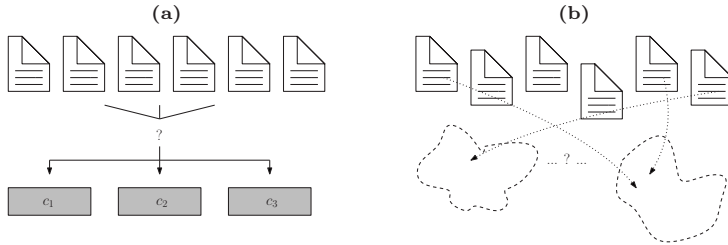
Als Beispiel für die beständige Forschung in diesem Bereich kann TREC – Text REtrieval Conference – genannt werden. Diese wird seit 1992 organisiert und bündelt Kompetenzen und Forschung im Bereich Information Retrieval. Dabei werden auch für verschiedene Bereiche Wettbewerbe abgehalten (vgl. z. B. [71]).

In diesem Kapitel sollen zunächst die Grundzüge des Document Clustering kurz dargestellt werden. Mit den entsprechenden Informationen kann ein Graph  $G = (V, E)$  mit Dokumenten als Knoten in  $V$  erzeugt werden. Werden die Kanten abhängig von der Ähnlichkeit der Dokumente gesetzt, kann der Graph in unterschiedliche Mengen zerlegt werden, die solchen Document Clustern entsprechen. Somit zerfällt das Document Clustering in zwei Schritte: Die Bestimmung einer Ähnlichkeitsfunktion zwischen den Dokumenten und dem eigentlichen Clustering. Letzteres ist ein graphentheoretisches Problem.

Hier soll nun der Fokus des Kapitels liegen. Es wird ein Clustering mit Hilfe einer Zerlegung eines solchen Dokumentengraphen in mehrfach pseudostabile Mengen dargestellt und das dadurch gewonnene Document Clustering evaluiert. Zusätzlich zu der Einteilung in Klassen können auch solche Dokumente identifiziert werden, die zwischen solchen Klassen liegen.

Konkret sollen hier verschiedene Dokumente innerhalb von MEDLINE (vgl. <http://pubmed.gov>) analysiert werden. Das Medical Literature Analysis and Retrieval System Online ist eine bibliografische Datenbank, die vom National Center for Biotechnology Information gepflegt wird und die wesentliche wissenschaftliche Publikationen aus dem Bereich der Medizin, Psychologie und des Gesundheitssystems beinhaltet. Dazu werden verschiedene Heuristiken und Schranken untersucht. Das ist insbesondere im Bereich der Medizin- und Bioinformatik ein derzeit weit genutztes Feld, wie Zhu u. a. beurteilen: „Mining biomedical texts for knowledge discovery and hypothesis generation has become a very active field“ [S. 1944–86]. Der große Vorteil von MEDLINE ist auch die Verschlagwortung der Dokumente mittels MeSH (Medical Subject Headings).

Zunächst soll der Fall betrachtet werden, in dem Dokumente in bereits definierte Kategorien sortiert werden. Dieses Problem wird auch als *Text Classification Problem* bezeichnet. Manning u. a. definieren das Problem formal wie folgt (vgl. [S. 237ff–62]):



**Abbildung 7.1:** Schematische Darstellung von *Text Classification* (a) und *Text Clustering* (b).

Gegeben sei eine Beschreibung  $d \in \mathbb{X}$  von Dokumenten, wobei  $\mathbb{X}$  der Raum aller Dokumente sei. Die Kategorien oder Klassen  $\mathbb{C} = \{c_1, \dots, c_j\}$  seien vorgegeben.

Manning u. a. gehen davon aus, dass es bereits eine zugeordnete Trainingsmenge  $\mathbb{D}$  mit Elementen  $\langle d, c \rangle \in \mathbb{X} \times \mathbb{C}$  gibt.

Gesucht ist nun eine Klassifizierungsfunktion

$$y : \mathbb{X} \rightarrow \mathbb{C}$$

die Dokumente in die Klassen abbildet. Diese kann nach dem Vorschlag von Manning u. a., wenn eine Trainingsmenge gegeben ist, durch einen selbstlernenden Algorithmus mittels maschinellem Lernen gefunden werden. Alternativ kann hier auch eine einfache gewichtete Zuordnungsfunktion genutzt werden, wenn eine Distanz- oder Ähnlichkeitsfunktion der Dokumente zu den Klassen in  $\mathbb{C}$  gegeben ist.

Als Beispiel kann die Klassifizierung von Patenten nach der Europäische Patentklassifikation (European Classification System, ECLA) angeführt werden. Hier gibt es verschiedene aktuelle Ansätze (vgl. z. B. [25]).

Ein *Clusteringverfahren* verfolgt das Ziel, eine Menge von Objekten in Teilmengen, sogenannte Cluster, zu partitionieren. Objekte innerhalb eines Clusters sollen zueinander mehr Ähnlichkeit haben als zu solchen, die in anderen Clustern liegen. Die Teilmengen sind vorher nicht bekannt. Es findet also keine Klassifizierung in vorher festgelegte Klassen statt. Zu den beiden Verfahren vgl. auch Abb. 7.1.

Manning u. Schütze stellen die verschiedenen Möglichkeiten von Document Clustering dar. So kann die Sprache der Dokumente identifiziert werden,



der Autor oder eben das Thema der Texte (vgl. [S. 575 63]). Als Meta-informationen zu einem Dokument stehen neben dem Autor, dem Titel auch der Verlag und die eventuell die Zeitschrift oder Buchserie, das Veröffentlichungsjahr, eine Zusammenfassung und oft auch Schlagworte zur Verfügung. In der Praxis werden oft verschiedene Metainformationen der Dokumente genutzt, um ein Clustering zu erzeugen.

Je nach Anwendungsfall können die Objekte zu einem eindeutigen Cluster oder zu mehreren gehören, eventuell sogar nur mit einer gegebenen Wahrscheinlichkeit. Manning u. a. bezeichnen dies als *hard clustering* oder *soft clustering* (vgl. [S. 322 62]). Hier soll zunächst angenommen werden, dass die Cluster disjunkt sind, Objekte also einem eindeutigen Cluster zugeordnet werden können. Es wird sich aber zeigen, dass der hier vorgestellte neue Ansatz zu einem soft clustering führt.

Manning u. a. verwenden hier dediziert keine selbstlernenden Algorithmen: „Clustering is the most common form of unsupervised learning. No supervision means that there is no human expert who has assigned documents to classes.“ [S. 321 62]. Wichtig für diesen Anwendungsfall ist die Existenz einer Ähnlichkeitsfunktion zwischen den Dokumenten. Es muss auch zwischen *flachem Clustering* und *hierarchischem Clustering* unterschieden werden. Ersteres generiert nur unabhängige Cluster ohne Relationen zu anderen Clustern. Hierarchische Verfahren produzieren eine Baumstruktur über einer Dokumentenmenge und damit eine Hierarchie von Clustern, indem die jeweils ähnlichsten Cluster verschmolzen werden.

Die Cluster Hypothese ist essentiell für alle Ansätze des Document Clustering: „Documents in the same cluster behave similarly with respect to relevance to information needs.“ [S. 322 62] Damit können auch die verschiedensten Anwendungsfelder, wie oben schon genannt, bedient werden.

Manning u. a. definieren Document Clustering wie folgt (vgl. [S. 326 62]):

**Definition 7.1** (K-Document Clustering). Gegeben sind eine Menge von Dokumenten  $\mathbb{D} = \{d_1, \dots, d_N\}$ , eine gewünschte Anzahl von Clustern  $K$  und eine Funktion  $q$ , die die Qualität eines Clustering bewertet. Gesucht ist eine Zuordnung  $\gamma : D \rightarrow \{1, \dots, K\}$  die  $q$  minimiert.

Die Abbildung  $\gamma$  sollte in aller Regel surjektiv sein, d. h. kein Cluster bleibt leer. Die Funktion wird in aller Regel in Abhängig einer Ähnlichkeitsfunktion zwischen den Dokumenten erstellt. Dieses Verfahren wird auch *K-Clustering* genannt. Es teilt alle Dokumente in  $K$  Cluster von jeweils ähnlichen Dokumenten auf, sucht also ein flat clustering in  $K$  Cluster mit minimalen Kosten.

Alternativ kann auch eine minimale Anzahl von Clustern gesucht werden:

**Definition 7.2** (Document Clustering). Gegeben eine Menge von Dokumenten  $\mathbb{D} = \{d_1, \dots, d_N\}$  und eine Ähnlichkeitsfunktion  $sim : \mathbb{D} \times \mathbb{D} \rightarrow \mathbb{R}^+$  sowie ein  $\epsilon \in \mathbb{R}^+$ . Gesucht ist eine minimale Anzahl von Clustern, so dass für je zwei Dokumente  $x, y$  innerhalb eines Clusters gilt, dass  $sim(x, y) \geq \epsilon$ .

In diesem Ansatz ist die Anzahl der Cluster also nicht vorgegeben. Dies liefert für viele Anwendungsfälle zusätzliche Informationen, da die inhärente Datenstruktur nicht aufgelöst wird. Dieses Problem soll im Folgenden betrachtet werden.

Doch zunächst sollen noch einmal alle nötigen Arbeitsschritte bestimmt werden. Jain u. a. führen folgende Schritte an, um Document Clustering zu betreiben (vgl. [S. 266f 51]):

1. *Pattern Representation*: Strukturen und Informationen auslesen, die die Dokumente angemessen repräsentieren.
2. Eine *Ähnlichkeitsfunktion* definieren.
3. Das eigentliche *Clustering* ausführen.
4. Optional die entsprechenden Daten abstrahieren, d. h. beispielsweise die Cluster entsprechend menschenlesbar zu benennen.
5. Ebenfalls optional ist der Schritt, die Ausgabe zu validieren.

Die letzten beiden Schritte sind lediglich im abschließenden Praxisteil relevant. Ebenfalls soll von einer geeigneten Repräsentation der Daten ausgegangen werden.

Ist die Ähnlichkeitsfunktion gegeben, so kann das eigentliche Clustering als graphentheoretisches Problem formuliert werden. So soll noch kurz ein Überblick über die verschiedenen Möglichkeiten, Ähnlichkeiten zwischen Dokumenten zu definieren, gegeben werden um dann den Aspekt der Graphentheorie näher zu beleuchten.

## 7.2 Dokumentenähnlichkeit

Die Frage nach der Ähnlichkeit von zwei Textdokumenten stellt sich nicht nur im Text Mining, sondern auch im Information Retrieval. Beide Begriffe gehen fließend ineinander über. Letzteres stellt sich eher der Frage nach dem Wiederauffinden von Information, Text Mining geht wiederum

eher auf die Frage ein, wie bisher unbekannte Informationen gefunden werden können. Document Clustering fällt damit in beide Bereiche. Ein gutes Document Clustering sollte ähnliche Dokumente in den selben Cluster sortieren und unähnliche in separate. Dazu muss die Ähnlichkeiten zwischen Dokumenten messbar sein. Bestimmte Eigenschaften, sogenannte *features*, der Elemente bzw. Dokumente in  $\mathbb{D}$  werden zu diesem Zweck mit einer Funktion in eine reelle Zahl abgebildet.

Das *Vector Space Model* bzw. *Vektorraum-Retrieval* nimmt zu diesem Zweck an, dass alle Elemente in  $\mathbb{D}$  Vektoren in einem *Feature Space*  $\mathbb{D}^N$  sind. Dann kann die Ähnlichkeit einfach als Distanz zwischen den Vektoren berechnet werden (vgl. [S. 84f 27]).

Eine Möglichkeit ist es, den gewichteten Vektor der Worte eines Textes als eine Koordinate in einem  $n$ -dimensionalen Vektorraum darzustellen (vgl. [52, S. 275]). In diesem Zusammenhang ist allerdings wichtig, dass seltene Wörter höher gewichtet werden sollten und Standardwörter (sogenannte „stop words“) ignoriert werden. Dabei entspricht die *term frequency document frequency* der Anzahl der Verwendungen über einem oder allen Dokumenten. Das Verfahren, das Wörter mit einer kleinen document frequency belohnt, wird *inverse document frequency* genannt (vgl. [85, S. 99]).

