

Zur Komplexität von Reduzierbarkeitsproblemen über H-Comparabilitygraphen

Dissertation

Zur Erlangung des akademischen Grades

**doctor rerum naturalium
(Dr. rer. nat.)**

von Herrn Dipl.-Wirtsch.-Math. Michael Andresen

geb. am 10. November 1979 in Hoyerswerda

genehmigt durch die Fakultät für Mathematik
der Otto-von-Guericke-Universität Magdeburg

Gutachter: apl. Prof. Dr. Heidemarie Bräsel

Prof. Dr. Ingo Althöfer

eingereicht am: 20. Mai 2009

Verteidigung am: 03. November 2009

Michael Andresen

Zur Komplexität von Reduzierbarkeitsproblemen über H-Comparabilitygraphen
Dissertationsschrift

e-mail: Michael.Andresen@Web.de

Danksagung

Ich danke Frau apl. Prof. Dr. Heidemarie Bräsel von ganzem Herzen für die hervorragende Betreuung dieser Arbeit, und für die Unterstützung und Ermutigung zur Präsentationen von Ergebnissen auf Tagungen und Workshops. Ich danke allen Kollegen am Institut für Algebra und Geometrie für die tatkräftige administrative Unterstützung. Meinen Eltern danke ich für die Unterstützung während des gesamten Studiums. Julia danke ich für ihre Geduld an meiner Seite.

Abstract

This thesis investigates the complexity of the following decision problem. Determine, whether a given sequence of an open-shop scheduling problem on an arbitrary set of operations is irreducible, or not.

With the notion of so-called H-comparability graphs the corresponding complementary problem co-IRREDUCIBILITY can be posed as follows: ‘Is there a comparability graph G^* with $G_{SIJ} \subseteq G^* \subsetneq G$ ’, where G_{SIJ} denotes the induced subgraph of the Hamming graph $K_n \times K_m$ corresponding to the set of operations, and G denotes the H-comparability graph corresponding to the given sequence. This problem, however, is no special case of the COMPARABILITY-GRAPH-SANDWICH problem which is known to be NP-complete.

IRREDUCIBILITY is known to belong to co-NP. It is investigated, under which conditions IRREDUCIBILITY belongs to NP, and therefore to $NP \cap co-NP = ZPP^*$, or even P. An algorithm is presented which reduces reducible sequences nondeterministically in polynomial time, and fails to recognize the irreducibility of irreducible sequences only under very narrow conditions. Based on reasonable doubts on the existence of instances meeting these narrow conditions, it is conjectured that IRREDUCIBILITY belongs to NP. If true, from this conjecture follows that either IRREDUCIBILITY belongs to P, or to NP-incomplete = $NP - (P + NP\text{-complete})$. The details pointing to IRREDUCIBILITY belonging to P are discussed.

Zusammenfassung

In dieser Arbeit wird das folgende Entscheidungsproblem hinsichtlich seiner Komplexität untersucht. Gegeben ist ein Plan eines Open-Shop Schedulingproblems über einer beliebigen Operationenmenge. Ist dieser Plan irreduzibel?

In einer Formulierung mittels sogenannter H-Comparabilitygraphen lautet die Problemstellung für das komplementäre Problem co-IRREDUCIBILITY: “Existiert ein Comparabilitygraph G^* mit $G_{SIJ} \subseteq G^* \subsetneq G$ ”, wobei G_{SIJ} der zu der Operationenmenge SIJ gehörende induzierte Teilgraph des Hamming-Graphen $K_n \times K_m$, und G der H-Comparabilitygraph zu dem gegebenen Ausgangsplan ist. Trotz der Ähnlichkeit ist dieses Problem kein Spezialfall des COMPARABILITY-GRAPH-SANDWICH Problems, dessen NP-Vollständigkeit bekannt ist.

Bekannt ist die Zugehörigkeit von IRREDUCIBILITY zu co-NP. Untersucht werden die Bedingungen, unter denen IRREDUCIBILITY in NP, und damit in $NP \cap co-NP = ZPP^*$, oder sogar in P liegt. Es wird ein Algorithmus vorgestellt, der reduzierbare Pläne nichtdeterministisch reduziert, und irreduzible Pläne nur unter sehr engen Voraussetzungen nicht als irreduzibel erkennen kann. Basierend auf begründeten Zweifeln an der Existenz von Instanzen, die diese engen Voraussetzungen erfüllen, wird vermutet, dass IRREDUCIBILITY in NP liegt. Bei Richtigkeit dieser Vermutung liegt IRREDUCIBILITY entweder in P, oder in NP-incomplete = $NP - (P + NP\text{-complete})$. Die Hinweise auf die Zugehörigkeit zu P werden diskutiert.

Keywords: irreducible sequences, H-comparability graphs, open-shop, complexity, ZPP*, NP-incomplete

Inhaltsverzeichnis

Einleitung	1
I Grundlagen	3
1 Graphen und Comparabilitygraphen	5
1.1 Graphen und Kantenmengen	5
1.2 Der Begriff des Comparabilitygraphen	7
1.2.1 Orientierungen und transitive Orientierungen	7
1.2.2 Die Γ -Relation	9
1.2.3 Implikationsklassen und Farbklassen	10
1.2.4 Die Γ -Graph	11
1.2.5 Transitive Hülle und transitive Reduktion	12
1.2.6 Das Dreieckslemma	13
1.2.7 Implikationsklassen als Bausteine transitiver Orientierungen	15
1.2.8 Anzahl transitiver Orientierungen	17
1.3 Charakterisierung von Comparabilitygraphen	22
1.3.1 Echte Implikationsklassen	22
1.3.2 Pseudotransitive Orientierungen	23
1.3.3 Bipartiter Graph	23
1.3.4 Dreieckssehnen	24
1.3.5 Verbotene Teilgraphen	24
1.3.6 Partiiell geordnete Mengen	25
1.4 Weitere Eigenschaften von Comparabilitygraphen	26
1.4.1 Entfernen von Knoten und Kanten aus Comparabilitygraphen	27
1.4.2 Modulare Dekomposition	30
1.4.3 Fast alle Comparabilitygraphen sind prim	31
1.4.4 Perfektheit und Superperfektheit	32
2 Komplexitätstheoretische Grundlagen	35
2.1 Die Zeitkomplexität von Algorithmen und Problemen	35
2.2 Die Komplexitätsklassen P und NP	37
2.3 Die Komplexitätsklasse NP-complete	38
2.4 Die Komplexitätsklassen co-NP und NP-incomplete	40
2.5 Weitere Komplexitätsklassen	42
3 Open-Shop Scheduling	45
3.1 Das allgemeine Shop Schedulingproblem	45
3.1.1 Technologie und Organisation	46
3.1.2 Die $\alpha \beta \gamma$ -Notation	47
3.1.3 Das Problem $O C_{\max}$	47
3.1.4 Plan und Schedule	48
3.2 Das Blockmatrizenmodell	49
3.3 Plangraphen und H-Comparabilitygraphen	51

3.3.1	Das disjunktive Graphenmodell	51
3.3.2	Der Plangraph zu einem Plan	53
3.3.3	Der H-Comparabilitygraph zu einem Plan	54
3.3.4	Reguläre Kanten und Schrägkanten	55
3.3.5	Ähnliche Pläne	57
3.3.6	Problemformulierungen für $O \parallel C_{\max}$	61
3.4	Die Komplexität von Schedulingproblemen	62
4	Das Problem IRREDUCIBILITY	65
4.1	Reduzierbarkeit und Irreduzibilität	65
4.1.1	Reduzierbarkeit von Plänen	65
4.1.2	Die Bedeutung irreduzibler Pläne	67
4.1.3	Die Probleme IRRED und REDUCING	67
4.1.4	Notwendige Bedingungen für Irreduzibilität	69
4.1.5	Verallgemeinerungen der Reduzierbarkeit	73
4.2	Planimplikationsklassen und der Faktorgraph	74
4.2.1	Der Begriff der Planimplikationsklasse	75
4.2.2	Planimplikationsklassen und Reduzierung	77
4.2.3	Der Faktorgraph	78
4.2.4	Trennende Kantenmengen	79
4.2.5	Stabile Schrägkanten	81
4.2.6	Planimplikationsklassen als Bausteine von Plänen	82
4.2.7	Erweiterte Planimplikationsklassen	86
4.3	Immer-transitive Kanten und stabile Schrägkanten	87
4.3.1	Implikationsklassen und Planimplikationsklassen	87
4.3.2	Das erweiterte Dreieckslemma	88
4.3.3	Nichttrivial-stabile Schrägkanten	90
4.3.4	Zwei relevante Planimplikationsklassen	94
4.4	Zur Komplexität verwandter Probleme	96
4.4.1	GRAPH SANDWICH, SUBGRAPH und COMP-DEL	96
4.4.2	Spezielle Operationenmengen	98
4.4.3	Das SEQUENCE-SUBGRAPH Problem	100
4.5	Schranken für die Anzahl von Schrägkanten	101
4.5.1	Untere Schranke	101
4.5.2	Obere Schranke	104
II	Zur Reduzierbarkeit von Plänen	107
5	Ansätze zur Reduktion von Plänen	109
5.1	Normale Pläne als Ausgangspunkt	109
5.2	Naives Umkehren von Planimplikationsklassen	110
5.3	Erkennung stabiler Schrägkanten	112
5.4	Erweitertes Dreieckslemma für Planimplikationsklassen	113
5.5	Eigenschaften von Comparabilitygraphen	113
5.5.1	Bestimmung von Mengen entferntbarer Schrägkanten?	114
5.5.2	Verbotene Teilgraphen	114
5.5.3	Eigenschaften von Modulen	116
6	Ein Reduktionsalgorithmus	119
6.1	Faktorgraph und Konsequenzgraph	119
6.1.1	Zulässige und zulässig erweiterbare Entfernungsmengen	119
6.1.2	Der Faktorgraph und seine Bedeutung	120
6.1.3	Der Konsequenzgraph und seine Bedeutung	121
6.2	Der Reduktionsgraph	124
6.2.1	Der Begriff des Reduktionsgraphen	124

6.2.2	Konflikte im Reduktionsgraphen	125
6.3	Ein Reduktionsschritt	129
6.3.1	Die Erweiterung von zulässigen Entfernungsmengen	129
6.3.2	Die Bestimmung von Konflikten	131
6.3.3	Getarnte stabile Schrägkanten	135
6.4	REDUCING1 und REDUCING2	136
6.4.1	Der Algorithmus REDUCING1	136
6.4.2	Der Algorithmus REDUCING2	142
6.4.3	Enumeration aller reduzierenden Pläne	148
6.5	Konstruktion irreduzibler Pläne durch Hinzufügen von Schrägkanten	149
6.6	Eine Nachbarschaft zwischen irreduziblen Plänen	154
7	Komplexität von IRREDUCIBILITY	159
7.1	REDUCIBILITY und Spezialfälle von IRRED	159
7.2	Umkehren von Planimplikationsklassen	159
7.3	Nichttrivial-stabile Schrägkanten	160
7.4	Getarnte stabile Schrägkanten	160
7.5	IRRED nicht in NP?	163
	Schlussbemerkungen	165
A	REDUCING1 und REDUCING2	167
A.1	Teilprozeduren	167
A.2	REDUCING1	170
A.3	REDUCING2	171
	Symbolverzeichnis	173
	Index	175
	Literaturverzeichnis	183

Einleitung

Bei der Suche nach Lösungen für Optimierungsprobleme ist eine Beschränkung des Suchraums auf effiziente Lösungen hilfreich. Für viele Optimierungsprobleme können solche effizienten Lösungen charakterisiert, und von nicht-effizienten Lösungen unterschieden werden (“efficiency frontier”). Für Probleme aus der Schedulingtheorie ist eine solche Unterscheidung oft nicht bekannt.

Aus der Frage der Beschreibung von effizienten Lösungen für das Open-Shop Schedulingproblem hat sich die Theorie der Reduzierbarkeit von Plänen entwickelt. Ein (Ablauf-) Plan ist eine strukturelle Lösung des Open-Shop Problems. Er enthält Informationen über die Reihenfolgen, in der einzelne Operationen abgearbeitet werden. Zusammen mit den numerischen Werten einer gegebenen Aufgabenstellung kann aus ihm ein Zeitplan (Schedule) abgeleitet werden. Zu den numerischen Werten eines Problems gehören vor allem die Bearbeitungszeiten der einzelnen Operationen.

Theorie der Reduzierbarkeit von Plänen

Der Begriff der Reduzierbarkeit geht zurück auf KLEINAU (1992) [41]. Sie erkannte, dass bestimmte Pläne für jede Wahl der numerischen Parameter besser als andere Pläne sind. Ein Plan A reduziert einen Plan B , $A \preceq B$, wenn sein Zielfunktionswert $f_P(A)$ für jede Wahl der Bearbeitungszeiten P nicht größer als der Zielfunktionswert $f_P(B)$ von B ist, $f_P(A) \leq f_P(B)$. Pläne, die nicht weiter (streng) reduzierbar sind, heißen irreduzibel. Sie stellen die effizienten Lösungen für das Open-Shop Problem dar.

Der Komplexitätsstatus des Problems der Erkennung irreduzibler Pläne konnte seitdem nur für einige Spezialfälle geklärt werden. So ist die Irreduzibilität sowohl für Pläne mit nur zwei Maschinen (BRÄSEL UND KLEINAU (1996) [16]), als auch für Pläne über Operationenmengen mit “baumartiger” Struktur (TAUTENHAHN (2000) [64]) in polynomieller Zeit erkennbar. Wesentliche Impulse für die Bestimmung der Komplexität dieses Problems sind auch bei WILLENUS (2000) [66] enthalten.

BRÄSEL, HARBORTH, TAUTENHAHN UND WILLENUS (1999) [13] erkannten die Möglichkeit der kompakten und anschaulichen Beschreibung der Reduzierbarkeit mittels sogenannter H-Comparabilitygraphen. Für vollständige Operationenmengen besteht ein H-Comparabilitygraph aus dem Hamming-Graphen (H-Graphen) $K_n \times K_m$ und allen zusätzlichen Kanten aus der transitiven Hülle zu einer azyklischen Orientierung auf dem Hamming-Graphen.

Das Ziel dieser Arbeit besteht in der Einordnung der Komplexität des Entscheidungsproblems der Erkennung der Irreduzibilität eines gegebenen Open-Shop Plans in die Komplexitätsklassen innerhalb von $NP \cup co - NP$. Dabei steht insbesondere die Frage der Existenz eines nichtdeterministischen Verfahrens mit polynomiell beschränkter maximaler Laufzeit im Vordergrund.

Kapitelübersicht

Die Arbeit ist in zwei Teile gegliedert. Teil I (Kapitel 1 bis 4) umfasst die Beschreibung des Problems IRREDUCIBILITY mit allen dazu notwendigen Grundlagen. In *Kapitel 1 (Graphen und Comparabilitygraphen)* werden die Eigenschaften von Comparabilitygraphen dargestellt. *Kapitel 2 (Komplexitätstheoretische Grundlagen)* enthält einen Überblick über die komplexitätstheoretischen Aspekte der Fragestellung. Ein besonderes Augenmerk liegt hier auf die Betrachtung randomisierter Verfahren mit polynomiell beschränkter maximaler Laufzeit. In *Kapitel 3 (Open-Shop Scheduling)* wird die Modellierung bestimmter Schedulingprobleme durch graphentheoretische Modelle beleuchtet. Insbesondere die Beschreibung von Open-Shop Plänen durch H-Comparabilitygraphen wird detailliert

dargelegt. *Kapitel 4 (Das Problem IRREDUCIBILITY)* enthält schließlich eine Darstellung der Theorie der Reduzierbarkeit mit Hilfe von H-Comparabilitygraphen, und die Beschreibung der zentralen Fragestellung.

Teil II dieser Arbeit (Kapitel 5 bis 7) umfasst die Untersuchung des Komplexitätsstatus von IRREDUCIBILITY. In *Kapitel 5 (Ansätze zur Reduktion von Plänen)* wird eine Reihe von Ideen zur polynomiellen Reduktion von Plänen diskutiert. Anhand verschiedenster Eigenschaften von H-Comparabilitygraphen werden dabei die Schwierigkeiten bei der Bestimmung der Komplexität von IRREDUCIBILITY erörtert.

In *Kapitel 6 (Ein Reduktionsalgorithmus)* werden zwei Ausprägungen eines Verfahrens zur konstruktiven Lösung des Problems IRREDUCIBILITY vorgestellt. Während einer der beiden Algorithmen (**Reducing1**) ein polynomielles Verfahren mit kleiner Versagenswahrscheinlichkeit $\varepsilon \geq 0$ ist, löst der andere Algorithmus (**Reducing2**) das Problem ohne Fehler, aber unter Umständen nicht in polynomiell beschränkter Laufzeit. Es werden Bedingungen diskutiert, unter denen beide Algorithmen identisch sind.

Schließlich sind in *Kapitel 7 (Komplexität von IRREDUCIBILITY)* die Ergebnisse der Untersuchungen der vorangegangenen Kapitel zur Komplexität von IRREDUCIBILITY zusammenfassend dargestellt.

Darüber hinaus enthält die Arbeit einen *Anhang*, in dem die in Kapitel 6 beschriebenen Algorithmen **Reducing1** und **Reducing2** etwas detaillierter dargestellt sind.

Teil I
Grundlagen

Kapitel 1

Graphen und Comparabilitygraphen

In diesem Kapitel stellen wir die zentralen graphentheoretischen Begriffe und Konzepte dieser Arbeit ausführlich vor. Nach einigen grundlegenden Definitionen und Bezeichnungen (Abschnitt 1.1) widmen wir uns im zweiten Abschnitt dem Begriff der transitiven Orientierung. Anschließend betrachten wir verschiedene Charakterisierungen von Graphen, die transitive Orientierungen besitzen (Abschnitt 1.3), und stellen danach einige ausgewählte Eigenschaften dieser Graphenklasse vor (Abschnitt 1.4).

1.1 Graphen und Kantenmengen

Graphen und Komponenten

Ein *Graph* $G = (V, E)$ besteht aus einer Menge von *Knoten* $V = \{v_1, \dots, v_n\}$ und einer Menge $E \subseteq V \times V$ von *Kanten* (oder *Bögen*), die verschiedene Knoten miteinander verbinden können. Sind mehrere Kanten (*parallele Bögen*, *Multikanten*) zwischen zwei Knoten zugelassen, so bezeichnet man einen Graphen auch als *Multigraphen*. Eine Kante von einem Knoten v zurück zum selben Knoten v wird als *Schlinge* bezeichnet. Ein Graph, der weder multiblen Kanten, noch Schlingen enthält heißt *schlicht* oder *einfach*.

Ferner heißt ein Graph $G = (V, E)$ *zusammenhängend*, wenn von jedem Knoten $v \in V$ ein Weg zu jedem anderen Knoten $w \in V$ existiert. Andernfalls zerfällt G in verschiedene *Zusammenhangskomponenten* oder einfach nur *Komponenten*.

In dieser Arbeit betrachten wir ausschließlich zusammenhängende schlichte Graphen. Die Auffassung von der Kantenmenge $E \subseteq V \times V$ als binärer Relation zwischen den Knoten ist daher wohldefiniert. Ferner betrachten wir ausschließlich Graphen mit endlichen Knoten- und Kantenmengen.

Die Sprechweise „ein Graph $G = (V, E)$ “ meint in dieser Arbeit immer einen schlichten ungerichteten Graphen. Aus Gründen der Übersichtlichkeit werden wir generell darauf verzichten, diese beiden Eigenschaften jedesmal erneut zu nennen.

Gerichtete und ungerichtete Kanten

Mit $e = ab \in E$ wollen wir eine *gerichtete* Kante von Knoten $a \in V$ zu Knoten $b \in V$ bezeichnen. Die zugehörige *Umkehrkante* von b nach a sei mit $e^{-1} = ba$ (gelegentlich auch mit \bar{e}) bezeichnet. Existiert in einem Graphen $G = (V, E)$ zu einer Kante $e = ab \in E$ auch die Umkehrkante $e^{-1} \in E$, so sagen wir, G enthält die *ungerichtete* Kante $\hat{e} = \widehat{ab}$ zwischen a und b . Existiert in einem Graphen G zu jeder Kante e die Umkehrkante e^{-1} , so nennen wir G einen *ungerichteten Graphen*, andernfalls ist G ein *gerichteter Graph*.

Für eine ungerichtete Kante $\hat{e} = \{e, e^{-1}\} = \widehat{e^{-1}} \subseteq E$ schreiben wir gelegentlich abkürzend $\hat{e} \in E$. Diese Konvention erlaubt uns, bei ungerichteten Graphen eine einzelne Kante \hat{e} als Element der Kantenmenge zu betrachten, obwohl wir sie formal als Menge zweier gerichteter Kanten e und e^{-1} beschreiben.

Wege und Nichtkanten

Ein (gerichteter) *Weg* oder *Pfad* in einem Graphen von einem Knoten a zu einem Knoten b ist eine Folge von Kanten $e_1 = v_1v_2, e_2 = v_2v_3, \dots, e_l = v_lv_{l+1}$ mit $v_1 = a$ und $v_{l+1} = b$. Er kann wahlweise durch die Kanten, e_1, \dots, e_l , oder durch die Folge der Knoten $a = v_1, v_2, \dots, v_l, v_{l+1} = b$ beschrieben werden, und hat wahlweise die *Kantenlänge* l oder die *Knotenlänge* $l + 1$. Wir werden die Länge von Wegen in aller Regel durch die Anzahl ihrer Kanten beschreiben.

Mit $\hat{e} \notin E$ bezeichnen wir eine ungerichtete Kante, die *nicht* in der Kantenmenge E zu einem Graphen $G = (V, E)$ liegt. D.h. weder e , noch e^{-1} liegen in E . Wir werden solche Kanten gelegentlich als *Nichtkanten* in G bezeichnen. Diese sprachliche Regelung wird sich später als äußerst hilfreich erweisen.

Im Rahmen dieser Arbeit wollen wir einen Graphen $G = (V, E)$ grundsätzlich als ungerichtet betrachten. D.h. wann immer eine Kante $e = ab \in E$ existiert, dann existiert in G auch ihre Umkehrkante $e^{-1} \in E$. Dadurch entfällt u.a. auch die Notwendigkeit der formalen Unterscheidung zwischen gerichteten und ungerichteten Wegen. Die Definition der gerichteten Wege deckt beide Möglichkeiten ab.

Zwei Knoten $a, b \in V$, die durch eine Kante $\hat{ab} \in E$ verbunden sind, oder zwei Kanten $\hat{e}_1, \hat{e}_2 \in E$, die einen gemeinsamen Knoten haben, heißen *adjazent*. Eine Kante *inzidiert* mit jedem ihrer beiden Knoten. Die Anzahl der ungerichteten Kanten, die mit einem Knoten a inzidieren, liefern die *Valenz* oder den *Grad* von a .

Teilgraph und induzierter Teilgraph

Ein *Teilgraph* oder *Untergraph* von $G = (V, E)$ ist ein Graph $G' = (V', E')$ mit $V' \subseteq V$ und $E' \subseteq E$. Gelegentlich wird dabei auch $V' = V$ gefordert. Die Frage, ob $V' = V$ oder $V' \subset V$ gilt, markiert manchmal den Unterschied zwischen den Begriffen Teilgraph und Untergraph. Wir verwenden beide Begriffe synonym (und meinen $V' \subseteq V$).

Sei $V^* \subseteq V$ eine Teilmenge der Knoten und $E^* \subseteq E$ eine (ungerichtete) Teilmenge der Kanten von $G = (V, E)$. Dann ist $G_{V^*} = (V^*, E(V^*))$ der durch die Knotenmenge V^* *induzierte Teilgraph* von G , und $G_{E^*} = (V(E^*), E^*)$ der durch die Kantenmenge E^* *induzierte Teilgraph* von G . Dabei ist $E(V^*)$ die durch V^* *induzierte Kantenmenge*, $E(V^*) = \{ab \in E : a, b \in V^*\} \subseteq E$, d.h. $E(V^*)$ enthält genau die Kanten aus E , für die sowohl der Anfangs- als auch der Endknoten in V^* liegt. $V(E^*)$ ist analog die durch E^* *induzierte Knotenmenge*, $V(E^*) = \{a, b \in V : ab \in E^*\}$.

Symmetrischer Abschluss und Semigraph

Zu einer beliebigen Kantenmenge $A \subseteq E$ bezeichnen wir mit $A^{-1} = \{e^{-1} : e \in A\}$ (bzw. mit \bar{A}) die Menge der Umkehrkanten zu den Kanten aus A . Mit $\hat{A} = \{e, e^{-1} : e \in A\} = A \cup A^{-1}$ sei ferner der *symmetrische Abschluss* oder die *symmetrische Hülle* der Kantenmenge A bezeichnet. \hat{A} enthält die Menge der ungerichteten Kanten zu den Kanten aus A (oder A^{-1}). Den Graphen $\hat{G}_A = (V, \hat{A})$ nennt man den *Semigraphen* zum Graphen $G_A = (V, A)$. Er ist ungerichtet und enthält alle Kanten e und e^{-1} für die gilt, dass wenigstens eine der beiden in A enthalten ist.

Wie bereits erwähnt, beschäftigen wir uns in dieser Arbeit ausschließlich mit einfachen und zusammenhängenden Graphen. Insbesondere spielen *isolierte Knoten*, also Knoten, die mit keiner Kante inzidieren, keine Rolle. Darüberhinaus ist durch die Struktur der Graphen, die wir zur Beschreibung von Schedulingproblemen verwenden werden, die Menge der Knoten klar vorgegeben und unabänderlich. Daher können wir in diesen Fällen einen Graphen vollständig und eindeutig durch die Angabe der Kantenmenge beschreiben. Wir werden also abkürzend häufig die Kantenmenge A mit dem durch sie induzierten Untergraphen $G_A = (V(A), A)$ identifizieren. Für den zu A gehörenden Semigraphen $\hat{G}_A = (V(A), \hat{A})$ werden wir dann jedoch $[A] = (V(A), \hat{A})$ schreiben. Diese abweichende Notation bietet uns mehr Platz und Übersichtlichkeit für Variationen der Kantenmenge A : Beispielsweise kann der Semigraph $\hat{G}_{(P_1+P_2)^{tr}} = \left(V(P_1+P_2)^{tr}, \widehat{(P_1+P_2)^{tr}} \right)$ zu einer Kantenmenge $A = (P_1+P_2)^{tr}$, so etwas übersichtlicher als $\left[(P_1+P_2)^{tr} \right]$ geschrieben werden.

Differenz und Vereinigung von Kantenmengen

Die Vereinigung (oder Summe) von zwei Kantenmengen A und B werden wir der Einfachheit halber in aller Regel mit $A \cup B = A + B$ beschreiben. Analog, schreiben wir $A \setminus B = A - B$ für die Differenz der Kantenmengen. Dies erlaubt uns beispielsweise das Umkehren der Kantenmengen P_i und P_j in einem Graphen einfach mit $A - P_i - P_j + P_i^{-1} + P_j^{-1} = (A \setminus (P_i \cup P_j)) \cup (P_i^{-1} \cup P_j^{-1})$ zu bezeichnen.

Mit $G - \hat{A} = (V, E - \hat{A})$ sei der Teilgraph von $G = (V, E)$ bezeichnet, der durch *Entfernen* oder *Löschen* der ungerichteten Kanten aus $\hat{A} \subseteq E$ entsteht. Wird durch das Entfernen von \hat{A} kein Knoten aus V isoliert, so gilt $[E - \hat{A}] = G - \hat{A}$. Von diesem Fall kann in dieser Arbeit ausnahmslos ausgegangen werden.

Mit $G - \hat{e}$ sei analog der Teilgraph bezeichnet, der durch *Entfernen* der Kante \hat{e} entsteht. Das Konzept der Kontraktion von Kanten spielt in dieser Arbeit keine Rolle, muss aber deutlich von dem einfachen Entfernen von Kanten unterschieden werden.

Komplement und bipartiter Graph

Der *Komplementgraph* oder das *Komplement* $G^c = (V, E^c)$ (oder $\bar{G} = (V, \bar{E})$) zu einem schlichten Graphen $G = (V, E)$ enthält genau die Kanten, die in E nicht enthalten sind, $E^c = V \times V - E$.

Ein Graph $G = (V, E)$ heißt *bipartit*, wenn seine Knotenmenge $V = V_1 + V_2$ eine Zerlegung in zwei Mengen V_1 und V_2 hat, so dass sämtliche Kanten in E sowohl mit einem Knoten aus V_1 , als auch mit einem Knoten aus V_2 inzidieren. D.h. die induzierten Kantenmengen E_{V_1} und E_{V_2} sind leer. Den *vollständigen bipartiten Graph* auf den Knotenmengen V_1 und V_2 mit $|V_1| = n$ und $|V_2| = m$ bezeichnen wir mit $K_{n,m} = (V_1 + V_2, \{ij : i \in V_1, j \in V_2\})$.

Kreis, Clique und stabile Knotenmenge

Abschließend seien mit $K_n = (\{1, \dots, n\}, \{ij : i, j \in \{1, \dots, n\}\})$ der *vollständige Graph* auf n Knoten, und mit C_n der *Kreis* mit n Knoten und n Kanten bezeichnet. Tritt der K_n als Teilgraph eines Graphen G auf, so nennt man diesen Teilgraphen auch *Clique*. Eine *stabile* oder *unabhängige Knotenmenge* in G ist eine Clique im Komplement G^c .

Auf der Grundlage dieser Bezeichnungen können wir uns nun ausführlich mit einer speziellen Klasse von Graphen befassen—den Comparabilitygraphen.

1.2 Der Begriff des Comparabilitygraphen

In diesem Abschnitt werden wir die Begriffe *transitive Orientierung* und *Comparabilitygraph* einführen (Abschnitt 1.2.1), und einige Konzepte zur Beschreibung von transitiven Orientierungen betrachten (Abschnitte 1.2.2 bis 1.2.8).

Übersichten zu diesen Zusammenhängen, die über die hier dargestellten hinaus gehen, können z.B. bei GOLUMBIC (1980) [33] oder SIMON (1992) [60] nachgelesen werden.

1.2.1 Orientierungen und transitive Orientierungen

Bevor wir jedoch den Begriff der *transitiven Orientierung* definieren können, müssen wir klären, was wir unter einer *Orientierung* auf einem schlichten ungerichteten Graphen verstehen wollen.

Definition 1.1 (Orientierung) Sei $G = (V, E)$ ein Graph. Eine Kantenmenge $A \subseteq E$ mit $A + A^{-1} = E$ und $A \cap A^{-1} = \emptyset$ heißt eine *Orientierung* von G .

Eine Orientierung A enthält also von jedem (gerichteten) Kantenpaar $\{e, e^{-1}\} \subseteq E$ genau eine der gerichteten Kanten e oder e^{-1} . Oder, alternativ: A enthält von jeder ungerichteten Kante $\hat{e} \in E$ genau eine der beiden möglichen "Richtungen" oder Orientierungen.

Interpretation von Orientierungen

Es gibt mehrere Möglichkeiten der Interpretation einer Orientierung A auf dem Graphen $G = (V, E)$. Zum Einen können wir sie als Teilmenge der gerichteten Kanten eines Graphen auffassen—also als Kantenmenge $A \subseteq E$. Diese Interpretation entspricht der Definition. Andererseits können wir A auch als Zuordnung oder Fixierung einer “Richtung” für jede der Kanten eines ungerichteten Graphen betrachten. Diese Interpretation ist formal völlig verschieden von der ersten. Sie ist jedoch sehr anschaulich und hilfreich bei der Interpretation von Konzepten im Zusammenhang mit der Umorientierung von Teilmengen orientierter Kanten. Zum Dritten können wir A auch als (induzierten) gerichteten, oder *orientierten*, Teilgraphen $G_A = (V, A)$ vom ungerichteten Semigraphen $[A] = G = (V, A + A^{-1})$ auffassen. Diese Interpretation wird uns später insbesondere erlauben, einen Sachverhalt (z.B. eine Lösung für das Open-Shop Scheduling-Problem) direkt mit einer Orientierung zu verknüpfen, ohne in jedem Argumentationsschritt den Umweg über den zugehörigen Semigraphen erläutern zu müssen. Wir werden alle drei Interpretationen verwenden: die erste und die letzte in formaler Darstellung von Zusammenhängen (weitgehend synonym), und die mittlere zur Anschauung.

Transitive Orientierungen und Comparabilitygraphen

Definition 1.2 (transitive Orientierung) Eine transitive Orientierung auf einem Graphen $G = (V, E)$ ist eine Orientierung $T \subseteq E$ mit der Eigenschaft, dass $ab, bc \in T$ impliziert, dass auch $ac \in T$ gilt.

D.h. in einer transitiven Orientierung T existiert zu jedem Weg der Kantenlänge 2 auch die direkte *transitive* Kante. Als direkte Folgerung gilt sogar, dass zu jedem Weg der Länge $l \geq 2$ in T alle transitiven Kanten in T enthalten sein müssen. D.h. jeder Weg der Kantenlänge l in T korrespondiert mit einem vollständigen Graphen K_{l+1} in $[T]$. Umgekehrt korrespondiert auch jede Clique K_n in $[T]$ mit einem Weg der Knotenlänge n in jeder transitiven Orientierung von $[T]$. Das liegt daran, dass transitive Orientierungen per Definition *kreisfrei* bzw. *azyklisch* sind: Würde es in einer transitiven Orientierung T einen Kreis $v_1v_2, v_2v_3, \dots, v_{l-1}v_l, v_lv_1$ geben, so müssten nach Definition auch alle transitiven Kanten—insbesondere auch v_1v_l —existieren. Also müsste T sowohl die Kante v_1v_l als auch ihre Umkehrkante v_lv_1 enthalten, und wäre damit keine Orientierung.

Definition 1.3 (Comparabilitygraph) Ein Graph $G = (V, E)$ auf dem eine transitive Orientierung existiert, ist ein Comparabilitygraph.

Ein Beispiel für einen Comparabilitygraphen G und eine transitive Orientierung von G wird in Abbildung 1.1 auf Seite 10 gegeben. G (linke Seite) besitzt genau zwei verschiedene transitive Orientierungen T_1 und T_2 . In der abgebildeten Orientierung (rechte Seite) sind die Kanten vw und vz transitive Kanten. Die Kante vz ist dabei sowohl über den Knoten x als auch über den Knoten y transitiv.

Die Bezeichnung Comparability- oder auch *Vergleichbarkeitsgraph* für transitiv orientierbare Graphen hat ihren Ursprung im Zusammenhang mit partiellen Ordnungen. Comparabilitygraphen werden daher oft auch als *PO-Graphen* (partially orderable) bezeichnet. Wir werden diese Zusammenhänge im Abschnitt 1.3.6 näher erläutern.

Ist $T \subseteq E$ eine transitive Orientierung auf einem Graphen G , so ist ihre *Umkehrorientierung* T^{-1} auch eine transitive Orientierung auf G . Wir nennen zwei transitive Orientierungen T_1 und T_2 *verschieden*, wenn $T_1 \neq T_2$ und $T_1 \neq T_2^{-1}$ gilt. Die Menge aller transitiven Orientierungen auf einem Graphen G bezeichnen wir mit $\mathcal{T}_G = \{T_1, \dots, T_t, T_1^{-1}, \dots, T_t^{-1}\}$. Für einen Graphen G mit t verschiedenen transitiven Orientierungen gilt $|\mathcal{T}_G| = 2t$.

Definition 1.4 (eindeutig orientierbar) Ein Comparabilitygraph $G = (V, E)$ der keine zwei verschiedenen transitiven Orientierungen besitzt, für den also $\mathcal{T}_G = \{T_1, T_1^{-1}\}$ gilt, ist *eindeutig orientierbar* oder *prim*.

Werden Comparabilitygraphen als PO-Graphen bezeichnet, so nennt man solche eindeutig transitiv orientierbaren Graphen *UPO-Graphen* (uniquely partially orderable). Eine besondere Eigenschaft solcher speziellen Comparabilitygraphen werden wir im Abschnitt 1.4.3 kennen lernen.

Der Rang eines Knotens

Eine azyklische Orientierung $A \subseteq E$ besitzt mindestens eine Quelle und mindestens eine Senke. Eine *Quelle* in einer Orientierung ist ein Knoten $q \in V$, für den sämtliche inzidierende Kanten $\widehat{qw} \in \hat{A}$ die Form $qw \in A$ haben. Für eine *Senke* $s \in V$ gilt umgekehrt $ws \in A$ für alle Kanten $\widehat{ws} \in \hat{A}$. Bezüglich einer azyklischen Orientierung $A \subseteq E$ können wir außerdem den Rang eines Knotens definieren.

Definition 1.5 (Rang) Sei $A \subseteq E$ eine azyklische Orientierung auf einem Graphen $G = (V, E)$. Der Rang $r(v)$ eines Knotens $v \in V$ ist die Anzahl der Knoten auf einem längsten einfachen Weg von einer Quelle zu v .

Ein längster einfacher Weg von einem Knoten $v \in V$ zu einem Knoten $w \in V$ ist dabei ein Weg von v nach w , der keinen Knoten mehrfach enthält, und nicht weiter verlängert werden kann, ohne diese Eigenschaft zu verlieren. Mit Hilfe des Rangs können wir den Knoten in einer kreisfreien Orientierung eine eindeutige "Rangordnung" zuordnen. Insbesondere liefern die Ränge eine Darstellung einer transitiven Orientierung: Ist $A \in \mathcal{T}_G$ eine transitive Orientierung von $G = (V, E)$, dann gilt für adjazente Knoten $v, w \in V$ entweder $r(v) < r(w)$ oder $r(v) > r(w)$.

1.2.2 Die Γ -Relation

Wir führen nun eine Relation auf den Kanten eines Graphen ein, die eine Beschreibung von transitiven Orientierungen ermöglicht.

Betrachten wir einen Comparabilitygraphen $G = (V, E)$, also einen ungerichteten Graphen, auf dem mindestens eine transitive Orientierung $T \in \mathcal{T}_G$ existiert. Es ist naheliegend, dass nicht jede beliebige Orientierung $A \subseteq E$ die Eigenschaft der Transitivität besitzt. Das bedeutet, dass die Kanten aus E nicht völlig unabhängig voneinander orientiert werden können, will man eine transitive Orientierung konstruieren. In der Tat ist es so, dass die Orientierung einer einzelnen Kante in aller Regel die Orientierung von anderen Kanten nach sich zieht.

Konkret impliziert die Nichtexistenz einer Kante zwischen zwei Knoten x und z , die durch einen Weg der Länge 2 über einen dritten Knoten y verbunden sind, dass die Orientierung einer dieser beiden Kanten zu y auch die Orientierung der anderen Kante festlegt. Es ist also $\widehat{xy}, \widehat{yz} \in E$, aber $\widehat{xz} \notin E$. Wird nun die Kante \widehat{xy} von x nach y orientiert, oder, formal richtiger, xy zur Orientierung hinzugefügt, dann kann die Kante \widehat{yz} nicht mehr von y nach z orientiert werden. Sie muss von z nach y orientiert werden. Andernfalls kann die Orientierung nicht mehr transitiv werden, da immer die transitive Kante von x nach z fehlen wird ($\widehat{xz} \notin E$). Damit impliziert die Orientierung der Kante \widehat{xy} von x nach y die Orientierung der Kante \widehat{yz} von z nach y .

Dieser Sachverhalt kann durch eine binäre Relation zwischen gerichteten Kanten beschrieben werden. Diese Relation ist die sogenannte Γ -Relation.

Definition 1.6 (Γ -Relation) Sei $G = (V, E)$ ein Graph, und seien $ab, cd \in E$. Die Γ -Relation auf E wird folgendermaßen definiert:

$$ab\Gamma cd \Leftrightarrow \begin{cases} a = c, \widehat{bd} \notin E \\ \vee b = d, \widehat{ac} \notin E \\ \vee ab = cd \end{cases} .$$

Mit $\Gamma(e) = \{e' \in E - e : e'\Gamma e\}$ sei die Γ -Nachbarschaft der Kante $e \in E$ bezeichnet. Sie enthält die Menge der von e verschiedenen Kanten aus E , die in Γ -Relation zu e stehen.

Die Γ -Relation ist offensichtlich reflexiv und symmetrisch, aber sie ist nicht transitiv, d.h. sie ist keine Äquivalenzrelation. Um das zu verdeutlichen, betrachten wir z.B. die schwarzen Kanten in Abbildung 1.1. Die gerichtete Kante vw liegt sowohl in Γ -Relation zu der Kante vx ($\widehat{wx} \notin E$) als auch zu vz ($\widehat{wz} \notin E$), d.h. $\Gamma(vw) = \{vx, vz\}$. Eine dieser beiden Kanten, nämlich vx , liegt ihrerseits in Γ -Relation zur Kante vy ($\widehat{xy} \notin E$). Damit gilt also $vw\Gamma vx\Gamma vy$. Es gilt jedoch nicht $vw\Gamma vy$, da die (graue) Kante \widehat{wy} in E existiert.

Für zwei Kanten ab und cd , die nicht in Γ -Relation zueinander stehen, schreiben wir auch $ab \not\Gamma cd$.

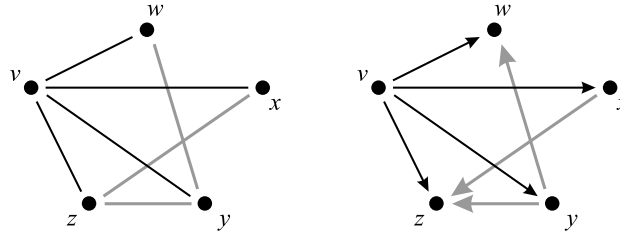


Abbildung 1.1: Die Γ -Relation. Dargestellt ist ein Graph G (linke Seite) und eine transitive Orientierung von G (rechte Seite). Es gelten: $\Gamma(vw) = \{vx, vz\}$, $\Gamma(vx) = \{vw, vy\}$, $\Gamma(yz) = \{xz, yw\}$. Die vier schwarzen (ungerichteten) Kanten bilden eine Farbklassse, die drei grauen eine andere.

Symmetrien der Γ -Relation

Neben der bekannten Symmetrie einer binären Relation, also in diesem Fall $ab\Gamma cd \Leftrightarrow cd\Gamma ab$, besitzt die Γ -Relation eine weitere bemerkenswerte symmetrische Eigenschaft. Wir haben die Γ -Relation als eine Relation zwischen gerichteten Kanten eingeführt. Bei ihrer Definition spielt jedoch die Nichtexistenz einer ungerichteten Kante die entscheidende Rolle. Daher folgt, dass zwei Kanten genau dann in Relation zueinander stehen, wenn auch ihre Umkehrkanten in Relation zueinander stehen, $ab\Gamma cd \Leftrightarrow ab^{-1}\Gamma cd^{-1}$, bzw. $ab\Gamma cd \Leftrightarrow ba\Gamma dc$.

1.2.3 Implikationsklassen und Farbklassen

Die Γ -Relation ist, wie bereits erwähnt, keine Äquivalenzrelation. Betrachten wir jedoch den transitiven Abschluss dieser Relation, so können wir eine Äquivalenzrelation mit Hilfe der Γ -Relation beschreiben.

Definition 1.7 (Γ -Kette) Sei $G = (V, E)$ ein Graph. Eine Γ -Kette oder ein Γ -Weg zwischen zwei Kanten $e', e'' \in E$ ist eine Kantenmenge $K = \{e_1, \dots, e_r\} \subseteq E$ mit $e' = e_1\Gamma e_2\Gamma \dots \Gamma e_r = e''$. Zwei Kanten, die durch eine Γ -Kette miteinander verbunden sind, heißen Γ -verbunden.

Definition 1.8 (Γ^* -Relation) Sei $G = (V, E)$ ein Graph. Zwei Kanten $e', e'' \in E$ liegen in Γ^* -Relation zueinander, $e'\Gamma^*e''$, wenn sie Γ -verbunden sind, d.h. wenn in G eine Γ -Kette von e' nach e'' existiert.

Implikationsklassen und Farbklassen

Die Γ^* -Relation ist eine Äquivalenzrelation. D.h. sie zerlegt die (gerichtete) Kantenmenge E eines Graphen $G = (V, E)$ in Äquivalenzklassen, für die gilt, dass zwei beliebige Kanten e_1 und e_2 aus einer solchen Äquivalenzklasse durch einen Γ -Weg miteinander verbunden sind. Für die Konstruktion einer transitiven Orientierung bedeutet dies, dass die Orientierung von \hat{e}_1 die Orientierung von \hat{e}_2 nach sich zieht. Wird e_1 als Orientierung von \hat{e}_1 gewählt, so muss auch e_2 als Orientierung von \hat{e}_2 gewählt werden, und umgekehrt. Also impliziert die Orientierung einer beliebigen Kante bereits die Orientierung aller anderen Kanten in derselben Äquivalenzklasse. Daher werden diese Äquivalenzklassen als *Implikationsklassen* bezeichnet.

Definition 1.9 (Implikationsklasse) Sei $G = (V, E)$ ein Graph. Eine Menge $I = I(e) = \{e' \in E : e'\Gamma^*e\} \subseteq E$ für ein $e \in E$ nennen wir eine Implikationsklasse von G . I heißt echt, falls $I \cap I^{-1} = \emptyset$ gilt.

Zwei Kanten liegen also genau dann in einer gemeinsamen Implikationsklasse, wenn sie Γ -verbunden sind. Die ungerichtete Kantenmenge $\hat{I} = I + I^{-1}$ zu einer Implikationsklasse I nennen wir die *Farbklassse* von I .

Mit $I(e)$ und $\hat{I}(e)$ bezeichnen wir die Implikations- bzw. die Farbklassse, die die Kante e enthält. Jede Kante $e \in E$ in einem schlichten Graphen $G = (V, E)$ induziert die eindeutig bestimmte Implikationsklasse $I(e)$ und die zugehörige Farbklassse $\hat{I}(e)$. Eine Implikationsklasse I wird auf der anderen Seite durch jede ihrer Kanten $e \in I$ induziert.

Für eine Implikationsklasse $I(e)$ ist $I(e)^{-1} = I(e^{-1})$ die *Umkehrimplikationsklasse*. Zwei Implikationsklassen I_1 und I_2 heißen *verschieden*, falls $I_1 \neq I_2$ und $I_1 \neq I_2^{-1}$ gilt.