Kombiniert ergibt sich das TF.IDF-Maß (*term frequency-inverse document frequency*). Dieses kombinierte Verfahren hat einige Vorteile bei der Zusammenstellung von Suchergebnissen, aber auch im Information Retrieval. Wörter mit einem hohen Wert im TF.IDF-Maß würden einzelne Dokumente besonders gut beschreiben (vgl. [4, S. 29ff]). [6] beschreiben die Bedeutung dieses Verfahrens für Vorschlagverfahren für Forschungsartikel: „TF-IDF was the most frequently applied weighting scheme.“ [6, S. 305]. Die Berechnung kann für  $N$  Dokumente in  $\mathbb{D}$  nach [68, S. 8f] folgendermaßen erfolgen: Es sei  $f_{ij}$  die Anzahl des Wortes  $i$  in Dokument  $j$ . Dann ist die term frequency  $TF_{ij}$  wie folgt mit einer Normalisierung über die maximale Anzahl eines Wortes als Abbildung nach  $[0, 1]$  definiert:

$$TF_{ij} = \frac{f_{ij}}{\max_k f_{kj}}$$

Die inverse document frequency wiederum wird als

$$IDF_i = \log_2 \left( \frac{N}{n_i} \right)$$

definiert. Dabei ist  $n_i$  die Anzahl des Wortes  $i$  in allen  $N$  Dokumenten. Dann ist das TF-IDF-Maß definiert durch  $TF_{ij} \times IDF_i$ . Es ergeben sie

dann für jedes Dokument ein reelwertiger Vektor, der genutzt werden kann, um eine Vektordistanz zu berechnen.

Auch andere Ansätze werden verwendet, um am Ende eine Vektordistanz zu berechnen. Oft werden sogenannte Kernel-Methoden verwendet, um die Dimension  $n$  zu reduzieren (vgl. [75, S. 11ff]). Mit Hilfe der euklidischen Norm findet sich dann beispielsweise die Form

$$sim(x_i, x_j) = \sqrt{\sum_k (x_{i,k} - x_{j,k})^2}$$

Ebenfalls gebräuchlich Kosinus-Ähnlichkeit, die sich wie folgt definiert:

$$sim(x_i, x_j) = (x'_i \cdot x'_j) = \sum_k x'_{i,k} \cdot x'_{j,k}$$

wobei  $x'$  der normalisierte Vektor  $x' = \frac{x}{|x|}$  ist.

Jain u. a. beschreiben, warum dieser Ansatz häufig genutzt wird: „The Euclidean distance has an intuitive appeal as it is commonly used to evaluate the proximity of objects in two or three-dimensional space. It works well when a data set has 'compact' or 'isolated' clusters“ [S. 272 51]. Ein anderer Ansatz ist die sogenannte Manhattan-Distanz, die sich aus der euklidischen Norm ergibt:

$$sim(a, b) = \sum_i |a_i - b_i|$$

Hier wird die Ähnlichkeit über die Summe der absoluten Differenzen der Einzelkoordinaten der Vektoren definiert. Strehl u. a. untersuchten verschiedene Normen in Hinblick auf Webseiten Clustering und kamen zu dem Resultat, die Manhattan Distanz sei insbesondere invariant gegenüber Verschiebungen (vgl. [S. 60 78]). Sherwood u. a. können in ihren Untersuchungen ergänzen, die Manhattan Distanz repräsentiere „more accurately [] differences in [...] high-dimensional data“ [S. 46 74].

Eine weitere, oft angewendete Distanz ist die Tanimoto bzw. Jaccard Distanz. Sie kann gut eingesetzt werden, wenn man unterschiedlich große Mengen mit verschiedenen Elementen hat. Im Textmining bietet sich dabei beispielsweise an, Texte einfach in Vektoren zu überführen. Da auf MEDLINE Listen  $M$  mit Schlagworten, den sogenannten MeSH-Terms, vorliegen, bietet es sich an die Ähnlichkeitsfunktion entsprechend zu wählen:

$$\text{sim}(a, b) = \frac{|M_a \cap M_b|}{|M_a \cup M_b|} \quad \forall a, b \in \mathbb{D}$$

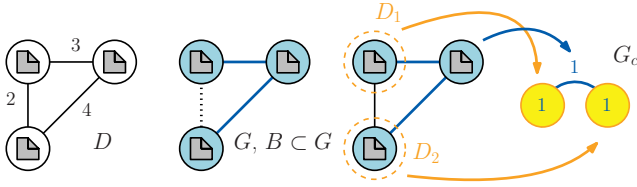
Dabei ist der Bildbereich dieser Funktion  $[0, 1]$  (vgl. [S. 52 48]).

### 7.3 Ein graphentheoretischer Ansatz des Clusterings

Die Zerlegung eines Graphen in Cluster wurde in der Literatur schon viel diskutiert. So urteilt Schaeffer: „the field of graph clustering has grown quite popular and the number of published proposals for clustering algorithms as well as reported applications is high“ [S. 27 72]. In diesem Artikel gibt die Autorin eine gute Übersicht über die häufigsten Clusteringalgorithmen. Es kann grundsätzlich unterschieden werden zwischen dem Clustering von gewichteten Graphen und gerichteten Graphen. Ungerichtete und ungewichtete Graphen werden in aller Regel in Cliques oder stark zusammenhängende Subgraphen (vgl. [43]) zerlegt.

Schaeffer weist auf einen wichtigen Punkt hin: „not all graphs have a structure with natural clusters. Nonetheless, a clustering algorithm outputs a clustering for any input graph.“ [S. 27 72] So ist es unerlässlich, gründlich die Eingabedaten zu betrachten, da im schlimmsten Fall eine zufällige Partition in Cluster generiert wird. Wird im ersten Schritt eine dem Problem angemessene Ähnlichkeitsfunktion gewählt, wird auch der Graph, der diese Daten repräsentiert in nicht zufällige Cluster zerfallen. So gibt es auch keine allgemeingültige Definition, um Graphen in Cluster zu zerlegen: „Unfortunately, no single definition of a cluster in graphs is universally accepted, and the variants used in the literature are numerous“ [S. 33 72]. Dabei reicht die Bandbreite von Zerlegungen des Graphen in bestimmte Subgraphen, wie Cliques oder stark zusammenhängende Subgraphen oder aber auch, bis hin zur Vereinfachung des Graphen durch Zusammenfassung von sogenannten natürlichen Clustern, wie es Edachery u. a. vorschlagen (vgl. [22]).

An dieser Stelle soll nun ein leicht anderer Ansatz gewählt werden. Er wurde in seinen Grundzügen schon von Schaeffer (vgl. [S. 34 72]) beschrieben: Zunächst wird die Ähnlichkeitsfunktion auf das Intervall  $[0, 1]$  eingeschränkt, „where one corresponds to a 'full' edge, intermediate values to 'partial' edges, and zero to there being no edge between two vertices.“ (ebd.) So können Kanten ab einer bestimmten Ähnlichkeit existieren und bis zu einer gewissen, gegebenen, Ähnlichkeit gibt es „Teilkanten“. Hier sollen Kanten allerdings Unähnlichkeit und nicht Ähnlichkeit ausdrücken,



**Abbildung 7.2:** Darstellung zu Beispiel 7.1. Diese zeigt zunächst die Dokumente mit ihren Ähnlichkeiten, den daraus resultierenden Dokumentengraphen, eine Zerlegung in mehrfach pseudostabile Mengen und die zugehörige gewichtete Graph des damit einhergehenden Clusterings.

d. h. es soll nach Clustern mit möglichst wenigen Kanten gesucht werden. Dann kann der Graph in stabile bzw. pseudostabile Mengen zerlegt werden. Vgl. hierzu auch [21].

Gegeben sei eine Menge von Dokumenten  $\mathbb{D} = \{d_1, \dots, d_N\}$ , eine Ähnlichkeitsfunktion  $sim : \mathbb{D} \times \mathbb{D} \rightarrow \mathbb{R}^+$  und ein  $\epsilon \in \mathbb{R}^+$ . Diese Funktion kann auch auf das Intervall  $[0, 1]$  eingeschränkt werden, dazu muss eine normalisierte Funktion  $sim' : D \times D \rightarrow [0, 1]$  als  $sim'(x, y) = \frac{sim(x, y)}{\max sim(x, y)}$  definiert werden. Dann kann ein Graph wie folgt konstruiert werden:

$$G = (V, E) \text{ mit } V = \mathbb{D} \text{ und}$$

$$E = \{(d_i, d_j) \mid sim(d_i, d_j) \leq \epsilon\}$$

Das heißt, dass alle Knoten Dokumenten entsprechen und Kanten zwischen ihnen nur existieren, wenn sie sich weniger ähnlich als ein Grenzwert  $\epsilon$  sind. Somit sind alle unähnlichen Dokumente mit Kanten verbunden.

Nun ist es möglich, die Dokumente mittels einer Graphenfärbung in Cluster zu zerlegen. Dies ist zulässig, da alle Dokumente in einer Farbklasse, also einem Cluster, sich ähnlicher sind als der Schwellenwert  $\epsilon$ . Dieser Ansatz ist nicht sonderlich innovativ, ist er doch der Zerlegung in Cliques sehr ähnlich.

Nimmt man jedoch eine weitere Schranke  $\iota$  mit  $0 < \iota < \epsilon$  hinzu, so kann man eine weitere Kantenmenge hinzufügen:

$$B = (V, E') \text{ mit } E' = \{(d_i, d_j) \mid \iota \leq sim(d_i, d_j) \leq \epsilon\}$$

Es wird also eine bestimmte Kantenmenge blau gefärbt. Alle Knoten sind zunächst blau. Es gilt offensichtlich  $B \subset G$ . Damit unterscheiden die Kanten zwischen „unähnlichen“ Dokumenten mit schwarzen Kanten und zwischen „nicht ganz unähnlichen“ Dokumenten mit blauen Kanten. Formal kann man es wie folgt definieren:

**Definition 7.3** (PS-Document Clustering). (Gegeben sei der oben definierte Graph  $G$  und  $B \subset G$ . Eine Lösung für minPS'-a liefert ein Clustering der Dokumente in mehrfach pseudostabile Mengen in  $\zeta(G)$  Cluster und Dokumenten, die in Pfaden zwischen den Clustern  $D$  liegen.

In der Praxis ist dies keine Einschränkung. Ohne weiteres kann der Graph ungerichtet sein, da für zwei Dokumente  $d_i, d_j$  immer gilt, dass  $\text{sim}(d_i, d_j) = \text{sim}(d_j, d_i)$  ist. Da jeder Clusteringalgorithmus immer entscheiden muss, ob zwei Dokumente nun Teil des selben Clusters sind, kann auch das Kantengewicht immer entfallen – denn es kann jederzeit eine Kante für die Entscheidung gesetzt werden. Somit ist es lediglich eine Frage der Modellierung und eine Einschränkung auf einfache Graphen ebenso wie auf einfache Graphen mit einem blauen Teilgraphen möglich.

Sucht man nun eine minimale Zerlegung des Graphen  $G$  in mehrfach pseudostabile Mengen ohne die azyklische Bedingung und ohne, dass Knoten in einem Pfad eindeutig einem Pfad zugeordnet werden können (minMPS'-a), so kann man in der Praxis folgende Vorteile erwarten:

- In stabilen Mengen dieser Zerlegung sind alle Dokumente erlaubt, die sich nicht unähnlicher sind als die Schranke  $\epsilon$ .
- Dokumente, die mit einer oder mehreren Clustern Ähnlichkeit eine gewisse, von der Schranke  $\iota$  abhängige, Ähnlichkeit haben, können in den Pfaden zwischen den zugehörigen stabilen Mengen liegen und sind somit zählbar.
- Minimale Cluster, die nur aus wenigen Dokumenten bestehen, können somit zuverlässig eliminiert werden.

Diese zusätzlichen Informationen lassen auch eine weitaus tiefere Analyse der entstandenen Cluster zu. Ist  $P_i$  eine Zerlegung des Graphen  $G$  in mehrfach pseudostabile Mengen, so kann – Definition 3.5 erweiternd – recht einfach der *gewichtete Graph des Clusterings* generiert werden:

**Definition 7.4** (Gewichteter Graph des Clusterings). Gegeben sei eine Lösung von minPS mit Wert  $\zeta(G)$  auf  $G$  und  $B \subset G$ . Es sei nun

$$G_c = (V_c, E_c) \text{ mit } V_c = \{D_j^i \in P_i\}, d(D_j^i) = |D_j^i| \text{ und} \\ d(D_j^i, D_k^i) = o_{j,k}^i, E_c = \{(D_j^i, D_k^i), d(D_j^i, D_k^i) > 0\}$$

Somit entsprechen die Knoten den Clustern, wobei ihr Maß die Anzahl der Dokumente im Cluster ist. Die Kanten geben mit ihrem Gewicht die Dokumente zwischen den Clustern an. So kann die generierte Zerlegung der Dokumente in Cluster auch angemessen visualisiert werden, vgl. dazu die Abb. in den folgenden Abschnitten.

**Beispiel 7.1.** Gegeben seien drei Dokumente mit einer Ähnlichkeit, vgl. Abb. 7.2. Es sei  $\epsilon = 2,5$  und  $\iota = 5$ . Es ergibt sich ein Graph mit blauen Knoten und zwei blauen Kanten. Eine Kante ist schwarz. Eine Zerlegung in mehrfach pseudostabile Mengen findet also zwei Cluster mit jeweils einem Dokument und einen Pfad zwischen beiden Clustern, der ein Dokument beinhaltet. Der gewichtete Graph des Clusterings findet sich in Abb. 7.2 rechts. Jeder Knoten entspricht einem Cluster und das Gewicht der Kanten der Anzahl von Pfaden zwischen den Clustern.  $\triangle$

## 7.4 Neue Ansätze

### 7.4.1 Lineare Programmierung

Auch für dieses Problem kann ein ganzzahliges lineares Programm hergeleitet werden. Es sei  $G = (V, E)$  ein Graph mit einem Teilgraph  $B \subset G$  mit blauen Knoten und Kanten. Es sei  $T$  die Menge aller Pfade der Länge drei in  $B$ .

In dem folgenden Ansatz sei  $y_k$  die Variable, die angibt, ob eine Farbe  $k$  genutzt wird. Ist  $y_k = 0$ , so wird die Farbe  $k$  nicht genutzt.  $x_{i,k}$  gibt an, ob der Knoten  $i \in G$  mit der Farbe  $k$  gefärbt wurde. Die Farbe  $k = 0$  wird dabei für Knoten verwendet, die in einem Pfad  $p$  liegen.

Bedingung minMPS-a-IP.1 stellt nun sicher, dass jeder Knoten mindestens eine Farbe oder die Farbe  $k = 0$  zugewiesen bekommt. Für jeden Knoten  $i$



$$\begin{array}{ll}
\text{[minMPS-a-IP]} & \min \sum_{k=1}^n y_k \\
\text{s.d.} & \sum_{k=1}^n x_{i,k} = 1 \quad \forall i = 0, \dots, n \\
& \hspace{15em} (\text{minMPS-a-IP.1}) \\
& x_{i,k} - y_k \leq 0 \quad \forall i = 0, \dots, n, \forall k = 1, \dots, n \\
& \hspace{15em} (\text{minMPS-a-IP.2}) \\
& x_{i,k} + x_{j,k} \leq 1 \quad (i, j) \in E(G), \forall k = 1, \dots, n \\
& \hspace{15em} (\text{minMPS-a-IP.3}) \\
& x_{i,0} \leq 0 \quad \forall i \notin B(G) \\
& \hspace{15em} (\text{minMPS-a-IP.4}) \\
& x_{i,k} \geq 0 \quad (\text{minMPS-a-IP.5}) \\
& y_k \leq 1 \quad (\text{minMPS-a-IP.6}) \\
& x_{i,k} + x_{j,k} + x_{v,0} - 2 \leq 0 \quad (i, v, j) \in T, \forall k = 1, \dots, n \\
& \hspace{15em} (\text{minMPS-a-IP.7}) \\
& x_{i,0} + x_{j,0} + x_{v,0} \leq 1 \quad (i, v, j) \in T, \forall k = 1, \dots, n \\
& \hspace{15em} (\text{minMPS-a-IP.8}) \\
& x_{i,k}, y_k \in \mathbb{Z}
\end{array}$$

muss gelten, dass für jede Farbklasse  $k$  gilt, dass  $x_{i,k} - y_k \leq 0$ . Ist Knoten  $i$  nicht in der Farbe  $k$ , so gilt die Ungleichung minMPS-a-IP.2 immer. Liegt er jedoch in der Farbe  $k$ , so muss  $y_k = 1$  sein, damit die Ungleichung gilt. Zwei benachbarte Knoten  $i, j$  dürfen für alle Farben  $k > 0$  nicht in einer Farbklasse liegen, also muss  $x_{i,k} + x_{j,k} \leq 1$  sein, siehe Bedingung minMPS-a-IP.3. Bedingung minMPS-a-IP.4 stellt sicher, dass kein nicht blauer Knoten in der Farbe  $k = 0$  liegt. minMPS-a-IP.6 und minMPS-a-IP.7 wiederum stellt sicher, dass wenn ein Knoten  $v$  in der Farbe  $k = 0$  liegt, alle mit ihm durch blaue Kanten verbundenen blauen Knoten eine andere Farbe haben.

In der Praxis kann minMPS-a-IP genauso wie minMPS-IP in Abschnitt 6.3 nur auf kleinen Instanzen mit akzeptabler Laufzeit eingesetzt werden.

## 7.4.2 Ein Greedy-Ansatz

Gegeben sei ein Graph  $G = (V, E)$  mit einem Teilgraph  $B \subset G$  von blauen Kanten und Knoten. Ausgehend von einer (nicht notwendigerweise minimalen) Graphenfärbung  $f : V \rightarrow F$  mit  $F \subset \mathbb{N}$  kann man einen Greedy-Algorithmus entwerfen, der alle möglichen Pfade zwischen die stabilen Mengen legt. Da beim Document Clustering in der Regel keine perfekten Graphen vorliegen, muss eine geeignete Heuristik zur Graphenfärbung genutzt werden. Alternativ kann auch der komplementäre Graph  $\overline{G}$  erstellt werden und ein Algorithmus zur Zerlegung in Cliques genutzt werden. Diese liefert eine Färbung auf  $G$ . Beide Ansätze sollen in zwei folgenden Abschnitten noch genauer beschrieben werden.

Der Greedy-Ansatz versucht nun iterativ jede Farbe zu eliminieren oder zumindest zu reduzieren.

- Für jede Farbklasse  $i$  wird jeder Knoten  $u$  in dieser betrachtet:
  - Ist dieser Knoten noch kein Endknoten eines Pfades  $p$ , so schaue, ob es zwei Knoten  $v, w \in G$  gibt, die durch blaue Kanten mit  $u$  verbunden sind und noch Teil von zwei jeweils verschiedenen Farbklassen sind.
  - Ist dies alles gegeben, so entferne  $u$  aus  $i$  und erstelle einen neuen Pfad  $p = [v, u, w]$ .

---

### Algorithmus 7 GREEDY-DC

---

**Eingabe:** Graph  $G$  mit einer Färbung  $f$  und einer Liste  $T = (t_1, \dots, t_{t_C})$  von möglichen Pfaden.

**Ausgabe:** Zerlegung  $P$  von  $G$  in MPS-a

```
|1| Sortiere alle Farbklassen  $f_1, \dots, f_{|F|}$  aufsteigend nach ihrer Größe
|2| for jede Farbklasse  $f_i$  in  $F$  do
|3|    $T_i \leftarrow$  alle  $t \in T$  mit einem mittleren Knoten in  $f_i$ 
|4|   for jedes  $t_i = (a, b, c)$  in  $T_i$  do
|5|     if  $f(a) \neq f(c)$  und  $ende(b) = false$  then
|6|        $ende(a) \leftarrow true$ 
|7|        $ende(c) \leftarrow true$ 
|8|        $f(b) = 0$ 
|9|     end if
|10|   end for
|11| end for
|12| return  $P$ , wobei  $f$  die stabilen Mengen angibt und  $f_0$  alle Pfade.
```

---

Dies liefert eine gültige Lösung für minMPS<sup>1</sup>-a. Jeder Pfad  $(a, b, c)$  liegt in  $B(G)$  und hat mindestens einen dedizierten End- und Anfangsknoten in zwei verschiedenen stabilen Mengen.

Wie gut eine Lösung dieser Heuristik mit den verschiedenen Ansätzen ist, wird in Abschnitt 7.5 untersucht.

### 7.4.3 Graphenfärbung

Das Problem, zu einem gegebenen ungerichteten und ungewichteten Graphen  $G = (V, E)$  eine minimale Knotenfärbung zu finden, ist im Allgemeinen  $\mathcal{NP}$ -vollständig. Da beim Document Clustering keine perfekten Graphen betrachtet werden, muss eine Näherungslösung über eine geeignete Heuristik berechnet werden.

Kosowski und Manuszewski haben eine detaillierte Übersicht über verschiedene Heuristiken erstellt und diese Ansätze diskutiert (vgl. [58]). Dabei führen sie die meisten auf einen Greedy-Ansatz zurück, vgl. Algorithmus 8.

---

**Algorithmus 8** GREEDY-COLORING nach [S. 8 58]

---

**Eingabe:** Graph  $G = (V, E)$ , Knotenfolge  $K = \{v_1, \dots, v_n\}$

**Ausgabe:** Graphenfärbung  $f : V \rightarrow \mathbb{N}$

|1| **for**  $i = n \dots 1$  **do**

|1|     Wähle für  $f(v_n)$  die kleinste freie Farbklasse.

|2| **end for**

|3| **return**  $f$

---

Nun muss eine geeignete Strategie gefunden werden, um die Knotenfolge  $K$  zu bestimmen.

Ein anderer Ansatz erfolgt über unabhängige Mengen. Dabei wird eine disjunkte Überdeckung der gegebenen Knotenmenge mit möglichst großen unabhängigen Mengen gesucht. Dieser Ansatz, der an dieser Stelle genutzt werden soll, wird von Kosowski und Manuszewsk als *greedy independent sets* (GIS) eingeführt. Hier wird nach maximalen stabilen Mengen gesucht. Alle Knoten werden in einer bestimmten Reihenfolge analysiert und der Knoten  $v_i$  wird mit Farbe  $c$  gefärbt, wenn  $v_i$  mit keinem Knoten in  $c$  benachbart ist. Ist die Farbe  $c$  allen möglichen Knoten zugewiesen, so werden diese Knoten aus  $G$  entfernt und der Algorithmus auf den verbliebenen Graphen mit Farbe  $c + 1$  angewandt.

---

**Algorithmus 9** GIS-COLORING nach [S. 14 58]

---

**Eingabe:** Graph  $G = (V, E)$

**Ausgabe:** Graphenfärbung  $f : V \rightarrow \mathbb{N}$

```
[1]  $c \leftarrow 1$ 
[2]  $f(n) = -1 \forall n \in V$ 
[3] while  $\exists n \in V$  mit  $f(n) = -1$  do
[3]    $available \leftarrow \{v \in V : f(n) = -1\}$ 
[4]   while  $available \neq \emptyset$  do
[4]      $v \leftarrow n \in available$  mit dem kleinsten  $\delta(v)$  in dem von  $available$ 
       induzierten Teilgraphen
[4]      $f(v) = c$ 
[4]     lösche  $v$  aus  $available$ 
[5]   end while
[5]    $c = c + 1$ 
[6] end while
[7] return  $f$ 
```

---

Ist zu einem Graphen  $G = (V, E)$   $n$  die Anzahl der Knoten in  $V$  und  $m$  die Anzahl der Kanten in  $E$ , so kann der Algorithmus 9 mit einer Laufzeit in  $O(mn)$  implementiert werden. Allerdings kann auch eine sublineare Güte in  $O(n/\log n)$  garantiert werden (vgl. [S 274f 54]). Diese Garantie wurde einige Male verbessert (vgl. z. B. [40]).

In der Praxis konnte zu vielen Heuristiken kein Beweis über eine bestimmte Güte erfolgen und die Entscheidung muss auf Beobachtungen und Messungen beruhen. So hat die von Kosowski und Manuszewski als SLF-Methode eingeführte Heuristik<sup>1</sup> zwar eine bessere Laufzeit von  $O((m+n)\log n)$ , aber nur eine garantierte Güte von  $O(n)$ .

Der Saturierungsgrad  $\rho(v)$  eines Knoten  $v \in G$  ist die Anzahl verschiedener Farben zu der dieser Knoten adjazent ist, d. h.  $\min_{v \in N(u)} f(v)$  (vgl. [S. 252 13]).