In Abbildung 1.1 bilden die (schwarzen) gerichteten Kanten vw , vx , vy und vz eine Implikationsklasse I_1 , und die (grauen) Kanten xz , yz und yw eine Implikationsklasse I_2 . Damit enthält der (ungerichtete) Graph im linken Teil der Abbildung genau 4 Implikationsklassen, I_1 , I_2 , I_1^{-1} und I_2^{-1} , die zu zwei verschiedenen Farbklassen \hat{I}_1 (schwarze Kanten) und \hat{I}_2 (graue Kanten) gehören.

Zerlegung der Kantenmenge

In einer echten Implikationsklasse I gilt, dass zu keiner Kante $e \in I$ die Kante e^{-1} in I liegt. Gilt andererseits $e^{-1} \in I$ für ein $e \in I$, so gilt es für alle $e \in I$, und es folgt sofort $I(e) = I(e^{-1}) = I(e)^{-1} = \hat{I}(e)$. Der Grund hierfür ist die symmetrische Eigenschaft der Γ -Relation bzgl. der Umkehrkanten. Es kann sogar der Fall auftreten, dass eine Implikationsklasse I eines Graphen $G = (V, E)$ die gesamte Kantenmenge E enthält, $I = I^{-1} = \hat{I} = E$. Wir werden solche nicht-echten Implikationsklassen in den Abschnitten 1.3.1 und 1.3.4 noch näher betrachten. An dieser Stelle sei lediglich erwähnt, dass jeder ungerade Kreis C_n ($n \geq 5$) nur aus genau einer Implikationsklasse besteht.

Ist auf der anderen Seite die Γ -Nachbarschaft einer Kante e leer, $\Gamma(e) = \emptyset$, so besteht $I(e)$ allein aus der Kante e , $I(e) = \{e\}$. Ein Beispiel hierfür ist der vollständige Graph K_n ($n \geq 2$), bei dem jede Kante ihre eigene Implikationsklasse bildet.

Die Menge der Implikationsklassen eines Graphen $G = (V, E)$ bezeichnen wir mit $\mathcal{I}_G = \{I_1, \dots, I_k, I_1^{-1}, \dots, I_k^{-1}\}$, und die Menge der Farbklassen von G mit $\mathcal{F}_G = \{\hat{I}_1, \dots, \hat{I}_k\}$. Es gilt $I_1 + \dots + I_k + I_1^{-1} + \dots + I_k^{-1} = E$, bzw. $\hat{I}_1 + \dots + \hat{I}_k = E$. Eine echte disjunkte Zerlegung der Kantenmenge E in ihre Implikationsklassen liegt bei dieser Darstellung formal genau dann vor, wenn jede Implikationsklasse I_i ($i = 1, \dots, k$) echt ist. Ist dagegen eine Implikationsklasse $I_j \in \mathcal{I}_G$ unecht, so gilt $I_j = I_j^{-1}$, und diese Kantenmenge ist damit doppelt in $I_1 + \dots + I_k + I_1^{-1} + \dots + I_k^{-1}$ vertreten.

Dennoch hat E auch in diesem Fall eine disjunkte Zerlegung in ihre Implikationsklassen, $E = I_1 + \dots + I_l$, mit $\mathcal{I}_G = \{I_1, \dots, I_l\}$. Da wir es in dieser Arbeit jedoch generell mit Comparabilitygraphen zu tun haben werden, für die $\{I_1, \dots, I_l\} = \{I_1, \dots, I_k, I_1^{-1}, \dots, I_k^{-1}\}$ gilt, ziehen wir die Auffassung, dass \mathcal{I}_G aus der Menge der Implikationsklassen I_1, \dots, I_k und ihren Umkehrimplikationsklassen $I_1^{-1}, \dots, I_k^{-1}$ besteht, der Auffassung, dass \mathcal{I}_G aus I_1, \dots, I_l besteht, vor. Diese Auffassung erschwert zwar formal die Zerlegung von E in die Implikationsklassen von G beim Auftreten von unechten Implikationsklassen, erleichtert aber andererseits die Darstellung von transitiven Orientierungen bei Comparabilitygraphen.

Die Zerlegung von E in ihre Farbklassen liefert unabhängig von der Auffassung von \mathcal{I}_G für jeden schlichten Graphen eine disjunkte Zerlegung der Kantenmenge E .

Kürzeste Wege

Abschließend wollen wir noch kurz einen interessanten Zusammenhang zwischen Farbklassen und kürzesten Wegen in einem Graphen herstellen. Ein kürzester Weg zwischen zwei Knoten $a = x_0$ und $b = x_l$ eines Graphen $G = (V, E)$ ist ein Weg $x_0, x_1, x_2, \dots, x_l$ mit minimaler Länge l . Ein solcher Weg enthält insbesondere keine "Umwege". D.h. alle x_i sind verschieden, und es gilt für alle $i = 0, \dots, l-2$ $\widehat{x_i x_{i+2}} \notin E$, da der Weg sonst verkürzt werden könnte. Damit gilt jedoch $x_i x_{i+1} \Gamma(x_{i+1} x_{i+2})^{-1}$ für alle $i = 0, \dots, l-2$, und damit insbesondere, dass alle Kanten $x_i x_{i+1}$ ($i = 0, \dots, l-1$) entweder in $I(x_0 x_1)$ oder in $I(x_1 x_2) = I(x_0 x_1)^{-1}$ liegen.

Lemma 1.10 (kürzeste Wege) [27] *Seien $a, b \in V$ zwei Knoten aus einer Zusammenhangskomponente eines Graphen $G = (V, E)$. Dann liegen alle Kanten entlang eines kürzesten Weges von a nach b in einer gemeinsamen Farbklassse.*

1.2.4 Der Γ -Graph

Wir wollen noch eine alternative Möglichkeit zur Beschreibung von Implikationsklassen betrachten. Im vorangegangenen Abschnitt haben wir Implikationsklassen als Äquivalenzklassen der Γ^* -Relation

kennen gelernt. Zwei gerichtete Kanten liegen demnach genau dann in einer Implikationsklassen, wenn sie durch eine Γ -Kette verbunden sind.

Diesen Sachverhalt können wir alternativ auch mit Hilfe des sogenannten Γ -Graphen darstellen.

Definition 1.11 (Γ -Graph) Sei $G = (V, E)$ ein Graph. Der Γ -Graph $G_\Gamma = (E, \Gamma)$ zum Graphen G enthält als Knotenmenge die Kantenmenge von G , und verbindet zwei Knoten e_1 und e_2 in G_Γ genau dann, wenn sie in G in direkter Γ -Relation liegen.

Der Γ -Graph zu einem Graphen $G = (V, E)$ bildet somit genau die direkten Γ -Relationen in G ab. Eine Γ -Kette $e_1\Gamma \dots \Gamma e_l$ in G liefert einen ungerichteten Weg e_1, \dots, e_l in G_Γ . Folglich sind zwei Kanten e_1 und e_2 in G genau dann Γ -verbunden, $e_1\Gamma^*e_2$, wenn die mit ihnen korrespondierenden Knoten im Γ -Graphen zu G durch einen Weg verbunden sind, d.h. wenn sie in der gleichen Zusammenhangskomponente liegen. Die Zusammenhangskomponenten von G_Γ liefern somit die Implikationsklassen von G . Es ist sorgfältig auseinander zu halten, dass die Implikationsklassen von $G = (V, E)$ in G eine Zerlegung der Kantenmenge E , in G_Γ jedoch eine Partition der Knotenmenge (in Zusammenhangskomponenten) liefern.

Die Auffassung einer Γ -Kette als ungerichteter Weg im Γ -Graphen zu G bietet gegenüber der Darstellung als Kette $e_1\Gamma \dots \Gamma e_l$ den Vorteil einer besseren Anschauung. Insbesondere das Hinzufügen von Kanten zu G , das bestehende Γ -Relationen zerstören kann, wird in G_Γ durch Löschen von Kanten sehr anschaulich dargestellt. Auf diesen Aspekt werden wir in Kapitel 4.3 noch näher eingehen.

Ein Beispiel für einen Γ -Graphen findet sich in Abbildung 1.4 auf Seite 23.

Γ -Relation und Nichtkanten

Jede Γ -Relation in $G = (V, E)$ ist durch eine Nichtkante $\hat{e} \notin E$ induziert. Umgekehrt kann eine Nichtkante $\hat{e} \notin E$ beliebig viele oder gar keine Γ -Relationen induzieren. Wird jedoch durch $\hat{e} \notin E$ eine Γ -Relation induziert, so ist wegen der symmetrischen Eigenschaften der Γ -Relation als Relation zwischen gerichteten Kanten, die von der Nichtexistenz von ungerichteten Kanten abhängt, die Anzahl der induzierten Γ -Relationen gerade. Folglich hat der Γ -Graph zu einem beliebigen Graphen $G = (V, E)$ immer eine gerade Anzahl von Kanten.

Echte Implikationsklassen und konsistenter Γ -Graph

Für eine echte Implikationsklasse $I \in \mathcal{I}_G$ gilt $I \cap I^{-1} = \emptyset$. Im Γ -Graphen zu $G = (V, E)$ lässt sich die Echtheit von I daran messen, ob in der durch I induzierten Knotenmenge (Komponente) in G_Γ zwei Knoten $e_1, e_2 \in E$ existieren, deren korrespondierende Kanten in G Umkehrkanten voneinander sind, $e_1 = e_2^{-1}$.

Um diese umständliche Sprechweise etwas zu entflechten, werden wir in Zukunft die Knoten von G_Γ mit den Kanten von G identifizieren. Eine Implikationsklassen in G ist demnach genau dann echt, wenn die zugehörige Komponente in G_Γ keine zwei Knoten enthält, die Umkehrkanten voneinander repräsentieren.

Definition 1.12 (konsistent) Sei $G_\Gamma = (E, \Gamma)$ der Γ -Graph zu einem Graphen $G = (V, E)$. Eine Zusammenhangskomponente $K \subseteq E$ in G_Γ heißt konsistent, wenn $K \cap K^{-1} = \emptyset$ gilt. Sind alle Komponenten in G_Γ konsistent, so heißt G_Γ konsistent.

Eine Komponente $K \subseteq E$ in G_Γ ist also konsistent, wenn $e^{-1} \notin K$ für alle $e \in K$ gilt. Damit ist eine Implikationsklasse in G genau dann echt, wenn sie im Γ -Graphen zu G eine konsistente Komponente induziert.

1.2.5 Transitiv Hülle und transitive Reduktion

Eine transitive Orientierung $T \subseteq E$ auf einem Graphen $G = (V, E)$ enthält per Definition bereits sämtliche transitive Kanten ac , die sich durch die Zugehörigkeit von ab und bc zu T ergeben. Eine beliebige Orientierung $A \subseteq E$ auf G können wir u. U. zu einer transitiven Orientierung "ausbauen", indem wir sämtliche noch nicht vorhandenen transitiven Kanten zu A hinzufügen.

Definition 1.13 (transitive Hülle) Sei A eine Kantenmenge. Mit $A^{tr} = A + \{ab : \exists a\text{-}b\text{-Weg in } A\}$ bezeichnen wir die transitive Hülle von A .

Die transitive Hülle oder der *transitive Abschluss* einer Kantenmenge ist eindeutig bestimmt. Sie ist für beliebige Kantenmengen definiert, aber natürlich besonders interessant für Orientierungen. Die transitive Hülle einer Orientierung A ist per Definition transitiv. Ob A^{tr} jedoch eine Orientierung ist, hängt davon ab, ob A kreisfrei ist. Enthält die Orientierung A einen Kreis $ab, bc, ca \in A$, so enthält A^{tr} die transitiven Kanten ba, cb und ac . Dann ist A^{tr} keine Orientierung auf $[A^{tr}]$ ($\widehat{ab}, \widehat{bc}, \widehat{ac} \in A^{tr}$). Ist A dagegen azyklisch, so werden bei der Bestimmung der transitiven Hülle höchstens Kanten ac zu A hinzugefügt, die nicht in der Kantenmenge E des Graphen $G = (V, E)$ enthalten sind, für die also nicht $ca \in A$ gelten kann. Damit ist A^{tr} eine Orientierung. Im Allgemeinen ist A^{tr} dann jedoch keine Orientierung auf G mehr, $A^{tr} \not\subseteq E$, sondern auf dem Graphen $[A^{tr}]$. Der Spezialfall $A^{tr} \subseteq E$ tritt genau dann ein, wenn A bereits transitiv ist. Dann gilt $A^{tr} = A$ und $G = (V, E) = [A^{tr}]$ ist ein Comparabilitygraph.

Ist A eine azyklische, aber nicht transitive Orientierung auf einem Comparabilitygraphen $G = (V, E) = [A]$, dann gilt $A \subsetneq A^{tr}$. $[A]$ ist dann ein echter Teilgraph von $[A^{tr}]$, $[A] \subsetneq [A^{tr}]$.

Während die transitive Hülle A^{tr} zu einer Orientierung A eindeutig bestimmt ist, kann es umgekehrt viele verschiedene Orientierungen A_1, \dots, A_s geben, mit $A^{tr} = A_1^{tr} = \dots = A_s^{tr}$. Die (bezüglich Kanteninklusion) kleinste dieser Orientierungen ist dagegen ebenfalls eindeutig bestimmt.

Definition 1.14 (transitive Reduktion) Sei A eine Orientierung. Mit $A^r = \{ab \in A : \nexists v \in V(A) \text{ mit } av, vb \in A\}$ bezeichnen wir die transitive Reduktion von A .

Die transitive Reduktion, oder auch das *transitives Redukt* A^r zu einer transitiven Orientierung A erhält man also, indem man aus A alle transitiven Kanten entfernt. Auch diese Definition kann auf beliebige Kantenmengen A angewandt werden, ist aber besonders bedeutsam für transitive Orientierungen.

Ist A eine transitive Orientierung, dann ist A^r die eindeutig bestimmte kleinste Kantenmenge mit $(A^r)^{tr} = A$. Für eine azyklische Orientierung A , in der es keine (gerichteten) Wege mit 3 oder mehr Knoten gibt, gilt $A = A^{tr} = A^r$. A ist dann eine transitive Orientierung, die keine transitiven Kanten enthält.

Sowohl die transitive Hülle als auch die transitive Reduktion zu einer azyklischen Orientierung A können in polynomialer Zeit (siehe Kapitel 2) berechnet werden.

1.2.6 Das Dreieckslemma

Eine wichtige Eigenschaft von Implikationsklassen ist die Gültigkeit des sogenannten Dreieckslemmas für Implikationsklassen. Dieses Dreieckslemma wird häufig auch als Golumbic's Dreieckslemma (GOLUMBIC (1980) [33]) bezeichnet, um es von anderen Dreieckslemmata aus anderen Feldern der Mathematik zu unterscheiden. Das Fundament für dieses bedeutende Theorem (Teil (i)) findet sich jedoch bereits bei GILLMORE UND HOFFMANN (1964) [30].

Für Comparabilitygraphen ermöglicht das Dreieckslemma den Schluss von lokalen Gegebenheiten in einem betrachteten Graphen $G = (V, E)$ auf andere Bereiche dieses Graphen. Den Wert dieser Eigenschaft werden wir später noch zu schätzen lernen, wenn wir die sogenannten Planimplikationsklassen als Teilstruktur von Implikationsklassen kennen lernen werden (in Kapitel 4), für die diese Eigenschaft nicht mehr gilt.

Kanonische Γ -Ketten

Um zukünftige Beweise etwas übersichtlicher gestalten zu können, führen wir zunächst den Begriff der kanonischen Γ -Kette ein (GOLUMBIC [33]). Nach Definition der Γ -Nachbarschaft, sind zwei Kanten ab und $a'b'$ aus E genau dann in $G = (V, E)$ Γ -verbunden, wenn eine Γ -Kette von ab nach $a'b'$ in G existiert, $ab = a_0b_0\Gamma a_1b_1\Gamma \dots \Gamma a_kb_k = a'b'$. Zu jeder dieser Γ -Verbindungen $a_ib_i\Gamma a_{i+1}b_{i+1}$ ($i = 0, \dots, k-1$) können wir nun o.B.d.A. die Kante $a_{i+1}b_i$ in die Kette einfügen, $a_ib_i\Gamma a_{i+1}b_i\Gamma a_{i+1}b_{i+1}$. Diese zusätzliche Kante $a_{i+1}b_i$ ist in jedem Fall identisch zu einer ihrer beiden

Nachbarn in der Kette. (Insbesondere gilt $a_{i+1}b_i \in E$.) Damit können wir von jeder Γ -Kette zwischen zwei Kanten ab und $a'b'$ ohne Einschränkung annehmen, dass sie von folgender Form ist:

$$ab = a_0b_0\Gamma a_1b_0\Gamma a_1b_1\Gamma a_2b_1\Gamma \dots \Gamma a_{k-1}b_{k-1}\Gamma a_k b_{k-1}\Gamma a_k b_k = a'b'.$$

Eine Γ -Kette dieser Form nennen wir *kanonisch*. Die besondere Bedeutung dieser Notation liegt darin, dass wir nun jede Γ -Relation in einer Γ -Kette gleich behandeln können. Wir müssen bei der Interpretation von $a_0b_0\Gamma a_1b_1$ nicht unterscheiden, ob vielleicht $a_0 = a_1$ und $b_0 \neq b_1$ gilt, oder aber $b_0 = b_1$ und $a_0 \neq a_1$. Wir interpretieren stattdessen die Kette $a_0b_0\Gamma a_1b_0\Gamma a_1b_1$, und können sofort $\widehat{a_0a_1} \notin E$ und $\widehat{b_0b_1} \notin E$ schließen. Dabei müssen wir nicht wissen, ob $a_1b_0 = a_0b_0$ oder aber $a_1b_0 = a_1b_1$ ist. (Da wir nur schlichte Graphen betrachten, gilt $\widehat{ab} \notin E$ auch für $a = b$.)

Golumbic's Dreieckslemma

Um das Verständnis einer Erweiterung von Golumbic's Dreieckslemma in Kapitel 4 (Satz 4.47) zu erhöhen, liefern wir zu Satz 1.15 auch einen Beweis.

Satz 1.15 (Dreieckslemma) [30],[33] *Sei $G = (V, E)$ ein Graph, und seien A, B , und C Implikationsklassen von G mit $A \neq B^{-1}$ und $A \neq C$. Ferner seien $ab \in B$, $ac \in C$ und $bc \in A$. Dann gelten*

- (i) $\exists b'c' \in A \Rightarrow ab' \in B$ und $ac' \in C$
- (ii) $\exists b'c' \in A, a'b' \in B \Rightarrow a'c' \in C$
- (iii) $a \notin V(A)$.

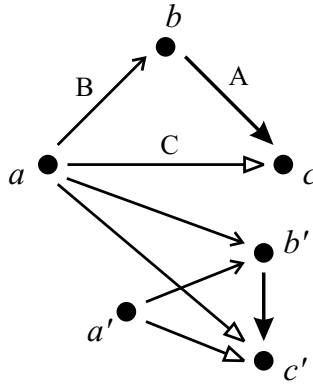


Abbildung 1.2: Das Dreieckslemma

Beweis. (Nach GOLUMBIC [33] und ANDRESEN [4]¹.) (i) Sei $b'c' \in A$. Dann existiert eine kanonische Γ -Kette $bc = b_0c_0\Gamma b_1c_0\Gamma b_1c_1\Gamma \dots \Gamma b_kc_k = b'c'$ in A . Wir zeigen nun per Induktion die Existenz von $ab_i \in B$ und $ac_i \in C$ für alle $i = 0, \dots, k$. Für $i = 0$ ist nichts zu tun, da nach Voraussetzung $ab_0 = ab \in B$ und $ac_0 = ac \in C$ gelten. Sei nun $i \geq 1$. Nach Induktionsvoraussetzung existieren $ab_{i-1} \in B$ und $ac_{i-1} \in C$. Aus der Existenz von $ac_{i-1} \in C$ und $b_i c_{i-1} \in A$ (aus Γ -Kette) folgt nun wegen $A \neq C$ die Existenz der ungerichteten Kante $\widehat{ab_i} \in E$. Andererseits folgt aus der direkten Γ -Nachbarschaft von $b_i c_{i-1}$ und $b_{i-1} c_{i-1}$ in der Γ -Kette $\widehat{b_{i-1}b_i} \notin E$. Damit liegen aber ab_i und $ab_{i-1} \in B$ in Γ -Relation zueinander, und beide folglich in der gleichen Implikationsklasse. Also ist $ab_i \in B$.

Aus $ab_i \in B$ folgt nun mit $b_i c_i \in A$ (Γ -Kette) und $A \neq B^{-1}$ die Existenz der ungerichteten Kante $\widehat{ac_i} \in E$. Da nun aber die Knoten c_i und c_{i-1} nicht benachbart sind ($b_i c_i \Gamma b_i c_{i-1}$), liegen ac_i und $ac_{i-1} \in C$ in Γ -Relation zueinander. Also ist auch $ac_i \in C$.

¹Der Beweis von (ii) bei GOLUMBIC [33] enthält einen Fehler, der bei ANDRESEN [4] korrigiert wird.

(ii) Angenommen, $B \neq C$. Wir betrachten nun eine neue kanonische Γ -Kette $ab = a_0b_0\Gamma a_1b_1\Gamma a_1b_1\Gamma \dots \Gamma a_l b_l = a'b'$ in B . Wir zeigen per Induktion die Existenz eines Dreiecks $\Delta a_j b_j c$ isomorph zu dem Dreieck Δabc für alle $j = 0, \dots, l$. Für $j = 0$ gilt nach Voraussetzung die Existenz von $\Delta a_0 b_0 c$. Sei nun also $j \geq 1$, und sei die Existenz von $\Delta a_{j-1} b_{j-1} c$ mit $a_{j-1} b_{j-1} \in B$, $a_{j-1} c \in C$ und $b_{j-1} c \in A$ bereits gezeigt. Die Existenz von $\widehat{a_j c}$, folgt aus $A \neq B^{-1}$ ($a_j b_{j-1} \in B$, $b_{j-1} c \in A$). Wegen $\widehat{a_j a_{j-1}} \notin E$ ($a_{j-1} b_{j-1} \Gamma a_j b_{j-1}$) liegt $a_j c$ in Γ -Relation zu $a_{j-1} c \in C$. D.h. $a_j c \in C$. Die Existenz von $\widehat{b_j c}$, andererseits, folgt aus der Annahme $B \neq C$ ($a_j b_j \in B$, $a_j c \in C$), und es gilt $b_j c \Gamma b_{j-1} c \in A$ ($b_j b_{j-1} \notin E$). Somit haben wir ein neues Dreieck $\Delta a_j b_j c$ mit $a_j b_j \in B$, $a_j c \in C$, und $b_j c \in A$ ($j = 0, \dots, l$), isomorph zu Δabc . Insb. existiert das Dreieck $\Delta a' b' c$ mit $a' b' \in B$, $a' c \in C$, und $b' c \in A$. Auf dieses Dreieck können wir nun Teil (i) bezüglich der Kanten $b' c \in A$ und $b' c' \in A$ anwenden, und erhalten $a' c' \in C$.

Sei nun $B = C$ angenommen. Aus $A \neq B^{-1}$ folgt die Existenz der Kante $\widehat{a' c'}$. Nach Teil (i) gilt ferner die Existenz von $a c' \in C$. Angenommen, $a' c'$ liegt nicht in C , $a' c' \in D \neq C$. Dann erfüllt das umgekehrte Dreieck $\Delta a' b' c'$ (mit $b' a' \in B^{-1}$, $c' b' \in A^{-1}$, und $c' a' \in D^{-1}$), bezüglich der Kanten $b' a' \in B^{-1}$ und $b a \in B^{-1}$ die Voraussetzungen von Teil (i) (es gelten $B^{-1} \neq (A^{-1})^{-1}$ und $B^{-1} = C^{-1} \neq D^{-1}$). Die Anwendung von Teil (i) liefert neben $b c' \in A$ insb. $a c' \in D$, im Widerspruch zu $a c' \in C \neq D$. Also folgt $a' c' \in C$.

(iii) $a \notin V(A)$ folgt direkt aus (i) ($A \neq C$). ■

Das Dreieckslemma besagt also folgendes. Angenommen, in einem beliebigen ungerichteten Graphen existiert ein Dreieck, dessen Kanten zu verschiedenen Implikationsklassen gehören. Existiert dann in diesem Graphen ein weiteres Dreieck, das, bezüglich der Zuordnung zu Implikationsklassen, mit dem Ausgangsdreieck in zwei Kanten übereinstimmt, so stimmt es auch in der dritten Kante überein. In jedem Graphen gibt es also viele ähnliche Dreiecke.

Anwendung auf echte Implikationsklassen

Als direkte Anwendung des Dreieckslemmas lässt sich folgern, dass echte Implikationsklassen transitive Orientierungen sind. Wegen der Bedeutung dieser Eigenschaft liefern wir auch hier den Beweis.

Satz 1.16 (Implikationsklassen sind transitiv) [33] *Sei $I \in \mathcal{I}_G$ eine Implikationsklasse eines Graphen $G = (V, E)$. Dann gilt entweder $I = \hat{I} = I^{-1}$, oder $I \cap I^{-1} = \emptyset$ und I und I^{-1} sind die einzigen transitiven Orientierungen von \hat{I} .*

Beweis. Angenommen, $I \cap I^{-1} \neq \emptyset$. Sei $ab \in I \cap I^{-1}$, d.h. $ab \Gamma^* ba$. Nun ist jede Kante $cd \in I$ Γ -verbunden mit $ab \in I$, $cd \Gamma^* ab$. Andererseits gilt dann auch $dc \Gamma^* ba$, und somit auch $cd \Gamma^* ab \Gamma^* ba \Gamma^* dc$. Also gilt $I = \hat{I}$. Sei nun $I \cap I^{-1} = \emptyset$ angenommen, und seien $ab, bc \in I$. Würde nun die Kante \widehat{ac} nicht existieren, so würde $ab \Gamma cb$ folgen—im Widerspruch zu $I \cap I^{-1} = \emptyset$. Also ist $\widehat{ac} \in E$. Sei J die Implikationsklasse, zu der ac gehört. Angenommen, $I \neq J$. Wegen $I \neq I^{-1}$ und $I \neq J$ ist das Dreieckslemma (Satz 1.15) auf das Dreieck Δabc anwendbar. Daraus folgt $a \notin V(I)$ (mit Teil (iii))—im Widerspruch zur Voraussetzung. Also gilt $I = J$, d.h. I ist transitiv. Mit I ist offensichtlich auch I^{-1} transitiv. Schließlich impliziert die Orientierung einer Kante aus \hat{I} die Orientierung aller anderen Kanten. D.h. für jede transitive Orientierung T auf \hat{I} gilt entweder $I \subseteq T$ oder $I^{-1} \subseteq T$. Damit sind I und I^{-1} die einzigen transitiven Orientierungen von \hat{I} . ■

Wir können also echte Implikationsklassen als transitive Orientierungen auf den Farbklassen auffassen. Die Aussage, dass eine echte Implikationsklasse transitiv ist, findet sich bereits bei GILMORE UND HOFFMAN (1964) [30]. Jedoch ist die Formulierung dort unabhängig von den Begriffen Implikationsklasse und Orientierung. Insbesondere findet sich dort noch nicht die Auffassung von Implikationsklassen als transitiven Orientierungen auf ihren Farbklassen.

1.2.7 Implikationsklassen als Bausteine transitiver Orientierungen

Hat eine Implikationsklasse $I \in \mathcal{I}_G$ einen nichtleeren Schnitt mit einer transitiven Orientierung $T \in \mathcal{T}_G$, so gilt, dass I dann vollständig zu T gehört, $I \subseteq T$. Der Grund hierfür ist, dass jede Kante $e \in I$ mit jeder Kante $e^* \in I \cap T$ aus dem Schnitt Γ -verbunden ist. Die Orientierung von \hat{e} wird eindeutig festgelegt durch die Orientierung von $\widehat{e^*}$ (siehe hierzu z.B. auch SIMON [60]).

Ist der Schnitt $T \cap I$ andererseits leer, so gilt $T \cap I^{-1} \neq \emptyset$, da T ja eine Orientierung ist. Damit gilt für jede transitive Orientierung $T \in \mathcal{T}_G$ und jede Implikationsklasse $I \in \mathcal{I}_G$ entweder $T \cap \hat{I} = I$ oder $T \cap \hat{I} = I^{-1}$.

Ein Spezialfall hierbei ist die Situation, dass eine Implikationsklasse $I \in \mathcal{I}_G$ genau eine *Orientierung* von G liefert. Dann gilt $I + I^{-1} = E$ und $I \cap I^{-1} = \emptyset$. Der Graph G ist in diesem Fall eindeutig transitiv orientierbar, also prim.

Satz 1.17 (Implikationsklassen und transitive Orientierungen) [33] *Sei $I \in \mathcal{I}_G$ eine Implikationsklasse eines Graphen $G = (V, E)$. Hat G eine transitive Orientierung T , dann gilt entweder $T \cap \hat{I} = I$ oder $T \cap \hat{I} = I^{-1}$, und in jedem Fall $I \cap I^{-1} = \emptyset$.*

Eine frühe Variante eines Teils dieser Aussage findet sich auch in diesem Fall bereits bei GILMORE UND HOFFMAN (1964) [30].

Aus Satz 1.17 folgt, dass jede transitive Orientierung T auf einem Graphen $G = (V, E)$ ausschließlich echte Implikationsklassen enthält. Diese Eigenschaft liefert eine Charakterisierung von Comparabilitygraphen: Ein Graph G ist genau dann ein Comparabilitygraph, wenn jede Implikationsklasse in G echt ist. Wir werden diesen Zusammenhang im Abschnitt 1.3.1 näher betrachten. Dort werden wir neben dieser auch noch einige andere Möglichkeiten der Charakterisierung von Comparabilitygraphen kennen lernen.

Kombination von Implikationsklassen

Nach Satz 1.16 gilt für jede Farbklasse \hat{I} , dass sie entweder nicht, oder aber eindeutig transitiv orientierbar ist. Im ersten Fall gilt $I = I^{-1} = \hat{I}$, und im zweiten Fall sind I und I^{-1} die einzigen transitiven Orientierungen. Zusammen mit Satz 1.17 ergibt sich daraus, dass jede transitive Orientierung T aus Orientierungen von Farbklassen, und damit aus Implikationsklassen, zusammengesetzt ist.

Wir können daher Implikationsklassen als fundamentale, nicht-spaltbare Bausteine oder Atome für transitive Orientierungen auffassen.

Folgerung 1.18 (Bausteine transitiver Orientierungen) *Sei $G = (V, E)$ ein Comparabilitygraph, und sei $\mathcal{F}_G = \{\hat{I}_1, \dots, \hat{I}_k\}$ die Menge seiner Farbklassen. Jede Farbklasse ist eindeutig transitiv orientierbar, und jede transitive Orientierung $T \in \mathcal{T}_G$ setzt sich aus Orientierungen auf den Farbklassen zusammen, $T = J_1 + \dots + J_k$ mit $J_i \in \{I_i, I_i^{-1}\}$ für alle $i = 1, \dots, k$.*

Andererseits liefert nicht jede beliebige Kombination von Orientierungen auf den Farbklassen eine transitive Orientierung von G . Als Gegenbeispiel betrachten wir ein Dreieck. Jede der drei Kanten bildet eine eigene Farbklasse, so dass sich $2^3 = 8$ mögliche Orientierungen ergeben. Von diesen sind jedoch nur 6 transitiv. Die anderen beiden bilden jeweils einen Kreis.

Potentiell transitive Orientierungen

Wir nennen eine beliebige Kombination $T^* = J_1 + \dots + J_k$ von Orientierungen $J_i \in \{I_i, I_i^{-1}\}$ auf den Farbklassen eines Graphen $G = (V, E)$ eine *potentiell transitive Orientierung*. Eine solche Kombination heißt *zulässig*, wenn sie zu einer transitiven Orientierung von G führt. Wenn G ein Comparabilitygraph ist, dann hängt die Antwort auf die Frage, ob T^* eine zulässige Kombination ist, lediglich davon ab, ob T^* Kreise enthält.

Satz 1.19 (potentiell transitive Orientierungen) *Sei $\mathcal{F}_G = \{\hat{I}_1, \dots, \hat{I}_k\}$ die Menge der Farbklassen eines Comparabilitygraphen $G = (V, E)$. Eine potentiell transitive Orientierung $T^* = J_1 + \dots + J_k$ mit $J_i \in \{I_i, I_i^{-1}\}$ ($i = 1, \dots, k$) ist genau dann transitiv, wenn sie kreisfrei ist.*

Beweis. Da G ein Comparabilitygraph ist, ist jede Implikationsklasse I_i ($i = 1, \dots, k$) nach Satz 1.17 sowohl echt als auch transitiv. Da Implikationsklassen Äquivalenzklassen auf E sind, ist T^* eine Orientierung von G . Da transitive Orientierungen azyklisch sind, bleibt lediglich zu zeigen, dass die Kreisfreiheit auch hinreichend für die Transitivität von T^* ist. Sei also T^* kreisfrei und nicht transitiv. D.h. es existieren $ab, bc \in T^*$ mit $ac \notin T^*$. Da alle Implikationsklassen transitiv sind, können ab und bc nicht in der gleichen Implikationsklasse liegen. Sei $I_1 = I(ab)$ und $I_2 = I(bc)$, mit $\hat{I}_1 \neq \hat{I}_2$. Da ab

und bc^{-1} andererseits auch nicht in Γ -Relation liegen ($I_1 \neq I_2^{-1}$), muss die Kante $\widehat{ac} \in E$ existieren, und damit $ca \in T^*$ gelten, da T^* eine Orientierung ist ($ac \notin T^*$). Damit enthält T^* jedoch einen Kreis. ■

Rekombination von transitiven Orientierungen

Jede transitive Orientierung eines Comparabilitygraphen $G = (V, E)$ ist also eine zulässige Kombination von transitiven Orientierungen auf seinen Farbklassen. Hat G mehrere verschiedene transitive Orientierungen, $|\mathcal{T}_G| > 2$, so können wir jede dieser Orientierungen aus jeder anderen durch Umkehren der Orientierungen auf einem Teil der Farbklassen erzeugen.

Folgerung 1.20 (Rekombination der Implikationsklassen) Sei $T = I_1 + \dots + I_k \in \mathcal{T}_G$ eine transitive Orientierung von $G = (V, E)$, und sei $M = \{i_1, \dots, i_m\} \subseteq \{1, \dots, k\}$. Wir nennen die Orientierung $T_M = T - I_{i_1} - \dots - I_{i_m} + I_{i_1}^{-1} + \dots + I_{i_m}^{-1}$, die durch Umkehren der Orientierung auf allen \hat{I}_{i_j} , $i_j \in M$, entsteht, eine Rekombination der Implikationsklassen von T . T_M ist genau dann zulässig, d.h. transitiv, wenn sie kreisfrei ist.

1.2.8 Anzahl transitiver Orientierungen

In diesem Teilabschnitt werden wir die Interaktionen zwischen Implikationsklassen untersuchen, und insbesondere die Anzahl der transitiven Orientierungen eines Comparabilitygraphen $G = (V, E)$ näher betrachten. Wir wissen bereits, dass diese Anzahl nach oben exponentiell durch die Anzahl der verschiedenen Farbklassen von G beschränkt ist. Hat G genau k verschiedene Farbklassen, $\mathcal{F}_G = \{\hat{I}_1, \dots, \hat{I}_k\}$, so kann G maximal 2^k transitive Orientierungen besitzen. In diesem Fall wäre jede beliebige Kombination der Orientierungen auf den Farbklassen kreisfrei. Jede potentiell transitive Orientierung wäre tatsächlich transitiv. Ein Beispiel für einen solchen Graphen ist in Abbildung 1.3 dargestellt.

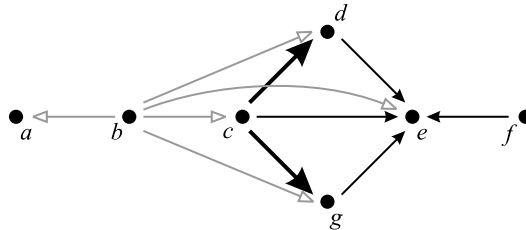


Abbildung 1.3: Der Graph $G = (V, E)$ besitzt 3 verschiedene echte Implikationsklassen (Darstellung durch verschiedene Kantensorten), die völlig unabhängig voneinander orientiert werden können, so dass jede der $2^3 = 8$ möglichen Kombinationen zu einer transitiven Orientierung auf G führt.

Andererseits haben wir bereits gesehen, dass es durchaus denkbar ist, dass einige Kombinationen von Implikationsklassen Kreise enthalten.

Um zu einer Beschreibung der Anzahl transitiver Orientierungen eines Comparabilitygraphen zu gelangen, müssen wir zunächst die möglichen Interaktionen zwischen den Implikationsklassen verstehen. Die Darstellung dieser strukturellen Zusammenhänge geht auf GOLUMBIC (1977a) [31] zurück (siehe auch GOLUMBIC (1980) [33]).

Zur Darstellung dieses Resultats (Satz 1.21) ist es notwendig, vorher noch ein paar Begriffe einzuführen. Diese werden für die weiteren Betrachtungen dieser Arbeit keine weitere Rolle spielen.

Simplex und Multiplex

Sei $G = (V, E)$ ein ungerichteter Graph. Ein vollständiger Teilgraph (V_S, S) auf $r + 1$ Knoten in G , bei dem jede ungerichtete Kante $\widehat{ab} \in S$ zu einer anderen Farbklassse von G gehört, heißt ein *Simplex* vom Rang r , oder auch *r-Simplex*. Jede ungerichtete Kante $\widehat{ab} \in E$ in G bildet beispielsweise ein Simplex vom Rang 1. Jedes r -Simplex enthält genau $\frac{1}{2}r(r + 1)$ ungerichtete Kanten aus $\frac{1}{2}r(r + 1)$

verschiedenen Farbklassen. Wir nennen ein Simplex *maximal*, wenn es nicht echt in einem größeren Simplex enthalten ist.

Das durch das r -Simplex S induzierte *Multiplex* (V_M, M) ist durch die folgende Kantenmenge definiert:

$$M = \bigcup_{I \cap S \neq \emptyset} \hat{I} = \{ab \in E : ab\Gamma^*xy \text{ für ein } xy \in S\}.$$

M ist also die Vereinigung aller $\frac{1}{2}r(r+1)$ in S enthaltenen Farbklassen. Jedes r -Simplex S induziert ein Multiplex M vom Rang r . Umgekehrt jedoch kann ein r -Multiplex M durch verschiedene r -Simplizes S_1 und S_2 erzeugt werden. In diesem Fall sind S_1 und S_2 isomorph. Ein Multiplex heißt *maximal*, wenn es nicht echt in einem größeren enthalten ist.

Ein maximales Multiplex wird durch ein oder mehrere maximale Simplizes generiert. Für zwei maximale Multiplizes M_1 und M_2 in einem Graphen G gilt entweder $M_1 \cap M_2 = \emptyset$ oder $M_1 = M_2$. Da die Kantenmenge E eines beliebigen Graphen $G = (V, E)$ eine Zerlegung in ihre Farbklassen besitzt, und ein Multiplex nichts weiter als die Vereinigung von verschiedenen Farbklassen ist, folgt damit, dass $E = M_1 + \dots + M_k$ auch eine eindeutige Zerlegung in die maximalen Multiplizes M_1, \dots, M_t von G hat (*M-Zerlegung*, GOLUMBIC [31]).

Jedes Simplex S vom Rang r eines Graphen G hat genau $(r+1)!$ transitive Orientierungen, und jede dieser transitiven Orientierungen ist auch transitiv auf dem durch S induzierten Multiplex M , sofern G ein Comparabilitygraph ist. D.h. jedes r -Multiplex M eines Comparabilitygraphen hat genau $(r+1)!$ transitive Orientierungen (GOLUMBIC [31], [33]).

Struktursatz von Golumbic

Der folgende Struktursatz von Golumbic fasst schließlich diese Zusammenhänge zusammen, und besagt, dass es eine eins-zu-eins-Beziehung zwischen den transitiven Orientierungen auf den maximalen Multiplizes von $G = (V, E)$ und denen von G gibt. Auch wenn wir hier nicht näher auf den Beweis dieser Aussage eingehen können, sei erwähnt, dass das Dreieckslemma (Satz 1.15) hierbei von fundamentaler Bedeutung ist. Mit $t(G)$ sei die Anzahl der transitiven Orientierungen von G bezeichnet, $t(G) = |\mathcal{T}_G|$.

Satz 1.21 (Anzahl transitiver Orientierungen) [31] *Sei $G = (V, E)$ ein Graph, und sei $E = M_1 + \dots + M_t$ die Zerlegung von E in seine maximalen Multiplizes.*

- (i) *Ist T eine transitive Orientierung von G , dann ist $T \cap M_i$ eine transitive Orientierung von M_i .*
- (ii) *Sind T_1, \dots, T_t jeweils transitive Orientierungen auf M_1, \dots, M_t , dann ist $T_1 + \dots + T_t$ eine transitive Orientierung auf G .*
- (iii) *$t(G) = t(M_1) \cdot t(M_2) \cdot \dots \cdot t(M_t)$.*
- (iv) *Ist G ein Comparabilitygraph und r_i der Rang von M_i , dann ist $t(G) = \prod_{i=1}^t (r_i + 1)!$.*

Anwendung auf potentiell transitive Orientierungen

Also überträgt sich jede beliebige transitive Orientierung auf einem maximalen Simplex auf das induzierte Multiplex, und jede transitive Orientierung auf einem Multiplex ist verträglich mit jeder anderen transitiven Orientierung auf jedem anderen Multiplex. Insbesondere entstehen also bei der Vereinigung von transitiven Orientierungen auf den Multiplizes M_i keine Kreise. Das bedeutet, dass eine beliebige Kombination von Orientierungen auf den Farbklassen eines Comparabilitygraphen nur dann einen Kreis enthalten kann, wenn dieser in einem maximalen Simplex liegt. D.h. jede Kombination, die in jedem maximalen Simplex kreisfrei ist, ist zulässig. Eine solche Kombination ist dann auch in jedem Simplex kreisfrei.

Satz 1.22 (potentiell transitive Orientierungen) *Sei $G = (V, E)$ ein Comparabilitygraph, und sei $\mathcal{I}_G = \{I_1, \dots, I_k, I_1^{-1}, \dots, I_k^{-1}\}$ die Menge der Implikationsklassen. Sei ferner $(V_{S_1}, S_1), \dots, (V_{S_l}, S_l)$ die Menge der verschiedenen maximalen Simplizes von G . Eine potentiell transitive Orientierung $T^* = J_1 + \dots + J_k$ mit $J_i \in \{I_i, I_i^{-1}\}$ für alle $i = 1, \dots, k$ ist genau dann transitiv, wenn sie in keinem maximalen Simplex von G einen Kreis enthält, d.h. wenn $T^* \cap S_j$ kreisfrei ist für alle $j = 1, \dots, l$.*

Beweis. Klar ist, dass $T^* \cap S_j$ kreisfrei für alle $j = 1, \dots, l$ ist, falls T^* transitiv ist. Es bleibt zu zeigen, dass die Kreisfreiheit von $T^* \cap S_j$ für alle j die Transitivität von T^* nach sich zieht. Angenommen, T^* ist nicht transitiv. Dann existieren in T^* zwei Kanten $ab, bc \in T^*$ mit entweder $\widehat{ac} \notin E$ oder $ca \in T^*$. Sei $A = I(ab)$ und $B = I(bc)$. Im ersten Fall ($\widehat{ac} \notin E$) würde $A = B^{-1}$ gelten, und $T^* = A + B + \dots$ damit eine unechte Implikationsklasse enthalten, $B, B^{-1} \subseteq T^*$ — im Widerspruch zur Voraussetzung (G ist Comparabilitygraph). Also ist $ca \in T^*$, und A und B sind verschieden, $\hat{A} \neq \hat{B}$. Sei $C = I(ca)$. Nach Annahme ist das Dreieck $\triangle abc$ kein 2-Simplex, da es nicht kreisfrei ist, d.h. $\hat{C} = \hat{A}$ oder $\hat{C} = \hat{B}$. Damit gilt entweder $A = C$ oder $B = C$. Im ersten Fall folgt $A = B^{-1}$ (Widerspruch) aus der Transitivität von A ($cb \in A$, da $ca \Gamma^* ab$) (Satz 1.16). Im zweiten Fall folgt analog $B = A^{-1}$ (Widerspruch) ($ba \in B$, da $bc \Gamma^* ca$). Damit muss T^* transitiv sein. ■

Ein Simplex vom Rang 2 wird naheliegenderweise als ein *3-farbiges Dreieck* bezeichnet. Der Graph $G = (V, E)$ in Abbildung 1.3 enthält kein solches 3-farbiges Dreieck, und damit auch kein Simplex höheren Rangs. Jedes Simplex in G hat den Rang 1 und besteht nur aus einer einzelnen Kante. Die Zerlegung von E in seine maximalen Multiplizes ist für G identisch mit der Zerlegung in seine Farbklassen, $E = \hat{I}_1 + \hat{I}_2 + \hat{I}_3$. Nach Satz 1.21 gilt damit $t(G) = 2 \cdot 2 \cdot 2 = 8$.

Folgerung 1.23 (3-farbige Dreiecke) Sei $G = (V, E)$ ein Comparabilitygraph, und sei $\mathcal{I}_G = \{I_1, \dots, I_k, I_1^{-1}, \dots, I_k^{-1}\}$ die Menge der Implikationsklassen. Enthält G keine 3-farbigen Dreiecke (Simplizes vom Rang 2), so ist jede potentiell transitive Orientierung tatsächlich transitiv, $t(G) = 2^k$.

Schranken für die Anzahl transitiver Orientierungen

Aus Golumbic's Struktursatz folgt ferner, dass die Anzahl transitiver Orientierungen auf einem Comparabilitygraphen mit k verschiedenen Farbklassen nicht nur nach oben (durch 2^k) beschränkt ist, sondern auch nach unten. Ein Comparabilitygraph mit k Farbklassen kann nicht beliebig viele zyklische Kombinationen von Orientierungen auf den Farbklassen enthalten. Anders gesagt: die Anzahl der unzulässigen Kombinationen von Orientierungen auf den Farbklassen ist nach oben beschränkt.

Nach Satz 1.21 und der Definition der Multiplizes ist für die Betrachtung von $t(G)$ die Zerlegung von E in seine maximalen Multiplizes äquivalent zu der Auswahl einer Menge maximaler Simplizes, die alle Farbklassen abdeckt. Ein Simplex vom Rang r hat $\frac{1}{2}r(r+1)$ ungerichtete Kanten aus verschiedenen Farbklassen. Damit gibt es $2^{\frac{1}{2}r(r+1)}$ potentiell transitive Orientierungen auf diesem r -Simplex. Von diesen sind nach Satz 1.21 $(r+1)!$ kreisfrei. Für wachsendes r gilt damit, dass der Anteil der kreisfreien potentiell transitiven Orientierungen an allen potentiell transitiven Orientierungen gegen Null strebt, $\lim_{r \rightarrow \infty} \frac{(r+1)!}{2^{\frac{1}{2}r(r+1)}} = 0$. Andererseits wächst die Anzahl dieser kreisfreien Orientierungen dennoch sehr schnell.

Ein Comparabilitygraph $G = (V, E)$ mit k verschiedenen Farbklassen hat daher besonders wenig transitive Orientierungen (geringer Anteil kreisfreier potentiell transitiver Orientierungen), wenn diese k Farbklassen eine Zerlegung in maximale Simplizes mit besonders großen Rängen haben. Die maximale Größe eines r -Simplexes in G ist dabei durch k begrenzt, da $\frac{1}{2}r_{\max}(r_{\max}+1) \leq k$ gelten muss.

Es ist folglich $r_{\max}(k) = \left\lfloor -\frac{1}{2} + \sqrt{\frac{1}{4} + 2k} \right\rfloor$. Von den k Farbklassen sind dann genau $\frac{1}{2}r_{\max}(r_{\max}+1)$ in einem maximalen r -Simplex gebunden. Für die verbleibenden $k - \frac{1}{2}r_{\max}(k)(r_{\max}(k)+1)$ Kanten können wir nun erneut die maximale Größe eines r -Simplexes bestimmen. Auf diese Weise entsteht eine Zerlegung von k in Summanden der Form $q_i \left(\frac{1}{2}i(i+1)\right)$ mit $q_i \in \mathcal{N}_0$, $i = 1, \dots, r_{\max}(k)$. Der Faktor q_i gibt dabei die Häufigkeit an, mit der ein maximaler Simplex vom Rang i in dieser Zerlegung der Farbklassen von G vertreten ist. Die auf diese Weise erzeugte Zerlegung von k ist ein guter Kandidat für einen Comparabilitygraphen mit minimaler Anzahl von transitiven Orientierungen bei gegebener Anzahl der Farbklassen.

Andererseits ist es denkbar, dass von einem Comparabilitygraphen $G = (V, E)$ nicht nur die Anzahl der Farbklassen, k , bekannt ist, sondern auch die Größe r eines größten r -Simplexes. Die angegebene Zerlegung von k lässt sich auch in diesem Fall leicht bestimmen. Ist r nun kleiner als $r_{\max}(k)$, so besitzt G weniger sehr große maximale Simplizes, sondern mehr Simplizes von mittlerer Größe, und damit tendenziell weniger potentiell transitive Orientierungen, die unzulässig sind. Die Anzahl transitiver Orientierungen wird hier größer ausfallen, als bei $r = r_{\max}(k)$.

Sei $t(k)$ die minimale Anzahl transitiver Orientierungen, die ein Comparabilitygraph mit genau k Farbklassen hat, und $t(k, r)$ die minimale Anzahl transitiver Orientierungen bei genau k Farbklassen

und r als maximalen Rang eines Simplexes.

Lemma 1.24 ($t(G)$ ist submultiplikativ) Seien $k_1, k_2 \in \mathcal{N}$. Es gilt $t(k_1) \cdot t(k_2) \geq t(k_1 + k_2)$.

Beweis. Sind $G_1 = (V_1, E_1)$ und $G_2 = (V_2, E_2)$ zwei Comparabilitygraphen mit k_1 bzw. k_2 Farbklassen, dann ist $G = (V_1 + V_2, E_1 + E_2)$ ein Comparabilitygraph mit $k_1 + k_2$ Farbklassen und $t(G) = t(G_1) \cdot t(G_2)$.² Werden nun G_1 und G_2 so gewählt, dass $t(G_1) = t(k_1)$ und $t(G_2) = t(k_2)$, dann folgt $t(k_1 + k_2) \leq t(G) = t(k_1) \cdot t(k_2)$. ■

Satz 1.25 (Schranken für $t(G)$) Sei $G = (V, E)$ ein Comparabilitygraph mit k Farbklassen. Ist r der Rang eines größten maximalen Simplexes in G , dann gilt

$$(r+1)! \cdot 2^{k-\frac{1}{2}r(r+1)} \geq t(G) \geq t(k, r) = \min_{\substack{q \in \mathcal{N}_0^r \\ \sum_{i=1}^r q_i \frac{1}{2}i(i+1) = k}} \left\{ \prod_{i=1}^r ((i+1)!)^{q_i} \right\}.$$

Dabei gibt q_i die Anzahl der i -Multiplizes in der durch das gewählte $q \in \mathcal{N}_0^r$ repräsentierten Zerlegung von E wieder.

Beweis. Nach Satz 1.21 gilt $t(G) = \prod_{j=1}^l t(M_j)$, wobei $E = M_1 + \dots + M_l$ die Zerlegung in maximale Multiplizes ist. Enthält G einen r -Simplex, so sind dadurch $\frac{1}{2}r(r+1)$ Farbklassen gebunden. Eine maximale Anzahl transitiver Orientierungen ergibt sich, wenn alle verbleibenden $k - \frac{1}{2}r(r+1)$ Farbklassen jeweils Simplizes vom Rang 1 bilden. Mit Folgerung 1.23 folgt $t(G) \leq (r+1)! \cdot 2^{k-\frac{1}{2}r(r+1)}$.

Da jede Zerlegung von E in seine maximalen Multiplizes mit einer Zuordnung der Farbklassen zu maximalen Simplizes korrespondiert, hat jede solche Zerlegung eine Darstellung von k als Summe von Summanden der Form $q_i \frac{1}{2}i(i+1)$. Dabei ist $\frac{1}{2}i(i+1)$ die Anzahl der Farbklassen, die durch ein i -Multiplex gebunden sind, und q_i die Häufigkeit, mit der ein Multiplex vom Rang i in der Zerlegung auftritt. Jede mögliche Zerlegung kann damit durch einen Vektor $q = (q_1, \dots, q_r) \in \mathcal{N}_0^r$, mit $\sum_{i=1}^r q_i \frac{1}{2}i(i+1) = k$, repräsentiert werden. Für eine gegebene Zerlegung q gilt nach Satz 1.21 $t(G) = \prod_{j=1}^l t(M_j) = \prod_{i=1}^r ((i+1)!)^{q_i}$. Da das Minimum dieses Ausdrucks über alle zulässigen q gebildet wird, ist auch die Zerlegung mit einer minimalen Anzahl von transitiven Orientierungen darunter. ■

Bemerkung 1.26 (Abschätzung für $t(G)$) In Satz 1.25 gilt vermutlich insbesondere die sehr grobe Abschätzung

$$t(G) \geq t(k) \geq (r_{\max}(k) + 1)! = \left(\left\lfloor -\frac{1}{2} + \sqrt{\frac{1}{4} + 2k} \right\rfloor + 1 \right)!.$$

Satz 1.25 besagt, dass die Anzahl der transitiven Orientierungen in einem Graphen $G = (V, E)$ mit genau k Farbklassen durch die Aufteilung der Farbklassen auf verschiedene Simplizes bestimmt ist. Daraus geht jedoch nicht hervor, ob zu jeder denkbaren Aufteilung von Farbklassen auf verschiedene Simplizes auch Graphen existieren, die diese Aufteilungen annehmen.

Lemma 1.27 (Zusammensetzen von Simplizes) Seien (V_{S_1}, S_1) und (V_{S_2}, S_2) zwei Simplizes mit Rängen r_1 und r_2 . Dann existiert ein zusammenhängender Graph $G = (V, E)$ mit $V_{S_1}, V_{S_2} \subseteq V$ und $S_1, S_2 \subseteq E$, dessen Kantenmenge in zwei Multiplizes M_1 und M_2 zerfällt, die durch S_1 und S_2 induziert werden. Es gilt dann $t(G) = t(S_1) \cdot t(S_2) = (r_1 + 1)! \cdot (r_2 + 1)!$.

Beweis. Sei $V_{S_1} = \{v_1, \dots, v_n\}$ und $V_{S_2} = \{w_1, \dots, w_m\}$, mit $n = r_1 + 1$ und $m = r_2 + 1$. Wir konstruieren $G = (V, E)$ folgendermaßen: Sei $V = V_{S_1} + V_{S_2} + x$, und sei $E = S_1 + S_2 + E_1 + E_2 + \dots + E_n$. Dabei sei $E_1 = \{\widehat{v_1 w_j} : j = 2, \dots, m\}$ und $E_i = \{\widehat{v_i w_j} : j = 1, \dots, m\} + \widehat{v_i x}$ ($i = 2, \dots, n$). D.h. der Knoten $v_1 \in V_{S_1}$ wird mit allen Knoten aus $V_{S_2} - w_1$ verbunden, und jeder weitere Knoten aus V_{S_1} wird sowohl mit *allen* Knoten aus V_{S_2} , als auch mit dem neuen Knoten x verbunden. (Es gilt $|E_1| = 2(m-1)$ und $|E_i| = 2(m+1)$ ($i = 2, \dots, n$), und damit insgesamt $|E| = 2(\frac{1}{2}r_1(r_1+1) + \frac{1}{2}r_2(r_2+1)) + 2r_2 + 2r_1(r_2+2)$.)

²Der Graph G kann auch als zusammenhängender Graph konstruiert werden (siehe Lemma 1.27). Für den Beweis ist das jedoch nicht ausschlaggebend.

Wegen $\widehat{v_1 w_1} \notin E$ liegt jede Kante $v_1 w_j$ aus E_1 in direkter Γ -Relation zu $w_1 w_j$, $v_1 w_j \Gamma w_1 w_j$ ($j = 2, \dots, m$). D.h. eine Kante $\widehat{v_1 w_j}$ aus E_1 liegt jeweils in derselben Farbklasse wie $\widehat{w_1 w_j}$. (Da (V_{S_1}, S_1) ein Simplex ist, gehört damit jede der $m - 1$ Kanten aus E_1 einer anderen Farbklasse an. Andererseits ist jedoch nicht jede der $\frac{1}{2}r_1(r_1 + 1)$ Farbklassen aus S_1 auch in E_1 vertreten!) Für die Kanten aus einem E_i mit $i = 2, \dots, n$ gilt, dass sie jeweils *alle* in der Farbklasse von $\widehat{v_i v_1}$ liegen: wegen $\widehat{v_1 x} \notin E$ gilt $v_i x \Gamma v_i v_1$, und wegen $\widehat{x w_j} \notin E$ gilt $v_i w_j \Gamma v_i x$ für alle $j = 1, \dots, m$. (Auch hier gilt, dass nicht jede der $\frac{1}{2}r_2(r_2 + 1)$ Farbklassen von S_2 auch in einem der E_i vertreten ist.)

Durch die Konstruktion ist ausgeschlossen, dass sich zusätzliche Simplizes ergeben. Damit ist G ein zusammenhängender Graph mit genau $\frac{1}{2}r_1(r_1 + 1) + \frac{1}{2}r_2(r_2 + 1)$ verschiedenen Farbklassen, für den $t(G) = t(S_1) \cdot t(S_2) = (r_1 + 1)! \cdot (r_2 + 1)!$ gilt. ■

Folgerung 1.28 (jede Zerlegung von k tritt auf) Sei durch $q \in \mathcal{N}_0^r$ mit $\sum_{i=1}^r q_i \frac{1}{2}i(i+1) = k$ eine Zerlegung von k verschiedenen Farbklassen in maximale i -Simplizes ($i = 1, \dots, r$) gegeben. Dann existiert ein zusammenhängender Comparabilitygraph $G = (V, E)$, dessen Zerlegung von E in seine maximalen Multiplizes dieser Darstellung von k genügt. Für $t(G)$ gilt dann $t(G) = \prod_{i=1}^r ((i+1)!)^{q_i}$. Insbesondere existiert ein Comparabilitygraph mit $r = r_{\max}(k) = \left\lfloor -\frac{1}{2} + \sqrt{\frac{1}{4} + 2k} \right\rfloor$.

Beweis. Nach Lemma 1.27 können zwei beliebige Simplizes so zu einem zusammenhängendem Comparabilitygraphen $G = (V, E)$ verknüpft werden, dass die maximalen Simplizes von G genau diesen beiden entsprechen. Diese Konstruktion kann nacheinander auf beliebig viele Simplizes angewendet werden. D.h. jede beliebige zulässige Zerlegung von k kann auftreten. Mit Satz 1.21 folgt dann $t(G) = \prod_{i=1}^r ((i+1)!)^{q_i}$. ■

Minimale Anzahl transitiver Orientierungen

Aus Satz 1.25 lässt sich ableiten, dass die minimale Anzahl transitiver Orientierungen in einem Graphen mit genau k Farbklassen erreicht wird, wenn die Farbklassen auf Simplizes von möglichst großem Rang verteilt werden. Ein Simplex mit großem Rang induziert bei gegebener Anzahl der verwendeten Farbklassen mehr zyklische potentielle transitive Orientierungen, und damit weniger transitive Orientierungen, als jede Aufteilung dieser Farbklassen in kleinere Simplizes.

Vermutung 1.29 (minimale Anzahl transitiver Orientierungen) Sei k die Anzahl der Farbklassen in einem Comparabilitygraphen, und sei $r = r_{\max} = \left\lfloor -\frac{1}{2} + \sqrt{\frac{1}{4} + 2k} \right\rfloor$ der größtmögliche Rang eines Simplexes. Seien ferner k_i und q_i für $i = r, r-1, \dots, 1$ folgendermaßen rekursiv bestimmt: Sei $k_r = k$ und $k_{i-1} = k_i \bmod q_i$ ($i = r, \dots, 2$), und sei $q_i = \left\lfloor \frac{k_i}{\frac{1}{2}i(i+1)} \right\rfloor$ ($i = r, \dots, 1$). Dann ist die minimale Anzahl transitiver Orientierungen eines Comparabilitygraphen mit k Farbklassen gegeben durch

$$t(k) = \begin{cases} \prod_{i=1}^r ((i+1)!)^{q_i} & , \quad k \notin \mathcal{K} \\ t(k - k^*) t(k^*) & , \quad k \in \mathcal{K} \setminus \{20\} \\ 14.400 & , \quad k = 20 \end{cases} .$$

\mathcal{K} beschreibt dabei die Menge der natürlichen Zahlen mit $k = \frac{1}{2}r_{\max}(k)(r_{\max}(k) + 1) + k^*$ für ein $k^* \in \mathcal{K}$ mit $k^* < k$. Das kleinste Element in \mathcal{K} ist 20.

Bemerkung 1.30 (minimale Anzahl transitiver Orientierungen) Sei r der Rang eines größten maximalen Simplexes in einem Comparabilitygraphen mit $k \notin \mathcal{K}$ Farbklassen. Nach Vermutung 1.29 hat die Kantenmenge E eines Comparabilitygraphen $G = (V, E)$ mit minimaler Anzahl transitiver Orientierungen eine Zerlegung

$$E = (M_1^r + \dots + M_{q_r}^r) + \dots + (M_1^1 + M_2^1 + \dots + M_{q_1}^1)$$

in jeweils größtmögliche maximale i -Multiplizes M_j^i .

r	2	3	4	5	6	7	
$\frac{1}{2}r(r+1)$	3	6	10	15	21	28	
$(r+1)!$	6	24	120	720	5.040	40.320	
k	2^k						
5	32	36	24				
		48	24				
10	1.024	432	288	120			
		768	384	120			
15	32.768	7.776	3.456	2.880	720		
		24.576	12.288	3.840	720		
20	1.048.576	186.624	55.296	14.400	17.280		
		786.432	393.216	122.880	23.040		
25	33.554.432	3.359.232	663.552	345.600	86.400	60.480	
		25.165.824	12.582.912	3.932.160	737.280	80.640	
30	1.073.741.824	60.466.176	7.962.624	1.728.000	518.400	725.760	161.280
		805.306.368	402.653.184	125.829.120	23.592.960	2.580.480	161.280
35	34.359.738.368	1.451.188.224	191.102.976	41.472.000	12.441.600	7.257.600	1.935.360
		25.769.803.776	12.884.901.888	4.026.531.840	754.974.720	82.575.360	5.160.960

Tabelle 1.1: Untere Schranke (obere Zeile) und obere Schranke (untere Zeile) für die Anzahl der transitiven Orientierungen auf einem Graphen mit genau k verschiedenen Farbklassen und einem maximalen Multiplex vom Rang r .

1.3 Charakterisierung von Comparabilitygraphen

Wir haben Comparabilitygraphen als Graphen definiert, die transitive Orientierungen besitzen. Daneben gibt es noch eine Reihe weiterer Möglichkeiten zur Charakterisierung von Comparabilitygraphen. In diesem Abschnitt wollen wir die wichtigsten dieser Charakterisierungen kurz betrachten.

Die für unsere Zwecke bedeutsamste Charakterisierung wird die bereits erwähnte Charakterisierung durch die Echtheit aller Implikationsklassen bleiben (Abschnitt 1.3.1). Die Erwähnung von alternativen Varianten soll lediglich die Vielfalt der Möglichkeiten bei der Beschreibung von transitiven Orientierungen aufzeigen.

1.3.1 Echte Implikationsklassen

Die zentrale Charakterisierung von Comparabilitygraphen geht auf GOLUMBIC (1977a) [31] zurück (siehe auch GOLUMBIC (1980) [33]).

Satz 1.31 (Charakt. durch echte Implikationsklassen) [31] *Ein Graph $G = (V, E)$ ist genau dann ein Comparabilitygraph, wenn jede seiner Implikationsklassen $I \in \mathcal{I}_G$ echt ist, $I \cap I^{-1} = \emptyset$.*

Ist $E = I_1 + \dots + I_l$ die Zerlegung der Kantenmenge eines Graphen $G = (V, E)$ in Implikationsklassen, dann ist G genau dann ein Comparabilitygraph, wenn $\mathcal{I}_G = \{I_1, \dots, I_k, I_1^{-1}, \dots, I_k^{-1}\}$ und $I_i \neq I_i^{-1}$ für alle $i = 1, \dots, k = \frac{l}{2}$ gilt. Der Γ -Graph von G besteht dann aus einer geraden Anzahl $l = 2k$ von Komponenten, und für jeden Knoten $e \in E$ in $G_\Gamma = (E, \Gamma)$ gilt, dass der Knoten e^{-1} in einer anderen Komponente von G_Γ liegt.

Folgerung 1.32 (Charakt. durch konsistenten Γ -Graphen) Ein Graph $G = (V, E)$ ist genau dann ein Comparabilitygraph, wenn sein Γ -Graph G_Γ konsistent ist.

GOLUMBIC (1977) [31], [32] beschreibt einen Algorithmus, der auf Grundlage dieser Charakterisierung in der Zeit $\mathcal{O}(\delta |E|)^3$ für einen Graphen $G = (V, E)$ entscheidet, ob G ein Comparabilitygraph ist, und ggf. eine transitive Orientierung von G bestimmt. Mit δ ist dabei die Maximalvalenz in G bezeichnet. Auf den bei GOLUMBIC [31], [32], [33] in diesem Zusammenhang benutzten Begriff der G -Zerlegung der Kantenmenge E wollen wir hier nicht näher eingehen. Es handelt sich dabei um eine algorithmische Darstellung der Bestimmung der Implikationsklassen eines Graphen.

Die Kantenmenge E in dem Graphen $G = (V, E)$ in Abbildung 1.4 besteht aus nur einer (unechten) Implikationsklasse $I = I^{-1} = E$. G ist damit kein Comparabilitygraph.

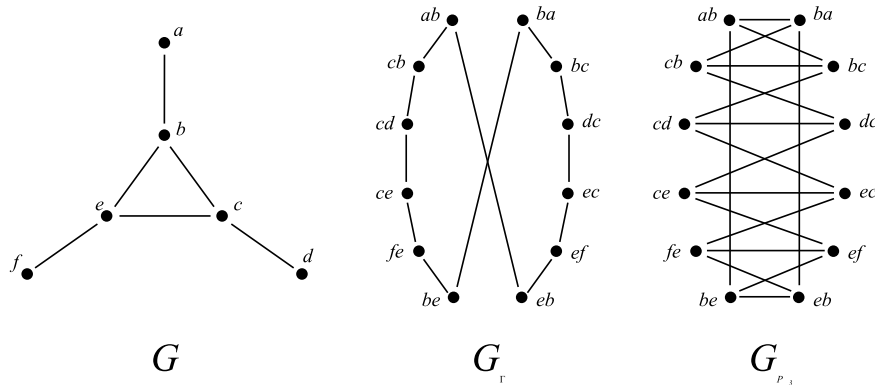


Abbildung 1.4: Der Γ -Graph und der Graph G_{P_3} zu einem Graphen G , der kein Comparabilitygraph ist.

1.3.2 Pseudotransitive Orientierungen

Eine transitive Orientierung ist eine Orientierung $A \subseteq E$ mit der Eigenschaft, dass die Zugehörigkeit der Kanten ab und bc zu A die Zugehörigkeit der transitiven Kante ac zu A impliziert. Eine Orientierung $A \subseteq E$ mit der Eigenschaft, dass $ab, bc \in A$ die Zugehörigkeit von entweder ac oder ca zu A impliziert, heißt *pseudo-* oder auch *quasitransitiv*.

Jede transitive Orientierung ist trivialerweise auch pseudotransitiv. Jeder Comparabilitygraph besitzt damit auch pseudotransitive Orientierungen. GHOUILA-HOURI (1962) [29] zeigte, dass hiervon auch die Umkehrung gilt.

Satz 1.33 (Charakt. durch pseudotransitive Orientierungen) [29] Ein Graph $G = (V, E)$ ist genau dann ein Comparabilitygraph, wenn er eine pseudotransitive Orientierung $Q \subseteq E$ besitzt.

Diese Charakterisierungen von Comparabilitygraphen ist eine der frühesten überhaupt. Ihr Beweis liefert konstruktiv eine transitive Orientierung T aus einer pseudotransitiven Orientierung Q (siehe z.B. auch SIMON (1992) [60]). Diese Konstruktion ist jedoch nicht unbedingt als Grundlage eines Algorithmusses zur Erkennung von Comparabilitygraphen geeignet, da ein konstruktiver Test auf Existenz einer pseudotransitiven Orientierung auf einem Graphen $G = (V, E)$ mindestens so schwierig ist, wie ein Test auf eine transitive Orientierung selbst.

1.3.3 Bipartiter Graph

Eine weitere sehr frühe Charakterisierung von Comparabilitygraphen geht ebenfalls auf GHOUILA-HOURI (1962) [29] zurück. Sie besagt, dass ein Graph $G = (V, E)$ genau dann eine transitive Orientierung besitzt, wenn ein von G abgeleiteter Hilfsgraph G_{P_3} bipartit ist. Der Hilfsgraph $G_{P_3} = (E, E_{P_3})$

³Die Beschreibung der Laufzeit von Algorithmen werden wir in Kapitel 2 näher betrachten.

wird dabei folgendermaßen definiert⁴: zwei Knoten $ab, cd \in E$ sind in G_{P_3} genau dann durch eine Kante verbunden, wenn in G entweder $b = c$ und $\widehat{ad} \notin E$, oder $a = d$ und $\widehat{bc} \notin E$ gilt. D.h. zwei Knoten sind G_{P_3} verbunden, wenn die Kanten, die sie repräsentieren in G einen gerichteten Weg der Knotenlänge 3 bilden, dessen transitive Kante nicht in G enthalten ist. Insbesondere ist jede Kante $ab \in E$ mit ihrer Umkehrkante $ba \in E$ in G_{P_3} verbunden. In Abbildung 1.4 ist ein Beispiel für den Graphen G_{P_3} zu einem Graphen $G = (V, E)$ gegeben.

Für G_{P_3} gilt genau wie für den Γ -Graphen, dass die Zusammenhangskomponenten die Implikationsklassen von G abbilden.

Satz 1.34 (Charakt. durch bipartiten Graphen) [29] *Ein Graph $G = (V, E)$ ist genau dann ein Comparabilitygraph, wenn der Graph G_{P_3} bipartit ist.*

Mit Hilfe dieser Charakterisierung lässt sich ein Comparabilitygraph $G = (V, E)$ in der Zeit $\mathcal{O}(\delta|E|)$ erkennen (siehe z.B. BANG-JENSEN UND GUTIN [7]), wobei auch hier δ die maximale Valenz in G bezeichnet.

1.3.4 Dreieckssehn

Eine der bedeutendsten Charakterisierungen von Comparabilitygraphen findet sich bei GILLMORE UND HOFFMAN (1964) [30].

Ein *nicht-einfacher Kreis* $K = x_0, x_1, \dots, x_{l-1}, x_0$ der Länge $l \geq 4$ in einem Graphen $G = (V, E)$ ist ein Kreis über die Knoten $x_0, \dots, x_l = x_0$, in dem Knoten mehrfach vorkommen dürfen. D.h. es wird nicht $x_i \neq x_j$ für alle $i, j = 0, \dots, l-1$ gefordert. Eine Kante $x_i x_{i+2} \in E$ in einem solchen Kreis heißt eine *Dreieckssehne*.

Für einen Comparabilitygraphen muss nun gelten, dass jeder ungerade Kreis mindestens eine Dreieckssehne enthalten muss.

Satz 1.35 (Charakt. durch Dreieckssehn) [30] *Ein Graph $G = (V, E)$ ist genau dann ein Comparabilitygraph, wenn er keine (nicht-einfachen) Kreise ungerader Länge ohne Dreieckssehne enthält.*

Der Graph G in Abbildung 1.4 enthält einen solchen ungeraden Kreis $K = a, b, c, d, c, e, f, e, b, a$ ohne Dreieckssehne, und ist damit kein Comparabilitygraph.

Für eine unmittelbare Anwendung in einem Algorithmus zur Identifizierung von Comparabilitygraphen ist diese Charakterisierung nicht geeignet. Sie spielt jedoch bei der Beschreibung von Eigenschaften transitiver Orientierungen eine herausragende Rolle.

1.3.5 Verbotene Teilgraphen

Der folgende Ansatz von GALLAI (1967) [27] bietet eine besonders aus theoretischer Sicht interessante Einsicht. Es handelt sich hierbei um eine Charakterisierung von Comparabilitygraphen durch eine *endliche* Liste von Graphenklassen, die ein Comparabilitygraph nicht enthalten darf.

Im Gegensatz zu allen anderen Alternativen zu der Echtheit von Implikationsklassen (Satz 1.31) werden wir auf diese Möglichkeit der Charakterisierung von Comparabilitygraphen noch einmal zurück kommen.

Wir bezeichnen die Menge der verbotenen Graphenklassen mit $\mathcal{C} = \mathcal{C}_1 + \mathcal{C}_2$. Die Graphen aus \mathcal{C}_1 sind in Abbildung 1.5 dargestellt, die Graphen aus \mathcal{C}_2 sind in Abbildung 1.6 und 1.7 durch ihre Komplemente dargestellt. Der Graph G aus Abbildung 1.4 entspricht dem Komplement des Graphen L_n für $n = 1$ aus Abbildung 1.6.

Satz 1.36 (Charakt. durch verbotene Teilgraphen) [27] *Ein Graph $G = (V, E)$ ist genau dann ein Comparabilitygraph, wenn er keinen induzierten Teilgraphen isomorph zu einem Graphen aus der Graphenfamilie $\mathcal{C} = \mathcal{C}_1 + \mathcal{C}_2$ (Abbildungen 1.5, 1.6 und 1.7) enthält.*

Die Familie \mathcal{C} der verbotenen Graphen enthält gewissermaßen genau die kleinsten Graphen, die keine Comparabilitygraphen sind. Während jeder Graph G aus \mathcal{C} selber kein Comparabilitygraph ist, ist jedoch jeder echte induzierte Teilgraph eines Graphen G aus \mathcal{C} ein Comparabilitygraph.

⁴Bei BANG-JENSEN UND GUTIN [7] findet sich für diesen Graphen die Bezeichnung G_{qtd} .

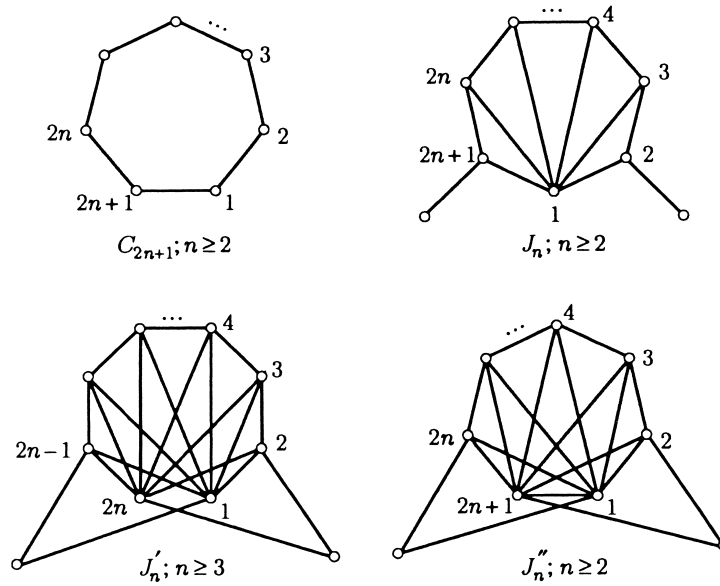


Abbildung 1.5: Liste \mathcal{C} - Teil 1: Die Graphen in \mathcal{C}_1 (TROTTER [62])

Der besondere Charme dieser Charakterisierung liegt darin, dass er die Möglichkeit eröffnet, zum Nachweis der Existenz von transitiven Orientierungen auf einer gegebenen Graphenstruktur, “nur” die Graphen in der Liste \mathcal{C} abzuarbeiten, und für jeden zu zeigen, dass er nicht als induzierter Teilgraph für die gegebene Struktur in Frage kommen kann.

1.3.6 Partiiell geordnete Mengen

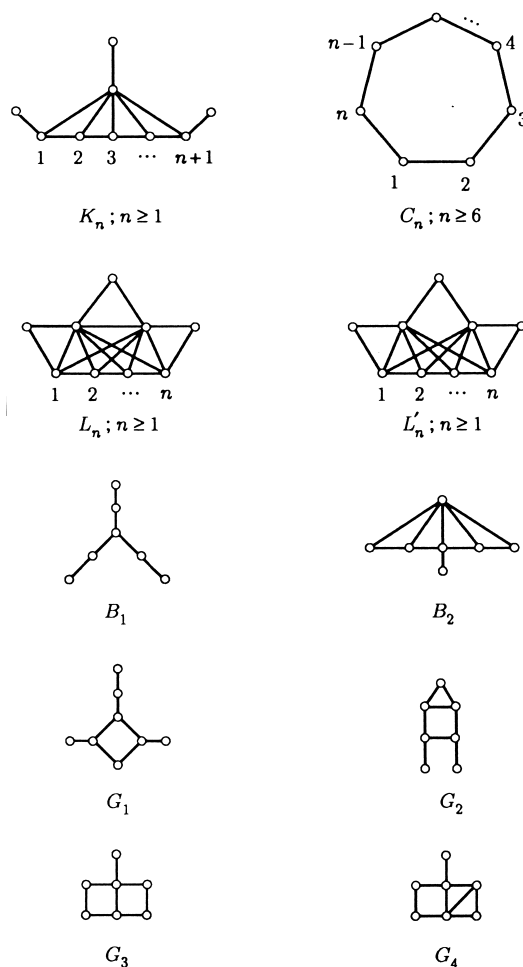
Abschließend wollen wir uns noch kurz der Auffassung von Comparabilitygraphen als Darstellung von partiellen Ordnungen widmen. Auf dieser Auffassung basiert der Begriff *Comparabilitygraph* für einen transitiv orientierbaren Graphen. Bei der Darstellung dieser Zusammenhänge orientieren wir uns an TROTTER (1992) [62]. Die Ursprünge der Theorie der geordneten Mengen gehen jedoch bereits auf DUSHNIK UND MILLER (1941) [25] zurück.

Eine *partiell geordnete Menge* (partially ordered set, poset) $\mathcal{P} = (X, P)$ besteht aus einer Grundmenge X und einer reflexiven, antisymmetrischen, und transitiven Relation $P \subseteq X \times X$. Eine Relation P mit diesen Eigenschaften heißt *partielle Ordnung* oder *Halbordnung*. Gilt für zwei Punkte $x, y \in X$ einer partiell geordneten Menge $\mathcal{P} = (X, P)$ entweder xPy oder yPx , so heißen sie *vergleichbar* (comparable). Andernfalls heißen sie *unvergleichbar* (incomparable).

Sind in einer partiellen Ordnung P alle Paare (x, y) vergleichbar, so nennt man P eine *lineare* oder *totale* Ordnung auf X . Jede partielle Ordnung P kann als Durchschnitt von verschiedenen linearen Ordnungen auf X dargestellt werden. Die minimale Anzahl der dazu notwendigen linearen Ordnungen definiert die *Dimension* von P .

Mit einer partiell geordneten Menge $\mathcal{P} = (X, P)$ können wir nun einen Vergleichbarkeits- oder *Comparabilitygraphen* $G(P) = (X, \hat{P}) = [P]$ assoziieren, der zwei verschiedene Punkte $x, y \in X$ genau dann miteinander verbindet, wenn sie bezüglich P vergleichbar sind. Ein solcher Graph, der eine partielle Ordnung abbildet, wird daher oft auch als *PO-Graph* bezeichnet. In $G(P)$ liefert $P \subseteq \hat{P}$ eine transitive Orientierung.

Satz 1.37 (Charakt. durch partielle Ordnungen) ([62]) *Ein Graph $G = (V, E)$ ist genau dann ein Comparabilitygraph, wenn es auf V eine partielle Ordnung P mit $\hat{P} = E$ gibt.*

Abbildung 1.6: Liste \mathcal{C} - Teil 2: Die Komplemente der Graphen in \mathcal{C}_2 (TROTTER [62])

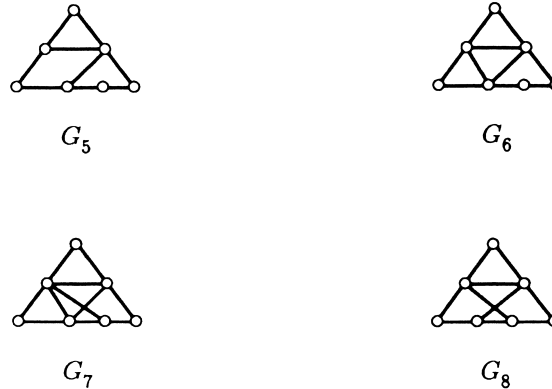
Damit können wir einen Comparabilitygraphen $G = (V, E)$ auch als Darstellung einer partiellen Ordnung auffassen. Hat G mehrere verschiedene transitive Orientierungen, so entspricht jede dieser Orientierungen einer anderen partiellen Ordnung auf der gleichen Grundmenge. Dabei induziert jede dieser verschiedenen partiellen Ordnungen den gleichen Comparabilitygraphen.

1.4 Weitere Eigenschaften von Comparabilitygraphen

In den vorherigen Abschnitten haben wir bereits viele Eigenschaften von Comparabilitygraphen und ihren transitiven Orientierungen kennen gelernt. In diesem Abschnitt wollen wir einige ausgewählte spezielle Eigenschaften dieser Graphenklasse nennen, die für unsere Zwecke relevant sind.

Von besonders großer Bedeutung sind die Umstände, unter denen wir Knoten und Kanten aus einem gegebenen Comparabilitygraphen so entfernen können, dass der verbleibende Teilgraph wieder ein Comparabilitygraph ist (Abschnitt 1.4.1). Insbesondere die Resultate zum Entfernen von Kanten bilden eine bedeutende Grundlage für die gesamte Arbeit.

Anschließend widmen wir uns kurz dem Konzept der modularen Dekomposition (Abschnitt 1.4.2), das vor allem aus algorithmischer Sicht eine große Rolle bei der Betrachtung von Comparabilitygraphen spielt. Danach wollen wir ein bekanntes asymptotisches Resultat zur Anzahl von transitiven Orientierungen zitieren (Abschnitt 1.4.3), und darlegen, dass Comparabilitygraphen sowohl perfekte, als auch superperfekte Graphen sind (Abschnitt 1.4.4).

Abbildung 1.7: Liste \mathcal{C} - Teil 2: Die Komplemente der Graphen in \mathcal{C}_2 (vervollständigt) (TROTTER [62])

1.4.1 Entfernen von Knoten und Kanten aus Comparabilitygraphen

Sei $G = (V, E)$ ein Comparabilitygraph. Seien ferner $v \in V$ ein Knoten und $e \in E$ eine Kante aus G . Die Frage, die wir uns nun stellen, lautet: Unter welchen Umständen sind die Graphen $G - v$ bzw. $G - e$ wieder Comparabilitygraphen?

Entfernen von Knoten

In Bezug auf das Entfernen von Knoten ist diese Frage schnell und einfach umfassend zu beantworten. Entfernt man aus einer transitiven Orientierung $T \in \mathcal{T}_G$ einen Knoten v und sämtliche zu v inzidierende Kanten, so ist die übrig bleibende Orientierung $T - v$ nach wie vor eine transitive Orientierung auf $G - v$ (siehe hierzu beispielsweise WILLENUS (2000) [66]).

Satz 1.38 (induzierter Teilgraph) [24],[66] *Jeder induzierte Teilgraph eines Comparabilitygraphen ist ein Comparabilitygraph.*

Diese Aussage folgt aus einem klassischen Theorem von DILLWORTH (1950) [24], und spiegelt wider, dass Comparabilitygraphen zu der Klasse der perfekten Graphen gehören. Auf diesen Begriff gehen wir im Abschnitt 1.4.4 noch näher ein.

Entfernen von Kanten aus transitiver Reduktion

Für Knoten eines Comparabilitygraphen gilt also, dass man sie in beliebiger Weise entfernen kann, ohne die Eigenschaft der transitiven Orientierbarkeit zu verlieren. Für Kanten gilt das nicht. Entfernt man beispielsweise aus einem Kreis der Länge 5 mit einer Dreieckssehne gerade diese Dreieckssehne, so ist der verbleibende Graph $G - e$ kein Comparabilitygraph mehr.

Das bedeutet jedoch nicht, dass aus einem Comparabilitygraphen keinerlei Kanten entfernt werden können. Im eben genannten Beispiel kann jede Kante außer der Dreieckssehne entfernt werden. Es gilt, dass in jedem Comparabilitygraphen mindestens eine Kante existiert, die entfernt werden kann, ohne die transitive Orientierbarkeit zu verletzen.

Satz 1.39 (Kante aus transitiver Reduktion) [66] *Sei $T \in \mathcal{T}_G$ eine transitive Orientierung auf $G = (V, E)$. Für eine Kante $e \in T$ ist $T - e$ genau dann eine transitive Orientierung auf $G - e$, wenn e eine nicht-transitive Kante in T ist, d.h. wenn $e \in T^r$ gilt.*

Der Satz 1.39 besagt, dass zu einer gegebenen transitiven Orientierung $T \in \mathcal{T}_G$ jede Kante aus der transitiven Reduktion T^r "gefährlos" gelöscht werden kann. Er besagt *nicht*, dass diese Kanten die einzigen Kanten sind, die aus G entfernt werden können, ohne die Eigenschaft der Comparabilität zu verlieren. Beispielsweise können auch solche Kanten entfernt werden, die zwar in T transitiv sind, aber bezüglich $T' \in \mathcal{T}_G$ zur transitiven Reduktion gehören.

Immer-transitive Kanten

Darüberhinaus trifft der Satz keine Aussage über Kanten, die in *jeder* transitiven Orientierung auf einem Graphen transitiv sind. Eine Aussage über die Entfernbarekeit einer beliebigen Kante $\hat{e} \in E$ aus $G = (V, E)$ ist bei ANDRESEN (2005) [3] enthalten (siehe auch [4]).

Definition 1.40 (immer-transitive Kante) Sei $G = (V, E)$ ein Comparabilitygraph, und sei $\mathcal{T}_G = \{T_1, \dots, T_k\}$ die Menge seiner transitiven Orientierungen. Eine Kante $\hat{e} \in E$ heißt immer-transitiv, wenn jede Orientierung T_i ($i = 1, \dots, k$) entweder e oder e^{-1} als transitive Kante enthält. Es gilt dann $\{e, e^{-1}\} \cap T_i^r = \emptyset$ für alle $i = 1, \dots, k$.

In Abbildung 1.8 ist ein Graph $G = (V, E)$ abgebildet, der eindeutig transitiv orientierbar ist. D.h. er besitzt genau zwei echte Implikationsklassen I und I^{-1} , die jeweils eine transitive Orientierung von G liefern. In jeder dieser beiden Orientierungen ist die Kante $\hat{e} = \widehat{bc}$ eine transitive Kante. Damit ist \hat{e} eine Kante, die bezüglich keiner transitiven Orientierung von G zur transitiven Reduktion gehört. Dennoch ist der Graph $G - \hat{e}$ ein Comparabilitygraph. Der Grund hierfür ist, dass die Entfernung von $e = bc$ und $e^{-1} = cb$ aus dem Γ -Graphen G_Γ ihre jeweiligen Komponenten aufspaltet. Diese Teilmengen der Implikationsklassen können sich im Graphen $G - \hat{e}$ "gegeneinander verdrehen". D.h. wir können die Orientierung auf einer der beiden Teilmengen umkehren, und auf der anderen unverändert lassen. In diesem Fall führt das zu einer (eindeutigen) transitiven Orientierung auf $G - \hat{e}$.

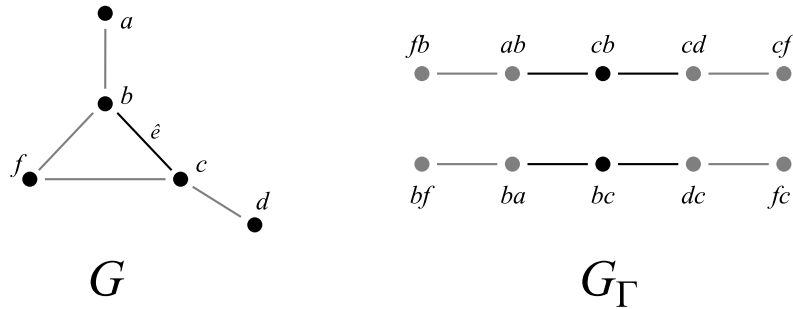


Abbildung 1.8: Ein (primer) Comparabilitygraph G mit einer Kante $\hat{e} = \widehat{bc}$, die entfernt werden kann, obwohl \hat{e} bezüglich der (einzigen) eindeutigen transitiven Orientierung $T \in \mathcal{T}_G$ nicht in der transitiven Reduktion liegt. Die Entfernung von \hat{e} spaltet jede der beiden Implikationsklassen in zwei $\dot{\Gamma}_{\hat{e}}$ -Komponenten.

$\dot{\Gamma}$ -Relation und $\dot{\Gamma}$ -Komponenten

Zur Beschreibung dieser Teilmengen der Implikationsklassen führen wir eine neue Relation ein.

Definition 1.41 ($\dot{\Gamma}_{\hat{e}}$ -Relation) Sei $G = (V, E)$ ein Graph, und sei $\hat{e} \in E$ eine Kante. Für zwei Kanten $ab, cd \in E$ ist die $\dot{\Gamma}$ -Relation bezüglich \hat{e} definiert durch

$$ab\dot{\Gamma}_{\hat{e}}cd \Leftrightarrow ab\Gamma cd \text{ und } \{ab, cd\} \cap \{e, e^{-1}\} = \emptyset.$$

D.h. zwei Kanten stehen genau dann in $\dot{\Gamma}_{\hat{e}}$ -Relation zueinander, wenn sie in Γ -Relation zueinander stehen, und keine der beteiligten Kante e oder e^{-1} ist. Als Anschauung können wir uns die *Kanten* im Γ -Graphen zu einem Graphen $G = (V, E)$ gefärbt vorstellen. Zu jeder Γ -Relation in G gehört eine Kante in G_Γ . Färben wir nun die Kanten, die eine $\dot{\Gamma}_{\hat{e}}$ -Relation repräsentieren mit einer anderen Farbe, als die Kanten, die "nur" eine Γ -Relation repräsentieren, so sind genau die *Kanten* in G_Γ , die mit den Knoten e und e^{-1} inzidieren, anders gefärbt, als alle übrigen (siehe Abbildung 1.8). Für alle Kanten aus den Farbklassen eines Graphen $G = (V, E)$, die \hat{e} nicht enthalten ist die $\dot{\Gamma}_{\hat{e}}$ -Relation identisch mit der Γ -Relation.

An dieser Stelle betrachten wir die $\dot{\Gamma}$ -Relation bezüglich einer fest vorgegebenen Kante $\hat{e} \in E$. Wir werden dieses Konzept der $\dot{\Gamma}_M$ -Relation später noch auf eine beliebige Kantenmenge $M \subseteq E$

ausdehnen, und auf die Menge der ‘‘Schragkanten’’ in einem Comparabilitygraphen zu einer Losung eines Schedulingproblems anwenden.

Analog zur Γ -Relation konnen wir auch fur die $\dot{\Gamma}$ -Relation die Begriffe $\dot{\Gamma}$ -Kette, $\dot{\Gamma}^*$ -Relation, $\dot{\Gamma}$ -Nachbarschaft und $\dot{\Gamma}$ -verbunden definieren. Zwei Kanten eines Graphen $G = (V, E)$ sind genau dann $\dot{\Gamma}_{\hat{e}}$ -verbunden, wenn es in G eine Γ -Kette zwischen diesen beiden Kanten gibt, die weder e noch e^{-1} enthalt.

Der transitive Abschluss $\dot{\Gamma}^*$ ist genau wie die Γ^* -Relation eine Aquivalenzrelation. Die zugehorigen Aquivalenzklassen nennen wir $\dot{\Gamma}$ -Komponenten.

Definition 1.42 ($\dot{\Gamma}_{\hat{e}}$ -Komponente) Sei $G = (V, E)$ ein Graph, und sei $\hat{e} \in E$ eine Kante. Eine Menge $\dot{I} = \dot{I}(e') = \{e'' \in E : e' \dot{\Gamma}_{\hat{e}}^* e''\}$ zu einer Kante $e' \in E$ nennen wir eine $\dot{\Gamma}$ -Komponente von G bezuglich \hat{e} .

Ein zentraler Begriff dieser Arbeit sind die sogenannten *Planimplikationsklassen* zu einem Plan, die wir in Kapitel 4 kennen lernen werden. Diese Planimplikationsklassen sind nichts anderes, als die $\dot{\Gamma}$ -Komponenten bezuglich der Schragkantenmenge des Comparabilitygraphen zu einem Plan.