Nun stehen zwei Algorithmen zur Auswahl, Algorithmus 9 mit einer garantiert sublinearen Güte und Laufzeit  $O(mn)$  und Algorithmus 10 mit einer weniger guten garantierten Güte aber besserer Laufzeit. Es soll im Folgenden evaluiert werden, welche Ansatz in der Laufzeit und im Ergebnis günstiger ist.

---

<sup>1</sup>Die allerdings schon 1997 von Brélaz als DESATUR-Methode beschrieben wurde, vgl. [13].

---

**Algorithmus 10** SLF-COLORING nach [S. 13f 58]

---

**Eingabe:** Graph  $G = (V, E)$

**Ausgabe:** Graphenfärbung  $f : V \rightarrow \mathbb{N}$

- |1|  $f(n) = -1 \forall n \in V$
  - |2| **while**  $\exists n \in V$  mit  $f(n) = -1$  **do**
  - |2|    $v \leftarrow$  der Knoten  $v$  mit dem größten Saturierungsgrad  $\rho(v)$  bzw. bei  
    Kreisen dem größten Grad  $\delta(v)$ .
  - |2|   Wähle für  $f(v)$  die kleinste freie Farbklassse.
  - |3| **end while**
  - |4| **return**  $f$
- 

### 7.4.4 Minimale Zerlegung in Cliques

In Abschnitt 6.4 wurde bereits darauf hingewiesen, dass eine Lösung des Cliquesüberdeckungsproblem auf dem komplementären Graphen  $\bar{G}$  deutlich schneller sein kann, als eine Heuristik zur Graphenfärbung auf dem Graphen  $G$ .

Wir können hier dem ersten Ansatz aus [S. 6 10] folgen:

---

**Algorithmus 11** TSENG CLIQUE-PARTITIONING ALGORITHM

---

**Eingabe:** Graph  $G = (V, E)$

**Ausgabe:** Cliqueszerlegung  $f : V \rightarrow \mathbb{N}$

- |1|  $N = \{n_1, \dots, n_m \in G \mid \delta(n_i) \neq 0 \text{ und } \delta(n_i) \leq \delta(n_{i+1})\}$
  - |2| **for**  $n \in N$  **do**
  - |3|    $u_1 \leftarrow n$
  - |4|    $u_2 \leftarrow \min_{\delta} N(n)$
  - |5|   Gruppiere  $u_1$  und  $u_2$  als neuen verbundenen Knoten  $u_3$
  - |6|   In  $G$  werden nun die Knoten  $u_1$  und  $u_2$  durch  $u_3$  ersetzt. Ein anderer  
    Knoten  $u_x$  ist genau dann mit  $u_3$  verbunden, wenn sowohl  $(u_1, u_x) \in$   
     $E(G)$  als auch  $(u_2, u_x) \in E(G)$ .
  - |7| **end for**
  - |8| Alle Knoten  $u \in G$ , die noch nicht eingruppiert sind, werden einzeln  
    einer Gruppe hinzugefügt.
  - |9| Für alle Knoten  $v, w \in V$  setze  $f(v) = f(w)$  wenn  $v, w \in u_x$  wobei  $u_x$   
    einer der vorher gruppierten Knoten ist.
  - |10| **return**  $f$
- 

In Zeile 1 werden alle Knoten in  $G$  die Nachbarknoten haben aufsteigend nach ihrem Knotengrad sortiert. Diese werden nun in dieser Reihenfolge

betrachtet und als  $u_1$  bezeichnet. Der Knoten  $u_2$  ist der Knoten in der Nachbarschaft  $N(u_1)$  mit dem kleinsten Knotengrad, vgl. Zeile 4. Diese beiden Knoten werden nun gruppiert. Abschließend werden alle noch nicht gruppierten Knoten in eine eigene Gruppe einsortiert. Im Worst Case hat dieser Algorithmus eine Laufzeit von  $O(n^3)$  (vgl. [S. 7 10]).

Weitere Ansätze finden sich beispielsweise bei Feder und Motwani (vgl. [26]), Gramm u. a. (vgl. [36]) und Benati u. a. (vgl. [7]). Eine andere Heuristik wurde von Kou u.a. vorgestellt (vgl. [59]).

Grundsätzlich ist das Vorgehen also an dieser Stelle wie folgt:

1. Generiere den komplementären Graphen  $\overline{G}$ .
2. Nutze Algorithmus 11 auf  $\overline{G}$ . Dies gibt eine Färbung auf  $G$ .
3. Nutze Algorithmus 7 auf  $G$ .

Es kann an dieser Stelle erwartet werden, dass dieser Ansatz eine bessere Laufzeit hat als der direkte Ansatz über eine approximierete Graphenfärbung in Abschnitt 7.4.2.

## 7.5 Document Clustering auf MEDLINE

Da in MEDLINE Informationen über Dokumente im zweistelligen Millionenbereich vorliegen, ist es technisch sehr schwierig, ein Clustering über alle verfügbaren Dokumente zu berechnen. Die Eingabedaten hätten eine nicht mehr überschaubare Größe und könnten auch nicht mit einem perfekten Graphen repräsentiert werden.

Überschaubare Teilprobleme ergeben sich, wenn man nur einzelne Zeitschriften oder eine Gruppe von Zeitschriften betrachtet. Auch so können signifikante Informationen extrahiert werden.

## 7.6 Implementierung und Auswertung

Aufgrund der Größe der Eingabedaten können keine exakten Lösungen angegeben werden, denn es liegen keine polynomiellen Ansätze vor. Der Ansatz durch lineare Programmierung konnte an dieser Stelle aufgrund der Laufzeit nur für kleine Instanzen ausgeführt werden.

Hier wird als Ausgangspunkt eine Näherung durch eine Graphenfärbung gewählt. Um die Lösung nach Möglichkeit weiter zu minimieren werden –

ausgehend von den kleinsten Farbklassen – alle möglichen Knoten aus diesen entfernt. Durch diesen Greedyansatz kann die Lösung weiter verbessert werden und es werden auch die gewünschten Pfade konstruiert.

Für die Implementierung wurde – ähnlich wie in Abschnitt 5.8 – die Sprache Python gewählt. Der objektorientierte Ansatz dieser Sprache vereinfacht die Programmierung komplexer Strukturen, wie sie insbesondere für die pseudostabilen Mengen nötig sind. Für die Programmierung einiger graphentheoretischer Aspekte wurde die freie Bibliothek NetworkX benutzt und für NLP und die Berechnung des TF.IDF-Maß die Bibliotheken NLTK (vgl. für eine genauere Darstellung der Funktion [42, S. 66] bzw. <http://www.nltk.org/>) und sklearn (vgl. <http://scikit-learn.org>).

Gearbeitet wurde mit Python 3.4 unter anderem auf einem Computer mit vier 2.4 GHz Prozessoren, 8 GB RAM und Linux Kernel 3.10. Es stand ein weiterer Rechner mit 48 Kernen und 251 GB RAM zur Verfügung. Das ganzzahlige lineare Programm wurde durch die GLPK (GNU Linear Programming Kit) in der Version 4.52 gelöst.

Die Dokumente mit ihren Metadaten und ihrer Ähnlichkeit wurden mit Hilfe der Software SCAIView im JSON-Format exportiert. Diese Software wird vom Fraunhofer SCAI entwickelt und bietet eine einfache Schnittstelle um Informationen aus dem Bereich Data- und Textmining, insbesondere auf MEDLINE, zu extrahieren<sup>2</sup>.

Die dedizierte Auswertung und Darstellung kann dann mit anderen Anwendungen, z. B. Cytoscape erfolgen.

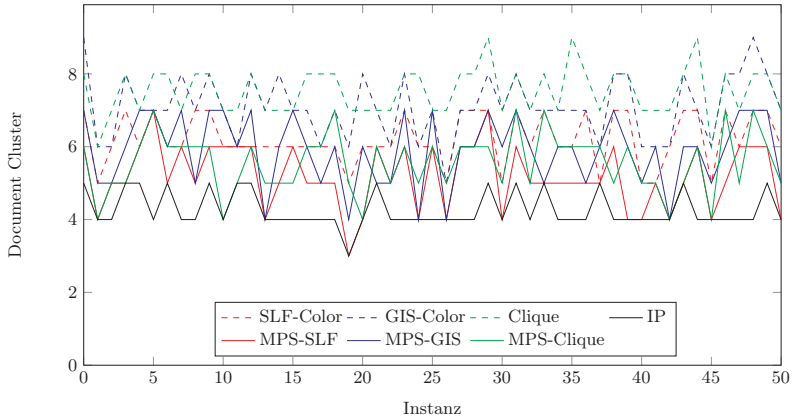
Um den hier vorgestellten Ansatz auszuwerten, sollen zunächst Zufallsinstanzen betrachtet werden. So können die Heuristiken und das lineare Programm zügig auf verschiedenen Instanzen getestet werden. Hier werden lediglich Laufzeiten und Ergebnisse evaluiert.

In Abschnitt 7.6.2 wird das Document Clustering in allen Schritten an der Zeitschrift *Alzheimer's & Dementia: The Journal of the Alzheimer's Association* vollzogen. Auch hier werden noch verschiedene Laufzeiten und Ergebnisse diskutiert.

In einem letzten Abschnitt wird versucht, Literaturergebnissen zu reproduzieren. Dieser Abschnitt folgt damit den Ergebnissen in [21] und macht deutlich, dass es sich bei PS-Document Clustering um eine Methode handelt, die – abhängig vom gewählten Ähnlichkeitsmaß – sinnvolle Ergebnisse liefern kann.

---

<sup>2</sup>vgl. <http://www.scaiview.com>



**Abbildung 7.3:** Ergebnisse verschiedener Heuristiken und Schranken für verschiedene Zufallsinstanzen des DC-Document Clusterings mit  $n = 15$ .

### 7.6.1 Auswertung auf Zufallsinstanzen

Zufallsgraphen werden oft nach dem Modell von Gilbert erstellt (vgl. [33]). Hier ist ein einfacher ungerichteter Graphen  $G = (V, E)$  mit  $(n(n-1))/2$  potentiell möglichen Kanten Teil des Modells  $\mathcal{G}(n, p)$ . Kanten werden mit Wahrscheinlichkeit  $0 < p < 1$  hinzugefügt. Die Kantenwahrscheinlichkeit wird oft abhängig von der Knotenwahrscheinlichkeit gewählt, d. h.  $p = p(n)$ .

Erdős und Rényi haben ein vergleichbares Modell  $\mathcal{G}(n, m)$  entworfen, in dem verschiedene Graphen mit genau  $n$  Knoten und  $0 < m < (n(n-1))/2$  Kanten gleich wahrscheinlich sind (vgl. [23]).

Die Laufzeit dieser Generatoren ist quadratisch. Batagel und Brandes haben für kleine  $p$  allerdings auch einen linearen Zeit von  $O(n + m)$ , wobei  $m$  die Anzahl der generierten Kanten ist, vorgestellt (vgl. [5]).

Es soll im folgenden  $p = 0,75$  gewählt werden. Mit einer weiteren Wahrscheinlichkeit von  $p' = 0,2$  werden Kanten dann blau gefärbt. Dies entspricht grob den in den folgenden realen Beispielen ermittelten Werten.

Die Ergebnisse für Instanzen mit  $n = 100$  Knoten finden sich in Abb. 7.3. Die Laufzeiten finden sich in Tabelle 7.1.



Es fällt auf, dass der Clique-Ansatz aus Algorithmus 11 die schlechteste Zerlegung in stabile Mengen liefert, aber der Greedy-Ansatz darauf die meisten Knoten in  $G$  eliminieren kann. Der SLF-Ansatz aus Algorithmus 10 liefert bessere Ergebnisse als der GIS-Ansatz, zusätzlich ist er deutlich schneller. GIS-Coloring hat eine deutlich schlechtere maximale Laufzeit als alle anderen Ansätze. Im Schnitt ist er ungefähr gleich schnell wie der Clique-Ansatz.

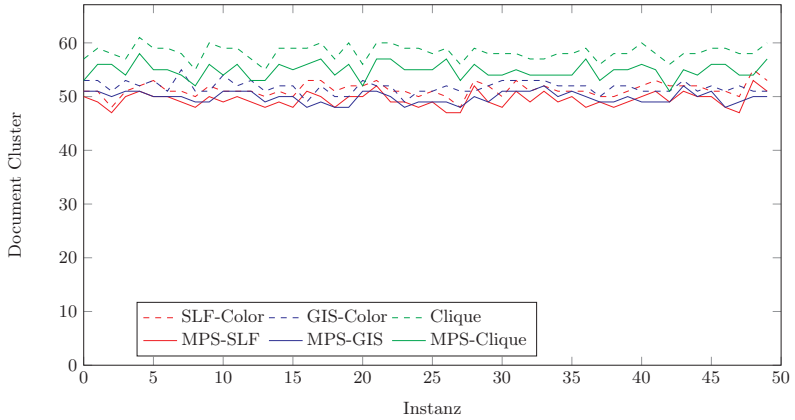
Noch deutlicher wird dieses Verhalten mit einer größeren Knotenzahl  $n = 200$ . Hier liefert die SLF-Strategie noch durchschnittlich bessere Ergebnisse bei besserer durchschnittlicher Laufzeit. Die maximale Laufzeit ist allerdings schon größer als bei der GIS-Strategie. Sowohl von der Laufzeit als auch von den Ergebnissen abgeschlagen ist die Clique-Heuristik. Vgl. hierzu auch Abb. 7.4.

Das die Clique-Strategie nicht optimal ist, wird bei Instanzen der Größe  $n = 500$  noch deutlicher. Dies wird in Abb. 7.5 illustriert. Interessant ist, dass bei diesen großen Instanzen die GIS-Strategie bessere Ergebnisse als die SLF-Strategie liefert. Auch ist ihre durchschnittliche Laufzeit deutlich kürzer geworden.

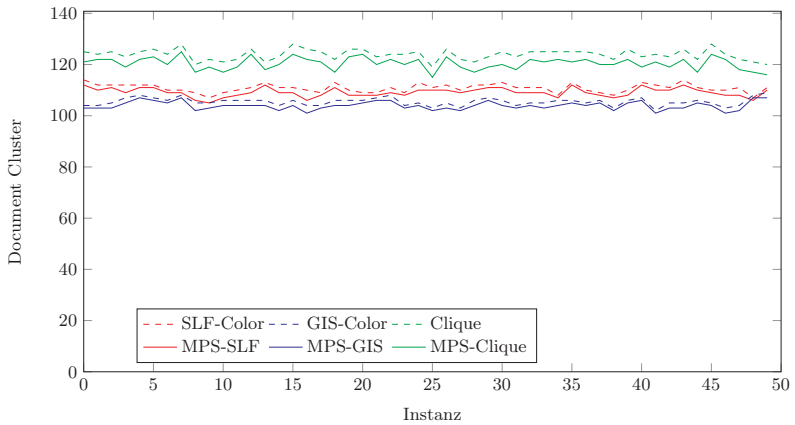
Es zeigt sich also, dass die Wahl der Näherungslösung auf der die Greedy-Heuristik arbeitet maßgeblich zur Lösung des Problems beiträgt. So soll im Folgenden eine einfache Färbeheuristik als Grundlage für den Greedy-Ansatz genutzt werden.

	SLF-Coloring	GIS-Coloring	Clique	
$\emptyset$	0,1245	0,1314	0,1325	n=100
max	0,1910	0,4492	0,1739	
min	0,0992	0,1074	0,1239	
$\emptyset$	0,3844	0,3868	0,4254	n=200
max	0,4390	0,4071	0,4888	
min	0,3474	0,3474	0,3806	
$\emptyset$	2,5415	2,3856	2,8392	n=500
max	2,8039	2,5964	3,1028	
min	2,3373	2,1965	2,4778	

**Tabelle 7.1:** Die Laufzeiten für die Algorithmen SLF-Coloring, GIS-COLORING und Clique für verschiedene Instanzen mit  $n = 100$ ,  $n = 200$  und  $n = 500$  in Sekunden.



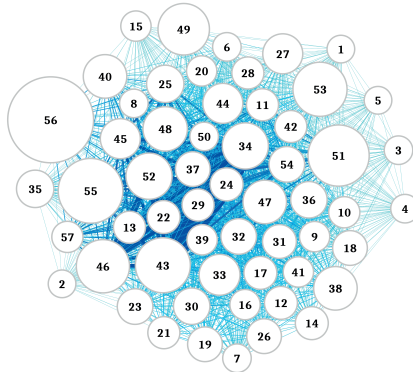
**Abbildung 7.4:** Ergebnisse verschiedener Heuristiken und Schranken für verschiedene Zufallsinstanzen des DC-Document Clusterings mit  $n = 200$ .



**Abbildung 7.5:** Ergebnisse verschiedener Heuristiken und Schranken für verschiedene Zufallsinstanzen des DC-Document Clusterings mit  $n = 500$ .

### 7.6.2 Auswertung eines einzelnen Journals

An dieser Stelle wird zunächst eine Auswertung der Zeitschrift *Alzheimer's & Dementia: The Journal of the Alzheimer's Association* dargestellt. Ent-



**Abbildung 7.6:** Die Gesamtzerlegung der Artikel aus der Zeitschrift *Alzheimer's & Dementia: The Journal of the Alzheimer's Association*. Die Nummern identifizieren die einzelnen Cluster. Die Größe der Knoten entspricht der Anzahl der Dokumente in dem jeweiligen Cluster. Die Kanten entsprechen in ihrer Breite und ihrem Farbton dem Gewicht dieser Kante. Die Darstellung erfolgte mit Hilfe des Algorithmus von Fruchterman und Reingold.

sprechend mit MESH-Termen versehene Artikel finden sich in MEDLINE ab dem Jahr 2008 und vollständig ab dem Jahr 2009.

Zunächst soll dargestellt werden, wie verschiedene Werte für  $\epsilon$  und  $\iota$  den Graphen  $G$  und  $B \subset G$  beeinflussen. In der folgenden Tabelle findet sich eine Übersicht. Der Graph  $G$  hat stets  $n = 715$  Knoten. Die Kantenanzahl  $n$  des Graphen  $G$  und die Kantenanzahl  $b$  des Graphen  $B$  ist entsprechend angegeben.

$\iota$	0,04			0,05			0,06		
$\epsilon$	0,05			0,07			0,07		
	e=112.325, b=27.828			e=144.078, b=31.753			e=155.711, b=25.782		
	Farben	Greedy	Laufzeit	Farben	Greedy	Laufzeit	Farben	Greedy	Laufzeit
SLF	29	27	1,3s	45	43	2,6s	55	51	3,7s
GIS	33	31	1,8s	55	49	2,3s	67	58	3,7s
Clique	55	52	2,8s	81	76	7,8s	105	99	11,1s

Interessant sind die verschiedenen Laufzeiten der Heuristiken und ihre Ergebnisse. In diesem Fall ist die GIS-Strategie nur bei kleinen Werten langsamer, je mehr Kanten es gibt, desto langsamer wird im Gegenzug die SLF-Strategie. Die Clique-Strategie ist sowohl von der Laufzeit als auch vom Ergebnis her weit abgeschlagen. Aufgrund der Laufzeit und des Ergebnisses wird an dieser Stelle das Resultat der GIS-Strategie mit den Werten  $\iota = 0,6$  und  $\epsilon = 0,7$  dargestellt.

In Abbildung 7.6 wird das Ergebnis der Zerlegung in mehrfach pseudo-stabile Mengen dargestellt. Es lässt sich gut erkennen, dass es keine völlig isolierten Themengebiete gibt. Trotzdem sind einige Cluster relativ isoliert. So beschäftigt sich Cluster 5 primär mit „Periodicals as Topic“, Cluster 3 mit „Patient Acceptance of Health Care“ und Cluster 4 mit „Tau Proteins/Genetics“. Mit 67 Artikeln ist Cluster 56 der umfangreichste. Er befasst sich Thematisch vor allem mit „Cohort Studies“. Cluster 55 behandelt mit 43 Artikeln vor allem das Thema „Neuropsychological Tests“, Cluster 53 mit 31 Artikeln vor allem „Disease Progression“, Cluster 51 mit 39 Artikeln das Thema „Risk Factors“, Cluster 43 vor allem das Thema „Amyloid beta-Peptides/cerebrospinal fluid“.

### 7.6.3 Reproduktion von Literaturergebnissen

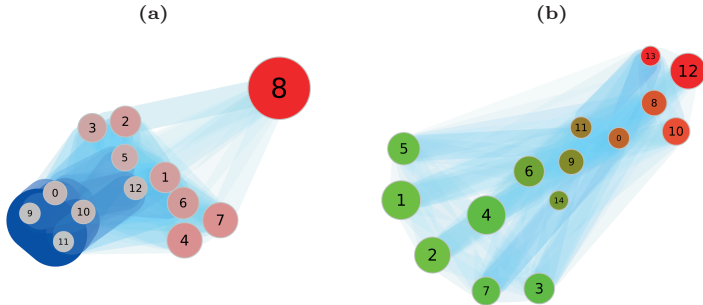
Um abschließend zu zeigen, dass mittels der hier eingeführten Methodik auch andere Ergebnisse reproduziert werden können, soll an dieser Stelle ein einfaches Beispiel diskutiert werden.

In der Arbeit von [49] wurden zwei Dokumentenmengen diskutiert, die mittels Suchanfrage in MEDLINE gefunden wurden. Das Ergebniss wurde ebenfalls von [81] mit einer neuen Methodik untersucht. Beide Publikationen nutzen eine Dokumentenmenge von 1660 Dokumenten die mittels der Suchanfragen „escherichia AND pili“ sowie „cerevisiae AND cdc\*“ mit jeweils 830 Dokumenten gefunden wurden. Leider war es nicht möglich, die exakte Dokumentenmenge zu finden, allerdings konnte mit einer Einschränkung der Suche auf alle Dokumente bis zum Ende des Jahres 2001 insgesamt 1628 Dokumente gefunden werden. Die erste Suchanfrage bezieht sich auf das Darmbakterium Escherichia coli, während die Zweite Hefe betrifft.

Also gibt es  $n = 1628$  Knoten bzw. Dokumente im Graphen. Die Anzahl der Kanten hängt von der Wahl von  $\iota$  und  $\epsilon$  sowie der Wahl der Ähnlichkeitsfunktion ab. Es sollen zwei Funktionen evaluiert werden. Die erste Funktion nutzt ein Distanzmodell  $d_V$  auf der Zusammenfassung des Textes, das auf Vektorraum-Retrieval basiert. Da gleiche Zeitschriften ebenfalls bevorzugt werden sollten, wurde  $d_J(x, y) = \{0, 1\}$  gesetzt.  $d_J(x, y) = 1$ , wenn  $x$  und  $y$  in der selben Zeitschrift publiziert wurden. Damit ist

$$d_1(x, y) = \frac{d_V(x, y) + d_J(x, y)}{2}$$

Für den zweiten Ansatz soll nur  $d_2 = d_V$  gesetzt werden.



**Abbildung 7.7:** (a) Die Zerlegung der Dokumentenmenge mit Ähnlichkeitsmaß  $d_1$ , dargestellt anhand des gewichteten Graphen des Clusterings. Die Zahlen identifizieren die Cluster. Die Größe eines Knotens steht in Relation zur Anzahl an Dokumenten in dem Cluster. Die Breite und Farbe der Kanten sind relativ zu ihrem Gewicht gesetzt. (b) Die Zerlegung der Dokumentenmenge mit Ähnlichkeitsmaß  $d_2$ , dargestellt anhand des gewichteten Graphen des Clusterings. Hier entspricht die Knotenfarbe der Anzahl an Dokumenten aus einer Suchanfrage (grün und rot).

Für  $d_1$  ergibt sich das Bild, dass in Abb. 7.7 als Graph der Zerlegung dargestellt wird. Es wurden 13 Cluster (nummeriert von 0 bis 12) berechnet. Dabei ist Cluster 8 mit 359 Dokumenten der größte Cluster, wohingegen Cluster 11 nur 5 Dokumente enthält. Da eine Zerlegung in zwei Suchanfragen gefunden werden muss, ist die Frage, ob thematisch ähnliche Cluster auch durch stark gewichtete Kanten zusammenfallen. Dies trifft zu, thematisch bildet Cluster 8 („Escherichia coli“, „Fimbriae“, etc.) eine Suchanfrage ab, die Clusters 0, 9, 10, 1 („Saccharomyces cerevisiae“, „Fungal proteins“) eine andere. Damit konnten die Ergebnisse von [81] reproduziert werden, da alle anderen Cluster mit beiden Suchanfragen zusammenhängen. Cluster 7 ist beispielsweise mit den Schlagworten „Molecular Sequence Data“ und „Escherichia coli“ verbunden.