Da die $\dot{\Gamma}_{\hat{e}}$ -Relation eine Aquivalenzrelation ist, bilden die $\dot{\Gamma}_{\hat{e}}$ -Komponenten eine Partition von $E - \hat{e}$. Diese Zerlegung ist auerhalb der Farbklasse $\hat{I}(e)$ identisch mit der Zerlegung in Implikationsklassen. Ob die Implikationsklasse $I(e)$, die e enthalt, tatsachlich in verschiedene $\dot{\Gamma}_{\hat{e}}$ -Komponenten und e zerfallt, hangt von den konkreten Eigenschaften der Kante e ab. $I(e)$ zerfallt genau dann, wenn das Entfernen des Knotens e im Γ -Graphen die entsprechende Zusammenhangskomponente dekomponiert.

Entfernen von beliebigen Kanten

Ist $\hat{e} \in E$ nun eine immer-transitive Kante, und zerfallt $I(e)$ nicht in mindestens zwei verschiedene $\dot{\Gamma}_{\hat{e}}$ -Komponenten, dann ist $G - \hat{e}$ kein Comparabilitygraph (ANDRESEN [3]). Zerfallt $I(e)$ dagegen, so ist $G - \hat{e}$ genau dann kein Comparabilitygraph, wenn es ein Paar von Kanten in $I(e)$ gibt, die fur die Immer-Transitivitat von \hat{e} verantwortlich sind, aber in einer gemeinsamen $\dot{\Gamma}$ -Komponente bezuglich \hat{e} liegen.

Satz 1.43 (Entfernen einer Kante) [3] Sei $G = (V, E)$ ein Comparabilitygraph, und sei $e = ab \in E$ eine Kante. Der Graph $G - \hat{e}$ ist genau dann kein Comparabilitygraph, wenn in G eine der beiden folgenden Konstellationen existiert:

- (i) \exists ein Knoten $x \in V$ mit $ax, xb \in I(e)$, und ax und xb liegen in einer gemeinsamen $\dot{\Gamma}_{\hat{e}}$ -Komponente;
- (ii) \exists zwei Knoten $y, z \in V$ mit $yb, zb \in I(e) = I_1$, $za, ay, zy \in I_2$ mit $\hat{I}_2 \neq \hat{I}_1$, und yb und zb liegen in einer gemeinsamen $\dot{\Gamma}_{\hat{e}}$ -Komponente.

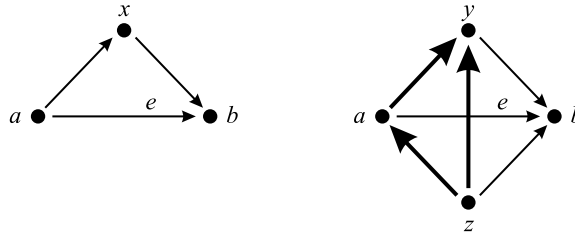


Abbildung 1.9: Eine immer-transitive Kante ab ist transitiv innerhalb ihrer Implikationsklasse $I(e)$ (links), oder sie erfullt zusammen mit einer zweiten Implikationsklasse die Konstellation (*) (rechts).

Der Satz 1.43 enthalt die Aussage von Satz 1.39 als Spezialfall. In [3] wird gezeigt, dass eine Kante $\hat{ab} \in E$ genau dann immer-transitiv ist, wenn sie eine der beiden Konstellationen aus Abbildung 1.9 erfullt. Insbesondere gibt es zu jeder immer-transitiven Kante, die nicht in ihrer eigenen Farbklasse transitiv ist, eine Menge von genau zwei verschiedenen Farbklassen, die fur die Transitivitat dieser Kante verantwortlich sind.

Satz 1.44 (immer-transitive Kanten) [3] Sei $G = (V, E)$ ein Comparabilitygraph. Eine Kante $\widehat{ab} \in E$ ist genau dann immer-transitiv, wenn ein Knoten $x \in V$ mit $ax, xb \in I(ab)$ existiert, oder wenn zwei Knoten $y, z \in V$ mit $yb, zb \in I(ab) = I_1$ und $za, ay, zy \in I_2$ mit $\hat{I}_2 \neq \hat{I}_1$ existieren.

Entfernen von Kantenmengen

Die Frage der transitiven Orientierbarkeit nach dem Entfernen einer *einzelnen* Kante aus einem Comparabilitygraphen ist damit umfassend geklärt. Leider ist dieser Ansatz nicht auf die *gleichzeitige* Entfernung mehrerer Kanten erweiterbar. Eine Kantenmenge ist nicht notwendigerweise entfernbar, wenn jede ihrer Kanten einzeln entfernbar ist. Ein Beispiel hierzu ist ein Kreis der Länge 5 mit 2 (beliebigen) Dreieckssehnern. Dieser Graph ist transitiv orientierbar. Die Orientierung ist dabei eindeutig, wenn die beiden Dreieckssehnern einen gemeinsamen Knoten haben. Das Entfernen jeder einzelnen Dreieckssehne führt nach Satz 1.35 zu einem Comparabilitygraphen. Werden jedoch beide Dreieckssehnern simultan entfernt, so ist der verbleibende ungerade Kreis kein Comparabilitygraph mehr.

Es gibt jedoch einige spezielle Kantenmengen, die problemlos entfernt werden können. Nach Golumbic's Struktursatz (Satz 1.21) können Multiplizes aus einem Comparabilitygraphen entfernt werden, ohne eine gegebene transitive Orientierung zu verändern. GOLUMBIC (1977a) [31] zeigte ferner, dass darüberhinaus auch einzelne Farbklassen gefahrlos entfernt werden können.

Satz 1.45 (Entfernen von Farbklassen) [31] Sei A eine Implikationsklasse eines Graphen $G = (V, E)$, und sei D eine Implikationsklasse von $E - \hat{A}$. Dann gilt eine der beiden folgenden Aussagen:

- (i) D ist eine Implikationsklasse in G , und A ist eine Implikationsklasse von $E - \hat{D}$.
- (ii) $D = B + C$, wobei B und C Implikationsklassen von G sind, und G enthält ein Dreieck mit Kanten aus den Implikationsklassen A, B und C .

Bemerkung 1.46 (Entfernen von Farbklassen) Satz 1.45 besagt insbesondere, dass D eine echte Implikationsklasse ist, wenn G ein Comparabilitygraph ist.

1.4.2 Modulare Dekomposition

In dieser Arbeit wollen wir transitive Orientierungen und Comparabilitygraphen grundsätzlich als aus Implikationsklassen zusammengesetzte Kantenmengen beschreiben. Man kann Comparabilitygraphen jedoch auch mit Hilfe von Knotenmengen algorithmisch beschreiben. Der zentrale Begriff in diesem Zusammenhang ist der eines *Moduls* in einem Graphen $G = (V, E)$.

Basierend auf einem Struktursatz von GALLAI (1967) [27] lässt sich die Knotenmenge eines Graphen $G = (V, E)$ rekursiv so partitionieren, dass mit Hilfe dieser Partition eine azyklische Orientierung von G bestimmt werden kann, die transitiv ist, falls G ein Comparabilitygraph ist. Diese *modulare Dekomposition* ermöglicht das Finden einer transitiven Orientierung eines Comparabilitygraphen in linearer Zeit (siehe Kapitel 2). Daher ist dieses Konzept aus algorithmischer Sicht von großer Bedeutung.

Wir werden diesen Ansatz hier nur kurz skizzieren, und den Zusammenhang zwischen Modulen und Implikationsklassen herstellen.

Definition 1.47 (Modul) Sei $G = (V, E)$ ein Graph. Eine nichtleere Teilmenge $X \subseteq V$ von Knoten heißt Modul, wenn für alle $v \in V - X$ gilt, dass v entweder zu allen $x \in X$ adjazent ist, oder zu keinem.

Definition 1.48 (starkes Modul) Zwei Module X und Y überlappen, falls $X \cap Y \neq \emptyset$ und weder $X \subseteq Y$ noch $Y \subseteq X$ gilt. Ein Modul, das kein anderes überlappt ist ein starkes Modul.

Definition 1.49 (quasimaximales Modul) Ein starkes Modul X in einem Graphen $G = (V, E)$ heißt quasimaximal (stark), wenn $X \subsetneq V$ gilt und kein starkes Modul Y mit $X \subsetneq Y \subsetneq V$ existiert.

Struktursatz von Gallai

Der Zusammenhang zwischen Modulen und transitiven Orientierungen ist durch den folgenden Struktursatz von GALLAI (1967) [27] gegeben.

Satz 1.50 (Gallai's Struktursatz) [27] *Sei $G = (V, E)$ ein Graph.*

- (i) *Ist G nicht zusammenhängend und sind G_1, \dots, G_q die Zusammenhangskomponenten von G , so sind die Farbklassen von G_1, \dots, G_q die Farbklassen von G . Es sei $X_i = V(G_i)$, $i = 1, \dots, q$ und $P = \{X_1, \dots, X_q\}$.*
- (ii) *Ist das Komplement G^c nicht zusammenhängend, (dann ist G zusammenhängend) und sind G_1^c, \dots, G_q^c die Komponenten von G^c mit Knotenmengen $X_i = V(G_i^c)$, $i = 1, \dots, q$, so gilt für jedes Indexpaar $1 \leq i < j \leq q$, dass jeder Knoten von X_i zu jedem Knoten von X_j adjazent ist. Diese Kanten zwischen X_i und X_j bilden eine Farbklassse \hat{I}_{ij} . Die von den \hat{I}_{ij} verschiedenen Farbklassen von G sind die Farbklassen der Graphen G_i . Wiederum sei $P = \{X_1, \dots, X_q\}$.*
- (iii) *Sind G und G^c zusammenhängend, so existiert eine eindeutige Partition $P = \{X_1, \dots, X_q\}$ von V mit folgenden Eigenschaften:*
 - (a) *Gibt es für ein Indexpaar $1 \leq i < j \leq q$ eine Kante von X_i nach X_j , so ist jeder Knoten von X_i adjazent zu jedem Knoten von X_j .*
 - (b) *Sämtliche Kanten von G , die verschiedene Mengen X_i verbinden, bilden eine einzige Farbklassse \hat{I} . Jeder Knoten von G ist zu mindestens einer Kante in \hat{I} inzident.*
 - (c) *Die von \hat{I} verschiedenen Farbklassen von G sind die Farbklassen der Graphen $G_i = G_{X_i}$ ($i = 1, \dots, q$).*
 - (d) *P ist keine Verfeinerung einer anderen Partitionierung von V , die obige Eigenschaften besitzt.*

Die Partition $P = \{X_1, \dots, X_q\}$ heißt kanonische Partition. Die Knotenmengen X_i ($i = 1, \dots, q$) der kanonischen Partition sind die quasimaximalen starken Module von G .

Module und Implikationsklassen

Wir wollen auf diesen fundamentalen Satz an dieser Stelle nicht näher eingehen. Wir greifen lediglich zwei darin enthaltene Aussagen heraus, um eine Verbindung zwischen Modulen und Implikationsklassen herstellen zu können.

Lemma 1.51 (Kanten zwischen Modulen) [27] *Seien X und Y zwei disjunkte Module eines Graphen $G = (V, E)$. Dann ist entweder jedes Element von X zu jedem Element von Y adjazent oder es ist kein Element von X zu einem Element von Y adjazent. Existieren Kanten zwischen X und Y , dann liegen alle Kanten von X nach Y in einer Implikationsklasse.*

Satz 1.52 (Farbklassen und Module) [27] *Sei $\hat{I} \in \mathcal{F}_G$ eine Farbklassse eines Graphen $G = (V, E)$. Dann ist die Menge der Knoten von \hat{I} ein Modul von G . Ist umgekehrt X ein Modul von G , dessen induzierte Kantenmenge eine Kante aus \hat{I} enthält, so liegen alle Kanten aus \hat{I} in $E(X)$.*

Nach Satz 1.52 können wir eine Implikationsklasse eines Graphen $G = (V, E)$ eindeutig mit einem Modul $X \subseteq V$ verknüpfen. Die Umkehrung gilt genau dann, wenn X ein starkes Modul ist.

Bei der *modularen Dekomposition* handelt es sich um eine rekursive Zerlegung der Knotenmenge V eines Graphen $G = (V, E)$ nach Satz 1.50 (Gallai's Struktursatz). Siehe hierzu beispielweise MCCONNELL UND SPINRAD (1999) [50] oder (2000) [51].

1.4.3 Fast alle Comparabilitygraphen sind prim

Ein primer Comparabilitygraph $G = (V, E)$ ist ein eindeutig transitiv orientierbarer Graph (siehe Abschnitt 1.2.1). "Eindeutig" bedeutet, dass es keine zwei *verschiedenen* transitiven Orientierungen auf G gibt. Alle Kanten aus E liegen in einer Farbklassse \hat{I} , und jede Implikationsklasse I und I^{-1} liefert eine transitive Orientierung von G , $\mathcal{T}_G = \{I, I^{-1}\} = \mathcal{I}_G$. MÖHRING (1984) [53] zeigte, dass "fast alle" Comparabilitygraphen eindeutig orientierbar sind. Wir wollen diese Aussage in diesem Abschnitt kurz erläutern, um in einem späteren Kapitel darauf Bezug nehmen zu können.

UPO-Graphen

In der Interpretation von G als PO-Graphen $G(P)$ zur partiellen Ordnung $\mathcal{P} = (V, P)$ (siehe Abschnitt 1.3.6) bedeutet die Eindeutigkeit der transitiven Orientierung, dass aus $G(P) = G(Q)$ entweder $P = Q$ oder $P = Q^{-1}$ folgt. D.h. außer der Umkehrung von P gibt es keine andere partielle Ordnung Q auf V , die denselben PO-Graphen induziert. Man nennt daher prime Comparabilitygraphen auch *UPO-Graphen* (uniquely partially orderable). Von AIGNER UND PRINS (1971) [2] stammen die folgenden Aussagen zu UPO-Graphen.

Satz 1.53 (Charakt. von UPO-Graphen) [2] *Ein Comparabilitygraph $G = (V, E)$ ist genau dann prim (UPO), wenn er höchstens eine nichttriviale Komponente hat, und es keine Partition $L_1 + L_2 = E$ gibt, mit*

$$\widehat{ab} \in L_1, \widehat{bc} \in L_2 \Rightarrow \widehat{ac} \in E.$$

Folgerung 1.54 (dreiecksfreie PO-Graphen) [2] *Sei $G = (V, E)$ ein zusammenhängender Graph, der keine Dreiecke enthält. Dann ist G genau dann eindeutig transitiv orientierbar, wenn er ein Comparabilitygraph ist.*

Satz 1.55 (hinreichende Bed. für UPO) [2] *Sei $G = (V, E)$ ein Comparabilitygraph, dessen Komplement zusammenhängend ist und keinen $K_{1,3}$ als induzierten Teilgraphen enthält. Dann ist G eindeutig transitiv orientierbar.*

UPO-Graphen und PO-Graphen

Die Aussage, dass “fast alle” Comparabilitygraphen eindeutig transitiv orientierbar sind, ist folgendermaßen zu verstehen. Sei $G(n)$ die Anzahl der Comparabilitygraphen, die auf n Knoten existieren, und $G(n, UPO)$ die Anzahl dieser Comparabilitygraphen, die eindeutig orientierbar sind. Dann meint die eindeutige Orientierbarkeit “fast aller” Comparabilitygraphen, dass der Anteil der primen Comparabilitygraphen an allen Comparabilitygraphen auf n Knoten mit wachsendem n gegen 1 konvergiert.

Satz 1.56 (PO-Graphen, asymptotisch) [53] *Fast alle Comparabilitygraphen sind prim, d.h.*

$$\lim_{n \rightarrow \infty} \frac{G(n, UPO)}{G(n)} = 1.$$

Satz 1.56 besagt also, dass unter allen Comparabilitygraphen auf n Knoten mit wachsendem n ein immer größer werdender Anteil davon eindeutig orientierbar ist. Damit lässt sich jedoch keine Aussage über *Teilklassen* von Comparabilitygraphen treffen. Insbesondere folgt für die Klasse der sogenannten H-Comparabilitygraphen, die wir in Kapitel 3 zur Beschreibung von Lösungen von Open-Shop Scheduling Problemen kennen lernen werden, nicht, dass für wachsende Anzahl der Knoten der Anteil der eindeutig orientierbaren Graphen zunimmt.

Aus Satz 1.56 folgt nicht unbedingt, dass bei beliebiger Wahl einer bestimmten Klasse von Comparabilitygraphen $G = (V, E)$ auf n Knoten die erwartete Anzahl von transitiven Orientierungen von G mit wachsendem n sinkt. In Abschnitt 1.2.8 haben wir gesehen, dass die Anzahl von transitiven Orientierungen sehr schnell mit der Anzahl der Farbklassen anwächst. Und da die mögliche Anzahl der Farbklassen mit der Anzahl der Knoten wächst, ist es denkbar, dass mit wachsendem n zwar ein immer kleinerer Anteil von Comparabilitygraphen verschiedene transitive Orientierungen besitzt, dieser jedoch in der Summe immer mehr Orientierungen repräsentiert. Aus diesem Grunde ist große Vorsicht bei der Interpretation von Satz 1.56 geboten.

1.4.4 Perfektheit und Superperfektheit

Zum Abschluss dieses Kapitels wollen wir noch kurz die Zusammenhänge zwischen perfekten, superperfekten und transitiv orientierbaren Graphen erläutern. Auf die herausragende Rolle, die perfekte Graphen in vielen anderen Zusammenhängen spielen, können wir dabei nicht näher eingehen.

Cliquenzahl und chromatische Zahl

Der Begriff der perfekten Graphen geht auf BERGE (1961) [8] zurück. Sie sind definiert als Graphen, bei denen die Cliquenzahl und die chromatische Zahl für jeden induzierten Teilgraphen übereinstimmen. Wir kennen bereits den Begriff Clique als vollständigen Teilgraphen eines Graphen $G = (V, E)$. Die *Cliquenzahl* $\omega(G)$ von G beschreibt nun die Anzahl der Knoten in einer größten Clique in G . Die *chromatische Zahl* oder *Färbungszahl* $\chi(G)$ eines Graphen ist die kleinste Anzahl von Farben in einer Knotenfärbung auf G . Eine *Knotenfärbung* eines Graphen $G = (V, E)$ ist eine Abbildung $f: V \rightarrow \{1, \dots, k\}$ der Knoten in eine Menge $\{1, \dots, k\}$ von Farben mit der Eigenschaft, dass benachbarte Knoten verschiedene Farben besitzen, $f(v) \neq f(w)$ für alle $\widehat{vw} \in E$.

In jeder Färbung eines Graphen $G = (V, E)$ muss jeder Knoten in einer Clique mit einer anderen Farben gefärbt sein. Daher gilt für jeden Graphen, dass die chromatische Zahl mindestens so groß wie die Größe einer maximalen Clique ist, $\chi(G) \geq \omega(G)$.

Definition 1.57 (perfekter Graph) *Ein Graph $G = (V, E)$ heißt perfekt, wenn $\chi(G_A) = \omega(G_A)$ für jeden induzierten Teilgraphen $A \subseteq V$ gilt.*

Berge-Graphen

Bereits Claude Berge vermutete, dass ein Graph genau dann perfekt ist, wenn weder er selbst, noch sein Komplement einen induzierten Kreis ungerader Länge ≥ 5 enthält. Graphen, die diese Eigenschaft erfüllen, werden *Berge-Graphen* genannt. Die Vermutung, dass ein Graph genau dann perfekt ist, wenn er ein Berge-Graph ist, wurde bekannt als die starke Perfekte-Graphen-Vermutung. Sie wurde inzwischen bewiesen (siehe CHUDNOVSKY ET AL. (2005) [21]), und wird nun als starkes Perfekte-Graphen-Theorem zitiert. Eine Folgerung aus dem starken Perfekte-Graphen-Theorem ist, dass ein Graph genau dann perfekt ist, wenn sein Komplement perfekt ist. Diese Aussage wurde bis zu ihrem Beweis durch FULKERSON (1971) [26] und LOVÁSZ (1972) [47] schwache Perfekte-Graphen-Vermutung genannt, und wird nun als schwaches Perfekte-Graphen-Theorem bezeichnet.

Satz 1.58 (schwaches Perfekte-Graphen-Theorem) [47] *Ein Graph $G = (V, E)$ ist genau dann perfekt, wenn sein Komplement G^c perfekt ist.*

Satz 1.59 (starkes Perfekte-Graphen-Theorem) [21] *Ein Graph $G = (V, E)$ ist genau dann perfekt, wenn weder er selbst, noch sein Komplement G^c einen ungeraden Kreis ≥ 5 als induzierten Teilgraphen enthält.*

Außer durch die Gleichheit der chromatischen Zahl $\chi(G_A)$ mit der Cliquenzahl $\omega(G_A)$ für jeden induzierten Teilgraphen $A \subseteq V$ können perfekte Graphen auch noch durch eine Reihe anderer Eigenschaften charakterisiert werden. Darauf wollen wir hier nicht näher eingehen (siehe hierzu beispielsweise GOLUMBIC [33],[34]).

Comparabilitygraphen und Perfektheit

Die Klasse der perfekten Graphen umfasst eine große Zahl von Graphenklassen. Bei HOUARDY (2006) [38] findet man eine Übersicht über 120 dieser Graphenklassen und ihre Beziehungen untereinander. Das *Information System on Graph Class Inclusions* [39] bietet eine Online-Übersicht über minimale Ober- und maximale Unterklassen von mehr als 1.000 verschiedenen Graphenklassen. Eine der ersten Graphenklassen, deren Perfektheit gezeigt werden konnte, war die Klasse der Comparabilitygraphen. Sie folgt bereits aus einem klassischen Theorem von DILLWORTH (1950) [24].

Satz 1.60 (Perfektheit) [24] *Comparabilitygraphen sind perfekt.*

Für Comparabilitygraphen ist der Nachweis der Perfektheit sehr anschaulich. Sei $T \in \mathcal{T}_G$ eine transitive Orientierung eines Comparabilitygraphen $G = (V, E)$. Wir bestimmen die dazugehörige partielle Ordnung $P = P(T)$ der Knoten, und färben die Knoten mit den entsprechenden Elementen aus P . Als Darstellung von P können wir beispielsweise die Ränge der Knoten bezüglich T bestimmen. Benachbarte Knoten können nicht den gleichen Rang haben, da sie bezüglich P vergleichbar sind—d.h. einer der beiden Knoten ist “kleiner” als der andere. (In T muss ein Knoten ein Vorgänger des

anderen sein.) Damit ist durch P eine Färbung von G mit den Farben $\{1, \dots, r_{\max}\}$ gegeben. Da jeder Weg in einer transitiven Orientierung zu einer Clique in G gehört, gilt insbesondere $r_{\max} = \omega(G)$, und damit $r_{\max} \leq \chi(G)$. Da P jedoch eine zulässige Färbung von G ist, gilt $r_{\max} = \chi(G) = \omega(G)$. Mit Satz 1.38 lässt sich diese Argumentation nun auf jeden induzierten Teilgraph von G übertragen.

Gewichtete Cliquenzahl und Intervall-Färbungszahl

Eine weitere Teilklasse der perfekten Graphen ist die Klasse der superperfekten Graphen. Während perfekte Graphen durch “einfache” Knotenfärbungen und Cliques definiert sind, werden bei der Betrachtung superperfekter Graphen gewichtete Varianten beider Begriffe herangezogen.

Sei $G = (V, E)$ ein Graph, und sei durch $w: V \rightarrow \mathbb{N}$ eine nichtnegative Gewichtung der Knoten gegeben. Bei einer *Intervall-Färbung* $f_w: V \rightarrow \mathcal{P}(\{1, \dots, k\})$ auf G wird jedem Knoten $v \in V$ eine Menge von $w(v)$ aufeinanderfolgenden Farben so zugeordnet, dass jeweils benachbarte Knoten keine gemeinsame Farbe erhalten. Es gilt also $|f_w(v)| = w(v)$ und $f_w(v) \cap f_w(w) = \emptyset$ für alle $v, w \in V$ mit $\widehat{vw} \in E$. Die *Intervall-Färbungszahl* $\chi_w(G)$ von G gibt die kleinste Gesamtzahl von Farben an, die bei einer Intervall-Färbung bezüglich w notwendig sind.

Die *gewichtete Cliquenzahl* $\omega_w(G)$ eines Graphen $G = (V, E)$ mit einer nichtnegativen Gewichtung w der Knoten, ist gegeben durch das größte Gewicht einer Clique in G . Trivialerweise muss in einer Intervall-Färbung die Anzahl der Farben für jede Clique mindestens so groß wie das Gewicht der Clique sein. Damit gilt insbesondere $\chi_w(G) \geq \omega_w(G)$. In Anlehnung an die Definition von perfekten Graphen, heißt ein Graph superperfekt, wenn in dieser Relation die Gleichheit gilt.

Definition 1.61 (superperfekter Graph) *Ein Graph $G = (V, E)$ heißt superperfekt, wenn für jede nichtnegative Gewichtung w der Knoten $\chi_w(G) = \omega_w(G)$ für jeden induzierten Teilgraphen $A \subseteq V$ gilt.*

Comparabilitygraphen und Superperfektheit

Klar ist, dass jeder superperfekte Graph auch perfekt ist. Die Definitionen der Begriffe Intervall-Färbung und Färbung stimmen überein, falls die Gewichtung $w(v) = 1$ für alle $v \in V$ gewählt wird. Weniger offensichtlich ist, dass für Comparabilitygraphen auch die Umkehrung dieser Aussage gilt.

Satz 1.62 (Superperfektheit) [39] *Comparabilitygraphen sind superperfekt.*

Die Idee für einen Beweis dieser Aussage werden wir im Abschnitt 3.3.3 skizzieren. Nach [39] sind Comparabilitygraphen sogar eine maximale Teilklasse in der Klasse der superperfekten Graphen. D.h., dass es keine Klasse von superperfekten Graphen gibt, die die Klasse der Comparabilitygraphen vollständig enthält.

Während das Problem der Erkennung von superperfekten Graphen aus komplexitätstheoretischer Sicht noch ungelöst ist, können perfekte Graphen in polynomieller Zeit erkannt werden (CHUDNOVSKY ET AL. (2005) [20]).

Damit beenden wir das Kapitel zu den graphentheoretischen Grundlagen dieser Arbeit. Die zentralen Begriffe “transitive Orientierung” und “Comparabilitygraphen” bleiben auch im weiteren Verlauf dieser Arbeit von entscheidender Bedeutung. Im folgenden Kapitel wollen wir eine kleine Einführung in die Komplexitätstheorie geben.

Kapitel 2

Komplexitätstheoretische Grundlagen

Die Komplexitätstheorie versucht, den Rechenaufwand von Algorithmen zu quantifizieren, und damit Aussagen über die relative Schwierigkeit von verschiedenen Problemen zu ermöglichen. Betrachtet werden dabei sowohl die benötigte Rechenzeit, als auch der benötigte Speicherbedarf. Wir beschränken uns hier ausschließlich auf den Aspekt der Zeitkomplexität von Algorithmen und Problemen.

Nach einer kurzen Einführung in einige grundlegende Begrifflichkeiten (Abschnitt 2.1), führen wir nacheinander die Komplexitätsklassen P und NP (Abschnitt 2.2), NP-complete (Abschnitt 2.3), sowie co-NP und NP-incomplete (Abschnitt 2.4) ein und setzen sie in Beziehung zueinander. Zuletzt betrachten wir noch eine Reihe weniger bekannte Komplexitätsklassen innerhalb von $NP \cup \text{co-NP}$ (Abschnitt 2.5).

Für eine detailliertere Darstellung der Zusammenhänge zwischen diesen Komplexitätsklassen sei auf WEGENER (2003) [65] verwiesen. Dort werden auch weitere Klassen oberhalb von NP und co-NP betrachtet (polynomielle Hierarchie). Eine Einführung in Strukturen innerhalb von P, sowie in Komplexitätsklassen, die in Abhängigkeit vom notwendigen Speicherplatz definiert sind, findet sich beispielsweise bei REISCHUK (1999) [58].

2.1 Die Zeitkomplexität von Algorithmen und Problemen

Als Maß für die Rechenzeit eines Verfahrens wird üblicherweise die maximale Rechenzeit bezüglich eines einheitlichen Kostenmaßes in Abhängigkeit von der Eingabelänge benutzt. Als Kostenmaß wird dabei die Anzahl der elementaren Rechenschritte verwendet. Zu diesen gehören dabei u. a. die Grundrechenarten, Vergleiche, oder die Auswahl von Elementen. Es wird unterstellt, dass jeder dieser elementaren Rechenschritte unabhängig von den technischen Gegebenheiten der benutzten Rechenmaschine dieselbe Zeit benötigt. Um die Rechenzeit eines Algorithmusses zu bestimmen, wird dann die maximale Anzahl der elementaren Rechenschritte gezählt. Diese Anzahl wächst monoton mit der Eingabelänge des vorliegenden Problems. Diese Eingabelänge umfasst einen oder auch mehrere Parameter, die die "Größe" einer bestimmten Problem Instanz beschreiben

Die \mathcal{O} -Notation

Da sich die Anzahl der elementaren Rechenoperationen nur sehr selten exakt bestimmen lässt, wird die Rechenzeit durch obere und untere Schranken beschrieben. Für diese Beschreibungen hat sich die \mathcal{O} -Notation bewährt, bei der die maximale Anzahl der Rechenschritte eines Verfahrens $g(n) : \mathbb{N} \rightarrow \mathbb{N}$ auf den Bestandteil der Funktion $g(n)$ reduziert wird, der das asymptotische Wachstum von $g(n)$ für große Eingabelängen n beschreibt. Das *Landausche Symbol* $\mathcal{O}(f(n))$ beschreibt die Menge aller Funktionen $g(n)$, für die eine Konstante $c > 0$ existiert, so dass $g(n) \leq c \cdot f(n)$ für alle $n \geq n_0$ gilt,

$$\mathcal{O}(f(n)) = \{g(n) : \exists c > 0 : g(n) \leq c \cdot f(n) \quad \forall n \geq n_0\}.$$

Ein Algorithmus mit einem Rechenaufwand $g(n) \in \mathcal{O}(f(n))$ hat die *Zeitkomplexität* $\mathcal{O}(f(n))$. Man sagt auch, der Algorithmus benötigt die (Lauf-) Zeit $\mathcal{O}(f(n))$ oder er hat den *Aufwand* $\mathcal{O}(f(n))$. Die Anzahl der Rechenschritte $g(n)$ wächst dann asymptotisch nicht schneller als $f(n)$, wobei von konstanten Faktoren abstrahiert wird. Ist $f(n)$ ein Polynom in n , so sagen wir, der Algorithmus hat eine polynomielle Laufzeit. Das Problem, das er löst, ist damit polynomiell lösbar.

Obwohl durch $\mathcal{O}(f)$ eine Menge von Funktionen beschrieben wird, wird die Zugehörigkeit von g zu $\mathcal{O}(f)$ häufig durch “ $g = \mathcal{O}(f)$ ” ausgedrückt. Diese Schreibweise ist formal nicht korrekt, aber üblich. Ebenso üblich ist es auch, diese Schreibweise auf Inklusionsbeziehungen zwischen verschiedenen \mathcal{O} -Mengen zu erweitern. So schreibt man etwa $n^2 + n = \mathcal{O}(n^2) = \mathcal{O}(n^3)$, und meint damit $n^2 + n \in \mathcal{O}(n^2)$, sowie $n^2 \in \mathcal{O}(n^3)$, bzw. $\mathcal{O}(n^2) \subseteq \mathcal{O}(n^3)$.

Die Zeitkomplexität eines Problems

Hat ein Algorithmus zur Lösung eines gegebenen Problems \mathcal{A} eine Zeitkomplexität von $\mathcal{O}(f(n))$, so lässt das—unter Berücksichtigung der vorhandenen Rechenleistung—Rückschlüsse auf die Rechenzeit zur Lösung einer konkreten Probleminstanz von \mathcal{A} zu. Damit lassen sich verschiedene Algorithmen zur Lösung von \mathcal{A} hinsichtlich ihres Aufwandes vergleichen. Die Schwierigkeit des Problems selber lässt sich so nur bedingt einschätzen. Klar ist lediglich, dass das Problem \mathcal{A} höchstens eine Zeitkomplexität von $\mathcal{O}(f(n))$ hat. Damit ist \mathcal{A} höchstens $f(n)$ -*schwierig* und mindestens $f(n)$ -*leicht*.

Ein schnellster Algorithmus zur Lösung eines Problems heißt *effizient*. Die Zeitkomplexität eines Problems \mathcal{A} wird bestimmt durch die Zeitkomplexität eines effizienten Algorithmusses zu dessen Lösung. Hat dieser einen Aufwand von $\mathcal{O}(f(n))$, so sagt man, \mathcal{A} ist in der Zeit $\mathcal{O}(f(n))$ lösbar.

Probleme und Probleminstanzen

Sorgfältig auseinanderhalten müssen wir in diesem Zusammenhang die Begriffe Problem und Probleminstanz. Während ein *Problem* eine allgemeine Beschreibung der Eingabeparameter zu einer Fragestellung bezeichnet, ist unter einer *Instanz* dieses Problems eine konkrete Ausprägung dieser Eingabeparameter zu verstehen. Betrachten wir zur Anschauung der Begriffe beispielsweise das folgende Problem.

Problem COMPARABILITY: Gegeben sei ein Graph $G = (V, E)$. Ist G ein Comparabilitygraph?

Eine Instanz dieses Problems ist ein konkreter Graph $G = (V, E)$, der auf seine transitive Orientierbarkeit überprüft werden soll.

Die Zeitkomplexität von COMPARABILITY

In Abschnitt 1.3 haben wir unterschiedliche Methoden zur Identifizierung von Comparabilitygraphen betrachtet, die zu verschiedenen Algorithmen zur Lösung des Problems COMPARABILITY führen. Bestimmt man z.B. in G die Implikationsklassen, und testet diese auf Echtheit, so lässt sich in der Zeit $\mathcal{O}(\delta|E|)$ überprüfen, ob G ein Comparabilitygraph ist, oder nicht (GOLUMBIC (1977) [31], [32]). Mit δ ist dabei die Maximalvalenz in G bezeichnet. Da ein Knoten $v \in V$ maximal $|V| - 1$ Nachbarn haben kann, ist δ nach oben durch $|V|$ beschränkt. Damit gilt $\mathcal{O}(\delta|E|) = \mathcal{O}(|V||E|)$. Bestimmt man alternativ den Graphen G_{P_3} zu G , und testet diesen auf Bipartitheit, so lässt sich die transitive Orientierbarkeit von G ebenfalls in $\mathcal{O}(\delta|E|)$ verifizieren.

Schneller lässt sich das Problem COMPARABILITY jedoch durch die Verwendung der modularen Dekomposition lösen. Die ersten linearen Algorithmen zur Erkennung von Comparabilitygraphen stammen von MCCONNELL UND SPRINRAD (1994) [49],[50] und COURNIER UND HABIB (1994) [23]. Vorgestellt wurde dort jeweils ein Algorithmus mit einer Laufzeit $\mathcal{O}((|V| + |E|) \log |V|)$.

Bei MCCONNELL UND SPRINRAD (2000) [51] wird ein Algorithmus mit einer Laufzeit von $\mathcal{O}(|V| + |E| \log(|V|))$ vorgestellt, der eine azyklische Orientierung bestimmt, die transitiv ist, falls G ein Comparabilitygraph ist. Eine Implementierung dieses Verfahrens mit der angegebenen Laufzeit findet sich bei MÖRIG (2006) [54]. Dieser Algorithmus löst damit nicht das Problem COMPARABILITY. Er liefert jedoch eine Lösung für das Problem TRANSITIVE ORIENTATION, bei dem nicht nach der Existenz einer transitiven Orientierung gefragt wird, sondern eine transitive Orientierung auf einem

Comparabilitygraphen gesucht ist. Bei MCCONNELL UND MONTGOLFIER (2005) [48] findet sich für dieses Problem ein noch schnellerer Algorithmus mit einer Laufzeit $\mathcal{O}(|V| + |E|)$.

2.2 Die Komplexitätsklassen P und NP

Algorithmen mit polynomiellm Aufwand sind für praktische Anwendungen von besonderer Bedeutung. Die damit lösbaren Probleme können exakt gelöst werden, sofern die Problemgröße n nicht zu groß ist. Bei Problemen, die nicht polynomiell lösbar sind, ist das nur bei sehr kleinen Problemgrößen möglich. Allerdings sind auch bereits polynomielle Algorithmen mit einem Polynomgrad von 3 oder mehr für große Problemgrößen oft nicht mehr geeignet.

Randomisierte Algorithmen

Bei allen bisherigen Betrachtungen zu der Komplexität von Algorithmen und Problemen haben wir stillschweigend unterstellt, dass ein Algorithmus ein Problem auch wirklich löst. D.h. dass der Algorithmus die Fragestellung des Problems tatsächlich beantwortet. Die Frage, die hinter dieser Überlegung steht, zielt dabei nicht darauf ab, dass wir zur Lösung eines Problems \mathcal{A} natürlich nur Algorithmen betrachten wollen, die \mathcal{A} zu lösen versuchen, und nicht etwa solche, die ein Problem \mathcal{B} lösen, aber keine Aussage über \mathcal{A} treffen. Wir wollen in Zukunft auch Algorithmen betrachten, die das Problem \mathcal{A} zwar zu lösen versuchen, dabei aber möglicherweise Fehler machen, oder keine Lösung liefern.

Um diesen Unterschied zu verdeutlichen, betrachten wir das Problem CLIQUE: Sei ein Graph $G = (V, E)$ gegeben. Existiert in G eine Clique der Größe k ? Ein Algorithmus, der dieses Entscheidungsproblem vollständig löst, liefert für jede Instanz eine korrekte Antwort JA oder NEIN. "Korrekt" bedeutet in diesem Zusammenhang, dass ein Graph G genau dann eine k -Clique enthält, wenn der Algorithmus die Antwort JA liefert, und genau dann keine Clique der Größe k , wenn die Antwort NEIN lautet. Die Antwort des Algorithmusses ist nicht mit einer Fehlerwahrscheinlichkeit versehen.

Betrachten wir zur Lösung dieses Problems nun den folgenden Algorithmus: Wähle k verschiedene Knoten auf beliebige Weise aus V aus, und prüfe anschließend, ob die Menge der ausgewählten Knoten eine Clique in G bilden. Ist das der Fall, so wird als Antwort JA zurückgegeben. Ist das nicht der Fall, so lautet die Antwort NEIN. In diesem Fall ist der Algorithmus ein Verfahren mit *einseitigem Fehler* oder *Irrtum*: Während die Antwort JA immer korrekt ist, hat die Ausgabe NEIN eine *Irrtumswahrscheinlichkeit* $\varepsilon(n) < 1$. Wie groß diese Fehlerwahrscheinlichkeit ist, ist hierbei im Moment nicht von Belang. (Für dieses konkrete Beispiel liegt sie eher bei 1, als in der Nähe von 0.) Alternativ ist es denkbar, dass der Algorithmus als Antwort "?" zurückgibt, wenn die Menge der ausgewählten Knoten keine Clique in G bilden. Dann handelt es sich um einen Algorithmus mit *einseitigem Versagen*. Während eine Antwort JA immer korrekt ist, liefert das Verfahren mit einer *Versagenswahrscheinlichkeit* $\varepsilon(n) < 1$ keine definitive Antwort.

Da die Auswahl der k Knoten "beliebig" erfolgt, handelt es sich in jedem Fall um ein *randomisiertes* Verfahren. In jedem Schritt können Entscheidungen randomisiert getroffen werden. Um zu einer positiven Antwort für eine gegebene Eingabeinstanz G zu gelangen, muss das Verfahren ggf. mehrmals aufgerufen werden. Eine definitive negative Antwort ist auch bei sehr vielen Aufrufen nicht möglich. Hierzu ist möglicherweise eine gezielte Betrachtung aller möglichen Auswahlen von k Knoten aus V notwendig.

Determinismus und Nichtdeterminismus

Ein Algorithmus, der keine randomisierten Elemente enthält, heißt *deterministisch*. Ein deterministischer Algorithmus hat bei wiederholtem Aufruf mit identischen Eingabedaten immer exakt den gleichen Verlauf. Ein *randomisierter* Algorithmus kann in jedem Schritt zwischen verschiedenen Aktionen wählen, wobei es keine Vorschrift gibt, wie diese Wahl entschieden wird. *Nichtdeterminismus* ist ein Spezialfall von Randomisierung, bei dem einseitige Fehler mit einer Wahrscheinlichkeit kleiner als 1 erlaubt sind: Bei dem beschriebenen Algorithmus zur Lösung von CLIQUE werden k verschiedene Knoten beliebig ausgewählt. Dazu beginnt man beispielsweise mit einem Startknoten v_1 , und wählt nacheinander aus der jeweils verbleibenden Knotenmenge die Knoten v_2, v_3, \dots, v_k . Damit die Knotenmenge $\{v_1, \dots, v_k\}$ eine Clique sein kann, muss auch die Knotenmenge $\{v_1, \dots, v_{k-1}\}$ zwingend

eine Clique sein. Ebenso die Knotenmenge $\{v_1, \dots, v_{k-2}\}$, usw.. Damit kann der Algorithmus in jedem Schritt einen irreversiblen Fehler machen, und einen Knoten v_l wählen, der zusammen mit den Knoten v_1, \dots, v_{l-1} keine Clique bildet. Die Auswahl aller nachfolgenden Knoten ist dann irrelevant. Dieser einseitige Fehler kann nicht mehr durch einen weiteren Fehler (“in die andere Richtung”) korrigiert werden.

Gestaltet man die Auswahl der k Knoten jedoch beispielsweise so, dass nach der Auswahl eines Startknotens v_1 in jedem Schritt zufällig zwei Knoten v_{l_1} und v_{l_2} gewählt werden, und anschließend aus der Knotenmenge $\{v_1, \dots, v_{l-1}, v_{l_1}, v_{l_2}\}$ ein Knoten wieder gelöscht wird, so wäre es denkbar, dass am Ende eine k -Clique ausgewählt wird, obwohl die Menge der ausgewählten Knoten in einem Zwischenschritt keine Clique gebildet hat. In diesem Fall handelt es sich um einen randomisierten Algorithmus, der nicht nichtdeterministisch ist.

Definition 2.1 (P) Die Komplexitätsklasse P (*polynomial time*) enthält alle Entscheidungsprobleme, für deren Lösung ein deterministischer Algorithmus mit polynomieller maximaler Laufzeit existiert.

Die Klasse P enthält damit alle polynomiell lösbaren Entscheidungsprobleme. Diese Aussage impliziert, dass die Probleme dabei vollständig gelöst werden, also weder ein Irrtum, noch ein Versagen auftreten kann. Das Problem COMPARABILITY liegt in P , da es deterministisch in linearer Laufzeit (siehe oben) gelöst werden kann.

Komplexitätsklassen werden grundsätzlich für *Entscheidungsprobleme* definiert, bei denen die Antwort stets entweder JA oder NEIN lautet. Rationale *Optimierungsprobleme* mit nichtleerem Lösungsraum können jedoch mit Hilfe einer binären Suche immer in eine Kette von Entscheidungsproblemen transformiert werden, deren Länge polynomiell beschränkt ist, und zwar logarithmisch durch den Wert der Zielfunktion einer bekannten Lösung des Optimierungsproblems. Damit sind Optimierungsprobleme “im Wesentlichen gleich schwierig”, wie die zugehörigen Entscheidungsprobleme. Sie werden als komplexitätstheoretisch äquivalent betrachtet.

Definition 2.2 (NP) Die Komplexitätsklasse NP (*nondeterministic polynomial time*) enthält alle Entscheidungsprobleme, zu denen ein nichtdeterministischer Algorithmus mit polynomieller maximaler Laufzeit existiert, der jede zulässige Instanz auf mindestens einem erlaubten Rechenweg akzeptiert, und jede unzulässige Instanz auf allen erlaubten Rechenwegen ablehnt.

Ein Beispiel für einen Algorithmus aus NP ist das weiter oben beschriebene Verfahren zur Überprüfung, ob der Graph $G = (V, E)$ eine Clique der Größe k enthält. Dieser Algorithmus ist offensichtlich polynomiell beschränkt, da sowohl die Auswahl der Knoten, als auch die Überprüfung auf Existenz aller Kanten zwischen den ausgewählten Knoten in polynomieller Zeit durchführbar sind. Ferner gibt es zu jedem Graphen G , der eine k -Clique enthält, mindestens einen erlaubten Rechenweg, der eine solche Clique findet. Andererseits gibt es keine Möglichkeit für eine positive Antwort bei einem Graphen G , der keine k -Clique enthält. Also gilt $CLIQUE \in NP$.

Bemerkung 2.3 Da Determinismus ein Spezialfall von Nichtdeterminismus ist, gilt offensichtlich $P \subseteq NP$. Offen ist hingegen nach wie vor die Frage, ob auch $NP \subseteq P$, und damit $NP = P$, oder aber $NP \neq P$ gilt.

2.3 Die Komplexitätsklasse NP-complete

Im vorangegangenen Abschnitt haben wir die Komplexitätsklassen P und NP definiert. Die Klasse P ist dabei vollständig in NP enthalten. Das bedeutet, dass jedes Problem in P “höchstens so schwierig” ist, wie ein schwierigstes Problem in NP . Es gibt jedoch auch Probleme in NP , die “mindestens so schwierig” sind, wie jedes andere Problem in NP . Diese Probleme werden als NP-vollständig oder NP-complete bezeichnet. Die Komplexitätsklasse NP-complete bildet genau wie P eine Teilmenge von NP , und die Frage, ob $NP \neq P$ ist, ist eng mit der Frage verbunden, ob $P \cap NP\text{-complete} \neq \emptyset$ gilt. Um diese Klasse jedoch formal beschreiben zu können, ist es notwendig, zu klären, was damit gemeint ist, wenn ein Problem “mindestens so schwierig” wie ein anderes ist.

Polynomielle Reduktion

Ein Entscheidungsproblem A *reduziert* ein Problem B , wenn eine Problem Instanz von B mit polynomiellen Aufwand so in eine Problem Instanz von A überführt werden kann, dass die transformierte Instanz das Problem A genau dann mit JA bzw. NEIN beantwortet, wenn die Ausgangsinstanz das Problem B mit JA bzw. NEIN beantwortet. Das Problem B ist dann auf A *reduzierbar*, $B \alpha A$. Eine polynomielle Transformation wird dabei generell als schnelle Transformation betrachtet, deren Zeitaufwand als vernachlässigbar unterstellt wird. Der tatsächliche Grad des jeweiligen Polynoms ist dabei nicht von Belang. Mit dieser Auffassung ist das Problem A dann mindestens so schwierig zu lösen, wie das Problem B , da A als Verallgemeinerung von B betrachtet werden kann. Kann das Problem A gelöst werden, so kann das Problem B mindestens mit dem "gleichen Aufwand" gelöst werden.