Für  $d_2$  konnten die Ergebnisse aus [81] wiederum nicht vollständig reproduziert werden, vgl. Abb. 7.7. Es wurden 14 Cluster berechnet, die zwischen 2 und 158 Dokumente beinhalten. Da die Schlagworte zu den einzelnen Clustern nicht einfach ermittelt werden konnten, wurden die Knoten entsprechend des Anteils von Dokumenten aus einer Suchanfrage eingefärbt. Dabei fällt auf, dass einige Cluster eine geringe Entropie aufweisen, zum Beispiel Cluster 1, 5, 2, 7 und 3. Diese sind wiederum nicht mit stark gewichteten Kanten miteinander verbunden. Damit ergibt sich kein klares

Bild.

Da nur mit einem Ähnlichkeitsmaß die Ergebnisse aus [81] reproduziert werden konnten, ist die Frage zu stellen, ob der vorgestellte Ansatz valide ist. [18] weisen darauf hin, dass eine Preprozessierung und die Wahl der Ähnlichkeitsfunktion eine entscheidende Rolle für das Ergebnis spielen: „such decisions have profound effects on the results of real models for real data“ [18, S. 1]. Dies wurde auch in der Arbeit [21] herausgearbeitet. Ähnlich ist die Problematik versucht man zu evaluieren welches Thema ein Cluster eigentlich beschreibt. [34] etwa argumentieren, dass es kein grundsätzliches Einvernehmen in der Forschung gibt, wie so etwas zu messen oder zu bewerten sei. Klavans und Boyack fordern in [57] beispielsweise eine objektive und zuverlässige Methode um Clusterthemen zu messen. Allerdings setzen auch sie in weiteren Arbeiten ihren Schwerpunkt auf Ähnlichkeitsmaße (vgl. [12]). Damit wäre man bei der Frage von Gläser et al. in ihrer Arbeit „Same data – different results?“ angelangt: „Do we produce more than artefacts?“ [34]. Da diese Frage noch immer Gegenstand der Forschung ist, muss dem Konsens gefolgt werden: Ist die Methode für einen bestimmten, angepassten Fall valide, so wird für alle anderen Fälle ebenfalls angenommen, dass der Ansatz valide ist.

An dieser Stelle kann das Ergebnis also nur sein, dass der in dieser Arbeit vorgestellte Ansatz valide ist. Er hat für ein bestimmtes Ähnlichkeitsmaß die erwartete Ausgabe produziert. Die Frage, welches Ergebnis  $d_2$  liefert, muss unbeantwortet bleiben. Es kann angenommen werden, dass verschiedene Ähnlichkeitsmaße jeweils einen anderen subjektiven Blick auf die Dokumentenmenge liefern.

## 7.7 Zusammenfassung

In diesem Kapitel wurde ein neuer Ansatz für das Document Clustering vorgestellt. Er basiert auf einer Zerlegung des Dokumentgraphen in mehrfach pseudostabile Mengen. Dieses Problem ist  $\mathcal{NP}$ -vollständig. Es wurde sowohl ein Ansatz durch lineare Programmierung als auch verschiedene Greedy-Ansätze vorgestellt.

Dieser neue Ansatz für das Document Clustering durch mehrfach pseudostabile Mengen liefert ein soft-Clustering. Durch die nun verfügbaren Zusatzinformationen in den Kanten zwischen den Clustern, können diese ebenfalls nach ihrer Ähnlichkeit untereinander sortiert werden. Es wurde bei der Analyse von realen Suchanfragen auf MEDLINE evaluiert, ob

diese auch durch das Clustering separiert werden. Dabei stellte sich heraus, dass das Ergebnis im Wesentlichen vom gewählten Ähnlichkeitsmaß abhängt, die Methode des PS-Document Clusterings aber valide ist.





## Fazit und Ausblick

In dieser Arbeit wurden pseudostabile Mengen definiert und die theoretischen Grundlagen hergeleitet. Sie sind eine Verallgemeinerung von stabilen Mengen, die auch Knoten außerhalb der stabilen Mengen unter bestimmten Voraussetzungen zulassen. Dadurch entstehen Pfade zwischen den stabilen Mengen. Das komplementäre Problem sind Pseudocliquen.

Es wurde untersucht, wie die Strukturen eines Graphen  $G$  und der Kandidaten für die Pfade  $B \subset G$  bei einer Zerlegung in pseudostabile Mengen aussieht. Hier konnten bereits grundlegende Einschränkungen gemacht werden. Auch wurde der Zusammenhang zwischen stabilen und pseudostabilen Mengen untersucht.

Sowohl im Hinblick auf pseudostabile Mengen als auch mehrfach pseudostabilen Mengen wurde das Problem betrachtet, einen Graphen in eine minimale Überdeckung mit diesen Mengen zu zerlegen. Es wurde gezeigt, dass alle vorgestellten Minimierungsprobleme in ihrer azyklischen und zyklischen Variante  $\mathcal{NP}$ -vollständig sind. Es konnten für minPS allerdings auch hinreichende Bedingungen angegeben werden, unter welchen das Problem in P liegt. Es wurden verschiedene lineare Programme hergeleitet um das Problem zu lösen. Das komplementäre Problem, die Zerlegung des Graphen  $\overline{G}$  in Pseudocliquen (minPC bzw. minMPC), wurde ebenfalls untersucht und floss in die Problemlösung für minMPS'-a ein.

Sowohl für minPS als auch minMPS'-a wurden Praxisprobleme untersucht und Heuristiken und Schranken evaluiert. Das  $\mathcal{NP}$ -vollständige TMP entspricht dem Problem minPS. Es wurde unter Berücksichtigung der Ergebnisse von Dahlhaus, Horak, Miller und Ryan [16] und Beygang [8] bearbeitet.

Zunächst wurde die Struktur von trennbaren Zielen untersucht und durch diese Resultate gezeigt, dass das TMP äquivalent zu einer Zerlegung des zum Problem gehörenden Intervallgraphen in pseudostabile Mengen ist.

Mit Hilfe dieser neuen Formulierung konnten die Schranken und Heuristiken aus der Arbeit von Beygang wesentlich verbessert werden. Außerdem wurden neue polynomielle Algorithmen hergeleitet, die eine approximative Lösung des TMP berechnen und die Struktur pseudostabiler Mengen berücksichtigt. Eine weitere Heuristik konnte in ihrer Laufzeit optimiert werden. Sie erzielten fast immer bessere Resultate als die bekannten Algorithmen.

Trotzdem geben die Abweichungen Anlass für weitere Untersuchungen. Die Auswertungen werfen die Frage auf, ob es Kriterien gibt, unter denen trennbare Ziele nicht zu einer Minimierung der Gleiszahl beitragen. Diese Resultate könnten zu einer weiteren Optimierung der unteren Schranke beitragen.

Es bleibt auch die Frage offen, wie sich pseudostabile Mengen in die Theorie anderer Rangierprobleme einfügen oder sogar zu ihnen äquivalent sind. Es lässt sich vermuten, dass insbesondere für dem TMP ähnliche Probleme auch ähnliche Strukturen auftreten.

Mit Hilfe der Zerlegung eines Graphen in mehrfach pseudostabile Mengen wurde ein neuartiger Ansatz für das *Document Clustering* hergeleitet. Dieser ist innovativ, da er im Gegensatz zu einem Clustering mit stabilen Mengen bzw. Cliques ein *soft Clustering* generiert und äußerst nützliche Zusatzinformationen bereitstellt. Stabile Mengen liefert ein *hard Clustering*. Mehrfach pseudostabile Mengen erlauben Dokumenten auch „zwischen“ zwei Clustern zu liegen und generieren deswegen ein *soft Clustering*. Dadurch kann ein darstellender Graph generiert werden, der Kanten zwischen zwei Clustern darstellt. Sie enthalten die Information, wie viele Dokumente thematisch in beide Cluster gepasst hätten.

Dazu wurde neben einem Linearen Programm auch ein Greedy-Ansatz entworfen. Verschiedene Heuristiken zur Graphenfärbung bzw. zur Zerlegung von  $\bar{G}$  in Cliques lieferten die Grundlage dafür. Dabei liefern diese auch verschiedene Ergebnisse. Diese wurden sowohl auf Zufallsinstanzen als auch auf realen Instanzen diskutiert.

Es konnte dadurch gezeigt werden, dass ein nicht unerheblicher Anteil der kleinen Document Cluster aufgelöst werden konnte und viele kleine Cluster eine große Schnittmenge haben. Dieser innovative Ansatz gibt also zusätzlich Informationen über die Abhängigkeiten der einzelnen Document Cluster untereinander.

Auch hier bleiben allerdings offene Fragen: Können Schranken gefunden werden, die sinnvoll auf dieses komplexe Problem angewendet werden können? Gibt es Nebenbedingungen, die ein sinnvolles Clustering in polyno-

mieller Zeit ermöglichen? zusätzlich stellt sich die Frage, wie eine Dokumentenähnlichkeit sinnvoll berechnet werden kann.

Mit dieser Arbeit ist also nicht nur die theoretische Grundlage für die Arbeit mit pseudostabilen Mengen und Pseudocliquen gelegt, sondern auch eine Anwendung anhand zweier Probleme mittels verschiedener Ansätze diskutiert.



# Abbildungsverzeichnis

3.1	Darstellung pseudostabiler Tupel und mehrfach pseudostabiler Tupel. . . . .	17
3.2	Beispielzerlegung in pseudostabile Mengen. . . . .	22
3.3	Beispielzerlegung in mehrfach pseudostabile Mengen. . . . .	22
3.4	Illustration zu Lemma 3.13. . . . .	23
3.5	Illustration zu Lemma 3.13. . . . .	23
3.6	Illustration zu Lemma 3.13. . . . .	24
3.7	Illustration zu Lemma 3.14. . . . .	25
3.8	Darstellung zur Notation pseudostabiler Mengen. . . . .	27
3.9	Skizze zum Beweis von Lemma 3.18. . . . .	30
3.10	Skizze zum Beweis von Lemma 3.23. . . . .	33
3.11	Skizzen zu den Beispielen 3.1 und 3.2. . . . .	34
3.12	Skizze zu Beispiel 3.3. . . . .	39
4.1	Skizzen zu Beispiel 4.2. . . . .	46
4.2	Darstellung der Graphen aus Beispiel 4.3. . . . .	46
4.3	Skizze eines großen Diamanten. . . . .	58
5.1	Gleisplanschema Rbf Zürich-Limmattal. . . . .	67
5.2	Schematische Darstellung eines Rangierbahnhofes. . . . .	68
5.3	Abbildung zu Beispiel 5.1. . . . .	70
5.4	Darstellung der Gleisbelegung aus Beispiel 5.3. . . . .	71
5.5	Darstellung möglicher Rangierbewegungen. . . . .	72
5.6	Graphische Darstellung zu Beispiel 5.4. . . . .	75
5.7	Graphische Darstellung zu Beispiel 5.5. . . . .	76
5.8	Graphische Darstellung zu Beispiel 5.7. . . . .	78
5.9	Graphische Darstellung zu Beispiel 5.8. . . . .	85
5.10	Graphische Darstellung zu Lemma 5.22. . . . .	86
5.11	Graphische Darstellung zu Lemma 5.22. . . . .	87
5.12	Graphische Darstellung zu Lemma 5.22. . . . .	87
5.13	Ziele in Zufallsinstanzen der Größe $n = 200$ , $n = 150$ und $n = 100$ . . . . .	104
5.14	Ergebnisse verschiedener Heuristiken und Schranken für verschiedene Instanzen des TMP mit $n = 100$ . . . . .	105