Zwei Probleme A und B mit $A \alpha B$ und $B \alpha A$ heißen *komplexitätstheoretisch äquivalent*. Sie sind gewissermaßen gleich schwierig zu lösen: jeweils effiziente Algorithmen, die die Probleme vollständig lösen, unterscheiden sich "lediglich" durch einen polynomiellen Faktor. Als tatsächlich vernachlässigbar wirkt ein solcher Faktor z.B. bei Problemen, die nur mit exponentiellem Aufwand vollständig lösbar sind. Ist ein Problem polynomiell lösbar, so kann es natürlich auch mit polynomiellen Aufwand auf ein triviales Problem reduziert werden. Außerdem reduziert es ein solches triviales Problem. Damit ist jedes polynomiell lösbare Problem auf jedes andere reduzierbar. Sämtliche Probleme in P sind damit komplexitätstheoretisch äquivalent.

NP-vollständige Probleme

Es gibt auch in der Klasse NP Probleme, auf die jedes Problem aus NP reduziert werden kann. Diese bilden eine Klasse von Problemen, die untereinander ebenfalls komplexitätstheoretisch äquivalent sind.

Definition 2.4 (NPC) *Ein Entscheidungsproblem $A \in NP$ mit $B \alpha A$ für alle $B \in NP$, heißt NP-vollständig oder NP-complete. Die Komplexitätsklasse NPC (NP-complete) enthält alle NP-vollständigen Probleme.*

Die NP-vollständigen Probleme stellen damit innerhalb der Klasse NP die schwierigsten Probleme dar, da jedes einzelne mindestens so schwierig ist, wie jedes beliebige Problem aus NP. Das Optimierungsproblem zu einem Entscheidungsproblem aus NPC wird *NP-schwer* (*NP-hard*) genannt.

Das erste Problem, für das die NP-Vollständigkeit gezeigt werden konnte, war das Problem SAT (COOK (1971) [22], und unabhängig LEVIN (1973) [45]).

Problem SAT (SATISFIABILITY): Gegeben sei eine Menge von Variablen X_1, \dots, X_k , und eine Menge von Klauseln C_1, \dots, C_l , die aus Literalen $L_i \in \{X_i, \bar{X}_i\}$ bestehen, $C_j = (L_{j_1}, \dots, L_{j_{j^*}})$. Existiert eine Belegung der Variablen mit Wahrheitswerten, so dass in jeder Klausel mindestens ein Literal mit WAHR belegt ist?

Der NP-Vollständigkeitsbeweis für dieses Problem bildet eine wichtige Säule für die gesamte NP-Vollständigkeitstheorie. Ist erst einmal von einem Problem die Zugehörigkeit zu der Klasse NP-complete bekannt, so muss für weitere NP-complete-Beweise nicht mehr jedesmal die Reduzierbarkeit jedes Problems aus NP gezeigt werden. Wegen der Transitivität der α -Relation genügt es jeweils, die Reduzierbarkeit eines NP-vollständigen Problems auf das gegebene Problem zu zeigen.

Bemerkung 2.5 (Nachweis der NP-Vollständigkeit) *Ein Entscheidungsproblem A ist genau dann NP-vollständig, wenn die folgenden zwei Bedingungen erfüllt sind:*

- (i) $A \in NP$;
- (ii) *Es existiert ein $B \in NPC$ mit $B \alpha A$.*

Mittlerweile sind zahlreiche NP-vollständige Probleme bekannt. Zu den bekanntesten graphentheoretischen Problemen aus der Klasse NPC gehören beispielsweise die Probleme HC (Hamilton Circuit), VC (Vertex Cover), STABLE SET (unabhängige Knotenmenge) oder CHROMATIC NUMBER. Auch das im vorangegangenen Abschnitt betrachtete Problem CLIQUE liegt in NPC; ebenso wie die beiden folgenden Probleme, die für uns noch von besonderer Bedeutung sein werden.

Problem COMP-ED (Comparability-Editing): Gegen sei ein Graph $G = (V, E)$. Existiert eine Menge $F \subseteq V \times V$ mit $|F| \leq k$, so dass $G' = (V, E \Delta F)$ ein Comparabilitygraph ist? Dabei bezeichnet $E \Delta F = (E \setminus F) + (F \setminus E)$ die symmetrische Differenz der Kantenmengen E und F .

Problem SUBGRAPH: Gegeben seien zwei Graphen $G_1 = (V_1, E_1)$ und $G_2 = (V_2, E_2)$. Ist G_1 isomorph zu einem Teilgraphen von G_2 ?

Satz 2.6 (COMP-ED) [55] *Das Problem COMP-ED liegt in NPC.*

Satz 2.7 (SUBGRAPH) [28] *Das Problem SUBGRAPH liegt in NPC.*

In Kapitel 4 werden wir das Problem IRRED (IRREDUCIBILITY) als zentrales Problem dieser Arbeit beschreiben, und einen Zusammenhang zwischen IRRED und jedem der Probleme COMP-ED und SUBGRAPH erkennen können.

Die uns bisher bekannte Komplexitätswelt hat somit folgendes Aussehen (siehe Abbildung 2.1): Die Klasse NP enthält die Klassen P und NPC vollständig. Es ist nicht bekannt, ob $NP \neq P$ gilt. Im Falle von $NP = P$ liegt NPC trivialerweise vollständig in P. Im Falle von $P \neq NP$ gilt $NPC \cap P = \emptyset$. Da alle NP-vollständigen Problem komplexitätstheoretisch äquivalent sind, gilt sogar, dass der Nachweis der Existenz eines polynomiellen Algorithmusses für ein einziges NP-vollständiges Problem die Zugehörigkeit der gesamten Klasse NPC zu P, und damit $P = NP$, implizieren würde.

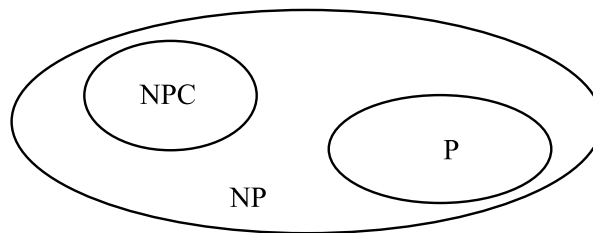


Abbildung 2.1: Die Komplexitätswelt in NP, falls $NP \neq P$ ist.

Ebenso gilt jedoch, dass der Nachweis der *Nicht*existenz eines polynomiellen Algorithmusses für ein einziges NP-vollständiges Problem $NPC \cap P = \emptyset$, und damit auch $P \neq NP$, implizieren würde. Bis heute konnte jedoch trotz intensiver Bemühungen weder das eine noch das andere gezeigt werden. Aber obwohl der Beweis noch aussteht, kann $NP \neq P$ als Antwort auf diese zentrale Frage vermutet werden. Für eine Begründung dieser Vermutung sei an dieser Stelle auf WEGENER (2003) [65] verwiesen. Daher muss der Nachweis der NP-Vollständigkeit eines Problems als ein sehr starkes Indiz für dessen Schwierigkeit gewertet werden.

2.4 Die Komplexitätsklassen co-NP und NP-incomplete

Die Klassen P und NPC bilden jeweils Äquivalenzklassen in NP bezüglich der komplexitätstheoretischen Äquivalenz. Im Falle von $NP \neq P$ enthält NP somit mindestens zwei verschiedene Äquivalenzklassen. Offen ist, ob es auch noch weitere Äquivalenzklassen gibt.

NP-unvollständige Probleme

Definition 2.8 (NPI) *Die Komplexitätsklasse NPI (NP-incomplete) enthält alle Entscheidungsprobleme, die in NP liegen, aber weder zu P, noch zu NPC gehören, $NPI = NP - (P + NPC)$.*

Die Frage, ob die Klasse NPI leer ist, ist nicht nur sehr eng mit der Frage $NP = P$ verbunden. Sie ist sogar äquivalent.

Satz 2.9 (NPI) [42] *Die Klasse NPI ist genau dann nicht leer, wenn $NP \neq P$ ist. Darüberhinaus enthält die Klasse NPI im Falle von $NP \neq P$ mehrere Probleme, die komplexitätstheoretisch nicht äquivalent sind.*

Damit existiert im Falle von $NP \neq P$ eine strukturiertere Komplexitätslandschaft innerhalb von NP als sie in Abbildung 2.1 dargestellt ist. Neben P und NPC existieren dann mindestens noch zwei weitere Äquivalenzklassen in NP. Dennoch ist bisher kein Problem bekannt, von dem gezeigt werden kann, dass es in NPI liegt.

Während es für die Zugehörigkeit eines Problems zu P oder NPC Beweisstrategien gibt, ist es allerdings alles andere als klar, wie eine Zugehörigkeit zu NPI gezeigt werden könnte. Als Kandidaten für NPI-Probleme wurden z.B. die folgenden Probleme betrachtet:

Problem LP (Linear Programming): Sei durch $\max_x \{c^T x : Ax \leq b\}$ ein lineares Optimierungsproblem und durch x_0 eine bekannte Lösung mit $c^T x_0 = k_0$ gegeben. Existiert ein x mit $Ax \leq b$ und $c^T x = k > k_0$?

Problem PRIMES: Gegeben sei eine natürliche Zahl q . Ist q eine Primzahl?

Problem FACT (Factorization): Gegeben seien zwei natürliche Zahlen q und k mit $k \leq q$. Hat q einen Primfaktor d mit $1 < d < k$?

Problem GI (Graph Isomorphism): Gegeben seien zwei Graphen $G_1 = (V_1, E_1)$ und $G_2 = (V_2, E_2)$. Sind G_1 und G_2 isomorph?

Von dem Problem LP konnte bereits 1979 gezeigt werden, dass es polynomiell lösbar ist (KHACHYAN [40]). Für das Problem PRIMES stellte MILLER (1976) [52] einen polynomiellen Algorithmus vor, der allerdings auf einer unbewiesenen zahlentheoretischen Hypothese basiert. AGRAWAL, KAYAL UND SAXENA [1] präsentierten 2002 einen polynomiellen Primzahltest. Damit gelten zur Zeit lediglich die Probleme FACT und GI als ernsthafte Kandidaten für NPI. Für beide Probleme ist es bisher nicht gelungen, polynomielle Verfahren zu entwickeln. Andererseits gibt es für beide Verfahren gute Gründe zu vermuten, dass sie nicht in NPC liegen. Gemeinsam ist den beiden Problemen dabei jeweils eine enge Verwandtschaft zu einem Problem mit bekanntem Komplexitätsstatus.

Das Problem FACT ist offensichtlich eng verwandt mit dem Problem PRIMES, und das Problem GI ist ein Spezialfall von SUBGRAPH für den Fall, dass die beiden Graphen G_1 und G_2 die gleiche Anzahl von Knoten haben. Allerdings gibt es einen bedeutenden Unterschied zwischen diesen beiden Verwandtschaften. Während das Problem PRIMES polynomiell lösbar ist, ist SUBGRAPH NP-vollständig. Auswirkung auf die Komplexität von FACT bzw. GI haben diese Tatsachen überraschenderweise jedoch nicht. Von FACT wird vermutet, dass es "ein wenig" schwieriger als PRIMES ist, aber eben nicht so schwierig, dass es in NPC liegen würde. Und von GI wird vermutet, dass "deutlich" einfacher zu lösen ist, als SUBGRAPH—ohne dabei jedoch polynomiell lösbar zu sein.

Komplementäre Probleme

Der Grund, warum man sowohl von FACT als auch von GI vermutet, dass sie nicht NP-vollständig sind, besteht darin, dass man für beide Probleme die Zugehörigkeit zu der folgenden Komplexitätsklasse zeigen konnte.

Definition 2.10 (co-NP) Die Komplexitätsklasse co-NP enthält alle Entscheidungsprobleme, deren Komplemente in NP liegen.

Ein Problem A liegt damit genau dann in co-NP, wenn \bar{A} in NP liegt. Das Komplement \bar{A} eines Entscheidungsproblems A ist dabei das Problem, dass für jede Instanz genau dann die Antwort JA liefert, wenn das Ausgangsproblem die Antwort NEIN liefert.

Im Abschnitt 2.2 haben wir einen nichtdeterministischen polynomiell begrenzten Algorithmus zur Lösung des Problems CLIQUE beschrieben. Das komplementäre Problem zu diesem Problem (co-CLIQUE) lautet: Hat der gegebene Graph $G = (V, E)$ keine Clique der Größe k (oder größer)? Bei der Beschreibung des angesprochenen Verfahrens haben wir die Bedeutung der Einseitigkeit des erlaubten Fehlers betont: Das Verfahren ist ggf. in der Lage, eine Aussage über die Existenz einer Clique zu treffen. Es ist jedoch keinesfalls in der Lage, eine Aussage, über die Nichtexistenz einer Clique in G zu treffen. Es ist auch kein alternativer NP-Algorithmus bekannt, der die Nichtexistenz einer Clique in G bestätigen könnte. Es ist nicht bekannt, ob das Problem co-CLIQUE in NP liegt.

Das Problem CLIQUE ist jedoch bei weitem nicht das einzigste NP-Problem, dessen Zugehörigkeit zu co-NP ungewiss ist. Es ist sogar so, dass es eine große Zahl von bekannten NP-Problemen gibt, für

deren Komplemente unklar ist, ob sie in NP liegen, oder nicht. Es ist jedoch kein Zufall, dass wir zur Anschauung dieser Tatsache auf ein NP-vollständiges Problem zurückgegriffen haben.

Satz 2.11 (co-NP) ([58],[65]) *Die Klassen NP und co-NP sind genau dann identisch, wenn ein Problem $A \in NPC$ mit $A \in co-NP$ existiert.*

Die Frage $NP \neq co-NP$ ist damit mit der Frage $NP \neq P$ verknüpft: Im Falle von $NP \neq co-NP$ gilt auch $NP \neq P$. Daher überrascht es nicht, dass bisher für kein NP-vollständiges Problem die Zugehörigkeit zu co-NP gezeigt werden konnte. Dies impliziert, dass es bisher beispielsweise auch nicht gelungen ist, ein NP-vollständiges Problem auf sein Komplement zu reduzieren.

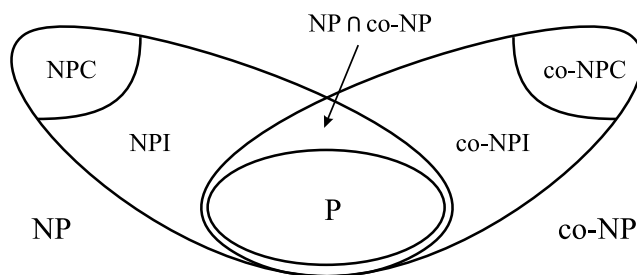


Abbildung 2.2: Die Komplexitätswelt in $NP \cup co-NP$, falls $NP \neq P$ ist.

Es ist nun nicht mehr überraschend, dass man “gute Gründe” hat, zu vermuten, dass die Probleme FACT und GI als Probleme aus $NP \cap co-NP$ nicht NP-vollständig sind. Besonders bemerkenswert ist das für das Problem GI, da im Gegensatz zu FACT das Referenzproblem SUBGRAPH nicht in P, sondern in NPC liegt. In der Vergangenheit konnte von den allermeisten NP-Problemen, deren Zugehörigkeit zu co-NP bewiesen wurde, später auch gezeigt werden, dass sie in P liegen. Für FACT und GI ist das bis jetzt nicht gelungen. Damit ist die Komplexitätswelt in $NP \cup co-NP$ in etwa so strukturiert, wie in Abbildung 2.2 illustriert. Es ist nicht bekannt, ob P (im Falle von $NP \neq P$) eine echte Teilmenge von $NP \cap co-NP$ ist. Ebenso wenig ist bekannt, ob die Klasse $NPI = NPI \cap co-NP$ im Falle von $NP \neq co-NP$ leer oder nicht leer ist. Allerdings wird—nicht zuletzt wegen der Aussage in Satz 2.9—vermutet, dass alle Beziehungen in Abbildung 2.2 echte Teilmengenbeziehungen sind.

2.5 Weitere Komplexitätsklassen

Basierend auf dem Begriff der randomisierten Algorithmen lassen sich auch noch weitere Komplexitätsklassen definieren. Wir wollen diese Klassen an dieser Stelle nicht näher diskutieren, sondern lediglich ihre Beziehung zu den Klassen NP und co-NP darstellen.

Familien von randomisierten Komplexitätsklassen

Definition 2.12 (EP) *Die Komplexitätsklasse EP (exponential polynomial time) enthält alle Entscheidungsprobleme, für die ein randomisierter Algorithmus mit polynomieller maximaler durchschnittlicher Laufzeit existiert.*

Definition 2.13 (ZPP($\varepsilon(n)$)) *Die Komplexitätsklasse ZPP($\varepsilon(n)$) (zero-error probabilistic polynomial time) enthält alle Entscheidungsprobleme, für die ein randomisierter Algorithmus mit polynomieller maximaler Laufzeit existiert, der für jede Eingabe der Länge n eine durch $\varepsilon(n) < 1$ beschränkte Versagenswahrscheinlichkeit besitzt.*

Definition 2.14 (BPP($\varepsilon(n)$)) *Die Komplexitätsklasse BPP($\varepsilon(n)$) (bounded-error probabilistic polynomial time) enthält alle Entscheidungsprobleme, für die ein randomisierter Algorithmus mit polynomieller maximaler Laufzeit existiert, der für jede Eingabe der Länge n eine durch $\varepsilon(n) < \frac{1}{2}$ beschränkte Irrtumswahrscheinlichkeit besitzt.*

Definition 2.15 ($\text{RP}(\varepsilon(n))$) Die Komplexitätsklasse $\text{RP}(\varepsilon(n))$ (random polynomial time) enthält alle Entscheidungsprobleme, für die ein randomisierter Algorithmus mit polynomieller maximaler Laufzeit existiert, der abzulehnende Eingaben ablehnt und anzunehmende Eingaben der Länge n mit einer durch $\varepsilon(n) < 1$ beschränkte Irrtumswahrscheinlichkeit ablehnt.

Zu jeder Klasse \mathcal{C} ist die Klasse $\text{co-}\mathcal{C}$ dadurch definiert, dass die Komplemente ihrer Elemente in \mathcal{C} liegen. In der Klasse EP liegen die sogenannten “Las Vegas-Algorithmen”. Ein Algorithmus aus der Klasse $\text{ZPP}(\varepsilon(n))$ liefert keine falschen Ergebnisse, kann aber versagen (Ausgabe “?”). Ein $\text{BPP}(\varepsilon(n))$ -Algorithmus liefert dagegen immer ein Ergebnis, das jedoch falsch sein kann—und zwar *zweiseitig*. D.h. sowohl eine Antwort JA, als auch eine Antwort NEIN hat eine durch $\varepsilon(n)$ beschränkte Fehlerwahrscheinlichkeit. Die Beschränkung dieser zulässigen Fehlerwahrscheinlichkeit auf $\frac{1}{2}$ ist dabei notwendig, um sinnlose Algorithmen, wie z.B. ein Münzwurf, auszuschließen. In diese Gruppe fallen die sogenannten “Monte-Carlo-Algorithmen”. Im Gegensatz dazu haben $\text{RP}(\varepsilon(n))$ -Algorithmen nur eine einseitige Fehlerwahrscheinlichkeit. Ein Beispiel hierfür ist der Primzahltest von SOLOVAY UND STRASSEN (1977) [61], der in $\text{co-RP}(2^{-100})$ liegt. Ist eine Zahl n prim, so liefert der Algorithmus die Antwort “PRIME”. Ist n dagegen nicht prim, so liefert der Algorithmus mit einer Wahrscheinlichkeit von 2^{-100} dennoch die Antwort “PRIME”.

Für alle Klassen mit Versagens- oder Irrtumswahrscheinlichkeiten gilt, dass ein Problem jeweils nur dann in dieser Klasse liegt, wenn diese Versagens- oder Irrtumswahrscheinlichkeit für jede Instanz echt kleiner als die angegebene Schranke ist. Es reicht nicht, dass es mit einem Algorithmus gelöst werden kann, der für sehr viele oder sogar fast alle Instanzen eine korrekte Antwort findet, aber für einige wenige Instanzen mit Sicherheit ein falsches oder gar kein Resultat liefert. Bei einem $\text{RP}(\varepsilon(n))$ -Algorithmus muss eine zulässige Instanz auf mindestens einem erlaubten Rechenweg als zulässig verifizierbar sein. Bei einem $\text{ZPP}(\varepsilon(n))$ -Algorithmus muss damit nicht nur die Zulässigkeit von jeder zulässigen Instanz auf mindestens einem erlaubten Rechenweg erkennbar sein, sondern auch die Unzulässigkeit von jeder unzulässigen Instanz.

Satz 2.16 (EP) ([65]) *Es gilt $\text{EP} = \text{ZPP}(\frac{1}{2})$.*

Definition 2.17 (ZPP, ZPP*) Die Komplexitätsklasse ZPP enthält alle Entscheidungsprobleme aus $\text{ZPP}(\frac{1}{2})$. In ZPP^* liegen alle Probleme, die für eine Funktion $\varepsilon(n) < 1$ zu $\text{ZPP}(\varepsilon(n))$ gehören.

Definition 2.18 (RP, RP*) Die Komplexitätsklasse RP enthält alle Entscheidungsprobleme aus $\text{RP}(\frac{1}{2})$. In RP^* liegen alle Probleme, die für eine Funktion $\varepsilon(n) < 1$ zu $\text{RP}(\varepsilon(n))$ gehören.

Definition 2.19 (BPP, PP) Die Komplexitätsklasse BPP enthält alle Entscheidungsprobleme aus $\text{BPP}(\frac{1}{3})$. In PP liegen alle Probleme, die für eine Funktion $\varepsilon(n) < \frac{1}{2}$ zu $\text{BPP}(\varepsilon(n))$ gehören.

Die Klasse $\text{BPP}(\frac{1}{2})$ ist genauso unsinnig, wie die Klasse $\text{ZPP}(1)$, da beide jeweils alle Entscheidungsprobleme enthalten. Die Komplexitätsklassen ZPP, ZPP^* , BPP und PP sind genau wie P gegen Komplementbildung abgeschlossen. Diese Komplexitätsklassen lassen sich damit folgendermaßen organisieren: Alle setzen polynomielle maximale Laufzeiten der betreffenden Algorithmen voraus. Bei Problemen aus P darf ein Verfahren weder Fehler machen, noch versagen. Das Ergebnis muss stets richtig sein. Algorithmen zu Problemen aus ZPP oder ZPP^* sind ebenfalls fehlerfrei, dürfen aber versagen (Antwort “?”). Dagegen liefern Algorithmen zu Problemen aus den Klassen RP, RP^* oder co-RP , co-RP^* zwar stets eine Antwort; sie dürfen dabei jedoch einen einseitigen Fehler machen. Die “Richtung” dieser Einseitigkeit markiert dabei den Unterschied zwischen RP und co-RP , bzw. RP^* und co-RP^* . Bei BPP und PP sind schließlich bei allen Eingaben Fehler erlaubt.

Während ZPP, RP, co-RP und BPP Komplexitätsklassen mit beschränkter Versagens- oder Irrtumswahrscheinlichkeit sind, sind ZPP^* , RP^* , co-RP^* und PP Komplexitätsklassen mit unbeschränkten Versagens- oder Irrtumswahrscheinlichkeiten.

Randomisierung und Nichtdeterminismus

Da die Antwort “?” für ein Versagen durch ein Ablehnen mit möglichem Fehler ersetzt werden kann, gilt sowohl $\text{ZPP} \subseteq \text{RP}$, als auch $\text{co-ZPP} \subseteq \text{co-RP}$. Wegen $\text{ZPP} = \text{co-ZPP}$ gilt damit $\text{ZPP} \subseteq \text{RP} \cap \text{co-RP}$. Analog gilt natürlich auch $\text{ZPP}^* \subseteq \text{RP}^* \cap \text{co-RP}^*$. Für beide Klassen gilt sogar die Umkehrung dieser Teilmengenbeziehungen.

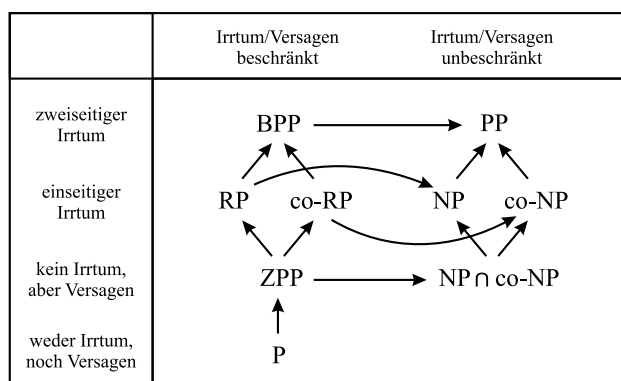
Satz 2.20 (ZPP, ZPP*) ([65]) *Es gelten $ZPP = RP \cap co-RP$ sowie $ZPP^* = RP^* \cap co-RP^*$.*

Auf die Klassen ZPP und ZPP* werden wir in Kapitel 7 noch einmal eingehen. Die Klasse ZPP* haben wir jedoch auch schon im vorangegangenen Abschnitt beschrieben—allerdings trug sie dort einen anderen Namen.

Satz 2.21 (RP*) ([65]) *Es gilt $NP = RP^*$.*

Damit gilt natürlich auch $co-NP = co-RP^*$, sowie $ZPP^* = NP \cap co-NP$. Zum Abschluss dieses Kapitels präsentieren wir eine Darstellung der Komplexitätswelt innerhalb der randomisierten Komplexitätsklassen.

Satz 2.22 (randomisierte Komplexitätsklassen) ([65]) *Zwischen den Klassen P , ZPP , RP , $co-RP$, BPP , $NP \cap co-NP$, NP , $co-NP$ und PP gelten die folgenden Teilmengenbeziehungen.*



Kapitel 3

Open-Shop Scheduling

In diesem Kapitel widmen wir uns einigen grundlegenden Aspekten der Schedulingtheorie. Wir konzentrieren uns dabei auf das sogenannte Open-Shop Schedulingproblem.

In Abschnitt 3.1 stellen wir die allgemeine Formulierung von Shop Schedulingproblemen vor. Anschließend betrachten wir das Blockmatrizenmodell zur Darstellung von Lösungen (Abschnitt 3.2). In Abschnitt 3.3 geben wir eine graphentheoretische Formulierung des Open-Shop Schedulingproblems an, und stellen eine Darstellung von Lösungen durch Comparabilitygraphen vor. Abschließend diskutieren wir in Abschnitt 3.4 kurz die Komplexität von Schedulingproblemen.

Einen Überblick über grundlegende Begriffe und Verfahren in der Schedulingtheorie findet man beispielsweise bei BRUCKER (1995, 2007) [17],[18], BŁAŻEWICZ et al. (2007) [9] oder PINEDO (1995) [56].

3.1 Das allgemeine Shop Schedulingproblem

Die Schedulingtheorie befasst sich mit Existenz- und Optimierungsproblemen an der Schnittstelle zwischen mathematischer Optimierung, diskreter Mathematik, und Operations Research. Die grundlegende Fragestellung lässt sich folgendermaßen formulieren: Wie kann eine gegebene Menge von Aufträgen A_1, \dots, A_n mit Hilfe einer Menge von Maschinen M_1, \dots, M_m so abgearbeitet werden, dass eine Zielfunktion, wie z.B. die Gesamtbearbeitungszeit oder die durchschnittliche Bearbeitungszeit minimiert wird.

Ein Schedulingproblem ist charakterisiert durch seine Maschinenumgebung, seine Jobcharakteristika, und seine Zielfunktion. Zu der Maschinenumgebung gehören unter anderem Informationen zu der Anzahl der vorhandenen Maschinen oder darüber, ob diese Maschinen identisch sind, oder aber verschiedene Arbeitsschritte eines Fertigungsprozesses erledigen.

Betrachtet man beispielsweise einen Prozess, bei dem jeder Auftrag nacheinander auf mehreren verschiedenen Maschinen bearbeitet wird, so spricht man von einem *Shop Problem*¹. Beispiele hierfür sind die Produktion von verschiedenen Modellen auf einer Fertigungsstraße in der Automobilindustrie, die Produktion von verschiedenen Gütern mit Hilfe der gleichen Maschinen innerhalb einer Fabrik, oder die Zuordnung von Lehrern und Klassen in einem Stundenplan. Neben den Shop Problemen werden in der Schedulingtheorie auch andere Problemklassen betrachtet, z.B. *Parallelmaschinenprobleme* oder *Einmaschinenprobleme*.

Zu den Jobcharakteristika gehören Informationen über eventuell vorhandene Vorrangbedingungen zwischen einzelnen Arbeitsschritten eines Prozesses, Verfügbarkeits- (*release dates*) oder Fälligkeitstermine (*due dates*) für einzelne Aufträge, oder ggf. die Unterbrechbarkeit von Operationen.

Mögliche Zielfunktionen können neben der bereits angesprochenen Minimierung der Gesamtbearbeitungszeit (*makespan*) oder der durchschnittlichen Bearbeitungszeit (*mean flow time*) beispiels-

¹Die Bezeichnung Shop Probleme hat ihren Ursprung darin, dass im Englischen einzelne Produktionsabteilungen häufig als Shop bezeichnet werden. In der Automobilindustrie spricht man beispielsweise vom “body shop” (Fertigung der Karosserie) oder vom “paint shop” (Lackierung).

weise auch die Minimierung von Verspätungen, oder die Minimierung von Strafen, die sich durch Verspätungen ergeben, sein.

3.1.1 Technologie und Organisation

Ein Shop Schedulingproblem kann folgendermaßen beschrieben werden. Gegeben sei eine Menge von *Aufträgen* (jobs) A_i mit $i \in I = \{1, \dots, n\}$ und eine Menge von *Maschinen* M_j mit $j \in J = \{1, \dots, m\}$. Jeder Auftrag A_i besteht aus einer Teilmenge der *Operationen* $O_i \subseteq \{O_{i1}, \dots, O_{im}\}$, wobei die Operation O_{ij} die Bearbeitung von Auftrag A_i auf Maschine M_j beschreibt.

Jeder Auftrag kann zu jedem festen Zeitpunkt nur von höchstens einer Maschine bearbeitet werden. Umgekehrt kann jede Maschine zu jedem Zeitpunkt höchstens einen Auftrag bearbeiten. Die Bearbeitung von A_i auf M_j darf u.U. unterbrochen (preemption) und zu einem späteren Zeitpunkt wieder aufgenommen werden. Die *Bearbeitungszeit* (processing time) p_{ij} von Auftrag A_i auf Maschine M_j ist deterministisch und bekannt. In der Literatur werden auch Schedulingprobleme behandelt, bei denen die Bearbeitungszeiten nur mit Ungewissheiten bekannt sind.

Vor dem Hintergrund der Approximierbarkeit von reellen Zahlen durch rationale Zahlen werden die Bearbeitungszeiten üblicherweise als ganzzahlig angenommen.

Operationenmenge

Die *Operationenmenge* $SIJ = O_1 + \dots + O_n$ eines Shop Problems besteht aus der Menge der Operationen aller Aufträge. Gilt für jeden Auftrag A_i , dass die Operationenmenge O_i aus allen Operationen O_{i1}, \dots, O_{im} besteht, so sprechen wir von einer *vollständigen* Operationenmenge SIJ . Andernfalls heißt sie *unvollständig*. Gelegentlich werden wir die Operationenmenge eines Problems nicht als Menge von Operationen O_{ij} beschreiben, sondern uns darauf beschränken, sie durch die Indizes $i \in I$ und $j \in J$ zu beschreiben. Wir identifizieren dann SIJ mit einer Teilmenge von $I \times J$, $SIJ = \{(i, j) \in I \times J: p_{ij} > 0\}$. Im Falle einer vollständigen Operationenmenge gilt $SIJ = I \times J$.

Eine unvollständige Operationenmenge $SIJ^* \subsetneq I \times J$ kann auch als eine vollständige Operationenmenge interpretiert werden, bei der jede nicht vorhandene Operation $O_{ij} \notin SIJ^*$ eine Bearbeitungszeit $p_{ij} = 0$ hat.

Technologische und organisatorische Reihenfolge

Die Reihenfolge der Bearbeitung eines Auftrags A_i auf den Maschinen nennt man die *technologische Reihenfolge* MO_i (machine order) oder die *Technologie* von A_i . Sie kann a-priori bekannt oder Teil des Problems sein. Die Reihenfolge der Aufträge auf einer Maschine M_j heißt die *organisatorische Reihenfolge* JO_j (job order) oder die *Organisation* auf M_j . Sie ist immer Teil des Problems.

Job-Shop

Ist bei einem Shop Problem die technologische Reihenfolge für alle Aufträge durch die Technologie des abgebildeten Prozesses vorgegeben, so spricht man von einem *Job-Shop Problem*. Untersuchungsgegenstand ist in diesem Fall lediglich die Organisation auf allen Maschinen.

Flow-Shop

Ein Spezialfall tritt ein, wenn alle Aufträge zu einer gemeinsamen Produktgruppe gehören und in der technologischen Reihenfolge übereinstimmen. In diesem Fall unterscheiden sich Aufträge, die zu verschiedenen Modellen dieser Produktgruppe gehören nur in den Bearbeitungszeiten. Man spricht dann von einem *Flow-Shop Problem*. Ein Beispiel hierfür ist die Serienproduktion von ähnlichen Modellen eines Produktes auf einer Fertigungsstraße.

Open-Shop

Gibt es dagegen weder Vorgaben zu den technologischen noch zu den organisatorischen Reihenfolgen, so spricht man von einem *Open-Shop Problem*². Hierbei ist es nicht wichtig, was wann und in

²Der Begriff Open-Shop rührt daher, dass es keinerlei Einschränkungen gibt, also alles "offen" ist.

welcher Reihenfolge geschieht. Es kommt lediglich darauf an, dass am Ende jeder Auftrag alle Bearbeitungsstationen durchlaufen hat. Ein Beispiel hierfür ist die Erstellung eines Stundenplans, der jeder Schulklasse für jede Unterrichtsstunde höchstens einen Lehrer, und jedem Lehrer für jede Unterrichtsstunde höchstens eine Klasse zuordnet. Aus symmetrischen Gründen ist es hierbei unerheblich, ob man dabei die Klassen als Aufträge und die Lehrer als Maschinen interpretiert, oder umgekehrt.

In der Literatur wird auch der Fall einer beliebigen Kombination aus Open-Shop und Job-Shop betrachtet. Bei einem solchen *Mixed-Shop Problem* ist die Technologie für einen Teil der Aufträge (oder einen Teil der Operationen einiger Aufträge) vorgegeben.

3.1.2 Die $\alpha \mid \beta \mid \gamma$ -Notation

Neben den bisher angesprochenen Problemen gibt es in der Schedulingtheorie eine gewaltige Anzahl weiterer Problemtypen. Zur Beschreibung von Schedulingproblemen hat sich eine Notation durchgesetzt, die jedes Problem durch drei Felder $\alpha \mid \beta \mid \gamma$ beschreibt. Durch α wird dabei die Maschinenumgebung eines Problems charakterisiert, durch β seine Jobcharakteristika, und durch γ seine Zielfunktion.

Seit ihrer Einführung durch GRAHAM ET AL. (1979) [37] ist diese Notation deutlich erweitert und verfeinert worden. Für eine Beschreibung sei beispielhaft auf BŁAŻEWICZ ET AL. (2007) [9] verwiesen.

Wir beschäftigen uns in dieser Arbeit ausschließlich mit dem Standard-Open-Shop Problem mit Makespan-Kriterium, $O \parallel C_{\max}$. Die Zielfunktion $C_{\max} = \max_i \{C_i\}$ (*Gesamtbearbeitungszeit* oder *makespan*) hängt von den Fertigstellungszeiten der einzelnen Aufträge A_i ab. Die *Fertigstellungszeit* (completion time) eines Auftrags A_i ($i = 1, \dots, n$) ist der Zeitpunkt, zu dem alle Operationen von A_i vollständig abgearbeitet sind. Sie wird mit C_i bezeichnet, und ist von der jeweiligen Lösung des Problems abhängig.

Bei der betrachteten Zielfunktion C_{\max} handelt es sich um eine sogenannte reguläre Zielfunktion. Dabei heißt eine Zielfunktion $f(C_1, \dots, C_n)$ *regulär*, wenn sie monoton wachsend in jeder Variable C_i ($i = 1, \dots, n$) ist, d.h. wenn $f(C_1, \dots, C_n) \leq f(C_1^*, \dots, C_n^*)$ gilt, falls $C_i \leq C_i^*$ für alle $i = 1, \dots, n$ gilt.³

3.1.3 Das Problem $O \parallel C_{\max}$

Das Klassifizierungssystem lässt einen gewissen Interpretationsspielraum bei der Abgrenzung zwischen einem Problem und einer Instanz dieses Problems. Daher wollen wir präzisieren, was wir unter einem Open-Shop Problem verstehen wollen.

Problem und Probleminstanz

Bei einem Open-Shop *Problem* ist eine Menge $\{A_i : i \in I\}$ von Aufträgen und eine Menge $\{M_j : j \in J\}$ von Maschinen gegeben. Jeder Auftrag A_i besteht aus einer gegebenen Menge $\{O_{ik} : k \in J_i \subseteq J\}$ von Operationen. Die Operationenmenge $SIJ = J_1 + \dots + J_n \subseteq I \times J$ beschreibt die Menge aller zu bearbeitenden Operationen. Eine *Probleminstanz* zu diesem Problem enthält darüberhinaus die Information über die Bearbeitungszeiten $p_{ij} > 0$ aller Operationen aus SIJ . Diese Bearbeitungszeiten sind ganzzahlig, deterministisch, und bekannt. Sie unterliegen also keinerlei stochastischen Prozessen oder Ungewissheiten. Gesucht sind die technologischen und organisatorischen Reihenfolgen MO_i ($i = 1, \dots, n$) und JO_j ($j = 1, \dots, m$) für alle Aufträge und alle Maschinen. Hinsichtlich dieser Reihenfolgen müssen keinerlei einschränkende Vorgaben berücksichtigt werden (Open-Shop). Es gelten jedoch die folgenden Bestimmungen (Shop Scheduling): Kein Auftrag A_i kann gleichzeitig von mehr als einer Maschine bearbeitet werden. D.h. zwei Operationen O_{ij} und O_{kl} können allenfalls dann gleichzeitig bearbeitet werden, wenn $i \neq k$ gilt. Analog kann keine Maschine M_j mehr als einen Auftrag gleichzeitig bearbeiten. D.h. zwei Operationen O_{ij} und O_{kl} können allenfalls dann gleichzeitig bearbeitet werden, wenn $j \neq l$ gilt. Wurde die Bearbeitung von Operation O_{ij} begonnen, kann sie nicht unterbrochen werden (kein *preemption*).

³Es gibt auch nicht-reguläre Zielfunktionen, wie beispielsweise $\sum (L_i^2)$ oder $\sum |L_i|$, bei denen jede Terminabweichung $L_i = C_i - d_i$ (d_i due date von Auftrag A_i) unabhängig vom Vorzeichen der Abweichung bestraft wird. Solche Zielfunktionen finden z.B. bei just-in-time-Problemen Anwendung.

Darüber hinaus gibt es keine Einschränkungen: Wartezeiten für Aufträge und Stillstandszeiten für Maschinen sind zulässig (weder *no-wait*, noch *no-idle*); für die Aufträge sind weder Bereitsstellungstermine r_i noch Fertigstellungstermine d_i gegeben.

Als zu minimierende Zielfunktion dient die Gesamtbearbeitungszeit (Makespan) $C_{\max} = \max_{i=1, \dots, n} \{C_i\}$, d.h. der Zeitpunkt, an dem die Bearbeitung aller Aufträge vollständig abgeschlossen ist.

Der angesprochene Interpretationsspielraum bei der Problemformulierung eines Problems der Form $O \parallel C_{\max}$ besteht im Folgenden: In unserer Formulierung ist die Operationenmenge SIJ ein Bestandteil der *Problembeschreibung* (und kein Bestandteil der Beschreibung einer *Probleminstanz*). Damit ist insbesondere die Anzahl der vorhandenen Maschinen bei der Beschreibung des Problems bekannt—und nicht etwa Teil des Problems. Ein Schedulingproblem mit unserer Beschreibung wäre damit besser klassifiziert durch $Om \parallel C_{\max}$. Da dies jedoch unüblich ist, werden wir daran festhalten, ein beliebiges Open-Shop Problem mit Makespan-Kriterium als ein Problem der Form $O \parallel C_{\max}$ zu beschreiben. Die Operationenmenge SIJ setzen wir dabei jedoch gemäß unserer Problembeschreibung immer als gegeben voraus.

Anwendung

Open-Shop Probleme treten in der Praxis unter anderem bei der Verwaltung von kapazitätsbeschränkten differenzierten Ressourcen auf. Ein Beispiel ist die bereits angesprochene Stundenplanung, bei der eine Menge von fachspezifischen Lehrern einer Menge von Klassen mit differenzierten fachlichen Bedürfnissen zugeordnet werden. Hat man dafür eine zulässige Lösung gefunden, ergibt sich ein ähnliches Problem bei der Raumplanung.⁴ Ein weiteres Beispiel ist die Wartung von technischen Anlagen.

Weitere Anwendungsgebiete finden sich zum Beispiel in der Halbleiterfertigung oder in der Satellitenkommunikation (siehe PRINS (1994) [57]).

Lösungsansätze für $O \parallel C_{\max}$

Das Open-Shop Problem $O \parallel C_{\max}$ ist, wie die meisten Schedulingprobleme, NP-schwer (Abschnitt 3.4), und damit für exakte Verfahren unzugänglich (Kapitel 2). Eine global beste Lösung lässt sich allenfalls für kleine Probleminstanzen finden. Für Shop-Probleme allgemein gilt darüber hinaus, dass nicht nur die Bestimmung von optimalen Lösungen sehr schwer ist. Sie sind auch schwer zu approximieren. WILLIAMSON ET AL. (1997) [67] haben gezeigt, dass im Falle von $P \neq NP$ kein polynomialer Algorithmus existiert, der eine approximierte Lösung S_{app} mit der Eigenschaft $C_{\max}(S_{app}) < \frac{5}{4}C_{\max}(S_{opt})$ konstruieren kann, wobei mit S_{opt} ein optimaler Plan bezeichnet sei.

Zur Lösung von Open-Shop Problemen müssen also approximative Verfahren herangezogen werden. Dabei finden zum Einen *konstruktive Heuristiken*, wie beispielsweise die sogenannten *Reihungsregeln* oder *Dispatching Rules*, aber auch abgerüstete *Branch & Bound*-Verfahren Anwendung. Zum Anderen liefern *iterative Verbesserungsverfahren*, wie *Lokale Suche*, *Threshold Accepting* oder *Simulated Annealing* sehr gute Lösungen. Ein notwendiges Element für iterative Heuristiken ist die Definition einer *Nachbarschaft* zwischen zulässigen Lösungen des Problems. Weitere Lösungsmethoden sind durch *dynamische Optimierung* oder *genetische Algorithmen* gegeben.

3.1.4 Plan und Schedule

Eine Lösung eines Open-Shop Problems ist ein *Plan* (engl. *sequence*). Er beschreibt eine zulässige Kombination von technologischer und organisatorischer Reihenfolge. Wir bezeichnen die Menge der zulässigen Pläne zu dem Open-Shop Problem mit einer gegebenen Operationenmenge SIJ mit \mathcal{S}_{SIJ} . Zu jedem Plan $S \in \mathcal{S}_{SIJ}$ ergibt sich bei Berücksichtigung der Bearbeitungszeiten ein Zeitplan oder

⁴Im Schulbetrieb ist das Problem der Raumplanung oft nicht sehr schwierig, da alle Lehrer/Klassen-Kombinationen sehr ähnliche Anforderungen an die Räume haben: die Klassenstärke muss bei der Auswahl der Räume in aller Regel nicht berücksichtigt werden, und zu berücksichtigende fachspezifische Anforderungen an die Räume sind eher die Ausnahme. Anders sieht das Problem der Raumplanung beispielsweise an einer Universität aus. Hier spielt insbesondere die stark variierende Gruppenstärke eine bedeutende Rolle.

Schedule. Dieser enthält über die Informationen zur Technologie und Organisation hinaus auch Informationen über die Anfangs- und Fertigstellungszeitpunkte aller Operationen. Ein Schedule ist eine zulässige Lösung einer Problem Instanz. Da die Operationen während der Bearbeitung nicht unterbrochen werden dürfen (no preemption), ergibt sich der Fertigstellungszeitpunkt C_{ij} (completion time) einer Operation O_{ij} durch seinen Anfangszeitpunkt plus seiner Bearbeitungszeit p_{ij} . Wir beschränken uns daher auf die Beschreibung eines Schedules durch die Fertigstellungszeitpunkte C_{ij} aller Operationen. Aus einem Schedule sind insbesondere die Fertigstellungszeiten $C_i = \max_{j=1, \dots, m} C_{ij}$ aller Aufträge A_i ($i = 1, \dots, n$) ablesbar. Ein Schedule ist *optimal*, wenn er die gegebene Zielfunktion minimiert.

Darstellung von Plänen und Schedules

Ein Plan kann durch die Angabe der Technologie und der Organisation dargestellt werden. Betrachten wir beispielsweise ein Open-Shop Problem mit unvollständiger Operationenmenge mit 3 Aufträgen A_1, A_2, A_3 und 4 Maschinen M_1, M_2, M_3, M_4 . Ein zulässiger Plan $B_1 \in \mathcal{S}_{SIJ}$ könnte dann folgendermaßen beschrieben werden:

$$\begin{array}{ll} \text{technologischer Reihenfolge:} & \text{organisatorischer Reihenfolge:} \\ A_1: & M_2 \rightarrow M_4 \rightarrow M_1 \\ A_2: & M_3 \rightarrow M_2 \rightarrow M_4 \rightarrow M_1 \\ A_3: & M_1 \rightarrow M_3 \rightarrow M_4 \end{array} \quad \begin{array}{l} M_1: A_3 \rightarrow A_1 \rightarrow A_2 \\ M_2: A_1 \rightarrow A_2 \\ M_3: A_2 \rightarrow A_3 \\ M_4: A_1 \rightarrow A_3 \rightarrow A_2 \end{array} .$$

Nehmen wir an, in unserem Beispiel seien die Bearbeitungszeiten durch die Matrix $P = [p_{ij}]$ gegeben. Dann kann ein Schedule beispielsweise durch eine Matrix $C = [C_{ij}]$ dargestellt werden.

$$P = \begin{bmatrix} 6 & 1 & 1 \\ 3 & 6 & 4 & 2 \\ 2 & & 1 & 3 \end{bmatrix} \quad C = \begin{bmatrix} 8 & 1 & 2 \\ 15 & 10 & 4 & 12 \\ 2 & & 5 & 8 \end{bmatrix}$$

Bei der vorliegenden Lösung sind die Aufträge A_1 und A_3 nach 8 Zeiteinheiten beendet, der Auftrag A_2 ist nach 15 Zeiteinheiten abgeschlossen. Mit einem Makespan-Kriterium als Zielfunktion ergibt sich für diesen Schedule ein Zielfunktionswert von 15. Aus den Matrizen P und C kann entnommen werden, dass diese Lösung optimal ist, da der Auftrag A_2 ohne Unterbrechung bearbeitet wird.

Semiaktive und aktive Schedules

Zu jedem Plan existieren beliebig viele Schedules. In dem betrachteten Beispiel können wir jede Operation und alle nachfolgenden Operationen beliebig verspäten. Konzentrieren wir uns andererseits jedoch auf die Betrachtung von Schedules, die keine solchen unnötigen Warte- oder Stillstandszeiten enthalten, so ist der angegebene Schedule eindeutig durch den Plan und die Bearbeitungszeiten bestimmt. Wir nennen einen Schedule *semiaktiv*, wenn keine Operation früher beendet werden kann, ohne den zugrundeliegenden Plan zu verändern. Ein semiaktiver Schedule heißt *aktiv*, wenn keine Stillstandszeit existiert, in der eine spätere Operation bearbeitet werden könnte.

Bemerkung 3.1 (Pläne und semiaktive Schedules) *Zu jedem Plan $S \in \mathcal{S}_{SIJ}$ existiert bei gegebenen Bearbeitungszeiten ein eindeutig bestimmter semiaktiver Schedule. Umgekehrt existiert zu jedem beliebigen Schedule ein eindeutig bestimmter Plan.*

3.2 Das Blockmatrizenmodell

Die Darstellung von Plänen erfolgt in der Literatur in aller Regel nicht durch die direkte Angabe der technologischen und organisatorischen Reihenfolgen, sondern üblicherweise durch eine vollständige lineare Ordnung aller Operationen, die die technologischen Reihenfolgen der Aufträge und die organisatorischen Reihenfolgen auf den Maschinen jeweils als Teilordnungen enthält.

Wir betrachten hier dagegen das sogenannte Blockmatrizenmodell nach BRÄSEL (1990) [10] ([11], [13], [15], oder auch [5], [6]), das eine sehr übersichtliche Beschreibung von Shop-Problemen erlaubt. Hierbei werden die wesentlichen Informationen eines Schedulingproblems und seiner Lösung durch einen Block von $n \times m$ -Matrizen über der Operationenmenge SIJ angegeben.

Die Matrizen P , MO , JO , LR und C

Die Matrix $P = [p_{ij}]$ enthält die Bearbeitungszeiten der Operationen $O_{ij} \in SIJ$. Ist die Operationenmenge unvollständig, so kann das durch Einträge $p_{ij} = 0$ dargestellt werden. Die Matrizen $MO = [mo_{ij}]$ und $JO = [jo_{ij}]$ enthalten die Informationen über die technologischen (machine order, MO) bzw. die organisatorischen Reihenfolgen (job order, JO).

Die Matrix $LR = [lr_{ij}]$ beschreibt die zulässige Kombination (wenn möglich) der technologischen und organisatorischen Reihenfolgen durch ein lateinisches Rechteck $LR = LR(n, m, r)$. Ein *lateinisches Rechteck* $LR(n, m, r)$ ist eine $(n \times m)$ -Matrix mit Einträgen aus $\{1, \dots, r\}$ mit der Eigenschaft, dass kein Eintrag mehrfach in einer Zeile oder Spalte vorkommt.

Schließlich enthält die Matrix $C = [c_{ij}]$ die Fertigstellungszeiten der Operationen $O_{ij} \in SIJ$ als Einträge. Sie beschreibt damit den semiaktiven Schedule zu dem Plan LR .

Die Rangmatrix

Da jede zulässige Kombination von Technologie und Organisation eine Lösung des Open-Shop Problems darstellt, können wir somit Pläne durch solche lateinischen Rechtecke beschreiben.

Definition 3.2 (Plan) Ein Plan $LR = LR(n, m, r) \in \mathcal{S}_{SIJ}$ ist ein lateinisches Rechteck vom Format $n \times m$ mit Einträgen aus $\{1, \dots, r\}$, das die Sequenzbedingung erfüllt: Zu jedem $lr_{ij} > 1$ existiert in Zeile i oder in Spalte j der Eintrag $lr_{ij} - 1$.

Jeder Eintrag lr_{ij} liefert den Rang der Operation $O_{ij} \in SIJ$. Die Matrix LR wird daher auch *Rangmatrix* genannt. Die Sequenzbedingung stellt die Eindeutigkeit der Beschreibung sicher. Unter allen lateinischen Rechtecken über SIJ , die eine bestimmte Lösung eines Schedulingproblems—d.h. eine bestimmte zulässige Kombination von Technologie und Organisation—repräsentieren, wollen wir ausschließlich jenes betrachten, bei dem kein Eintrag mehr durch einen geringeren ersetzt werden kann.

Der Zusammenhang zwischen den Matrizen

Zwischen diesen fünf Matrizen bestehen die folgenden Beziehungen. Der Plan LR ist durch MO und JO eindeutig bestimmt, und kann bei Kenntnis von MO und JO in der Zeit $\mathcal{O}(nm)$ ermittelt werden. Umgekehrt können MO und JO bei Kenntnis von LR in der Zeit $\mathcal{O}(nm)$ abgelesen werden. MO und JO sind hierbei jedoch nur dann eindeutig bestimmt, wenn wir uns auch hier auf den Vertreter mit minimalen Einträgen festlegen. Die Matrix C lässt sich eindeutig (in der Zeit $\mathcal{O}(nm)$) aus dem Plan LR und den Bearbeitungszeiten P bestimmen. Umgekehrt kann aus dem Schedule C bei Kenntnis von P der Plan LR herausgelesen werden.

$$\underbrace{P \quad LR}_C \left\{ \begin{array}{l} MO \\ JO \end{array} \right.$$

Bei einem Job-Shop Problem ist neben den Bearbeitungszeiten P auch die Matrix MO gegeben. Gesucht sind die Matrizen JO , LR und C . Jedoch können jeweils zwei dieser Matrizen bei Kenntnis der dritten leicht berechnet werden. Bei einem Open-Shop Problem sind weder MO noch JO bekannt. Betrachten wir noch einmal das Beispiel aus dem vorherigen Teilabschnitt.

Beispiel 3.3 (Blockmatrizenmodell) Es sind 3 Aufträge A_1, A_2 und A_3 , und 4 Maschinen M_1, M_2, M_3 und M_4 gegeben. Die Bearbeitungszeiten $P = [p_{ij}]$ sind gegeben durch

$$P = \begin{bmatrix} 6 & 1 & 1 \\ 3 & 6 & 4 & 2 \\ 2 & & 1 & 3 \end{bmatrix}.$$

Die weiter oben bereits beschriebene zulässige Lösung zu diesem Open-Shop Problem hat im Blockmatrizenmodell die folgende Darstellung durch die Matrizen $MO = [mo_{ij}]$ und $JO = [jo_{ij}]$,

$$MO = \begin{bmatrix} 3 & 1 & 2 \\ 4 & 2 & 1 & 3 \\ 1 & & 2 & 3 \end{bmatrix}, \quad JO = \begin{bmatrix} 2 & 1 & 1 \\ 3 & 2 & 1 & 3 \\ 1 & & 2 & 2 \end{bmatrix}.$$

Auftrag A_1 , beispielsweise, wird zuerst auf M_2 bearbeitet, danach auf M_4 , und schließlich auf M_1 (Zeile 1 in MO). Maschine M_4 bearbeitet zuerst A_1 , anschließend A_3 , und zuletzt A_2 (Spalte 4 in JO). Durch MO und JO ergibt sich der eindeutig bestimmte Plan $LR = [lr_{ij}] = B_1$. Aus LR und P lässt sich der zugehörige semiaktive Schedule $C = [c_{ij}]$ berechnen.

$$LR = \begin{bmatrix} 3 & 1 & & 2 \\ 5 & 2 & 1 & 4 \\ 1 & & 2 & 3 \end{bmatrix} = B_1, \quad C = \begin{bmatrix} 8 & 1 & & 2 \\ 15 & 10 & 4 & 12 \\ 2 & & 5 & 8 \end{bmatrix}.$$

■

Rangmatrix vs. lineare Ordnung

Der Vorteil der Darstellung eines Plans durch eine Rangmatrix gegenüber der in der Literatur üblichen Repräsentation durch eine lineare Ordnung auf der Operationenmenge SIJ , die alle Informationen zu der Technologie und Organisation als Teilordnungen enthält, besteht in der Eindeutigkeit der Darstellung. Während sehr viele lineare Ordnungen ein und denselben Plan beschreiben können, sind die Pläne, die durch zwei verschiedene Rangmatrizen repräsentiert werden, in keinem Fall identisch.⁵

Erkauft wird dieser Vorteil durch einen größeren Aufwand beim algorithmischen Umgang mit der Darstellung einer Lösung. Bei der Konstruktion einer neuen zulässigen Lösung aus einer gegebenen Rangmatrix muss beispielsweise mit einigem Aufwand sichergestellt werden, dass die manipulierte Matrix tatsächlich eine zulässige Kombination aus Technologie und Organisation repräsentiert—also die Sequenzbedingung erfüllt. Benutzt man dagegen eine lineare Ordnung zur Darstellung einer Lösung, so kann man sehr schnell, durch beliebiges Vertauschen von Operationen in dieser Ordnung, eine alternative zulässige Lösung generieren. Das Problem hierbei besteht dann jedoch darin, dass man möglicherweise keine neue Lösung generiert hat, sondern lediglich eine alternative Beschreibung der gegebenen Lösung. Will man sicherstellen, dass die erzeugte Lösung tatsächlich eine neue Lösung ist, so muss man bei der Darstellung durch lineare Ordnungen entweder auch ein Äquivalent zur Sequenzbedingung beachten, oder eine entsprechend große Anzahl von Lösungen erzeugen. In beiden Fällen entspricht der Gesamtaufwand etwa dem Aufwand beim Umgang mit Rangmatrizen.

Aus algorithmischer Sicht können beide Darstellungsformen daher als äquivalent betrachtet werden. Bei der theoretischen Beschreibung besitzt die Darstellung durch Rangmatrizen jedoch einige Vorteile. Wir können jedem Plan einen Comparabilitygraphen zuordnen, an dem wir schnell ablesen können, ob von zwei gegebenen Plänen einer vielleicht den anderen dominiert.

3.3 Plangraphen und H-Comparabilitygraphen

Viele Schedulingprobleme haben Interpretationen als graphentheoretische Probleme. Wir wollen in diesem Abschnitt eine sehr elegante graphentheoretische Formulierung für Shop Probleme beschreiben. Sie orientiert sich an der Auffassung eines Plans als Rangmatrix, und ermöglicht eine Identifizierung eines Plans mit einer transitiven Orientierung auf einem Comparabilitygraphen. Sie liefert nicht nur eine besonders anschauliche Darstellung von Lösungen des Shop Problems, sondern ermöglicht vor allem eine sehr elegante Beschreibung der Reduzierbarkeit von Plänen. Auf diesen zentralen Begriff dieser Arbeit werden wir in Kapitel 4 eingehen, wenn wir mit allen notwendigen Mitteln zur Beschreibung der Fragestellung dieser Arbeit vertraut sein werden.

Obwohl die graphentheoretische Formulierung grundsätzlich auch für Job-Shop oder Flow-Shop Probleme geeignet ist, wollen wir unsere Augenmerk nach wie vor auf dem Open-Shop Problem belassen.

3.3.1 Das disjunktive Graphenmodell

In der Literatur wird zur graphentheoretischen Beschreibung von Shop Problemen üblicherweise das sogenannte *disjunktive Graphenmodell* nach ROY UND SUSSMANN (1964) [59] herangezogen. Danach

⁵Zwei solche Pläne können allenfalls ähnlich sein—wenn einer der Umkehrplan des anderen ist. Auf diesen Begriff gehen wir im nächsten Abschnitt ein.

wird einer Instanz eines Shop Problems der folgende *disjunktive Graph* $G = (V, C, D)$ zugeordnet (siehe z.B. BRUCKER [17],[18]).

Konjunktive und disjunktive Kanten

Die Knotenmenge V ist gegeben durch die Menge der Operationen SIJ und jeweils einer Quelle $q \in V$ und einer Senke $s \in V$. Jeder der Knoten $O_{ij} \in SIJ$ wird mit der Bearbeitungszeit p_{ij} gewichtet. Die Knoten s und q erhalten das Gewicht Null. Die Kantenmenge des disjunktiven Graphen zerfällt in die Mengen C und D . Die Menge C enthält die Menge der gerichteten *konjunktiven* Kanten. Diese Kanten spiegeln die durch die Problemstellung gegebenen Vorrangbedingungen zwischen einzelnen Operationen wider. Die Quelle q ist mit jeder Operation, die keine Vorgänger hat, durch eine konjunktive Kante verbunden. Die Senke s mit jeder Kante, die keine Nachfolger hat. Ist a-priori nicht klar, welche Operationen jeweils keine Vorgänger bzw. Nachfolger haben werden, dann werden alle in Frage kommenden Operationen mit q bzw. s verbunden.

Die Menge D schließlich enthält die Menge der ungerichteten *disjunktiven* Kanten. Solche Kanten existieren genau dann zwischen zwei Operationen, wenn diese nicht gleichzeitig bearbeitet werden können.

Das Optimierungsproblem besteht nun darin, für jede noch ungerichtete “disjunktive” Kante eine Orientierung auszuwählen, so dass der resultierende Graph azyklisch ist, und die Zielfunktion minimiert wird. Bei einer C_{\max} -Zielfunktion entspricht der Zielfunktionswert dem Gewicht eines längsten Weges von der Quelle q zu der Senke s .

Anwendung auf Open-Shop Problem

Bei einem Job-Shop Problem enthält die Menge der konjunktiven Kanten die vorhandenen Informationen über die technologischen Reihenfolgen. Bei einem Open-Shop Problem ist diese Kantenmenge auf die Kanten von der Quelle q zu jeder Operation, und die Kanten von jeder Operation zu der Senke s beschränkt. Alle vorhandenen Kanten zwischen verschiedenen Operationen sind disjunktiv. Zwei Operationen O_{ij} und O_{kl} sind damit genau dann durch eine ungerichtete “disjunktive” Kante verbunden, wenn sie entweder zum selben Auftrag, oder zur selben Maschine gehören, $G = (V, C, D) = (SIJ + q + s, C + D)$ mit $C = \{qO_{ij} : O_{ij} \in SIJ\} \cup \{O_{ij}s : O_{ij} \in SIJ\}$ und $D = \{\widehat{O_{ij}O_{kl}} : O_{ij}, O_{kl} \in SIJ, i = k \vee j = l\}$.

Variation ohne zusätzliche Quelle oder Senke

Wir verwenden in dieser Arbeit eine leichte Variation des disjunktiven Graphenmodells, wie sie z.B. bei BRÄSEL ET AL. (1999) [13], BRÄSEL (2008) [11], WILLENUS (2000) [66] oder BŁAŻEWICZ ET AL. (2007) [9] beschrieben wird. Wir verzichten auf die Einführung der zusätzlichen Knoten q und s , und suchen entsprechend eine azyklische Orientierung mit möglichst niedrigem Gewicht auf einem schwersten Weg von einer Quelle zu einer Senke.

Der disjunktive Graph zu einem Open-Shop Problem hat dann die folgende Form, $G = (V, C, D) = (SIJ, C + D)$ mit $C = \emptyset$ und $D = \{\widehat{O_{ij}O_{kl}} : O_{ij}, O_{kl} \in SIJ, i = k \vee j = l\}$.

Der H-Graph zu einem Open-Shop Problem

Ist die Operationenmenge SIJ vollständig, d.h. gilt $SIJ = I \times J$ mit $|I| = n$ und $|J| = m$, so ist der disjunktive Graph zu einer *Instanz* eines Open-Shop Problem isomorph zu dem *Hamming-Graphen* $K_n \times K_m$. Da sich verschiedene Instanzen zu einem Open-Shop Problem lediglich durch die Bearbeitungszeiten unterscheiden, können wir den Hamming-Graphen $K_n \times K_m$ ohne Knotengewichtung dem Open-Shop *Problem* mit vollständiger Operationenmenge zuordnen.

Diesen Ansatz können wir nun auch auf unvollständige Operationenmengen erweitern. Der disjunktive Graph zu einem Open-Shop Problem ist dann isomorph zu einem induzierten Teilgraphen des $K_n \times K_m$.

Definition 3.4 (H-Graph) Sei $SIJ \subseteq I \times J$ eine Operationenmenge mit $|I| = n$ Aufträgen und $|J| = m$ Maschinen. Wir nennen die Einschränkung des Hamming-Graphen $K_n \times K_m$ auf die Knotenmenge SIJ den H-Graphen zu dem Open-Shop Problem auf SIJ , $G_{SIJ} = (SIJ, E_{SIJ})$ mit

$$E_{SIJ} = \left\{ \widehat{O_{ij}O_{kl}} : O_{ij}, O_{kl} \in SIJ, i = k \vee j = l \right\}.$$

Der H-Graph G_{SIJ} ist der eindeutig bestimmte disjunktive Graph zu dem Open-Shop Problem auf der Operationenmenge SIJ . Werden die Knoten in einem $n \times m$ -Raster angeordnet, so können wir von ‘‘Zeilen’’ und ‘‘Spalten’’ in G_{SIJ} sprechen. Jede Zeile von G_{SIJ} entspricht einem Auftrag A_i ($i = 1, \dots, n$) und jede Spalte einer Maschine M_j ($j = 1, \dots, m$). Wir werden noch sehen, dass diese Sprechweise nicht nur intuitiv ist, sondern durchaus auch sehr sinnvoll sein kann. Der Graph G_{SIJ} ist damit ein ungerichteter Graph, der in jeder Zeile und in jeder Spalte aus einem vollständigen Teilgraphen besteht. In Abbildung 3.1 ist der H-Graph (links) zu dem Problem aus Beispiel 3.3 illustriert.

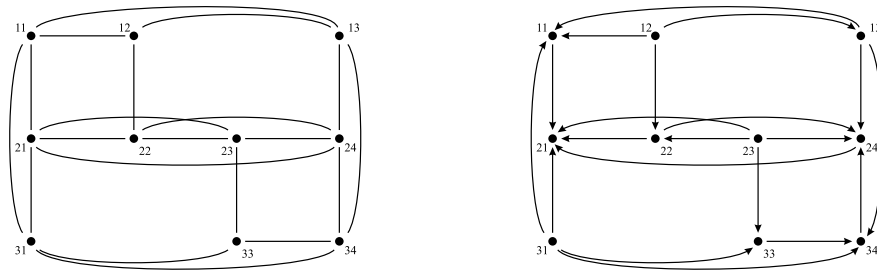


Abbildung 3.1: Der H-Graph G_{SIJ} und der Plangraph $G(B_1)$ zu dem Plan $B_1 \in \mathcal{S}_{SIJ}$.

3.3.2 Der Plangraph zu einem Plan

Der H-Graph G_{SIJ} zu einem Open-Shop Problem spiegelt die Aufgabenstellung wider. Jede azyklische Orientierung auf G_{SIJ} beschreibt eine zulässige Lösung des Open-Shop Problems, also einen Plan $S \in \mathcal{S}_{SIJ}$. Umgekehrt kann auch jedem Plan $S \in \mathcal{S}_{SIJ}$ eine azyklische Orientierung auf G_{SIJ} zugeordnet werden.

Der Plangraph $G(S)$

Nach dem Blockmatrizenmodell können wir einen Plan $S \in \mathcal{S}_{SIJ}$ als eine Rangmatrix über der Operationenmenge auffassen. Zur Erinnerung: Der Rang eines Knotens in einem Graphen bezüglich einer azyklischen Orientierung dieses Graphens ist die Länge eines längsten Weges von einer Quelle zu diesem Knoten (siehe Abschnitt 1.2.1). Wir können die durch S gegebenen Ränge der einzelnen Operationen daher in eine eindeutig bestimmte Orientierung der Kanten von G_{SIJ} übersetzen: $O_{ij} \rightarrow O_{kl} \Leftrightarrow [i = k \vee j = l] \wedge [s_{ij} < s_{kl}]$. Die Orientierung auf G_{SIJ} zu dem Plan S enthält dann die (gerichtete) Kante $O_{ij}O_{kl}$. Wir schreiben dann auch $O_{ij}O_{kl} \in S$. Diese Orientierung auf G_{SIJ} ist genau dann azyklisch, wenn S ein Plan ist, und somit die Sequenzbedingung aus Definition 3.2 erfüllt. Insbesondere können wir die Matrix S mit Einträgen über SIJ mit einer (orientierten) Kantenmenge auf G_{SIJ} identifizieren.

Definition 3.5 (Plangraph) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan. Durch $G(S) = (SIJ, S)$ wird eine azyklische Orientierung auf dem H-Graphen $G_{SIJ} = (SIJ, E_{SIJ})$ beschrieben. Wir nennen $G(S)$ den Plangraphen zu S .

Aus dem Plan- oder auch Ablaufgraphen $G(S)$ können alle Vorgänger- und Nachfolgerbeziehungen zwischen Operationen eines jeden Auftrags oder auf den einzelnen Maschinen abgelesen werden.

Der Plangraph $G(S)$ besteht aus horizontalen Kanten $O_{ij}O_{kl}$ für die $i = k$ und $s_{ij} < s_{kl}$ gilt, und aus vertikalen Kanten $O_{ij}O_{kl}$ mit $j = l$ und $s_{ij} < s_{kl}$. Die horizontalen Kanten beschreiben

die technologischen Reihenfolgen, und die vertikalen Kanten die organisatorischen Reihenfolgen einer Lösung.

In Abbildung 3.1 (rechts) ist der Plangraph zu dem Plan B_1 aus Beispiel 3.3 abgebildet.

Die Graphen G_{MO} , G_{JO} und $G_{MO,JO}$

Wir können auch die Matrizen MO und JO aus dem Blockmatrizenmodell mit Graphen assoziieren. Der Graph $G_{MO} = (SIJ, MO)$ ist ein gerichteter Graph auf der Operationenmenge SIJ , in dem zwei Operationen $O_{ij}O_{kl} \in SIJ$ genau dann durch eine Kante $O_{ij}O_{kl}$ verbunden sind, wenn $i = k$ und $s_{ij} < s_{kl}$ gilt. Der Graph beschreibt damit die Technologie zu einem Plan. Analog ist der Graph $G_{JO} = (SIJ, JO)$ definiert, der die organisatorischen Reihenfolgen zu einem Plan $S \in \mathcal{S}_{SIJ}$ beschreibt. In Abbildung 3.2 sind die Graphen G_{MO} und G_{JO} zu dem Plan B_1 aus Beispiel 3.3 abgebildet.

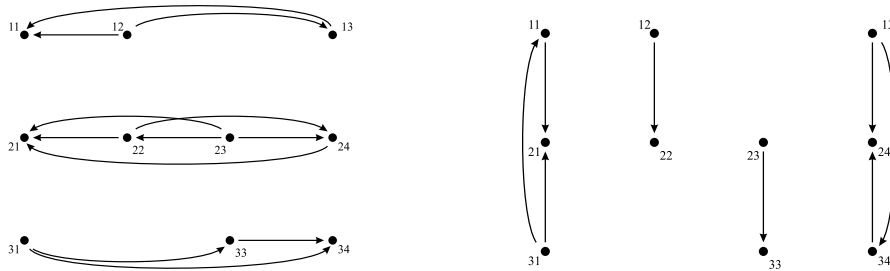


Abbildung 3.2: Die Graphen G_{MO} und G_{JO} zum Plan B_1 .

Der Graph $G_{MO,JO} = G_{MO} + G_{JO} = (SIJ, MO + JO)$ vereinigt die technologischen und organisatorischen Reihenfolgen. Er ist identisch zu dem Plangraphen $G(S)$ zu dem Plan S , der durch MO und JO beschrieben wird.

Aus Gründen der Übersichtlichkeit werden MO und JO häufig auch mittels ihrer transitiven Redukte MO^r und JO^r dargestellt.

Gewichtete und ungewichtete Varianten von G_{SIJ} und $G(S)$

Ein Schedule C zu einem Plan $S \in \mathcal{S}_{SIJ}$ kann einfach durch die Gewichtung der Knoten $O_{ij} \in SIJ$ in $G(S)$ mit den jeweiligen Bearbeitungszeiten p_{ij} mit einem Graphen assoziiert werden. Der resultierende ‘‘Schedulegraph’’ $G(C) = G(S, P)$ verhält sich damit so zum Plangraphen $G(S)$, wie der H-Graph $G_{SIJ}(P)$ für eine gegebene Instanz P zum H-Graphen G_{SIJ} des jeweiligen Problems. Obwohl diese zwei Paare von Graphen jeweils formal verschieden sind, und sorgfältig auseinander gehalten werden müssen, gibt es keine sprachliche Trennung zwischen den gewichteten und ungewichteten Varianten. Wir werden daher auch bei der Betrachtung von konkreten Probleminstanzen P das Optimierungsproblem als Suche nach einer azyklischen Orientierung auf dem H-Graphen G_{SIJ} formulieren—und nicht etwa als Suche auf dem Graphen $G_{SIJ}(P)$. Ebenso werden wir auch den gewichteten Graphen $G(S)$ zu einer Lösung als Plangraphen bezeichnen—und nicht etwa als Schedulegraph $G(S, P)$.

3.3.3 Der H-Comparabilitygraph zu einem Plan

Die Graphen G_{MO} und G_{JO} sind transitiv orientierte Graphen. Aus G_{MO} in Abbildung 3.2 können wir ablesen, dass die Operation O_{21} erst bearbeitet wird, nachdem die Operationen O_{23} , O_{22} und O_{24} abgearbeitet sind. Es existiert eine Kante von jedem dieser drei Vorgänger zu O_{21} . Ebenso können wir aus G_{JO} ablesen, dass auf Maschine 2 die Operation O_{12} noch vor der Operation O_{22} bearbeitet wird, da eine Kante von O_{12} nach O_{22} existiert. Aus dem Plangraphen $G(S) = G_{MO,JO}$ können wir jedoch nicht so einfach ablesen, dass O_{12} damit auch ein Vorgänger von O_{21} ist, da diese beiden Operationen nicht adjazent sind. Der Plangraph $G(S)$ ist also im Allgemeinen kein transitiv orientierter Graph. Um jedoch von den vielen Eigenschaften transitiv orientierbarer Graphen profitieren zu können, werden wir zur Beschreibung von Plänen auch den transitiven Abschluss von $G(S)$ betrachten.

Wenn wir $S \in \mathcal{S}_{SIJ}$ nicht nur als Rangmatrix über SIJ auffassen, sondern gleichzeitig auch mit der durch S induzierten Kantenmenge über der Knotenmenge SIJ identifizieren, können wir die transitive Hülle von $G(S)$ durch $G(S)^{tr} = (SIJ, S^{tr})$ oder auch einfach nur durch S^{tr} beschreiben. Der symmetrische Abschluss $(SIJ, S^{tr} + (S^{tr})^{-1}) = [S^{tr}]$ ist dann ein Comparabilitygraph, der eine transitive Orientierung besitzt, die eine Lösung des Open-Shop Problems auf der Operationenmenge SIJ repräsentiert. Da S eine Orientierung auf dem H-Graphen G_{SIJ} ist, ist G_{SIJ} ein Teilgraph von $[S^{tr}]$. Also enthält $[S^{tr}]$ insbesondere den auf SIJ eingeschränkten Hamming-Graphen $K_n \times K_m$ als Teilgraphen. Daher nennen wir $[S^{tr}]$ den H-Comparabilitygraphen zu S .

Definition 3.6 (H-Comparabilitygraph zum Plan S) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan. Der Graph $[S^{tr}] = (SIJ, S^{tr} + (S^{tr})^{-1})$ heißt H-Comparabilitygraph von S .

In Abbildung 3.3 ist die transitive Hülle B_1^{tr} und der H-Comparabilitygraph $[B_1^{tr}]$ zu dem Plan B_1 aus dem Beispiel 3.3 dargestellt. Im Abschnitt 3.3.5 werden wir noch eine etwas verallgemeinerte Betrachtung von H-Comparabilitygraphen diskutieren, die die Definition unabhängig von einem konkreten Plan machen wird. Bis dahin werden wir immer von dem H-Comparabilitygraphen zu einem konkreten Plan $S \in \mathcal{S}_{SIJ}$ sprechen.

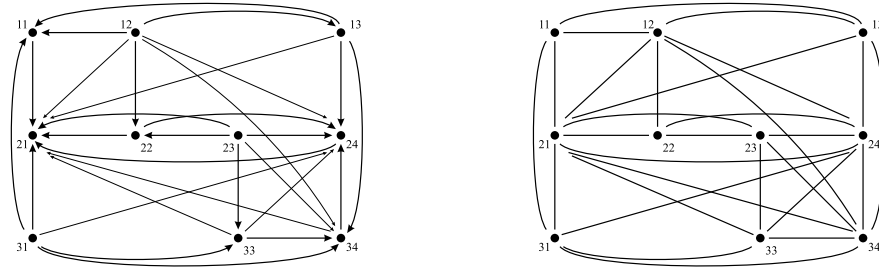


Abbildung 3.3: Die transitive Hülle B_1^{tr} und der H-Comparabilitygraph $[B_1^{tr}]$.

3.3.4 Reguläre Kanten und Schrägkanten

Die Kantenmenge eines H-Comparabilitygraphen zu einem Plan zerfällt in zwei Gruppen von Kanten, die vor dem Hintergrund des Schedulingproblems grundsätzlich verschiedene Eigenschaften besitzen. Die eine Gruppe besteht aus den Kanten, die auch in dem H-Graphen G_{SIJ} enthalten sind, die andere Gruppe enthält alle übrigen Kanten.

Reguläre Kanten und reguläre Wege

Die Kanten der ersten Gruppe sind, unabhängig von einer konkreten Lösung des Problems, immer Bestandteil des H-Comparabilitygraphen. Sie gehören zur Problembeschreibung, und werden als reguläre Kanten bezeichnet.

Definition 3.7 (reguläre Kante) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan, G_{SIJ} der H-Graph von SIJ , und $[S^{tr}] = (SIJ, E(S))$ der H-Comparabilitygraph zu S . Eine Kante $e \in E(S)$ heißt regulär, $e \in E_{reg}(S)$, wenn sie auch in dem Teilgraphen G_{SIJ} liegt. Insbesondere sind alle Kanten des Plangraphen $G(S) = (SIJ, S)$ regulär.

Die Menge der regulären Kanten $E_{reg}(S)$ umfasst damit sämtliche horizontalen und vertikalen Kanten—sowie deren Umkehrungen. Sie ist für alle Pläne $S \in \mathcal{S}_{SIJ}$ gleich, und nur von der Operationenmenge SIJ abhängig. Der Plangraph $G(S)$ besteht aus einer azyklischen Orientierung der regulären Kanten des H-Comparabilitygraphen $[S^{tr}]$.

Definition 3.8 (Weg, regulärer Weg) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan. Ein Weg W in S ist ein Weg auf dem Plangraphen $G(S) = (SIJ, S)$, $W \subseteq S$. Ein Weg $W \subseteq E(S)$ im H-Comparabilitygraphen $[S^{tr}] = (SIJ, E(S))$ von S heißt regulär (bezüglich S), wenn er ein Weg in S ist.

Ein regulärer Weg W im H-Comparabilitygraphen $[S^{tr}]$ ist also eine Kantenmenge der transitiven Orientierung S^{tr} von $[S^{tr}]$, die ausschließlich aus regulären Kanten besteht. D.h. es gilt $W \subseteq S$. Da wir den Plan S mit der Kantenmenge des Plangraphen $G(S)$ identifizieren, unterscheiden wir nicht zwischen einem Weg in einem Plan, und einem Weg auf dem zugehörigen Plangraphen.

Die transitive Hülle S^{tr} von S enthält in aller Regel jedoch zusätzliche Wege. Diese zusätzlichen Wege enthalten jedoch jeweils mindestens eine Kante, die weder horizontal, noch vertikal—und damit nicht regulär ist. Um diese zusätzlichen Wege in der transitiven Orientierung S^{tr} des H-Comparabilitygraphen $[S^{tr}]$ von den Wegen in S unterscheiden zu können, benutzen wir den Begriff des regulären Weges. Wir können so beispielsweise von einem regulären Weg W zu einem Plan S sprechen, ohne jedes Mal mit anführen zu müssen, dass wir W in diesem Zusammenhang ggf. als ein Objekt in einer Orientierung des H-Comparabilitygraphen von S betrachten.

Es sei an dieser Stelle in Erinnerung gerufen, dass wir Kantenmengen prinzipiell als gerichtet betrachten. In einem ungerichteten Graphen existiert zu jeder gerichteten Kante auch ihre Umkehrung. Wir wollen Wege daher prinzipiell ebenfalls als gerichtete Kantenmengen auffassen. Damit gilt, dass jeder Weg $W \subseteq S^{tr}$ in der transitiven Hülle von S auch ein Objekt im ungerichteten H-Comparabilitygraphen $[S^{tr}]$ ist. Anders als in S^{tr} existiert in $[S^{tr}]$ jedoch auch der umgekehrte Weg W^{-1} .

Schrägganten

Durch die Bestimmung der transitiven Hülle S^{tr} von $S \in \mathcal{S}_{SIJ}$ entstehen in aller Regel zusätzliche Kanten. Diese sind weder horizontal, noch vertikal. Wir nennen solche Kanten daher Schrägganten (*diagonal edges*). Ihre Existenz oder Nichtexistenz in S^{tr} ist damit von dem zugrundeliegenden Plan S abhängig.

Definition 3.9 (Schräggante) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan, G_{SIJ} der H-Graph von SIJ , und $[S^{tr}] = (SIJ, E(S))$ der H-Comparabilitygraph zu S . Eine Kante $e \in E(S)$, die nicht in dem Teilgraphen G_{SIJ} liegt, heißt Schräggante in $[S^{tr}]$, $e \in E_{diag}(S)$.

Die Kantenmenge eines H-Comparabilitygraphen zerfällt damit in die Menge der regulären Kanten E_{reg} und die Menge aller Schrägganten E_{diag} . Die Menge der regulären Kanten ist dabei unabhängig von einem konkreten Plan. Sie hängt lediglich von der zugrundeliegenden Operationenmenge ab. Daher unterscheiden sich die H-Comparabilitygraphen zu zwei Plänen S_1 und S_2 über derselben Operationenmenge SIJ nur durch ihre Schräggantenmengen.

Aufgrund der Definition der transitiven Hülle ist klar, dass zu jeder Schräggante in der transitiven Orientierung S^{tr} auf $[S^{tr}]$ ein regulärer Weg existiert, der diese Schräggante induziert.

Lemma 3.10 (Schrägganten und Wege) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan, und sei $e = ab$ eine Schräggante in S^{tr} . Dann existiert in S ein Weg $W \subseteq S$ von a nach b .

Teilweg und minimaler Teilweg

Mit Hilfe der Zerlegung der Kantenmenge eines H-Comparabilitygraphen in reguläre Kanten und Schrägganten sind wir in der Lage, jeden Sachverhalt auf einem der beiden Graphen $G(S)$ bzw. $[S^{tr}]$ auch durch den jeweils anderen Graphen beschreiben zu können.

Eine wichtige Eigenschaft von regulären Wegen ist, dass wir zu jedem regulären Weg einen regulären Teilweg beschreiben können, der abwechselnd aus horizontalen und vertikalen Kanten besteht. Das ist bedeutsam, weil wir unsere Argumentation häufig auf solche besonderen Wege beschränken werden.

Obwohl wir Wege in Plänen überwiegend als Kantenmengen, und nicht als Knotenmengen in den zugehörigen Plangraphen interpretieren, ist es hilfreich, den Begriff eines Teilweges mit Hilfe von Knotenmengen zu definieren.

Definition 3.11 (Teilweg) Sei W ein Weg in $S \in \mathcal{S}_{SIJ}$ mit der Knotenmenge $V_W \subseteq SIJ$. Der Weg W^* mit der Knotenmenge $V_{W^*} \subseteq SIJ$ ist ein Teilweg von W , $W^* \subseteq W$, wenn $V_{W^*} \subseteq V_W$ gilt, und W^* ebenfalls ein Weg in S ist.

Der Forderung, dass W^* seinerseits selbst ein Weg in S , also ein regulärer Weg in $[S^{tr}]$, sein soll, kommt dabei eine entscheidende Bedeutung zu. Nicht jede Teilmenge der Knoten eines Weges W in $S \in \mathcal{S}_{SIJ}$ beschreibt einen Teilweg von W in S . Zwar ist in $[S^{tr}]$ jede Teilmenge der Knoten eines Weges wieder ein Weg in $[S^{tr}]$. Einen Weg in S beschreibt eine solche Teilmenge der Knoten aber nur dann, wenn die zugehörige Kantenmenge der direkten Nachfolger der jeweiligen Knoten keine Schrägkanten enthält. Betrachten wir beispielsweise in Abbildung 3.3 den Weg W von Operation O_{12} über O_{14} und O_{34} nach O_{24} . Die Teilmenge $\{O_{12}, O_{34}, O_{24}\} \subseteq V_W$ beschreibt zwar einen Weg in B_1^{tr} , aber keinen regulären Weg, da die Kante $O_{12}O_{34} \in B_1^{tr}$ eine Schrägkante ist. Der einzige echte Teilweg in W von O_{12} nach O_{24} ist der Weg W^* mit $V_{W^*} = \{O_{12}, O_{14}, O_{24}\}$.

Lemma 3.12 (minimaler Teilweg) *Sei W ein Weg in $S \in \mathcal{S}_{SIJ}$ von a nach b . Dann existiert in S ein Teilweg $W^* \subseteq W$ von a nach b , der keine zwei aufeinanderfolgenden horizontalen oder vertikalen Kanten enthält. Wir nennen W^* den minimalen Teilweg von W , bzw. einen minimalen Weg in S von a nach b .*

Beweis. Sei $W \subseteq S$ ein beliebiger Weg in $S \in \mathcal{S}_{SIJ}$, und sei $V_W \subseteq SIJ$ die Menge seiner Knoten. Wir können W in Abschnitte von aufeinanderfolgenden horizontalen bzw. vertikalen Kanten unterteilen. Da die Graphen G_{MO} und G_{JO} transitiv orientiert sind, gilt für jeden dieser Abschnitte, dass der Startknoten v der jeweils ersten Kante und der Endknoten w der jeweils letzten Kante in $G(S)$ durch die Kante $vw \in S$ verbunden ist. Damit können wir jeden dieser Abschnitte auf genau eine Kante reduzieren, und erhalten so einen eindeutig bestimmten Teilweg W^* , der nicht weiter verkürzt werden kann. W^* besteht dann abwechselnd aus horizontalen und vertikalen Kanten. ■

Ecken und Eckkanten

Zur Beschreibung von einzelnen Abschnitten in minimalen regulären Wegen werden wir gelegentlich auch die beiden folgenden Begriffe benutzen.

Definition 3.13 (Ecke, Eckkante) *Sei $S \in \mathcal{S}_{SIJ}$ ein Plan. Ein minimaler regulärer Weg der Kantenlänge 2 heißt eine Ecke. Die durch diese Ecke induzierte Schrägkante nennen wir eine Eckkante.*

Ein minimaler regulärer Weg besteht damit ausschließlich aus Ecken. Jedoch ist nicht jede der durch einen solchen Weg induzierten Schrägkanten eine Eckkante. Das gilt nur, wenn der reguläre Weg nur aus einer Ecke besteht.

Reguläre Wege und Cliques

Bei der Betrachtung von transitiven Orientierungen in Kapitel 1.2.1 haben wir darauf hingewiesen, dass jeder Weg der Kantenlänge l in einer transitiven Orientierung T mit einem vollständigen Graphen K_{l+1} in $[T]$ korrespondiert, und umgekehrt. Das bedeutet, dass jeder Weg in S eine Clique in $[S^{tr}]$ induziert. Damit enthält $[S^{tr}]$ nicht nur Cliques in jeder Zeile oder Spalte, sondern auch solche, die Schrägkanten enthalten. Die Cliques, die mit minimalen regulären Wegen korrespondieren, haben die Eigenschaft, dass sie aus jeder Zeile oder Spalte höchstens zwei Knoten enthalten.

3.3.5 Ähnliche Pläne

In diesem Abschnitt wollen wir einige Zusammenhänge zwischen Plänen, ihren H-Comparabilitygraphen und beliebigen transitiven Orientierungen auf diesen Graphen betrachten. Dabei werden wir dem Reduktionsbegriff von Plänen, den wir in Kapitel 4 einführen wollen, schon ein wenig vorgreifen, und klären, was wir unter "ähnlichen" Plänen verstehen wollen. In diesem Zusammenhang werden wir eine etwas verallgemeinerte Auffassung von H-Comparabilitygraphen kennen lernen.

Planorientierung und Ähnlichkeit

Wir haben den H-Comparabilitygraphen $[S^{tr}]$ zu einem Plan $S \in \mathcal{S}_{SIJ}$ definiert, um eine graphentheoretische Beschreibung der Bearbeitungsreihenfolge der Operationen aus SIJ zu einer konkreten Lösung S des Open-Shop Schedulingproblems auf SIJ zu erhalten. Dabei betrachten wir den transitiven Abschluss aus den jeweiligen Reihenfolgen innerhalb der einzelnen Aufträge und auf den einzelnen

Maschinen. Der symmetrische Abschluss von dieser transitiven Hülle S^{tr} ist ein Comparabilitygraph, der die Eigenschaft hat, dass er eine transitive Orientierung S besitzt, bei der jede der Schrägkanten eine transitive Kante ist. Solche Orientierungen spielen eine besondere Rolle.

Definition 3.14 (Planorientierung) Sei $[S^{tr}]$ der H-Comparabilitygraph zu einem Plan $S \in \mathcal{S}_{SIJ}$. Eine transitive Orientierung $T \in \mathcal{T}_{[S^{tr}]}$ heißt Planorientierung von $[S^{tr}]$, wenn jede Schrägkante in T transitiv ist.

Nicht jede transitive Orientierung eines H-Comparabilitygraphen besitzt diese Eigenschaft. Der H-Comparabilitygraph zu dem Plan B_1 in Abbildung 3.3 besteht beispielsweise aus 2 verschiedenen Implikationsklassen, und hat daher genau $\frac{|\mathcal{T}_{[B_1^{tr}]}}{2} = 2$ verschiedene transitive Orientierungen. Von diesen 2 Orientierungen ist jedoch nur eine Planorientierung. In der anderen sind die Schrägkanten zwischen den Operationen O_{12} und O_{21} , bzw. O_{14} und O_{21} nicht transitiv.

Aus der Definition der Planorientierung folgt unmittelbar die folgende Aussage.

Lemma 3.15 (Planorientierung) Sei T eine Planorientierung auf $[S^{tr}]$. Dann beschreibt T einen Plan $T^* \in \mathcal{S}_{SIJ}$ mit $[(T^*)^{tr}] = [S^{tr}]$.

Definition 3.16 (ähnliche Pläne) Zwei Pläne $S_1, S_2 \in \mathcal{S}_{SIJ}$, für die $[S_1^{tr}] = [S_2^{tr}]$ gilt, heißen ähnlich, $S_1 \sim S_2$.

Umkehrplan

In Kapitel 1.2.1 haben wir vermerkt, dass wir zwei transitive Orientierungen T_1 und T_2 auf einem Comparabilitygraphen $G = (V, E)$ als verschieden ansehen wollen, wenn keine der beiden der jeweils anderen oder deren Umkehrung entspricht, $T_1 \neq T_2, T_2^{-1}$. Wir wollen auch im Zusammenhang mit H-Comparabilitygraphen eine Planorientierung und ihre Umkehrorientierung nicht als verschieden betrachten. Für S^{tr} und $(S^{tr})^{-1}$ gilt nämlich insbesondere, dass jede Kante, die in S^{tr} transitiv ist, auch in $(S^{tr})^{-1}$ transitiv ist. Also ist $(S^{tr})^{-1}$ ebenfalls eine Planorientierung von $[S^{tr}]$.

Definition 3.17 (Umkehrplan) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan. Der durch die Planorientierung $(S^{tr})^{-1}$ auf $[S^{tr}]$ induzierte Plan heißt Umkehrplan von S und wird mit S^{-1} bezeichnet. Es gelten $G(S^{-1}) = G(S)^{-1}$ und $(S^{-1})^{tr} = (S^{tr})^{-1}$. Insbesondere sind S und S^{-1} ähnliche Pläne, $S \sim S^{-1}$.

Die Bezeichnung des Umkehrplans von S mit S^{-1} ist wohldefiniert, da auch im Plangraphen die Umkehrorientierung von S der Orientierung des Umkehrplans entspricht, $(S)^{-1} = S^{-1}$.

Bei der Auffassung von S als Rangmatrix mit den Einträgen $1, \dots, r_{\max}$ kann die Rangmatrix von S^{-1} im Wesentlichen durch Ersetzen jedes Eintrags s_{ij} durch den Eintrag $r_{\max} + 1 - s_{ij}$ bestimmt werden. Einige Einträge müssen anschließend unter Umständen noch ein wenig nach unten angepasst werden, damit die Sequenzbedingung erfüllt ist. Auf diese Anpassungen wollen wir hier jedoch nicht näher eingehen (siehe hierzu beispielsweise BRÄSEL ET AL. [13] oder BRÄSEL [11]). Uns soll genügen, dass die Bestimmung der Rangmatrix zu S^{-1} deterministisch in polynomieller Zeit machbar ist.