5.15	Ergebnisse verschiedener Heuristiken und Schranken für verschiedene Instanzen des TMP mit $n = 150$ . . . . .	106
5.16	Ergebnisse verschiedener Heuristiken und Schranken für verschiedene Instanzen des TMP mit $n = 200$ . . . . .	106
5.17	Ergebnisse verschiedener Heuristiken und Schranken für verschiedene Instanzen des TMP mit $n = 300$ . . . . .	107
6.1	Graph zu Beispiel 6.1. . . . .	110
6.2	Graph zu Beispiel 6.3. . . . .	115
6.3	Graph zu Beispiel 6.4. . . . .	119
6.4	Graph zu Beispiel 6.5. . . . .	121
6.5	Graph zu Beispiel 6.5. . . . .	121
7.1	Schematische Darstellung von <i>Text Classification</i> und <i>Text Clustering</i> . . . . .	128
7.2	Darstellung zu Beispiel 7.1. . . . .	134
7.3	Ergebnisse verschiedener Heuristiken und Schranken für verschiedene Zufallsinstanzen des DC-Document Clusterings mit $n = 15$ . . . . .	144
7.4	Ergebnisse verschiedener Heuristiken und Schranken für verschiedene Zufallsinstanzen des DC-Document Clusterings mit $n = 200$ . . . . .	146
7.5	Ergebnisse verschiedener Heuristiken und Schranken für verschiedene Zufallsinstanzen des DC-Document Clusterings mit $n = 500$ . . . . .	146
7.6	Gesamtzerlegung der Artikel aus der Zeitschrift <i>Alzheimer's &amp; Dementia: The Journal of the Alzheimer's Association</i> . . . . .	147
7.7	Zerlegung einer Dokumentenmenge mit zwei Ähnlichkeitsmaßen $d_1$ und $d_2$ . . . . .	149

# Glossar

**Blauer Graph**  $G^B$ : Ein Graph  $G = (V, E)$  mit einem Teilgraphen  $G' = B \subset G$  von blauen Knoten und Kanten. Siehe S. 15

**COLORING-PS** Heuristik zur Lösung von minPS. Berechnet die Schranken  $u_{col}$  und  $l_{col}$ . Siehe S. 100, 104, 105, 107, 108

**Document Clustering** Definition siehe S. 125, 130

**Dynamisch gewichtetes Mengenüberdeckungsproblem** DGMÜP, Definition siehe S. 47

**Gewichteter Graph des Clusterings** Definition siehe S. 135

**Gewichtetes Mengenüberdeckungsproblem** MÜP, Definition siehe S. 13

**GIS-COLORING** Graphenfärbeheuristik mittels der Strategie des *greedy independent set*. Siehe S. 139

**Graph einer zulässigen Zerlegung in mehrfach pseudostabile Mengen** Siehe S. 110

**Graph einer zulässigen Zerlegung in pseudostabile Mengen** Definition siehe S. 44

**Graph einer Überdeckung** Definition siehe S. 19

**GREEDY-COLORING** Graphenfärbeheuristik mittels Greedy-Strategie. Siehe S. 139

**GREEDY-DC** Greedy Algorithmus zur Lösung des PS-Document Clustering Problems. Siehe S. 138

**GREEDY-PS** Heuristik zur Lösung von minPS. Siehe S. 95, 104, 105, 107, 108

**Großer Diamant** Definition siehe S. 58

**Jaccard Distanz** Siehe S. 132

**K-Document Clustering** Definition siehe S. 129

$l_B(G)$  Untere Schranke für minPS und minMPS. Siehe S. 57, 118

$l_C(G)$  Untere Schranke für minPS und minMPS durch die Cliquenzahl. Siehe S. 54, 113

$l_{col}$  Untere Schranke für das TMP. Siehe S. 102, 104

$l_p$  Untere Schranke durch das ganzzahlige lineare Programm zur Lösung für minPS. Siehe S. 63, 105

$l_{ps}$  Untere Schranke für minPS und das TMP. Siehe S. 93, 94, 105

$l_R(G)$  Untere Schranke für minPS und minMPS. Siehe S. 56, 116, 117

$l_{RB}(G)$  Untere Schranke für minPS, siehe S. 57, 118

**Mehrfach pseudostabile Menge** Definition siehe S. 21

**Mehrfache Pseudoclique** Definition siehe S. 122

**Mehrfach pseudostabiles Tupel** Definition siehe S. 16

**Mengenüberdeckungsproblem** MÜP, Definition siehe S. 12

**minMPS** Definition siehe S. 20, 109

**minMPS-a-IP** IP zur Lösung von PS-Document Clustering. Siehe S. 136

**minMPS-a** Definition siehe S. 20

**minPS** Definition siehe S. 19, 43

**minPS-IP** IP zur Lösung von minPS. Siehe S. 62

**minPS-a** Definition siehe S. 20

**minPS' und minMPS'** Definition siehe S. 20

**PS-Document Clustering** Definition siehe S. 135

**Pseudostabile Menge** Definition siehe S. 21

**Pseudostabiles Tupel** Definition siehe S. 16

**SLF-COLORING** Graphenfärbheuristik mittels der Strategie des *saturation largest-first*. Siehe S. 140

**SRSP** *Sorting of rolling Stock Problems*, siehe S. 69

**TF.IDF-Maß** *Term frequency-inverse document frequency*, siehe S. 131

**TMP** *Train Marshalling Problem*, siehe S. 73

**TSENG CLIQUE-PARTITIONING ALGORITHM** Cliquenüberdeckungsheuristik. Siehe S. 141

$u_{col}$  Siehe S. 102, 104, 105

$u_{greedy}$  Siehe S. 104, 105

$u_p$  Obere Schranke durch das ganzzahlige lineare Programm zur Lösung für minPS. Siehe S. 63, 105

**Vektorraum-Retrieval** Siehe S. 131



# Literaturverzeichnis

- [1] ADLBRECHT, J.-A. ; HÜTTLER, B. ; ZAZGORNIK, J. ; GRONALT, M.: The train marshalling by a single shunting engine problem. In: *Transportation Research Part C: Emerging Technologies* 58 (2015), S. 56–72
- [2] AGGARWAL, C. C. ; REDDY, C. K.: *Data clustering: algorithms and applications*. CRC press, 2013
- [3] AGGARWAL, C. C. ; ZHAI, C.: *Mining text data*. Springer Science & Business Media, 2012
- [4] BAEZA-YATES, R. ; RIBEIRO-NETO, B. u. a.: *Modern information retrieval*. Bd. 463. ACM press New York, 1999
- [5] BATAGELJ, V. ; BRANDES, U.: Efficient generation of large random networks. In: *Phys. Rev. E* 71 (2005)
- [6] BEEL, J. ; GIPP, B. ; LANGER, S. ; BREITINGER, C.: paper recommender systems: a literature survey. In: *International Journal on Digital Libraries* 17 (2016), Nr. 4, S. 305–338
- [7] BENATI, S. ; PUERTO, J. ; RODRÍGUEZ-CHÍA, A. M.: Clustering data that are graph connected. In: *European Journal of Operational Research* (2017)
- [8] BEYGANG, K.: *On the Solution of Some Railway Freight Car optimization Problems*. München : Dr. Hut, 2011
- [9] BEYGANG, K. ; KRUMKE, S. O. ; ZECK, C.: Train Marshalling Problem - Algorithms and Bounds - / University of Kaiserslautern. 2010 (Report in Wirtschaftsmathematik 132). – Forschungsbericht
- [10] BHASKER, J. ; SAMAD, T.: The clique-partitioning problem. In: *Computers & Mathematics with Applications* 22 (1991), Nr. 6, S. 1–11
- [11] BLUM, N.: *Einführung in Formale Sprachen, Berechenbarkeit, Informations- und Lerntheorie*. München : Oldenbourg Wissenschaftsverlag, 2007

- [12] BOYACK, K. W. ; KLAVANS, R.: Co-citation analysis, bibliographic coupling, and direct citation: Which citation approach represents the research front most accurately? In: *Journal of the American Society for Information Science and Technology* 61 (2010), Nr. 12, S. 2389–2404
- [13] BRÉLAZ, D.: New Methods to Color the Vertices of a Graph. In: *Commun. ACM* 22 (1979), April, Nr. 4, S. 251–256
- [14] BRUEGGEMANN, L. ; FELLOWS, M. ; FLEISCHER, R. ; LACKNER, M. ; KOMUSIEWICZ, C. ; KOUTIS, Y. ; PFANDLER, A. ; ROSAMOND, F.: Train Marshalling is Fixed Parameter Tractable. (2012)
- [15] CHVÁTAL, V.: *Linear programming*. New York : Freeman, 1983
- [16] DAHLHAUS, E. ; HORAK, P. ; MILLER, M. ; RYAN, J.: The train marshalling problem. In: *Discrete Applied Mathematics* 103(1-3) (2000), S. 41–54
- [17] DAHLHAUS, E. ; MANNE, F. ; MILLER, M. ; RYAN, J.: Algorithms for combinatorial problems related to train marshalling. In: *Proceedings of AWOCA 2000, Hunter Valley* (2000), S. 7–16
- [18] DENNY, M. J. ; SPIRLING, A.: Text preprocessing for unsupervised learning: why it matters, when it misleads, and what to do about it. In: *Available at SSRN: <https://ssrn.com/abstract=2849145>* (2017)
- [19] DI STEFANO, G. ; MAUE, J. ; MODELSKI, M. ; NAVARRA, A. ; NUNKESSER, M. ; BROEK, J. van d.: Models for Rearranging Train Cars / ARRIVAL Project. 2007 (ARRIVAL-TR-0089). – Forschungsbericht
- [20] DIESTEL, R.: *Graphentheorie*. 3. Auflage. Berlin : Springer-Verlag, 2006
- [21] DÖRPINGHAUS, J. ; SCHAAF, S. ; FLUCK, J. ; JACOBS, M.: Document Clustering using a Graph Covering with Pseudostable Sets. In: GANZHA, M. (Hrsg.) ; MACIASZEK, L. (Hrsg.) ; PAPRZYCKI, M. (Hrsg.): *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems* Bd. 11, IEEE, 2017 (Annals of Computer Science and Information Systems), 329–338
- [22] EDACHERY, J. ; SEN, A. ; BRANDENBURG, F. J.: Graph Clustering Using Distance-k Cliques. In: *Graph Drawing: 7th International Symposium, GD'99 Štířín Castle, Czech Republic September 15–19, 1999 Proceedings* (1999), S. 98–106

- [23] ERDÖS, P. ; RÉNYI, A.: On random graphs, I. In: *Publicationes Mathematicae (Debrecen)* 6 (1959), S. 290–297
- [24] FAIGLE, U. ; KERN, W. ; STILL, G.: *Kluwer Texts in the Mathematical Sciences*. Bd. 24: *Algorithmic principles of mathematical programming*. Dordrecht : Kluwer Academic Publishers, 2002
- [25] FALL, C. ; TÖRCSVÁRI, A. ; FIÉVET, P. ; KARETKA, G.: Automated categorization of German-language patent documents. In: *Expert Systems with Applications* 26 (2004), Nr. 2, S. 269 – 277
- [26] FEDER, T. ; MOTWANI, R.: Clique Partitions, Graph Compression and Speeding-Up Algorithms. In: *Journal of Computer and System Sciences* 51 (1995), Nr. 2, S. 261 – 272
- [27] FELDMAN, R. ; SANGER, J.: *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge University Press, 2007
- [28] FRANÇA, F. ; SOUZA, A. de: *Intelligent Text Categorization and Clustering*. Springer Berlin Heidelberg, 2008 (Studies in Computational Intelligence)
- [29] GAREY, M. ; JOHNSON, D.: *Computers and Intractability: A Guide to the Theory of NP-completeness*. W. H. Freeman, 1979
- [30] GAREY, M. R. ; JOHNSON, D. S.: *Computers and intractability: A guide to the theory of NP-completeness*. New York : Freeman, 1979
- [31] GATTO, M. ; MAUE, J. ; MIHALÁK, M. ; WIDMAYER, P.: Shunting for Dummies: An Introductory Algorithmic Survey. In: AHUJA, R. K. (Hrsg.) ; MÖHRING, R. H. (Hrsg.) ; ZAROLIAGIS, C. D. (Hrsg.): *Robust and Online Large-Scale Optimization*. Berlin, Heidelberg : Springer-Verlag, 2009, S. 310–337
- [32] GIGER, M.: Rangierbahnhof Limmattal. In: *SWISS ENGINEERING STZ AUTOMATE NOW!* 1-2 (2010), S. 93–94
- [33] GILBERT, E. N.: Random graphs. In: *The Annals of Mathematical Statistics* 30 (1959), Nr. 4, S. 1141–1144
- [34] GLÄSER, J. ; GLÄNZEL, W. ; SCHARNHORST, A.: Same data—different results? Towards a comparative approach to the identification of thematic structures in science. In: *Scientometrics* (2017), S. 1–18
- [35] GOLUMBIC, M. C.: *Algorithmic Graph Theory and Perfect Graphs*. Second edition. Amsterdam : Elsevier, 2004