Betrachten wir als Beispiel ein weiteres Mal den Plan B_1 aus Beispiel 3.3.

$$B_1 = \begin{bmatrix} 3 & 1 & 2 \\ 5 & 2 & 1 & 4 \\ 1 & 2 & 3 \end{bmatrix} \quad B_1^* = \begin{bmatrix} \mathbf{3} & 5 & 4 \\ 1 & \mathbf{4} & 5 & 2 \\ 5 & 4 & 3 \end{bmatrix} \quad B_1^{-1} = \begin{bmatrix} \mathbf{2} & 5 & 4 \\ 1 & \mathbf{3} & 5 & 2 \\ 5 & 4 & 3 \end{bmatrix}$$

Die Matrix B_1^* , die durch die Umformung $b_{1ij}^* = r_{\max} + 1 - b_{1ij}$ entsteht, beschreibt zwar die Umkehrorientierung von B_1 im Plangraphen $G(B_1)$, liefert aber keine Rangmatrix im engeren Sinne. Für die Operationen $b_{111}^* = 3$ und $b_{122}^* = 4$ gilt jeweils, dass der Eintrag $b_{1ij}^* - 1$ weder in der jeweiligen Zeile, noch in der jeweiligen Spalte existiert. Diese Einträge müssen daher vermindert werden, bis die Sequenzbedingung erfüllt ist.

Ähnliche Pläne durch andere Planorientierungen

Wir wissen inzwischen, dass es zu jedem Plan S sogenannte ähnliche Pläne geben kann. Solche Pläne zeichnen sich dadurch aus, dass ihre H-Comparabilitygraphen nicht nur isomorph zu dem von S sind. Bei gleicher Darstellung bzgl. SIJ sind sie sogar identisch. Ein Beispiel für einen ähnlichen Plan zu einem Plan S ist sein Umkehrplan S^{-1} . Daneben kann es jedoch weitere geben. Jede Planorientierung T^{tr} von $[S^{tr}]$ können wir eindeutig mit einem Plan T assoziieren, der ähnlich zu S ist. Auf die Frage, warum wir solche Pläne als “ähnlich” zu S bezeichnen, werden wir im folgenden Kapitel eingehen.

Der H-Comparabilitygraph zu dem Plan B_2 hat beispielsweise zwei verschiedene Planorientierungen. D.h. neben B_2 und B_2^{-1} gibt es einen weiteren Plan B_3 und dessen Umkehrung, die beide jeweils den gleichen H-Comparabilitygraphen induzieren, $[B_2^{tr}] = [B_3]$.

$$B_2 = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 3 & 1 \end{bmatrix} \quad B_3 = \begin{bmatrix} 4 & 3 & 1 \\ 1 & 2 & 3 \end{bmatrix}$$

Ähnliche Pläne durch Permutation oder Spiegelung

Man kann ähnliche Pläne somit auch als isomorphe Abbildungen des H-Comparabilitygraphen auf sich selbst, also als Automorphismen, betrachten. Eine hinreichende, aber nicht notwendige Bedingung für die Existenz von ähnlichen Plänen zu einem gegebenen Plan, ist die Existenz von Automorphismen durch Permutation von Zeilen oder Spalten, bzw. durch Spiegelung. Der H-Comparabilitygraph verfügt in diesem Fall über eine Symmetrie, über die der Plangraph oder die Rangmatrix nicht verfügt.

Lemma 3.18 (Automorphismen durch Permutation oder Spiegelung) *Sei $[S^{tr}]$ der H-Comparabilitygraph zu einem Plan $S \in \mathcal{S}_{SIJ}$. Existiert auf $[S^{tr}]$ ein Automorphismus, der durch eine Permutation von Zeilen oder Spalten, bzw. durch eine Spiegelung an einer Hauptdiagonalen erzeugt wird, so wird durch diesen Automorphismus ein ähnlicher Plan S^* zu S beschrieben, $S^* \sim S$.*

Beweis. Das Vertauschen von Zeilen oder Spalten entspricht der Umbenennung von Aufträgen oder Maschinen und hat keine Auswirkungen auf das zugrundeliegende Schedulingproblem. Ebenso hat das Spiegeln, das einem Vertauschen der Rollen von Aufträgen und Maschinen entspricht, keine Auswirkungen auf ein Open-Shop Problem.

Entsteht also auf diese Weise aus $G = [S^{tr}]$ ein Graph G^* mit $G^* = [S^{tr}]$, so ist die transformierte Orientierung $(S^{tr})^*$ eine Planorientierung auf $[S^{tr}]$: Jede Schrägkante von G ist auch in G^* eine Schrägkante. Jede reguläre Kante von G ist regulär in G^* . Und jede Schrägkante in G^* ist transitiv, da $(S^{tr})^*$ isomorph zu S^{tr} ist. Wegen $(S^{tr})^* \neq S^{tr}$ beschreibt $(S^{tr})^*$ somit einen Plan $S^* \neq S$ mit $S^* \sim S$. ■

Der H-Comparabilitygraph zu B_2 besteht beispielsweise aus dem $K_2 \times K_3$ und genau zwei Schrägkanten, die ein Kreuz bilden. Eine Vertauschung der beiden Zeilen liefert einen Automorphismus auf $[B_2^{tr}]$, die zu dem ähnlichen Plan B_3 führt. In $[B_4^{tr}]$ (vier Schrägkanten) können die ersten beiden Spalten vertauscht werden, und $[B_5^{tr}]$ kann an der Hauptdiagonalen gespiegelt werden. Beide H-Comparabilitygraphen haben genau wie $[B_2^{tr}]$ jeweils genau zwei verschiedene Planorientierungen. Der H-Comparabilitygraph zum Plan B_6 enthält alle sechs möglichen Schrägkanten. $[B_6^{tr}]$ entspricht damit dem vollständigen Graphen auf sechs Knoten. Hier beschreibt jede beliebige Kombinationen von Permutationen der Zeilen und Spalten einen Automorphismus. Von den $6! = 720$ transitiven Orientierungen auf $[B_6^{tr}] = K_6$ sind insgesamt genau 60 Orientierungen Planorientierungen (30 verschiedene Planorientierungen und ihre Umkehrorientierungen).

$$B_4 = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 3 & 5 \end{bmatrix} \quad B_5 = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 3 \\ 2 \end{bmatrix} \quad B_6 = \begin{bmatrix} 1 & 2 & 3 \\ 6 & 5 & 4 \end{bmatrix}$$

Die Bedingung aus Lemma 3.18 ist jedoch nur hinreichend, aber nicht notwendig für die Existenz von ähnlichen Plänen. Die Pläne B_7 und B_8 sind ähnlich, aber weder gilt $B_7 = B_8^{-1}$, noch geht einer der beiden durch Permutation oder Spiegelung aus dem anderen hervor.

$$B_7 = \begin{bmatrix} 5 & 6 & 1 & 2 \\ 1 & & 2 & \\ 4 & & 3 & \\ 2 & & & \end{bmatrix} \quad B_8 = \begin{bmatrix} 5 & 6 & 4 & 1 \\ 4 & & 3 & \\ 1 & & 2 & \\ 2 & & & \end{bmatrix}$$

Alternative Betrachtung von H-Comparabilitygraphen

Wir haben (in Abschnitt 3.3.3) den H-Comparabilitygraphen zu einem Plan als symmetrischen Abschluss der transitiven Hülle des Plans definiert. Anschließend haben wir festgestellt, dass jeder H-Comparabilitygraph aus dem H-Graphen G_{SIJ} und einer von dem Plan abhängigen Schrägkantenmenge besteht. Für die Schrägkanten gilt dabei, dass sie in mindestens einer transitiven Orientierung auf dem Graphen alle gleichzeitig transitiv sind. Eine transitive Orientierung mit dieser Eigenschaft haben wir Planorientierung genannt. Für eine vollständige Operationenmenge $SIJ = I \times J$ mit $|I| = n$ und $|J| = m$ bedeutet das, dass wir jedem Plan $S \in \mathcal{S}_{SIJ}$ einen Comparabilitygraphen G zuordnen können, der aus dem $K_n \times K_m$ und einer Schrägkantenmenge $E_{diag}(S)$ besteht. Falls G mehrere verschiedene Planorientierungen besitzt, existieren mehrere zu S ähnliche Pläne.

Neben dieser von einem konkreten Plan ausgehenden Auffassung von H-Comparabilitygraphen gibt es auch eine alternative Betrachtungsweise. Wir können nämlich auch umgekehrt jedem Graphen $G = (SIJ, E_{G_{SIJ}} + E_{diag})$ auf der Knotenmenge SIJ , der den H-Graphen G_{SIJ} als Teilgraphen enthält, und auf dem eine Planorientierung existiert, einen Plan zuordnen. Genauer gesagt, ordnen wir jeder Planorientierung $T^{tr} \in \mathcal{T}_G$ einen Plan $T \in \mathcal{S}_{SIJ}$ zu. So können wir beispielsweise einem Graphen G^* , der aus dem $K_2 \times K_3$ und zwei Schrägkanten besteht, die in G^* ein Kreuz bilden, zwei verschiedene Pläne T_1 und T_2 zuordnen, die isomorph zu den Plänen B_2 bzw. B_3 sind.

Eine notwendige Bedingung für eine solche Zuordnung ist dabei offenkundig die Existenz einer transitiven Orientierung auf G . D.h. G muss ein Comparabilitygraph sein. Die Existenz einer transitiven Orientierung auf G ist jedoch nicht hinreichend für die Existenz einer Planorientierung auf G . Ein Graph G^{**} beispielsweise, der aus dem $K_2 \times K_3$ und zwei Schrägkanten besteht, die *kein* Kreuz bilden, ist zwar unabhängig von der Anordnung der beiden Schrägkanten ein Comparabilitygraph, der jedoch in keinem Fall eine Planorientierung besitzt.

Allerdings können wir auch solchen Comparabilitygraphen G , die zwar den H-Graphen enthalten, aber keine Planorientierung besitzen, einen oder sogar mehrere Pläne zuordnen. Der Unterschied zwischen einer Planorientierung auf G , und einer beliebigen transitiven Orientierung ist, dass in einer beliebigen Orientierung $T \in \mathcal{T}_G$ einige der Schrägkanten vielleicht nicht transitiv sind. Eine solche nicht transitive Schrägkante gehört damit zur transitiven Reduktion T^r von T . Nach Satz 1.39 aus Abschnitt 1.4.1 können wir Kanten aus der transitiven Reduktion einer transitiven Orientierung jedoch löschen, ohne die Transitivität der Orientierung zu gefährden.

Das bedeutet, dass wir aus jeder transitiven Orientierung $T \in \mathcal{T}_G$ von G eine Planorientierung T^* auf einem Teilgraphen G^* von G erzeugen können, indem wir alle in T nicht transitiven Schrägkanten einfach entfernen. Dieser Teilgraph enthält dann nach wie vor den H-Graphen G_{SIJ} als Teilgraphen, da wir nur Schrägkanten entfernt haben. Der Plan $S_{T^*} \in \mathcal{S}_{SIJ}$, der durch die Planorientierung T^* auf G^* induziert wird, ist durch T eindeutig bestimmt, $S_{T^*} = S_T$. Wir können diesen Plan somit zum einem dem H-Comparabilitygraphen G^* , zum anderen aber auch dem Comparabilitygraphen G zuordnen.

Weiter oben haben wir beschrieben, dass der H-Comparabilitygraph zu dem Plan B_1 aus Beispiel 3.3 (Abbildung 3.3) zwei verschiedene transitive Orientierungen besitzt. Eine Orientierung ist eine Planorientierung und repräsentiert zusammen mit ihrer Umkehrung den Plan B_1 und seinen Umkehrplan $B_1^{-1} = B_2$. Die andere Orientierung enthält nur 7 der ursprünglich 9 Schrägkanten als transitive Kanten. Dieser Orientierung können wir den Plan B_9 , bzw. den Plan $B_{10} = B_9^{-1}$ zuordnen.

$$B_9 = \begin{bmatrix} 3 & 1 & 2 \\ 1 & 3 & 2 & 5 \\ 2 & & 3 & 4 \end{bmatrix} \quad B_{10} = B_9^{-1} = \begin{bmatrix} 1 & 4 & 3 \\ 5 & 2 & 4 & 1 \\ 4 & 3 & 2 \end{bmatrix}$$

Im nächsten Kapitel werden wir sehen, dass die Pläne B_9 und B_{10} auf eine bestimmte Art und Weise "besser" als die Pläne B_1 und B_2 sind.

Vor dem Hintergrund dieser Betrachtungen können wir den Begriff des H-Comparabilitygraphen ein wenig ausweiten.

Definition 3.19 (H-Comparabilitygraph) *Ein Comparabilitygraph $G = (V, E)$ auf der Knotenmenge $V = SIJ$, der den H-Graphen G_{SIJ} als Teilgraphen enthält, nennen wir einen H-Comparabilitygraphen.*

Anders ausgedrückt, ist jeder Graph, der aus dem H-Graphen G_{SIJ} und einer beliebigen Schrägkantenmenge E_{diag} besteht, genau dann ein H-Comparabilitygraph, wenn er ein Comparabilitygraph ist. Bei BRÄSEL, DHAMALA UND MATZKE (2007) [12] werden einige Eigenschaften solcher H-Comparabilitygraphen beschrieben.

Bemerkung 3.20 ($K_2 \times K_3$) *Der $K_2 \times K_3$ ist kein Comparabilitygraph. Damit ist jeder Graph, der aus dem H-Graphen G_{SIJ} und einer beliebigen Schrägkantenmenge E_{diag} besteht, und den $K_2 \times K_3$ als induzierten Teilgraphen enthält, auch kein Comparabilitygraph. Insbesondere ist daher die Schrägkantenmenge für jeden H-Comparabilitygraphen zu einer vollständigen Operationenmenge SIJ mit (o.B.d.A.) mindestens 2 Aufträgen und 3 Maschinen, nicht leer.*

Die eben diskutierten Zusammenhänge zwischen H-Comparabilitygraphen und ihren transitiven Orientierungen können wir zusammenfassend folgendermaßen formulieren.

Lemma 3.21 (H-Comparabilitygraph) *Sei G ein H-Comparabilitygraph auf SIJ . Dann gelten*

- (i) *Jede transitive Orientierung $T \in \mathcal{T}_G$ korrespondiert mit einem Plan $S_T \in \mathcal{S}_{SIJ}$.*
- (ii) *Der H-Comparabilitygraph $[S_T^{tr}]$ zu dem durch $T \in \mathcal{T}_G$ induzierten Plan $S_T \in \mathcal{S}_{SIJ}$ ist ein Teilgraph von G , $[S_T^{tr}] \subseteq G$.*
- (iii) *$[S_T^{tr}]$ ist genau dann kein echter Teilgraph, d.h. $[S_T^{tr}] = G$, wenn T eine Planorientierung auf G ist.*

3.3.6 Problemformulierungen für $O \parallel C_{\max}$

Am Anfang dieses Kapitels haben wir in Abschnitt 3.1.3 das Open-Shop Schedulingproblem mit C_{\max} -Zielfunktion folgendermaßen formuliert. Gegeben sei die Operationenmenge $SIJ \subseteq I \times J$ und Bearbeitungszeiten p_{ij} für alle Operationen $O_{ij} \in SIJ$. Für eine Probleminstanz von $O \parallel C_{\max}$ sind die Bearbeitungszeiten sowohl fest, als auch bekannt. Sie unterliegen also weder stochastischen Prozessen, noch ist ihre Kenntnis mit Unsicherheiten behaftet. Das Optimierungsproblem besteht nun darin, eine zulässige Kombination von technologischen Reihenfolgen innerhalb der Aufträge und organisatorischen Reihenfolgen auf den Maschinen zu bestimmen, die die Gesamtbearbeitungszeit $C_{\max} = \max_{i \in I} \{C_i\}$ minimiert. Das zugehörige Entscheidungsproblem fragt nach der Existenz einer zulässigen Kombination, die die Gesamtbearbeitungszeit einer gegebenen Lösung unterbietet.

Vor dem Hintergrund der graphentheoretischen Beschreibung dieses Problems in den letzten Abschnitten wollen wir nun noch einige Varianten der Problemformulierung sammeln.

Azyklische Orientierung auf disjunktivem Graph

Die direkte Übersetzung des eben geschilderten Sachverhalts in ein graphentheoretisches Problem lautet: Gegeben sei der disjunktive Graph $G = (V, C, D) = G_{SIJ}$ mit Knotengewichtungen $w_{ij} = p_{ij}$. Eine zulässige Lösung einer Instanz des Schedulingproblems wird dann durch eine azyklische Orientierung auf dem gewichteten H-Graphen G_{SIJ} repräsentiert. Das Gewicht eines schwersten Weges von einer Quelle zu einer Senke in einer solchen Orientierung entspricht der Gesamtbearbeitungszeit C_{\max} . Gesucht ist somit eine azyklische Orientierung auf G_{SIJ} , die das Gewicht eines schwersten Weges minimiert.

Diese Beschreibung ist für die algorithmische Umsetzung von Strategien zur Lösung des Problems geeignet.

Transitive Orientierung auf einem H-Comparabilitygraph

Eine dazu äquivalente Beschreibung ist die folgende. Gegeben sei die Operationenmenge SIJ und die Bearbeitungszeiten $P = [p_{ij}]$. Wir können jeder transitiven Orientierung auf einem H-Comparabilitygraphen über SIJ eine zulässige Lösung des Schedulingproblems zuordnen. Zur Lösung einer Problem Instanz ist somit ein H-Comparabilitygraph G mit einem minimalen schwersten Weg bezüglich der Knotengewichtung $w_{ij} = p_{ij}$ in einer transitiven Orientierung auf G gesucht.

Diese Beschreibung ist vor allem im Hinblick auf die Theorie der Reduzierbarkeit von Plänen, die wir im nächsten Kapitel aufgreifen wollen, hilfreich.

Intervallfärbung auf H-Comparabilitygraphen

Zuletzt wollen wir noch eine etwas ausgefallenerere Problemformulierung präsentieren, die im weiteren Verlauf dieser Arbeit keine weitere Rolle spielen wird.

In Kapitel 1.4.4 haben wir den Begriff der perfekten und superperfekten Graphen diskutiert, und bemerkt, dass Comparabilitygraphen sowohl perfekt, als auch superperfekt sind. Ein Graph $G = (V, E)$ ist perfekt, wenn für jeden induzierten Teilgraphen $A \subseteq V$ die Cliquenzahl $\omega(G)$ mit der chromatischen Zahl $\chi(G)$ übereinstimmt. Er ist superperfekt, wenn er diese Eigenschaft auch für jede nichtnegative Gewichtung w der Knoten besitzt. Dabei werden jedoch gewichtete Cliques und Intervall-Färbungen betrachtet. Bei einer Intervall-Färbung wird jedem Knoten $v \in V$ mit dem Gewicht $w(v)$ ein Intervall von $w(v)$ aufeinanderfolgenden Farben zugeordnet.

Betrachten wir nun das Open-Shop Schedulingproblem $O \parallel C_{\max}$, und betrachten wir hier zunächst den mit den Bearbeitungszeiten $P = [p_{ij}]$ gewichteten H-Comparabilitygraphen $G = [S^{tr}]$ zu einem Plan $S \in \mathcal{S}_{SIJ}$. Zu dem Plan S können wir mit Hilfe von P den Schedule $C = [c_{ij}]$ bestimmen. Färben wir nun in G jeden Knoten O_{ij} mit den Farben $\{c_{ij} - p_{ij} + 1, c_{ij} - p_{ij} + 2, \dots, c_{ij} - p_{ij} + p_{ij}\}$, so haben wir eine zulässige Intervall-Färbung von G : Jeder Knoten ist mit p_{ij} aufeinanderfolgenden Farben gefärbt, und benachbarte Knoten haben keine Farbe gemeinsam, da Operationen, die in $[S^{tr}]$ benachbart sind, in S unter keinen Umständen gleichzeitig bearbeitet werden. Die Anzahl der benötigten Farben entspricht der Gesamtbearbeitungszeit $C_{\max}(S)$ zu dem Plan S , und es gilt $C_{\max}(S) = \chi_w(G)$. BRÄSEL, DHAMALA UND MATZKE (2007) [12] zeigen, dass auch für beliebige H-Comparabilitygraphen sowohl die Intervall-Färbungszahl $\chi_w(G)$, als auch die gewichtete Cliquenzahl $\omega_w(G)$ polynomiell bestimmbar ist.

Das Optimierungsproblem $O \parallel C_{\max}$ besteht nun darin, zu einer gegebenen Operationenmenge SIJ und gegebenen Bearbeitungszeiten $P = [p_{ij}]$ einen H-Comparabilitygraphen G zu finden, der einen H-Comparabilitygraphen G^* mit minimaler Intervall-Färbungszahl $\chi_w(G^*)$ als Teilgraphen besitzt.

Diese Problemformulierung ist für praktische Anwendungen völlig ungeeignet. Sie liefert jedoch einen interessanten Zusammenhang zwischen den Lösungen eines Open-Shop Problems und Intervall-Färbungen auf superperfekten Graphen.

3.4 Die Komplexität von Schedulingproblemen

In den vorangegangenen Abschnitten haben wir uns mit der Beschreibung von Shop Schedulingproblemen im Allgemeinen und der des Open-Shop Problems $O \parallel C_{\max}$ im Besonderen befasst. In diesem Abschnitt wollen wir uns kurz mit der Komplexität von solchen Problemen auseinandersetzen.

Komplexitätshierarchien zwischen verschiedenen Klassen von Schedulingproblemen

Basierend auf der $\alpha \mid \beta \mid \gamma$ -Notation (Abschnitt 3.1.2) ist eine Darstellung von Komplexitätshierarchien zwischen verschiedenen Klassen von Schedulingproblemen möglich (PINEDO [56] oder BŁAŻEWICZ ET AL. [9]). Diese ermöglichen den Vergleich zweier verwandter Schedulingprobleme hinsichtlich ihrer Zeitkomplexität. Von besonderem Interesse ist in diesem Zusammenhang die jeweiligen Grenze zwischen polynomiell lösbaren Problemen und NP-vollständigen Problemen. Bei BRUCKER UND KNUST (2002) [19] ist eine umfangreiche Beschreibung dieser Grenze für eine Vielzahl von Problemklassen angegeben.

Die meisten Schedulingprobleme sind NP-schwer

Einen aussagekräftigen Überblick über die Bedeutung der Komplexität bei der Betrachtung von Schedulingproblemen liefert nach wie vor die Arbeit von LAGEWEG ET AL. [43] von 1981. In dieser Studie haben die Autoren 4.536 verschiedene (Entscheidungs-) Probleme der Form $\alpha | \beta | \gamma$ auf ihren Komplexitätsstatus hin untersucht. Die Kenntnis der Komplexitätshierarchien erlaubte dabei Aussagen über große Mengen von Problemen, ohne alle Probleme einzeln überprüfen zu müssen. Bereits 1981 konnte dadurch für mehr als 84% der betrachteten Probleme (3.821) die NP-Vollständigkeit gezeigt werden. Andererseits konnte nur von etwa 9% der untersuchten Probleme (417) die Zugehörigkeit zu P gezeigt werden. Für die verbleibenden 7% der Probleme (298) war der Komplexitätsstatus noch nicht bekannt.

Auch wenn sich in den vergangenen mehr als 25 Jahren die Anzahl der durch die $\alpha | \beta | \gamma$ -Klassifizierung beschreibbaren Probleme zweifellos erhöht hat, hat sich an dem grundlegenden Bild nichts geändert. Die überwiegende Mehrzahl aller Schedulingprobleme ist NP-schwer. Polynomiell lösbare Probleme sind große Ausnahmen, und in aller Regel sehr starke Vereinfachungen von den jeweiligen allgemeinen Problemen. Selbst in der Klasse der Einmaschinenprobleme $1 | \beta | \gamma$ liegt die Grenze in der Komplexitätshierarchie zwischen P und NPC nicht sehr weit oben.

Die Komplexität von Open-Shop Problemen

Die Grenze zwischen P und NPC in den Komplexitätshierarchien liegt damit über alle Problemklassen hinweg relativ weit unten (vgl. BRUCKER UND KNUST [19]). Für Open-Shop Probleme mit C_{\max} -Zielfunktion stellt sich diese Grenze beispielsweise folgendermaßen dar. Während das Open-Shop Problem mit maximal 2 zur Verfügung stehenden Maschinen $O | m = 2 | C_{\max}$ bzw. $O2 || C_{\max}$ noch polynomiell lösbar ist (GONZALES UND SAHNI (1976) [36]), ist es bei 3 zur Verfügung stehenden Maschinen bereits NP-schwer ([36]). Damit gilt auch für das allgemeine Open-Shop Problem $O || C_{\max}$ die NP-Vollständigkeit.

Polynomiell lösbar sind jedoch die beiden folgenden wichtigen Vereinfachungen. Zum Einen wird das Open-Shop Problem auch für mehr als 3 Maschinen wieder "leicht", wenn man die Unterbrechung der Bearbeitung von Aufträgen zulässt (preemption). Zum Anderen führt auch die Einschränkung auf Einheitsbearbeitungszeiten für alle Operationen zu einem polynomiell lösbaeren Problem (vgl. [19]).

Auf der anderen Seite wird auch das einzige polynomiell lösbare allgemeine Open-Shop Problem, $O2 || C_{\max}$, bereits durch die Hinzunahme von Bereitstellungszeiten (release-dates) NP-schwer. Es ist $O2 | r_i | C_{\max} \in \text{NPC}$ (LAWLER, LENSTRA UND RINNOY KAN (1981) [44]).

Nachdem wir nun mit der Beschreibung des Open-Shop Problems $O || C_{\max}$ und einigen Eigenschaften dieser Beschreibung vertraut sind, können wir uns im folgenden Kapitel der Formulierung der eigentlichen Fragestellung dieser Arbeit widmen.

Kapitel 4

Das Problem IRREDUCIBILITY

In den vorangegangenen Kapiteln haben wir drei sehr verschiedene Themengebiete beleuchtet: Comparabilitygraphen in Kapitel 1, Komplexitätstheorie in Kapitel 2 und Open-Shop Scheduling in Kapitel 3. Es existieren viele Fragestellungen die diese Themen paarweise miteinander verknüpfen. Auf einige davon sind wir bereits eingegangen: Fragen zur Komplexität von Problemen stellen sich für alle algorithmischen Probleme, insbesondere natürlich auch für Probleme aus der Graphentheorie oder der Schedulingtheorie. Und bei der Beschreibung von Open-Shop Plänen haben wir auf Comparabilitygraphen zurückgegriffen.

In diesem Kapitel wollen wir nun eine Fragestellung aus der Schnittmenge aller drei genannten Themenfelder vorstellen: das Problem IRREDUCIBILITY. Der Beschreibung und Betrachtung dieses Problems ist der verbleibende Teil dieser Arbeit gewidmet.

In Abschnitt 4.1 werden wir den Begriff der Reduzierbarkeit von Plänen einführen, und eine erste Problembeschreibung von IRREDUCIBILITY und einem eng verwandten Problem vorstellen. Anschließend werden wir das Konzept der sogenannten Planimplikationsklassen kennen lernen (Abschnitt 4.2), dem eine entscheidende Bedeutung bei der Betrachtung der Reduzierbarkeit von Plänen zukommt. In Abschnitt 4.3 werden wir auf die strukturellen Gemeinsamkeiten von immer-transitiven Kanten und sogenannten stabilen Schrägkanten eingehen. In Abschnitt 4.4 werden wir die Komplexität einiger Probleme aus dem Umfeld von IRREDUCIBILITY betrachten, und in Abschnitt 4.5 schließlich werden wir uns mit der minimalen und maximalen Anzahl von Schrägkanten in H-Comparabilitygraphen beschäftigen.

4.1 Reduzierbarkeit und Irreduzibilität

Bei der Diskussion der Komplexitätstheorie in Kapitel 2 haben wir einen Reduktionsbegriff zwischen algorithmischen Problemen beschrieben. Wenn ein Problem A auf ein Problem B reduziert werden kann, $A \alpha B$, dann ist A ein Spezialfall des allgemeineren Problems B . Nun wollen wir einen völlig anderen Reduzierbarkeitsbegriff zwischen Lösungen des Open-Shop Problems betrachten. Während das reduzierende Problem B im Falle von $A \alpha B$ eine Verallgemeinerung von A ist, beschreibt im Falle der Reduzierbarkeit zwischen Open-Shop Plänen der reduzierende Plan eine Verbesserung des reduzierten Plans.

Wir betrachten das Open-Shop Problem $O \parallel C_{\max}$ mit Makespan-Kriterium über einer gegebenen Operationenmenge $SIJ \subseteq I \times J$. Einen Plan $S \in \mathcal{S}_{SIJ}$ können wir wahlweise durch eine Rangmatrix über der Operationenmenge, durch den Plangraphen $G(S)$ oder durch eine Planorientierung auf dem H-Comparabilitygraphen $[S^{tr}]$ beschreiben.

4.1.1 Reduzierbarkeit von Plänen

Die Theorie der Reduzierbarkeit von Plänen zu Open-Shop Problemen wurde eingeführt von KLEIN-AU (1992) [41], und aufgegriffen z.B. von BRÄSEL UND KLEINAU (1996) [16], BRÄSEL, HARBORTH, TAUTENHAHN UND WILLENIUS (1999) [13] und [14], und WILLENIUS (2000) [66].

Reduzierbarkeit

Betrachten wir zwei Pläne A und B über einer Operationenmenge SIJ , so kann es vorkommen, dass einer der beiden Pläne “besser” als der andere Plan ist. Der Plan A *reduziert* den Plan B , $A \preceq B$, wenn die Gesamtbearbeitungszeit von Plan A für keine Instanz, d.h. für keine mögliche Wahl von Bearbeitungszeiten, größer als die von Plan B ist. Dann gilt also $C_{\max}(A) \leq C_{\max}(B)$ für jede Wahl der Bearbeitungszeiten. Der Plan A ist in diesem Fall in dem Sinne “besser” als der Plan B , als dass er diesen *dominiert*. Für jede Instanz über SIJ ist der resultierende Schedule zu Plan A mindest so gut, wie der Schedule zu Plan B , in vielen Fällen möglicherweise sogar besser. Wir nennen A auch einen *reduzierenden* Plan von B .

Gilt für $C_{\max}(A) = C_{\max}(B)$ für jede Wahl der Bearbeitungszeiten, so nennen wir A und B *ähnlich*, $A \sim B$. In diesem Fall gilt sowohl $A \preceq B$, als auch $A \succeq B$. In Abschnitt 3.3.5 haben wir zwei Plänen S_1 und S_2 ähnlich genannt, $S_1 \sim S_2$, wenn ihre H-Comparabilitygraphen übereinstimmen, $[S_1^{tr}] = [S_2^{tr}]$. Wir werden noch sehen, dass diese beiden Definitionen äquivalent sind.

$$B_1 = \begin{bmatrix} 3 & 1 & 2 \\ 5 & 2 & 1 & 4 \\ 1 & & 2 & 3 \end{bmatrix} \quad B_9 = \begin{bmatrix} 3 & 1 & 2 \\ 1 & 3 & 2 & 5 \\ 2 & & 3 & 4 \end{bmatrix}$$

In Kapitel 3 haben wir die Pläne B_1 und B_9 kennen gelernt (Seite 58 bzw. 60). Dort haben wir gesehen, dass der H-Comparabilitygraph $[B_1^{tr}]$ aus zwei verschiedenen Implikationsklassen besteht. Er besitzt damit genau zwei verschiedene transitive Orientierungen, die die Pläne B_1 bzw. B_1^{-1} und B_9 bzw. B_9^{-1} beschreiben. Für die Pläne B_1 und B_9 gilt nun, dass jeder Weg in B_9 ein Teilweg eines Weges in B_1 oder B_1^{-1} ist. Damit gilt insbesondere, dass das Gewicht eines schwersten Weges in B_9 höchstens so groß wie das Gewicht eines schwersten Weges in B_1 sein kann. Umgekehrt bedeutet das, dass $C_{\max}(B_9) \leq C_{\max}(B_1)$ für jede Wahl der Bearbeitungszeiten gilt. Der Plan B_9 reduziert den Plan B_1 .

Irreduzible Pläne

In Abschnitt 3.3.5 haben wir bereits bemerkt, dass jeder Plan A ähnlich zu seinem Umkehrplan A^{-1} ist, $A \sim A^{-1}$, aber umgekehrt aus $A \sim B$ nicht $A = B$ oder $A = B^{-1}$ folgt. Reduziert ein Plan A einen Plan B , der nicht ähnlich zu A ist, $A \not\sim B$, dann wird B von A *streng reduziert*, $A \prec B$. In diesem Fall gilt, dass es mindestens eine Wahl von Bearbeitungszeiten gibt, für die $C_{\max}(A) < C_{\max}(B)$ gilt. Die minimalen Elemente bezüglich dieser Reduzierbarkeits-Relation werden irreduzibel genannt. Sie können als *effiziente* Lösungen des Open-Shop Problems betrachtet werden.

Definition 4.1 (irreduzibler Plan) *Ein Plan $A \in \mathcal{S}_{SIJ}$, der von keinem Plan streng reduziert wird, heißt irreduzibel.*

Zwei irreduzible Pläne A und B über derselben Operationenmenge SIJ sind im Allgemeinen nicht aufeinander reduzierbar. Für sie gilt, dass es jeweils mindestens eine Problem Instanz gibt, bezüglich der einer der Pläne echt besser als der andere ist, $C_{\max}(A) < C_{\max}(B)$ oder $C_{\max}(A) > C_{\max}(B)$.

Den folgenden Abschnitten schon etwas vorgreifend, wollen wir hier noch kurz festhalten, dass der Plan B_9 den Plan B_1 sogar streng reduziert, und darüberhinaus irreduzibel ist.

Universell-optimale Mengen

Betrachtet man alle irreduziblen Pläne zu einer Operationenmenge, so gibt es zu jedem irreduziblen Plan eine Menge von Instanzen, für die dieser Plan eine optimale Lösung liefert. Die Menge der irreduziblen Pläne enthält damit für jede mögliche Wahl der Bearbeitungszeiten eine optimale Lösung für das Open-Shop Problem $O \parallel C_{\max}$. Eine Teilmenge aller Pläne über SIJ mit dieser Eigenschaft wird *universell-optimale Menge* genannt. Auf die Eigenschaften von solchen universell-optimale Mengen werden wir im Abschnitt 4.1.5 noch einmal kurz eingehen.

4.1.2 Die Bedeutung irreduzibler Pläne

Ein Plan ist eine zulässige Lösung des Open-Shop Schedulingproblems $O \parallel C_{\max}$. Zu jedem Plan existiert mindestens ein reduzierender irreduzibler Plan. Für die Betrachtung des Optimierungsproblems wäre es daher wünschenswert, wenn man sich bei der Suche nach einer optimalen Lösung auf die Menge der effizienten Lösungen, d.h. der irreduziblen Pläne, beschränken könnte. Das gilt umso mehr, da die Anzahl der Pläne bereits für kleine Formate sehr groß wird (Tabelle 4.1), und der Anteil der irreduziblen Pläne sehr klein ist (Tabelle 4.2).

m	2	3	4	5	6	7
n						
2	14	204	5.016	185.520	9.595.440.	659.846.880
3		19.164	3.733.056	1.288.391.040	712.770.186.240	589.563.294.888.960
4			6.941.592.576	26.549.943.275.520	?	?

Tabelle 4.1: Anzahl zulässiger Pläne für eine vollständige Operationenmenge SIJ mit n Aufträgen und m Maschinen ([13])

n	m (%)	2	3	4	5	6
2		14.30	5.88	1.44	0.26	0.04
3			2.69	0.88	0.23	0.05
4				0.39	?	?

Tabelle 4.2: Anteil der irreduziblen Pläne an der Menge aller zulässigen Pläne (in %) ([13])

Vor diesem Hintergrund ist die Untersuchung der Eigenschaften irreduzibler Pläne und die Frage nach der Erkennung irreduzibler Pläne ein vielversprechender Ansatz auf dem Weg zu besseren Algorithmen für das Open-Shop Problem.

Wie wir in den kommenden Abschnitten noch sehen werden, ist die Fragestellung der Reduzierung eines Plans so eng mit der Frage nach der Erkennung der Reduzierbarkeit verbunden, dass es genügt, sich formal auf eine der beiden Fragestellungen zu konzentrieren.

4.1.3 Die Probleme IRRED und REDUCING

In diesem Abschnitt wollen wir die beiden Fragestellungen der Erkennung der Reduzierbarkeit (REDUCIBILITY) bzw. Irreduzibilität (IRREDUCIBILITY) eines Plans, und der Reduzierung (REDUCING) eines Plans formal beschreiben, und voneinander abgrenzen. Betrachten wir die folgenden drei Probleme:

Problem IRREDUCIBILITY (IRRED) (#1): Gegeben sei ein Plan $S \in \mathcal{S}_{SIJ}$ für das Open-Shop Problem $O \parallel C_{\max}$. Ist S irreduzibel?

Problem REDUCIBILITY: Gegeben sei ein Plan $S \in \mathcal{S}_{SIJ}$ für das Open-Shop Problem $O \parallel C_{\max}$. Ist S streng reduzierbar? D.h. existiert ein Plan $S^* \in \mathcal{S}_{SIJ}$ mit $S^* \prec S$?

Problem REDUCING: Gegeben sei ein Plan $S \in \mathcal{S}_{SIJ}$ für das Open-Shop Problem $O \parallel C_{\max}$. Finde, falls möglich, einen Plan $S^* \in \mathcal{S}_{SIJ}$ mit $S^* \prec S$.

Diese drei Probleme sind untereinander eng verwandt. Da bei REDUCIBILITY danach gefragt wird, ob der Plan $S \in \mathcal{S}_{SIJ}$ *nicht* irreduzibel ist, sind die Probleme REDUCIBILITY und IRRED komplementäre Probleme. Es ist $\text{co-IRRED} = \text{REDUCIBILITY}$, bzw. $\text{IRRED} = \text{co-REDUCIBILITY}$. Das Problem REDUCING ist das konstruktive Optimierungsproblem zum Entscheidungsproblem REDUCIBILITY.

Reduzierbarkeit zwischen zwei gegebenen Plänen

Ist neben dem Plan $S \in \mathcal{S}_{SIJ}$ ein zweiter Plan $S^* \in \mathcal{S}_{SIJ}$ gegeben, der mit S verglichen werden kann, dann kann in polynomieller Zeit überprüft werden, ob der Plan S^* den Plan S (streng) reduziert. Hierbei sind einmal mehr die H-Comparabilitygraphen zu den Plänen von Bedeutung.

Satz 4.2 (Reduzierbarkeit) [13],[14] *Seien A und B zwei Pläne auf der gleichen Operationenmenge SIJ . Dann reduziert der Plan A den Plan B , $A \preceq B$, genau dann, wenn der zu A gehörende H-Comparabilitygraph in dem zu B gehörenden H-Comparabilitygraphen enthalten ist, $[A^{tr}] \subseteq [B^{tr}]$. A reduziert B genau dann streng, $A \prec B$, wenn $[A^{tr}] \subsetneq [B^{tr}]$ gilt.*

Mit diesem Argument können wir leicht einsehen, dass der Plan B_9 auf Seite 60 den Plan B_1 auf Seite 58 streng reduziert: B_9^{tr} repräsentiert eine transitive Orientierung von $[B_1^{tr}]$, bei der zwei der neun Schrägkanten von $[B_1^{tr}]$ nicht transitiv sind. Für den H-Comparabilitygraphen $[B_9^{tr}]$ gilt damit $[B_9^{tr}] \subsetneq [B_1^{tr}]$.

Eine Halbordnung auf der Menge der H-Comparabilitygraphen

Die Reduzierbarkeitsrelation \prec induziert eine Halbordnung auf der Menge aller Pläne über der Operationenmenge SIJ . Die minimalen Elemente dieser Halbordnung sind die irreduziblen Pläne. Mit Hilfe von Satz 4.2 können wir diese Halbordnung in eine Halbordnung auf der Menge aller H-Comparabilitygraphen über SIJ übersetzen, die durch eine Teilmengenbeziehung beschrieben ist. Die minimalen Elemente in dieser Beschreibung sind gerade die H-Comparabilitygraphen, aus denen keine Menge der Schrägkanten mehr entfernt werden kann, ohne die Eigenschaft der transitiven Orientierbarkeit zu verlieren.

Vor dem Hintergrund der Überlegungen zu einer verallgemeinerten Auffassung von H-Comparabilitygraphen im vorherigen Kapitel (Abschnitt 3.3.5) können wir somit eine zweite Problemformulierung für IRRED angeben.

Problem IRREDUCIBILITY (IRRED) (#2): Gegeben sei ein Plan $S \in \mathcal{S}_{SIJ}$ für das Open-Shop Problem $O \parallel C_{\max}$. Sei $[S^{tr}]$ sein H-Comparabilitygraph, und sei G_{SIJ} der H-Graph zu der Operationenmenge SIJ . Existiert kein H-Comparabilitygraph G mit $G_{SIJ} \subseteq G \subsetneq [S^{tr}]$?

Eine Antwort auf die Frage “Existiert ein H-Comparabilitygraph G mit $G_{SIJ} \subseteq G \subsetneq [S^{tr}]$?” liefert eine Antwort für REDUCIBILITY = co-IRRED.

Ein NP-Test für REDUCIBILITY

Satz 4.2 liefert die Grundlage für einen nichtdeterministischen Test auf Reduzierbarkeit. Zur Erinnerung: Ein NP-Algorithmus liefert für jede zulässige Instanz auf mindestens einem Rechenweg eine positive Antwort, und lehnt jede unzulässige Instanz auf jedem Rechenweg ab (Kapitel 2).

Für das Problem REDUCIBILITY ist eine anzunehmende Instanz ein Plan S , der reduziert werden kann. Betrachten wir nun einen Algorithmus, der auf randomisierte Weise mit polynomiellen Aufwand irgendeinen Plan S' generiert, und anschließend mit Hilfe von Satz 4.2 prüft, ob S' den Plan S streng reduziert. Als Antwort gibt das Verfahren dann entweder “REDUZIERBAR” oder “NICHT REDUZIERBAR” zurück.

Wenn S reduzierbar ist, ist die Wahrscheinlichkeit, dass der erzeugte Plan S' ein reduzierender Plan von S ist, verschieden von Null. Genauer: Es gibt mindestens einen Rechenweg, der einen Plan S' mit $S' \prec S$ erzeugt. Gibt der Algorithmus als Antwort dennoch “NICHT REDUZIERBAR” zurück, so macht er einen Fehler. Für die Wahrscheinlichkeit dieses Fehlers gilt $\varepsilon < 1$.

Umgekehrt liefert ein solcher Algorithmus für jede abzulehnende Instanz auf jedem Rechenweg eine negative Antwort. Ist der Ausgangsplan S irreduzibel, dann kann kein Plan gefunden werden, der S streng reduziert. Die Antwort lautet stets “NICHT REDUZIERBAR”. Für abzulehnende Instanzen beträgt die Fehlerwahrscheinlichkeit somit Null.

Damit ist der beschriebene Algorithmus ein randomisiertes Verfahren mit einem einseitigen Fehler. Während die Antwort “REDUZIERBAR” immer korrekt ist, ist die Antwort “NICHT REDUZIERBAR” mit einer Wahrscheinlichkeit von $\varepsilon < 1$ falsch. Es handelt sich damit um einen nichtdeterministischen Algorithmus mit polynomieller maximaler Laufzeit für das Problem REDUCIBILITY.

Satz 4.3 (REDUCIBILITY) *Das Problem REDUCIBILITY liegt in NP.*

Folgerung 4.4 (IRRED) *Das Problem IRRED liegt in co-NP.*

Wir können also für einen reduzierbaren Plan S die Reduzierbarkeit nichtdeterministisch mit polynomiellen Aufwand beweisen. Da ein solcher Nachweis nur konstruktiv möglich ist, liefert ein solches Verfahren nicht nur eine Antwort für das Problem REDUCIBILITY, sondern auch für das Problem REDUCING.

IRRED \in NP ?

Können wir umgekehrt auch mit polynomiellen Aufwand nichtdeterministisch die Irreduzibilität eines Plans beweisen? D.h. gibt es für jeden irreduziblen Plan mindestens einen Rechenweg, auf dem mit polynomiellen Aufwand die Irreduzibilität nachgewiesen werden kann?

Die Konsequenzen der Antwort auf diese Fragen sind weitreichend. Könnten wir zeigen, dass IRRED nicht nur in co-NP, sondern auch in NP liegt, dann würde die Zugehörigkeit von IRRED zu $ZPP^* = NP \cap co-NP$ folgen. Damit würde IRRED entweder deterministisch polynomiell lösbar oder aber NP-unvollständig sein, sofern $P \neq NP$ gilt (Kapitel 2).

Der Beantwortung dieser Fragen dienen alle Untersuchungen im verbleibenden Teil dieser Arbeit.

4.1.4 Notwendige Bedingungen für Irreduzibilität

Es gibt eine Reihe von polynomiell überprüfbaren notwendigen Bedingungen für die Irreduzibilität eines Plans. Da viele davon bei Nichterfüllung konstruktiv einen reduzierenden Plan beschreiben, werden wir diese Bedingungen als hinreichende Bedingungen für die Reduzierbarkeit formulieren.

Eine besondere Bedeutung kommt dabei der Reduktion eines Plans durch das Umkehren von Implikationsklassen in der transitiven Hülle des Plangraphen zu.

Lemma 4.5 (H-Comparabilitygraph nicht prim) [66] *Sei $A \in \mathcal{S}_{SIJ}$ ein Plan, dessen H-Comparabilitygraph $[A^{tr}]$ nicht prim ist. Dann ist A entweder reduzierbar oder ähnlich zu einem irreduziblen Plan B mit $B \neq A$ und $B \neq A^{-1}$.*

Diese Reduktionsmethode markiert den Grundstein auf dem Weg zu einem Reduktionsalgorithmus. Alle weiteren Bedingungen beschreiben spezielle Teilstrukturen, die wir nur der Vollständigkeit halber nennen wollen.

Reduktion durch Umkehren von Implikationsklassen

In Abschnitt 3.3.5 haben wir gesehen, dass wir jeder transitiven Orientierung eines H-Comparabilitygraphen einen Plan zuordnen können. Hat der H-Comparabilitygraph G eine Planorientierung A^{tr} , dann ist $G = [A^{tr}]$ der H-Comparabilitygraph zu einem Plan $A \in \mathcal{S}_{SIJ}$. Ist durch B^{tr} eine transitive Orientierung auf $[A^{tr}]$ beschrieben, die keine Planorientierung ist, dann sind einige der Schrägkanten von $[A^{tr}]$ nicht transitiv in dieser Orientierung. Der H-Comparabilitygraph zu dem durch diese Orientierung beschriebenen Plan $B \in \mathcal{S}_{SIJ}$ ist dann ein echter Teilgraph von $[A^{tr}]$, $[B^{tr}] \subsetneq [A^{tr}]$. Nach Satz 4.2 reduziert damit der Plan B den Plan A streng, $B \prec A$.