- [36] GRAMM, J. ; GUO, J. ; HÜFFNER, F. ; NIEDERMEIER, R.: Data Reduction and Exact Algorithms for Clique Cover. In: *J. Exp. Algorithmics* 13 (2009), Februar, S. 2:2.2–2:2.15
- [37] GUPTA, U. ; LEE, D. ; LEUNG, J.-T.: An Optimal Solution for the Channel-Assignment Problem. In: *IEEE Transactions on Computer C-28*(11) (1979), S. 807–810
- [38] GUPTA, U. ; LEE, D. ; LEUNG, J.-T.: Efficient Algorithms for Interval graphs and Circular-Arc Graphs. In: *Networks* 12(4) (1982), S. 459–467
- [39] HAAHR, J. T. ; LUSBY, R. M.: A matheuristic approach to integrate humping and pullout sequencing operations at railroad hump yards. In: *Networks* 67 (2016), Nr. 2, S. 126–138
- [40] HALLDÓRSSON, M. M.: A still better performance guarantee for approximate graph coloring. In: *Information Processing Letters* 45 (1993), Nr. 1, S. 19 – 23
- [41] HANSMANN, R. S.: *Optimal Sorting of Rolling Stock*. Göttingen : Cuvillier, 2011
- [42] HARDENIYA, N.: *NLTK Essentials*. Packt Publishing, 2015
- [43] HARTUV, E. ; SHAMIR, R.: A clustering algorithm based on graph connectivity. In: *Information Processing Letters* 76 (2000), Nr. 4–6, S. 175 – 181
- [44] HILLER, W.: *Rangierbahnhöfe*. Berlin : VEB Verlag für Verkehrswesen, 1983
- [45] HIRSCH, L. ; DI NUOVO, A.: Document clustering with evolved search queries. (2017)
- [46] HOPCROFT, J. E. ; MOTWANI, R. ; ULLMANN, J. D.: *Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie*. 2. Auflage. München : Pearson Studium, 2002
- [47] HU, C. ; XU, Z. ; LIU, Y. ; MEI, L. ; CHEN, L. ; LUO, X.: Semantic link network-based model for organizing multimedia big data. In: *IEEE Transactions on Emerging Topics in Computing* 2 (2014), Nr. 3, S. 376–387
- [48] HUANG, A.: Similarity measures for text document clustering. In: *Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008)*, Christchurch, New Zealand, 2008, S. 49–56

- [49] ILIOPOULOS, I. ; ENRIGHT, A. ; OUZOUNIS, C.: Textquest: Document Clustering of MEDLINE. In: *Biocomputing 2001* (2000), S. 384
- [50] JAIN, A. ; DUBES, R.: *Algorithms for clustering data*. Prentice Hall PTR, 1988 (Prentice-Hall Advanced Reference Series)
- [51] JAIN, A. K. ; MURTY, M. N. ; FLYNN, P. J.: Data Clustering: A Review. In: *ACM Comput. Surv.* 31 (1999), 9, Nr. 3, S. 264–323
- [52] JANNIDIS, F. ; KOHLE, H. ; REHBEIN, M.: *Digital Humanities: Eine Einführung*. J.B. Metzler, 2017
- [53] JOHNSON, D. B.: Finding All the Elementary Circuits of a Directed Graph. In: *SIAM J. Comput.* 4 (1975), Nr. 1, S. 77–84
- [54] JOHNSON, D. S.: Approximation algorithms for combinatorial problems. In: *Journal of Computer and System Sciences* 9 (1974), Nr. 3, S. 256 – 278
- [55] KARAA, W. B. A. ; ASHOUR, A. S. ; SASSI, D. B. ; ROY, P. ; KAUSAR, N. ; DEY, N.: Medline text mining: an enhancement genetic algorithm based approach for document clustering. In: *Applications of Intelligent Optimization in Biology and Medicine*. Springer, 2016, S. 267–287
- [56] KARP, R. M.: Reducibility Among Combinatorial Problems. In: *Complexity of Computer Computations*, 1972, S. 85–103
- [57] KLAUVANS, R. ; BOYACK, K. W.: Toward an objective, reliable and accurate method for measuring research leadership. In: *Scientometrics* 82 (2010), Nr. 3, S. 539–553
- [58] KOSOWSKI, A. ; MANUSZEWSKI, K.: Classical coloring of graphs. In: *Contemporary Mathematics* 352 (2004), S. 1–20
- [59] KOU, L. T. ; STOCKMEYER, L. J. ; WONG, C. K.: Covering Edges by Cliques with Regard to Keyword Conflicts and Intersection Graphs. In: *Commun. ACM* 21 (1978), Februar, Nr. 2, S. 135–139
- [60] KRUMKE, S. O. ; NOLTEMEIER, H.: *Graphentheoretische Konzepte und Algorithmen*. 2. Auflage. Wiesbaden : Vieweg + Teubner, 2009
- [61] LEE, C.-J. ; HSU, C.-C. ; CHEN, D.-R.: A Hierarchical Document Clustering Approach with Frequent Itemsets. In: *International Journal of Engineering and Technology* 9 (2017), Nr. 2, S. 174
- [62] MANNING, C. ; RAGHAVAN, P. ; SCHÜTZE, H.: *Introduction to Information Retrieval*. Cambridge University Press, 2008

- [63] MANNING, C. ; SCHÜTZE, H.: *Foundations of Statistical Natural Language Processing*. MIT Press, 1999
- [64] MATETI, P. ; DEO, N.: On Algorithms for Enumerating All Circuits of a Graph. In: *SIAM J. Comput.* 5 (1976), Nr. 1, S. 90–99
- [65] MU, T. ; GOULERMAS, J. Y. ; KORKONTZELOS, I. ; ANANIADOU, S.: Descriptive document clustering via discriminant learning in a co-embedded space of multilevel similarities. In: *Journal of the Association for Information Science and Technology* 67 (2016), Nr. 1, S. 106–133
- [66] PAVLOPOULOS, G. A. ; PROMPONAS, V. J. ; OUZOUNIS, C. A. ; ILIOPOULOS, I.: Biological information extraction and co-occurrence analysis. In: *Biomedical Literature Mining* (2014), S. 77–92
- [67] PSALLIDAS, F. ; BECKER, H. ; NAAMAN, M. ; GRAVANO, L.: Effective Event Identification in Social Media. In: *IEEE Data Eng. Bull.* 36 (2013), Nr. 3, S. 42–50
- [68] RAJARAMAN, A. ; ULLMAN, J. D.: *Mining of Massive Datasets*. Cambridge University Press, 2011
- [69] REISCHUK, K. R.: *Komplexitätstheorie, Band I: Grundlagen*. 2. Auflage. Stuttgart, Leipzig : Teubner, 1999
- [70] RINALDI, F. ; RIZZI, R.: Solving the train marshalling problem by inclusion–exclusion. In: *Discrete Applied Mathematics* 217, Part 3 (2017), S. 685 – 690
- [71] ROBERTS, K. ; SIMPSON, M. S. ; VOORHEES, E. M. ; HERSH, W. R.: Overview of the TREC 2015 Clinical Decision Support Track. In: *TREC*, 2015
- [72] SCHAEFFER, S. E.: Graph clustering. In: *Computer Science Review* 1 (2007), Nr. 1, S. 27 – 64
- [73] SCHRIJVER, A.: *Theory of linear and integer programming*. Chichester u.a. : Wiley, 1986
- [74] SHERWOOD, T. ; PERELMAN, E. ; HAMERLY, G. ; CALDER, B.: Automatically Characterizing Large Scale Program Behavior. In: *SIGOPS Oper. Syst. Rev.* 36 (2002), 10, Nr. 5, S. 45–57
- [75] SRIVASTAVA, A. N. ; SAHAMI, M.: *Text mining: Classification, clustering, and applications*. CRC Press, 2009

- [76] STANCHEV, L.: Semantic document clustering using a similarity graph. In: *Semantic Computing (ICSC), 2016 IEEE Tenth International Conference on IEEE*, 2016, S. 1–8
- [77] STEFANO, G. D. ; KOČI, M. L.: A Graph Theoretical Approach To The Shunting Problem. In: *Electronic Notes in Theoretical Computer Science* 92 (2004), S. 16–33
- [78] STREHL, A. ; GHOSH, J. ; MOONEY, R.: Impact of similarity measures on web-page clustering. In: *Workshop on artificial intelligence for web search (AAAI 2000)* Bd. 58, 2000, S. 64
- [79] SUTANTO, T. ; NAYAK, R.: The ranking based constrained document clustering method and its application to social event detection. In: *International Conference on Database Systems for Advanced Applications* Springer, 2014, S. 47–60
- [80] TARJAN, R. E.: Enumeration of the Elementary Circuits of a Directed Graph. In: *SIAM J. Comput.* 2 (1973), Nr. 3, S. 211–216
- [81] THEODOSIOU, T. ; DARZENTAS, N. ; ANGELIS, L. ; OUZOUNIS, C. A.: PuReD-MCL: a graph-based PubMed document clustering methodology. In: *Bioinformatics* 24 (2008), Nr. 17, 1935–1941. <http://dx.doi.org/10.1093/bioinformatics/btn318>. – DOI 10.1093/bioinformatics/btn318
- [82] TIERNAN, J. C.: An efficient search algorithm to find the elementary circuits of a graph. In: *Commun. ACM* 13 (1970), Nr. 12, S. 722–726
- [83] VOSSEN, G. ; WITT, K.: *Grundkurs Theoretische Informatik: Eine anwendungsbezogene Einführung - Für Studierende in allen Informatik-Studiengängen*. Springer Fachmedien Wiesbaden, 2016
- [84] WANG, C. ; SONG, Y. ; EL-KISHKY, A. ; ROTH, D. ; ZHANG, M. ; HAN, J.: Incorporating world knowledge to document clustering via heterogeneous information networks. In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* ACM, 2015, S. 1215–1224
- [85] ZHAI, C. ; MASSUNG, S.: *Text Data Management and Analysis: A Practical Introduction to Information Retrieval and Text Mining*. Association for Computing Machinery and Morgan & Claypool Publishers, 2016 (ACM Books)
- [86] ZHU, S. ; ZENG, J. ; MAMITSUKA, H.: Enhancing MEDLINE document clustering by incorporating MeSH semantic similarity. In: *Bioinformatics* 25 (2009), Nr. 15, S. 1944–1951

- [87] ZHU, Y. ; ZHU, R.: Sequence Reconstruction under some Order-Type Constraints. In: *Scientia Sinica (A)* 26(7) (1983), S. 702–713
- [88] ZIONTS, S.: *Linear and integer programming*. Englewood Cliffs : Prentice-Hall, 1974



# Erklärung

Ich versichere, dass ich die von mir vorgelegte Dissertation selbständig angefertigt, die benutzten Quellen und Hilfsmittel vollständig angegeben und die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken im Wortlaut oder dem Sinn nach entnommen sind, in jedem Einzelfall als Entlehnung kenntlich gemacht habe; dass diese Dissertation noch keiner anderen Fakultät oder Universität zur Prüfung vorgelegen hat; dass sie – abgesehen von unten angegebenen Teilpublikationen – noch nicht veröffentlicht worden ist, sowie, dass ich eine solche Veröffentlichung vor Abschluss des Promotionsverfahrens nicht vornehmen werde. Die Bestimmungen der Promotionsordnung sind mir bekannt. Die von mir vorgelegte Dissertation ist von Prof. Dr. Rainer Schrader betreut worden.

Köln, den 9. November 2017    *Jens Dörpinghaus*

## Teilpublikationen

DÖRPINGHAUS, J. ; SCHAAF, S. ; FLUCK, J. ; ZIMMERMANN, M. : Document Clustering using a Graph Covering with Pseudostable Sets. In: Ganzha, M. (Hrsg.) ; Maciaszek, L. (Hrsg.) ; Paprzycki, M. (Hrsg.): *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems*, Bd. 11, IEEE, 2017 (Annals of Computer Science and Information Systems), 329–338