Wir erinnern daran, dass jede transitive Orientierung $T \in \mathcal{T}_G$ auf einem Graphen G eine Kombination von Implikationsklassen $I_i \in \mathcal{I}_G = \{I_1, \dots, I_k, I_1^{-1}, \dots, I_k^{-1}\}$ ist (Abschnitt 1.2.7). Benutzen wir die Planorientierung $A^{tr} = I_1 + \dots + I_k$ als Bezugspunkt, dann ist jede transitive Orientierung $T \in \mathcal{T}_{[A^{tr}]}$ auf $[A^{tr}]$ eine Rekombination von transitiven Orientierungen auf den Farbklassen von $[A^{tr}]$, $T = J_1 + \dots + J_k$ mit $J_i \in \{I_i, I_i^{-1}\}$. Also lässt sich der Plan A durch das Umkehren von einzelnen Implikationsklassen in den Plan B transformieren.

Für den Plan B lässt sich nun erneut die Frage stellen: Existieren auf $[B^{tr}]$ neben den Planorientierungen B^{tr} und $(B^{tr})^{-1} = (B^{-1})^{tr}$ transitive Orientierungen, die den Plan B weiter reduzieren? Daran schließt sich die Frage an, wie man weitere transitive Orientierungen auf einem Comparabilitygraphen finden kann. Eine Möglichkeit besteht darin, eine oder mehrere Implikationsklassen einer gegebenen Orientierung einfach umzukehren, und anschließend zu überprüfen, ob die resultierende potentiell transitive Orientierung kreisfrei ist (Satz 1.19).

Umkehren von Implikationsklassen durch Entfernen von einzelnen Schrägkante

Eine besonders in unserem Zusammenhang wirksamere Methode zum Umkehren von Implikationsklassen besteht darin, Schrägkanten einzeln zu entfernen. Nach Satz 1.39 können wir eine Schrägkante, die in der transitiven Reduktion einer transitiven Orientierung liegt, bedenkenlos löschen. Eine Schrägkante liegt genau dann in der transitiven Reduktion, wenn sie nicht transitiv ist. Betrachten wir nun eine gegebene (ungerichtete) Schrägkante $\hat{e} \in E_{diag}(A)$ in $[A^{tr}]$. Wenn es auf $[A^{tr}]$ eine transitive Orientierung gibt, die weder e noch e^{-1} als transitive Kante enthält, dann ist nach Satz 1.39 der Graph $[A^{tr}] - \hat{e}$ ein Comparabilitygraph.

D.h. wir können umgekehrt durch Testen von $[A^{tr}] - \hat{e}$ auf Existenz einer transitiven Orientierung überprüfen, ob auf $[A^{tr}]$ eine transitive Orientierung existiert, die einen reduzierenden Plan B induziert, der die Schrägkante \hat{e} nicht enthält. Ist $[A^{tr}] - \hat{e}$ kein Comparabilitygraph, dann kann A nicht durch das Entfernen der Schrägkante \hat{e} reduziert werden. Ist dagegen $[A^{tr}] - \hat{e}$ ein Comparabilitygraph, und ist $T \in \mathcal{T}_{[A^{tr}] - \hat{e}}$ eine transitive Orientierung, so ist T entweder eine Rekombination der unveränderten Implikationsklassen von $[A^{tr}]$, oder T besteht aus einer Mischung aus unveränderten Implikationsklassen von $[A^{tr}]$ und $\hat{\Gamma}_{\hat{e}}$ -Komponenten von $[A^{tr}]$ (Satz 1.43, [3],[4]). In beiden Fällen induziert die transitive Orientierung einen Plan $B \in \mathcal{S}_{SIJ}$, der den Ausgangsplan A streng reduziert.

Da die Bestimmung einer transitiven Orientierung auf einem Graphen $G - \hat{e}$ mit polynomiellern Aufwand möglich ist (Abschnitt 2.1), und die Anzahl der Schrägkanten in einem H-Comparabilitygraphen mit n Aufträgen und m Maschinen durch $\mathcal{O}(n^2m^2)$ beschränkt ist, kann mit polynomiellern Aufwand getestet werden, ob ein gegebener Plan $A \in \mathcal{S}_{SIJ}$ durch Entfernen *irgendeiner* Schrägkante reduziert werden kann.

Satz 4.6 (Reduktion durch Entfernen von einzelnen Schrägkanten) [66] *Sei $A \in \mathcal{S}_{SIJ}$ ein Plan. Existiert in $[A^{tr}]$ eine Schrägkante $\hat{e} \in E_{diag}$, für die $[A^{tr}] - \hat{e}$ ein Comparabilitygraph ist, dann liefert jede transitive Orientierung von $[A^{tr}] - \hat{e}$ einen Plan, der A streng reduziert.*

Mit Hilfe dieser Überlegung sind wir nun in der Lage, einen gegebenen Plan $A \in \mathcal{S}_{SIJ}$ sukzessive mit polynomiellern Aufwand zumindest soweit zu reduzieren, bis keine der Schrägkanten mehr einzeln entfernt werden kann. Dabei werden in jedem Reduktionsschritt höchstens $|E_{diag}(A)| = \mathcal{O}(n^2m^2)$ viele Schrägkanten betrachtet. Wird in einem Schritt ein Plan $B \in \mathcal{S}_{SIJ}$ gefunden, der den Ausgangsplan A für diesen Schritt reduziert, dann dient der Plan B als neuer Ausgangsplan für den nächsten Schritt—und nicht etwa der Graph $[A^{tr}] - \hat{e}$. Dabei entfallen neben \hat{e} u.U. noch weitere Schrägkanten. Da in jedem Reduktionsschritt jedoch mindesten eine Schrägkante entfernt werden kann, ist die Anzahl der Reduktionsschritte ebenfalls durch die Anzahl der Schrägkanten des Ausgangsgraphen begrenzt.

Am Ende dieser sukzessiven Reduktionen steht ein Plan $B \in \mathcal{S}_{SIJ}$, für den gilt, dass $[B^{tr}] - \hat{e}$ für keine Schrägkante $e \in E_{diag}(B)$ ein Comparabilitygraph ist. Nach Satz 1.39 sind dann auf $[B^{tr}]$ alle transitiven Orientierungen Planorientierungen.

Bemerkung 4.7 (Reduktion durch Umkehren von Implikationsklassen) *Ein Plan $A \in \mathcal{S}_{SIJ}$ kann mit polynomiellern Aufwand auf einen Plan $B \in \mathcal{S}_{SIJ}$ reduziert werden, der nicht weiter durch Umkehren von Implikationsklassen streng reduziert werden kann. Der H-Comparabilitygraph $[B^{tr}]$ ist dann entweder prim, oder zu B existieren ähnliche Pläne, die verschieden von B^{-1} sind.*

Das bedeutet natürlich nicht, dass mit polynomiellern Aufwand *alle* reduzierenden Pläne $B \in \mathcal{S}_{SIJ}$ erzeugt werden können, die nicht weiter durch Umkehren von Implikationsklassen streng reduziert werden können. Die Anzahl solcher Pläne ist typischerweise sehr groß und im Allgemeinen nur exponentiell durch die Anzahl der potentiellen transitiven Orientierungen von $[A^{tr}]$ begrenzt, d.h. durch 2^k , wenn $[A^{tr}]$ k verschiedene Implikationsklassen besitzt. Die Bemerkung sagt lediglich, dass wir *irgendeinen* dieser Pläne B finden können. Es ist jedoch so, dass es für *jeden* dieser reduzierenden Pläne B einen zulässigen Reduktionsweg gibt, der zu diesem Plan führt.

Veranschaulichen wir uns die Reduktion durch das Umkehren von Implikationsklassen an einem Beispiel.

Beispiel 4.8 (Reduktion durch Umkehren von Implikationsklassen) *Betrachten wir den Plan $S \in \mathcal{S}_{SIJ}$ mit*

$$A = \begin{bmatrix} 1 & 2 & & & 6 \\ & 3 & 4 & 1 & 5 \\ 2 & & 3 & & \end{bmatrix}.$$

In Abbildung 4.1 ist der Plangraph $G(A)$ von A abgebildet. Der Übersicht halber werden wir Pläne in Zukunft häufig nur durch die Menge der Schrägkanten des zugehörigen H -Comparabilitygraphen abbilden. Der H -Comparabilitygraph $[A^{tr}]$ zum Plan A besteht aus allen horizontalen und vertikalen Kanten zwischen den vorhandenen neun Operationen und den in Abbildung 4.1 abgebildeten 14 Schrägkanten. Einen (streng) reduzierenden Plan von S erkennt man dann daran, dass die Menge der Schrägkanten dieses Plans jeweils eine (echte) Teilmenge der Schrägkanten von $[A^{tr}]$ ist. Ebenfalls aus Gründen der Übersicht werden wir bei Beispielen dieser Art zukünftig auf die Bezeichnung der Operationen O_{ij} durch den Zeilen- und Spaltenindex verzichten. Wir werden stattdessen die Knoten des H -Comparabilitygraphen einfach durchnummerieren. Das erleichtert die Beschreibung von Schrägkantenmengen. Der H -Comparabilitygraph $[A^{tr}]$ besteht aus acht Implikationsklassen I_1, \dots, I_8

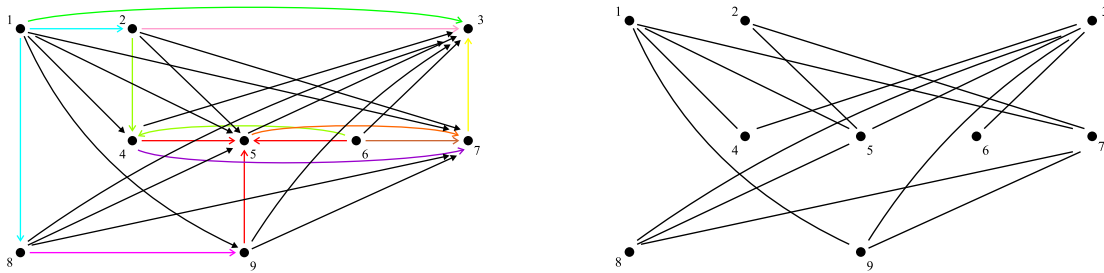


Abbildung 4.1: Der Plangraph $G(A)$ und die Menge der Schrägkanten von $[A^{tr}]$.

und ihren Umkehrungen $I_1^{-1}, \dots, I_8^{-1}$ (Abbildung 4.2). Es gilt $A = I_1 + \dots + I_8$. Nach dem Struktursatz von Golumbic (Satz 1.21 auf Seite 18) sind von den $8^2 = 256$ potentiellen transitiven Orientierungen genau $6 \cdot 2^5 = 192$ kreisfrei. Jede dieser 192 transitiven Orientierungen liefert einen Plan, der S reduziert. Dabei können jedoch jeweils mehrere Orientierungen denselben Plan induzieren. In diesem Fall

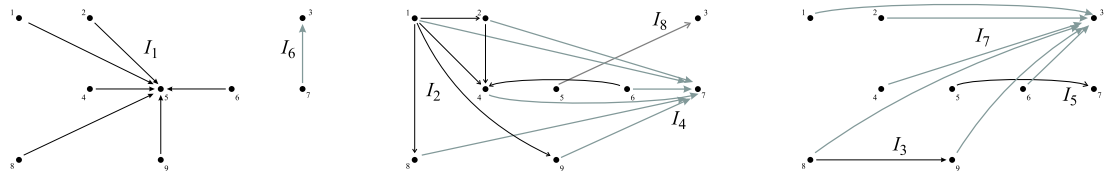


Abbildung 4.2: Die Zerlegung der Kantenmenge des Plangraphen in Implikationsklassen $A = I_1 + \dots + I_8$.

existieren 72 reduzierende Pläne, die durch Umkehren von Implikationsklassen aus A erzeugt werden können. Unter diesen 72 Plänen sind A und A^{-1} die einzigen Planorientierungen von $[A^{tr}]$. D.h. durch die Rekombination von Implikationsklassen können 35 verschiedene streng reduzierende Pläne (und ihre Umkehrungen) erreicht werden. Unter diesen insgesamt 36 verschiedenen reduzierenden Plänen befinden sich auch 5 irreduzible Pläne (Abbildung 4.3). Betrachten wir einen Reduktionspfad genauer: Kehren wir in $G(A)$ die Implikationsklassen I_1 und I_4 um, gelangen wir zu einer kreisfreien potentiellen transitiven Orientierung, bei der lediglich die Schrägkanten $1.4 \in E_{diag}(A)$, $1.9 \in E_{diag}(A)$ und $5.3 \in E_{diag}(A)$ transitiv sind. Alle übrigen Schrägkanten liegen somit in der transitiven Reduktion dieser Orientierung und können gelöscht werden. Die verbleibende Planorientierung auf einem Teilgraphen von $[A^{tr}]$ enthält also nur noch 3 der ursprünglich 14 Schrägkanten. Der zugehörige Plan ist der erste der irreduziblen Pläne in Abbildung 4.3.

Da wir normalerweise nicht wissen können, welche Menge der Implikationsklassen umgekehrt werden muss, um in einem Schritt zu einem irreduziblen Plan zu gelangen, betrachten wir nun die sukzessive Entfernung von einzelnen Schrägkanten. Nehmen wir beispielsweise an, wir entfernen aus $[A^{tr}]$ die Schrägkante $\widehat{1.5} \in E_{diag}([A^{tr}])$. Der Graph $[A^{tr}] - \widehat{1.5}$ besitzt 6 verschiedene Implikationsklassen. Von den $2^6 = 64$ potentiellen transitiven Orientierungen sind 36 kreisfrei. Eine dieser Orientierungen ist $S = I_1 + I_2^{-1} + I_3^{-1} + I_4 + I_5 + I_6 + I_7 + I_8$. Die Schrägkantenmenge des H -Comparabilitygraphen von S ist in Abbildung 4.4 (links) dargestellt. Während $[A^{tr}] - \widehat{1.5}$ noch 13 Schrägkanten und 6 verschiedene

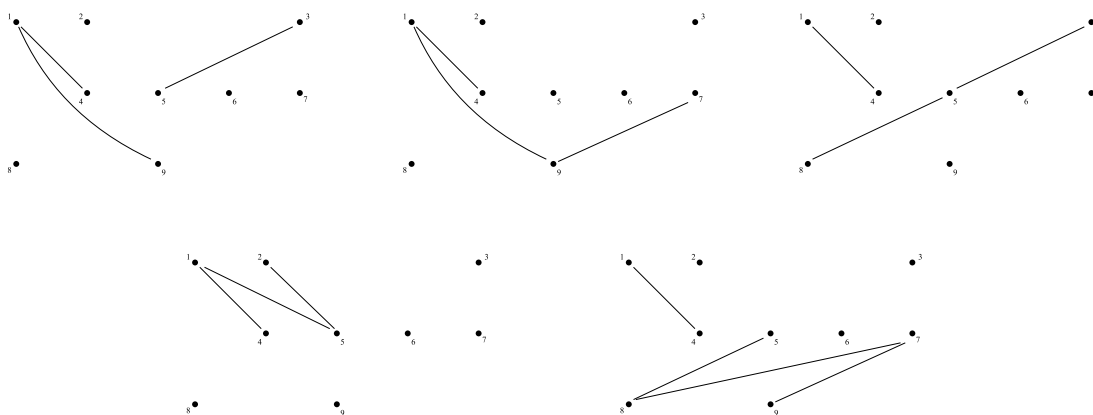


Abbildung 4.3: Die Schrägkantenmengen der 5 irreduziblen Pläne, die durch Umkehren der Implikationsklassen von A erzeugt werden können.

Implikationsklassen besitzt, hat $[S^{tr}]$ nur noch 8 Schrägkanten und 3 verschiedene Implikationsklassen. Entfernen wir nun aus $[S]$ beispielsweise die Schrägkante $\widehat{4.3}$, so gelangen wir zu einem Graphen

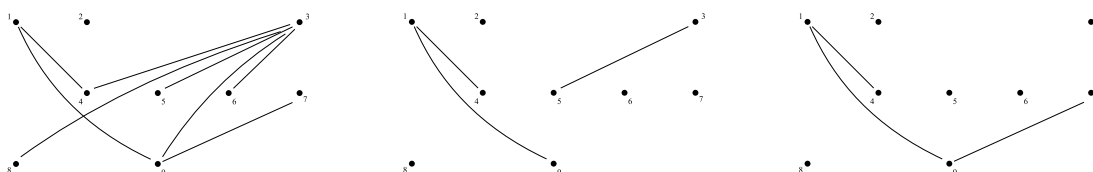


Abbildung 4.4: Die Schrägkantenmenge eines Plans zu einer transitiven Orientierung von $[A^{tr}] - \widehat{1.5}$, und die der daraus durch Umkehren von Implikationsklassen erreichbaren irreduziblen Pläne.

mit nur noch 7 Schrägkanten und genau 2 verschiedenen Implikationsklassen. Jede der beiden transitiven Orientierungen von $[S] - \widehat{4.3}$ führt zu einem der beiden irreduziblen Pläne in Abbildung 4.4 (mitte und rechts). Durch diesen Reduktionspfad sind wir schließlich auch bei dem irreduziblen Plan $S^* = I_1^{-1} + I_2 + I_3 + I_4^{-1} + I_5 + I_6 + I_7 + I_8$ (mitte) angelangt. Die beiden irreduziblen Pläne in Abbildung 4.4 unterscheiden sich übrigens nur durch die Orientierung von $\widehat{I_5}$.

In dem eben beschriebenen Reduktionspfad haben wir genau zwei Reduktionsschritte durchgeführt. Wir haben zunächst aus $[A^{tr}]$ die Schrägkante $\widehat{1.5}$ entfernt. Anschließend haben wir aus der Planorientierung $[S]$ zu einer transitiven Orientierung von $[A^{tr}] - \widehat{1.5}$ die Schrägkante $\widehat{4.3}$ entfernt, und sind nach der Auswahl einer transitiven Orientierung auf $[S] - \widehat{4.3}$ zu einem irreduziblen Plan gelangt. ■

Reduzierbare Teilstrukturen in Plänen

Einige weitere notwendige Bedingungen für die Irreduzibilität eines Plans ergeben sich aus den folgenden Sätzen (BRÄSEL, HARBORTH, TAUTENHAHN UND WILLENIUS (1999) [13]).

Satz 4.9 (Rang $nm - 2$) [13] Sei $A \in \mathcal{S}_{SIJ}$ ein Plan über der vollständigen Operationenmenge mit n Aufträgen und m Maschinen. Hat eine Operation in A den Rang $nm - 2$, dann ist A streng reduzierbar auf einen Plan $B \in \mathcal{S}_{SIJ}$.

Satz 4.10 (Umkehren spezifischer Kante) [13] Sei $A \in \mathcal{S}_{SIJ}$ ein Plan mit der folgenden Eigenschaft: Die Operation O_{ij} ist die letzte Operation in Zeile i , O_{kj} ist die erste Operation von Zeile k , und O_{kl} ist der direkte Nachfolger von O_{ij} in Spalte j . Dann wird A durch den Plan B , der durch Umkehren der Kante $O_{ij}O_{kj}$ entsteht, reduziert, $B \preceq A$. Existiert eine Spalte $l \neq j$ mit $a_{kl} \leq a_{ij} + 3$ oder $a_{kj} \leq a_{il} + 3$, dann reduziert B den Plan A streng, $B \prec A$.

Satz 4.11 (Nachfolger in Zeile oder Spalte) [13] *Sei $A \in \mathcal{S}_{SIJ}$ ein Plan und sei O_{ij} eine Operation in S mit mindestens einem Nachfolger, für die gilt, dass keiner der Nachfolger in Zeile i oder Spalte j einen direkten Vorgänger außerhalb von Zeile i bzw. Spalte j hat. Dann ist A streng reduzierbar auf einen Plan $B \in \mathcal{S}_{SIJ}$.*

Satz 4.12 (gleiche Startmaschine) [13] *Sei $A \in \mathcal{S}_{SIJ}$ ein Plan, bei dem die jeweils erste Operation jeder Zeile $i = 1, \dots, n$ in der gleichen Spalte j ($j \in \{1, \dots, m\}$) liegt. Dann ist A streng reduzierbar auf einen Plan $B \in \mathcal{S}_{SIJ}$.*

Satz 4.13 (Entfernen von Operationen) [13] *Sei $A \in \mathcal{S}_{SIJ}$ ein Plan über einer Operationenmenge mit n Aufträgen und m Maschinen. Es ist in $\mathcal{O}(n^2m^2)$ entscheidbar, ob A durch Entfernen einer Operation und Neueinfügen als Quelle oder Senke streng reduziert werden kann.*

Reduktion über eine einzelne Kante

Bei KLEINAU (1992) [41] ist eine Bedingung beschrieben, die die Reduktion eines Plans $A \in \mathcal{S}_{SIJ}$ durch Umkehren einer einzelnen Kante auf einem maximalen Weg in A beschreibt.

Satz 4.14 (Reduktion über eine Kante) [41] *Sei $A \in \mathcal{S}_{SIJ}$ ein Plan und sei $e = ab$ eine (o.B. d.A.) horizontale Kante eines maximalen Weges in A . Existieren in A Wege von dem direkten Vorgänger von b in seiner Spalte zu a , und von b zu dem direkten Nachfolger von a in seiner Spalte, dann kann A durch Umkehren von e streng reduziert werden.*

Die Formulierung dieser Bedingung für eine vertikale Kante $e = ab$ ist völlig analog; es wird lediglich die Rolle von Zeilen und Spalten vertauscht.

Als eine direkte Folgerung dieser Bedingung ergibt sich, dass das Problem IRRED für ein Job-Shop Scheduling Problem polynomiell lösbar ist. Bei einem Job-Shop Problem sind die technologischen Reihenfolgen vorgegeben. D.h. die horizontalen Kanten des Plangraphen sind fixiert. Lediglich die vertikalen Kanten können umorientiert werden. KLEINAU [41] konnte zeigen, dass jede Reduktion eines Job-Shop Plans eine solche Reduktion über eine Kante ist.

Satz 4.15 (Reduktion für Job-Shop Pläne) [41] *Ein Plan $A \in \mathcal{S}_{SIJ}$ ist unter Beibehaltung seiner Technologie genau dann reduzierbar, wenn er sich über eine Kante reduzieren lässt.*

Folgerung 4.16 (IRRED für Job-Shop) [41] *Das Problem IRRED ist für Job-Shop Probleme polynomiell lösbar.*

Zyklische Pläne und lateinische Quadrate

Abschließend wollen wir hier auch noch eine hinreichende Bedingung für die Irreduzibilität eines Plans anführen. KLEINAU [41] hat gezeigt, dass sogenannte zyklische Pläne irreduzibel sind. Dabei heißt ein Plan $A \in \mathcal{S}_{SIJ}$ über einer Operationenmenge mit n Aufträgen und m Maschinen *zyklisch*, wenn jede Zeile von A eine zyklische Permutation von $\{1, \dots, m\}$, oder jede Spalte eine zyklische Permutation von $\{1, \dots, n\}$ ist.

Satz 4.17 (zyklische Pläne) [41] *Jeder zyklische Plan $A \in \mathcal{S}_{SIJ}$ ist irreduzibel.*

Neben den zyklischen Plänen liefert auch jedes lateinische Rechteck $LR(n, m, r)$ mit $n = m = r$ einen irreduziblen Plan (WILLENUS [66]).

Satz 4.18 (lateinische Quadrate) [66] *Jeder rangminimale Plan $A \in \mathcal{S}_{SIJ}$ über einer vollständigen Operationenmenge mit $m = n$ ist irreduzibel.*

4.1.5 Verallgemeinerungen der Reduzierbarkeit

In diesem Abschnitt wollen wir kurz auf zwei mögliche Verallgemeinerungen des Reduzierbarkeitsbegriffes eingehen. Zum Einen lässt sich die Reduzierbarkeit auch für andere Zielfunktionen als das Makespan-Kriterium formulieren. Und zum Anderen kann der Begriff der Reduzierbarkeit eines Plans durch einen einzelnen anderen Plan auf einen Dominanzbegriff erweitert werden, bei dem ein Plan durch eine Menge anderer Pläne reduziert werden kann.

Erweiterung auf andere Zielfunktionen

Alle in Kapitel 3 erwähnten Zielfunktionen $f(C_1, \dots, C_n)$ sind reguläre Zielfunktionen (Abschnitt 3.1.2). Sie sind monoton wachsend in den Fertigstellungszeiten C_i der Aufträge A_i , $i = 1, \dots, n$. Gilt nun für zwei Pläne $A, B \in \mathcal{S}_{SIJ}$ und für jede Wahl von Bearbeitungszeiten für jeden Auftrag $C_i(A) \leq C_i(B)$, dann folgt $f(A) \leq f(B)$ für jede reguläre Zielfunktion $f(C_1, \dots, C_n)$. Wir sagen dann, dass der Plan A den Plan B *allgemein reduziert*. Die allgemeine Reduzierbarkeit ist, bei Erweiterung des Modells, genau wie die Reduzierbarkeit mit Hilfe von Comparabilitygraphen beschreibbar. Sei $V_i(A)$ die Menge der Operationen, die in einem Plan $A \in \mathcal{S}_{SIJ}$ Vorgänger einer Operation von Auftrag i sind. Dann kann die allgemeine Reduzierbarkeit folgendermaßen charakterisiert werden (WILLENUS (2000) [66]).

Satz 4.19 (allgemeine Reduzierbarkeit) [66] *Ein Plan $A \in \mathcal{S}_{SIJ}$ reduziert einen Plan $B \in \mathcal{S}_{SIJ}$ genau dann allgemein, wenn $[A^{tr}] \subseteq [B^{tr}]$ und $V_i(A) \subseteq V_i(B)$ für jeden Auftrag $i \in \{1, \dots, n\}$ gilt.*

Neben den Gemeinsamkeiten im Beschreibungsmodell gibt es eine weitere Verbindung zwischen der Reduzierbarkeit und der allgemeinen Reduzierbarkeit.

Satz 4.20 (allgemein-irreduzible Pläne) [66] *Sei $A \in \mathcal{S}_{SIJ}$ ein irreduzibler Plan. Dann existiert ein allgemein-irreduzibler Plan, der ähnlich zu A ist.*

Ausgehend von Satz 4.19 beschreibt WILLENUS [66] auch ein Graphenmodell, mit dem die allgemeine Reduzierbarkeit genau wie die Reduzierbarkeit allein durch eine Teilgraphenrelation beschrieben werden kann. Auf die Darstellung dieses Modells wollen wir hier jedoch verzichten.

Die Dominanz-Relation

Eine zweite Verallgemeinerung der Reduzierbarkeit ist die sogenannte Dominanz eines Plans durch eine Menge von Plänen. Wir haben bereits festgestellt, dass die Menge der nichtähnlichen irreduziblen Pläne eine universell-optimale Menge bilden. Für jede Wahl der Bearbeitungszeiten enthält sie eine optimale Lösung für das Schedulingproblem. TAUTENHAHN (1996) [63] konnte zeigen, dass es Operationenmengen gibt, für die die Menge der nichtähnlichen irreduziblen Pläne keine minimale universell-optimale Menge ist.

$$B_{11} = \begin{bmatrix} 2 & 1 & 3 & 4 \\ 4 & 3 & 1 & 2 \end{bmatrix} \quad B_{12} = \begin{bmatrix} 1 & 2 & 4 & 3 \\ 3 & 4 & 2 & 1 \end{bmatrix} \quad B_{13} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 3 & 4 & 1 & 2 \end{bmatrix}$$

Betrachten wir beispielsweise die Pläne B_{11} , B_{12} und B_{13} . Jeder dieser drei Pläne ist irreduzibel und die H-Comparabilitygraphen zu diesen Plänen besitzen jeweils genau zwei Schrägkanten. Der Graph $[B_{13}^{tr}]$ hat dabei jeweils eine seiner beiden Schrägkanten mit den Graphen $[B_{11}^{tr}]$ bzw. $[B_{12}^{tr}]$ gemeinsam. Für diese Pläne lässt sich nun leicht folgendes zeigen (WILLENUS [66]): Für jede Wahl von Bearbeitungszeiten gilt $C_{\max}(B_{11}) \leq C_{\max}(B_{13})$ oder $C_{\max}(B_{12}) \leq C_{\max}(B_{13})$. D.h. es gibt keine mögliche Wahl der Bearbeitungszeiten, für die der irreduzible Plan B_{13} einen besseren Zielfunktionswert liefert als irgendein anderer Plan. Der Plan B_{13} wird durch die beiden Pläne B_{11} und B_{12} *dominiert*. Bei der Suche nach einer minimalen universell-optimale Menge für die vollständige Operationenmenge mit 2 Aufträgen und 4 Maschinen brauchen wir damit den Plan B_{13} nicht berücksichtigen.

Bei WILLENUS [66] findet sich ein Algorithmus, mit dessen Hilfe sich polynomiell entscheiden lässt, ob ein gegebener Plan $S \in \mathcal{S}_{SIJ}$ durch zwei gegebene Pläne $A, B \in \mathcal{S}_{SIJ}$ dominiert wird.

4.2 Planimplikationsklassen und der Faktorgraph

In diesem Abschnitt werden wir eine Teilstruktur der Implikationsklassen von H-Comparabilitygraphen kennen lernen, die von fundamentaler Bedeutung für die Theorie der Reduzierbarkeit von Open-Shop Plänen ist. Diese sogenannten *Planimplikationsklassen* können in demselben Sinn als Bausteine von reduzierenden Plänen betrachtet werden, wie Implikationsklassen als Bausteine von transitiven Orientierungen aufgefasst werden können (Abschnitt 1.2.7).

Im vorangegangenen Abschnitt haben wir die Reduktion durch das Umkehren von Implikationsklassen betrachtet. Dass man auf diesem Weg nicht unbedingt bis hinunter zu einem irreduziblen Plan

gelangt, zeigt das folgende kleine Beispiel. Wir haben im Zusammenhang mit ähnlichen Plänen auf Seite 59 den Plan B_5 kennen gelernt.

$$B_5 = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 3 & \\ 2 & & \end{bmatrix} \quad B_{14} = \begin{bmatrix} 1 & 4 & 2 \\ 2 & 3 & \\ 3 & & \end{bmatrix} \quad B_{15} = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & \\ 3 & & \end{bmatrix}$$

Der H-Comparabilitygraph zu diesem Plan enthält genau zwei Schrägkanten, und besteht aus genau zwei verschiedenen Implikationsklassen. Kehrt man eine davon um (Plan B_{14}), gelangt man erneut zu einer Planorientierung desselben H-Comparabilitygraphen. Jede transitive Orientierung von $[B_5^{tr}]$ liefert einen zu B_5 ähnlichen Plan. Man kann B_5 also nicht durch Umkehren einer Implikationsklasse reduzieren. Das bedeutet jedoch nicht, dass B_5 irreduzibel ist. Der H-Comparabilitygraph zu dem Plan B_{15} enthält nur eine der beiden Schrägkanten von $[B_5^{tr}]$. B_{15} reduziert also B_5 streng.

Man kann von B_5 zu B_{15} gelangen, indem man eine der beiden Implikationsklassen in zwei Komponenten zerlegt, und nur eine dieser beiden Komponenten umkehrt, während die andere unverändert bleibt. Die Beschreibung solcher Teilstrukturen von Implikationsklassen und ihrer Eigenschaften ist daher ein wichtiger Schritt auf dem Weg zum Verständnis der Reduzierbarkeitstheorie.

4.2.1 Der Begriff der Planimplikationsklasse

Der Begriff der Planimplikationsklasse (engl. *sequence implication class*) wurde von WILLENUS (2000) [66] geprägt. Bereits bei KLEINAU (1992) [41] findet sich jedoch ein erster Hinweis auf diese Struktur und ihre Bedeutung. Es handelt sich dabei jeweils um Teilmengen der regulären Kanten eines Plans $S \in \mathcal{S}_{SIJ}$. Sie können daher sowohl als Objekte im Graphen G_{SIJ} bzw. im Plangraphen $G(S)$, als auch als Objekte im H-Comparabilitygraphen $[S^{tr}]$ betrachtet werden. Abhängig von dem Graphen, als dessen Teilstruktur man eine Planimplikationsklasse betrachtet, gibt es verschiedenen Möglichkeiten der Definition.

Definition als Äquivalenzklasse auf dem Plangraphen

Betrachten wir zunächst eine Definition als Zerlegung der Kantenmenge des Plangraphen $G(S)$ (nach WILLENUS [66]). Es sei daran erinnert, dass der Plangraph $G(S)$ eine kreisfreie Orientierung des H-Graphen $G_{SIJ} = (SIJ, E_{reg})$ ist. Er erhält im Gegensatz zum H-Comparabilitygraphen $[S^{tr}]$ keine Schrägkanten, sondern nur reguläre Kanten, $S \subseteq E_{reg}$.

Definition 4.21 (Γ_S -Relation) Sei $G(S) = (SIJ, S)$ der Plangraph und $[S^{tr}]$ der H-Comparabilitygraph zu einem Plan $S \in \mathcal{S}_{SIJ}$. Für zwei Kanten ab und cd in $G(S)$ wird die Γ_S -Relation folgendermaßen definiert:

$$ab\Gamma_S cd \Leftrightarrow ab\Gamma cd \text{ in } [S^{tr}].$$

Zwei reguläre Kanten liegen damit genau dann in Γ_S -Relation zueinander, wenn sie im H-Comparabilitygraphen in Γ -Relation zueinander liegen. Natürlich können wir sämtliche Begriffe aus dem Umfeld der Γ -Relation in analoger Weise auch auf diese neue Relation übertragen. Wir nennen zwei Kanten daher auch Γ_S -verbunden, wenn sie durch eine Γ_S -Kette in Verbindung stehen, oder sprechen von der Γ_S -Nachbarschaft einer Kante. Ebenso wie bei der Γ -Relation oder der $\dot{\Gamma}$ -Relation sind wir nicht allein an der Relation interessiert, sondern vor allem auch an dem transitiven Abschluss dieser Relation. Entsprechend beschreiben wir durch die Γ_S^* -Relation die Existenz einer Γ_S -Kette zwischen zwei Kanten. Diese Relation ist nun genau wie in den anderen Fällen eine Äquivalenzrelation, an deren Äquivalenzklassen wir interessiert sind.

Definition 4.22 (Planimplikationsklasse) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan, und sei durch den Plangraphen $G(S)$ eine kreisfreie Orientierung auf $G_{SIJ} = (SIJ, E_{reg}(S))$ gegeben. Eine Menge $P = P(e') = \{e'' \in E_{reg}(S) : e'\Gamma_S^* e''\}$ zu einer Kante $e' \in E_{reg}(S)$ nennen wir eine Planimplikationsklasse von S .

Eine Planimplikationsklasse bezüglich eines Plans $S \in \mathcal{S}_{SIJ}$ ist demnach eine Menge von regulären Kanten, die im H-Comparabilitygraphen $[S^{tr}]$ untereinander Γ -verbunden sind. Da Planimplikationsklassen in Abhängigkeit von einem gegebenem Plan $S \in \mathcal{S}_{SIJ}$ definiert sind, und G_{SIJ} ein Teilgraph von $[S^{tr}]$ ist, ist es sinnvoll, sie als Teilmenge der Kantenmenge des H-Comparabilitygraphen $[S^{tr}]$ zu betrachten. Aus der Definition der Γ_S -Relation folgt sofort die folgende Eigenschaft.

Lemma 4.23 (mehrere Kanten) [66] *Besitzt eine Planimplikationsklasse eines Plan $S \in \mathcal{S}_{SIJ}$ mindestens zwei Kanten, dann besitzt sie mindestens eine horizontale und mindestens eine vertikale Kante.*

Wir bezeichnen die Menge aller Planimplikationsklassen auf einem H-Comparabilitygraphen $[S^{tr}]$ mit $\mathcal{P}_{[S^{tr}]} = \{P_1, \dots, P_l\}$. Sie liefern eine Zerlegung der Menge der regulären Kanten, $P_1 + \dots + P_l = S + S^{-1} = [S] = E_{reg}(S)$. Da es sich bei Planimplikationsklassen genau wie bei Implikationsklassen um orientierte Kantenmengen handelt, die bezüglich einer transitiven Orientierung auf $[S^{tr}]$ definiert sind, gilt analog $\mathcal{P}_{[S^{tr}]} = \{P_1, \dots, P_k, P_1^{-1}, \dots, P_l^{-1}\}$, mit $P_1 + \dots + P_k = S$ und $P_1^{-1} + \dots + P_k^{-1} = S^{-1}$. Wir nennen $\mathcal{P}_S = \{P_1, \dots, P_k\}$ die Menge der Planimplikationsklassen von S .

Planimplikationsklassen und Implikationsklassen

In $[S^{tr}]$ kann die Γ_S -Relation dann als Teilrelation der Γ -Relation betrachtet werden. Zwei Kanten, die Γ_S -verbunden sind, sind auch Γ -verbunden.

Lemma 4.24 (Planimplikationsklassen und Implikationsklassen) [66] *Sei $S \in \mathcal{S}_{SIJ}$ ein Plan, und sei $P \subseteq E_{reg}(S)$ eine Planimplikationsklasse in $[S^{tr}] = (SIJ, E_{reg}(S) + E_{diag}(S))$. Dann existiert eine Implikationsklasse $I \in \mathcal{I}_{[S^{tr}]}$ mit $P \subseteq I$.*

Die Umkehrung dieser Aussage gilt nicht. In dem Beispiel zu Beginn dieses Abschnitts besteht eine transitive Orientierung des H-Comparabilitygraphen $[B_5^{tr}]$ aus zwei Implikationsklassen. Eine dieser beiden Klassen enthält alle mit O_{11} inzidierenden Kanten: vier reguläre Kanten und eine Schrägkante. Die vier regulären Kanten bilden eine Planimplikationsklasse. Die andere Implikationsklasse enthält alle übrigen Kanten: ebenfalls vier reguläre Kanten und die zweite Schrägkante. Hier sind jedoch nicht alle regulären Kanten paarweise durch eine Γ -Kette von regulären Kanten verbunden. Es gibt beispielsweise keine Γ_S -Kette zwischen den Kanten $O_{12}O_{22}$ und $O_{22}O_{21}$. Diese Implikationsklasse zerfällt in zwei Planimplikationsklassen und eine Schrägkante.

$$B_5 = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 3 & \\ 2 & & \end{bmatrix}$$

Definition als $\dot{\Gamma}$ -Komponente auf dem H-Comparabilitygraphen

Alternativ können wir Planimplikationsklassen auch als $\dot{\Gamma}$ -Komponenten bezüglich einer Kantenmenge $F = E_{diag}(S) \subseteq E(S)$ im H-Comparabilitygraphen interpretieren (nach ANDRESEN [3]). In Abschnitt 1.4.1 haben wir bereits den Spezialfall dieser Relation für den Fall, dass die Menge F nur aus einer Kante $\hat{e} \in E$ besteht, betrachtet.

Definition 4.25 ($\dot{\Gamma}_F$ -Relation) *Sei $G = (V, E)$ ein Graph, und sei $F \subseteq E$ eine Teilmenge der Kanten mit $F = \hat{F}$. Für zwei Kanten $ab, cd \in E$ ist die $\dot{\Gamma}$ -Relation bezüglich F definiert durch*

$$ab\dot{\Gamma}_F cd \Leftrightarrow ab\Gamma cd \text{ und } \{ab, cd\} \cap F = \emptyset.$$

Entsprechend nennen wir zwei Kanten auch in diesem Fall $\dot{\Gamma}_F$ -verbunden, wenn sie durch eine $\dot{\Gamma}_F$ -Kette verbunden sind. Auch den Begriff der $\dot{\Gamma}$ -Komponente als Bezeichnung für die Äquivalenzklassen der transitiven Hülle $\dot{\Gamma}_F^*$ der $\dot{\Gamma}_F$ -Relation haben wir bereits in Abschnitt 1.4.1 kennen gelernt.

Definition 4.26 ($\dot{\Gamma}_F$ -Komponente) *Sei $G = (V, E)$ ein Graph, und sei $F \subseteq E$ eine Teilmenge der Kanten. Eine Menge $\dot{P} = \dot{P}(e') = \{e'' \in E: e'\dot{\Gamma}_F^* e''\}$ zu einer Kante $e' \in E$ nennen wir eine $\dot{\Gamma}$ -Komponente von G bezüglich F .*

Der Bezug zu Planimplikationsklassen auf H-Comparabilitygraphen wird durch die folgende Überlegung unmittelbar deutlich.

Lemma 4.27 ($\dot{\Gamma}_F$ und Γ_S) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan, und sei $[S^{tr}] = (SIJ, E_{reg}(S) + E_{diag}(S))$ sein H -Comparabilitygraph. Dann gilt

$$e' \Gamma_S e'' \Leftrightarrow e' \dot{\Gamma}_{E_{diag}(S)} e''.$$

Beweis. Zwei Kanten e' und e'' liegen genau dann in direkter $\dot{\Gamma}_F$ -Relation zueinander, wenn sie in Γ -Relation zueinander liegen, und keine der beiden Kanten zu $F \subseteq E$ gehört. Für $F = E_{diag}(S)$ ist die $\dot{\Gamma}_F$ damit identisch zur Γ_S -Relation. Es gilt $e' \dot{\Gamma}_{E_{diag}(S)} e''$ genau dann, wenn e' und e'' reguläre Kanten mit $e' \Gamma e''$ sind, d.h. wenn auch $e' \Gamma_S e''$ gilt. ■

Damit ergibt sich eine alternative Möglichkeit zur Definition von Planimplikationsklassen.

Bemerkung 4.28 (Planimplikationsklasse, alternative Definition) Sei $[S^{tr}] = (SIJ, E_{reg}(S) + E_{diag}(S))$ der H -Comparabilitygraph zu einem Plan $S \in \mathcal{S}_{SIJ}$. Eine Planimplikationsklasse $P \in \mathcal{P}_{[S^{tr}]}$ von $[S^{tr}]$ ist eine $\dot{\Gamma}_F$ -Komponente bezüglich $F = E_{diag}(S)$.

4.2.2 Planimplikationsklassen und Reduzierung

In Abschnitt 1.2.4 sind wir bei der Beschreibung der Γ -Relation darauf eingegangen, dass die Γ -Relation zwischen zwei Kanten durch die Nichtexistenz einer Kante induziert wird. Wir haben in diesem Zusammenhang auch von der Existenz einer Nichtkante gesprochen, die Γ -Relationen induzieren kann, aber nicht muss.

Planimplikationsklassen als Atome

Werden nun aus einem Graphen Kanten entfernt, dann hat das keinen Einfluss auf bestehende Γ -Relationen. Bereits bestehende Γ -Relationen, und damit insbesondere auch bereits bestehende Γ_S -Relationen können durch das Entfernen von Schrägkanten nicht zerstört werden. Das bedeutet, zwei Kanten, die in einem H -Graphen G mit $G_{SIJ} \subseteq G \subseteq [S^{tr}]$ in einer gemeinsamen Planimplikationsklasse liegen, liegen auch in jedem Teilgraphen G^* mit $G_{SIJ} \subseteq G^* \subseteq G$ in einer gemeinsamen Planimplikationsklasse. Planimplikationsklassen können durch das Hinzufügen von Nichtkanten zu $[S^{tr}]$ nicht zerstört werden. In diesem Sinne können sie gewissermaßen als atomare Bausteine von Plänen interpretiert werden.

Lemma 4.29 (Planimplikationsklassen und Reduzierung) [66] Seien $S, S^* \in \mathcal{S}_{SIJ}$ zwei Pläne mit $S^* \preceq S$. Ist P eine Planimplikationsklasse von S . Dann existiert zu S^* eine Planimplikationsklasse P^* mit $P \subseteq P^*$ oder $P^{-1} \subseteq P^*$.

Auf der anderen Seite können durch das Hinzufügen von Nichtkanten jedoch neue Γ -Relationen oder sogar neue Γ_S -Relationen entstehen. Das bedeutet, zwei Planimplikationsklassen P_1 und P_2 von $S \in \mathcal{S}_{SIJ}$ können durch das Entfernen einer Schrägkante miteinander verschmelzen—in der Form $P_1 + P_2$ oder in der Form $P_1 + P_2^{-1}$.

Eine hinreichende Bedingung für Irreduzibilität

Eine unmittelbare Folgerung von Lemma 4.29 ist, dass ein Plangraph, der nur noch aus einer Planimplikationsklasse besteht, nicht weiter streng reduziert werden kann. Da die Bestimmung der Planimplikationsklassen mit gleichem Aufwand wie die der Implikationsklassen möglich ist, erhalten wir so eine schnell überprüfbare hinreichende Bedingung für Irreduzibilität.

Satz 4.30 (Plan mit 1 Planimplikationsklasse) [66] Besteht ein Plan $S \in \mathcal{S}_{SIJ}$ nur aus einer Planimplikationsklasse, dann ist S irreduzibel.

Zu Beginn dieses Kapitels haben wir den Begriff der Reduzierbarkeit an den beiden Plänen B_1 und B_9 veranschaulicht, und anschließend behauptet, dass B_9 sogar irreduzibel sei (Seite 66). Mit Satz 4.30 können wir diese Behauptung nun beweisen. Auch für die irreduziblen Pläne im Beispiel 4.8 (Abbildung 4.3 auf Seite 72) kann die Irreduzibilität jeweils mit diesem Argument nachgewiesen werden.

In der Tat stellt sich heraus, dass diese Bedingung ein äußerst wirkungsvolles Kriterium zum Nachweis der Irreduzibilität eines Plans ist. Beispielsweise lässt sich Satz 4.17 (zyklische Pläne) mit

Hilfe von Satz 4.30 beweisen. Es ist sogar alles andere als einfach, einen irreduziblen Plan zu finden, der nicht nur aus einer Planimplikationsklasse besteht. Solche Pläne gibt es jedoch, und wir werden noch auf sie zu sprechen kommen.

Dieses Kriterium liefert somit keinen polynomiellen Test auf Irreduzibilität. Es ist nur hinreichend, aber nicht ungedingt notwendig. Es liefert leider auch keinen NP-Test auf Irreduzibilität, der $\text{IRRED} \in \text{NP} \cap \text{co-NP}$ nach sich ziehen würde: Während es sehr viele anzunehmende (d.h. irreduzible) Instanzen gibt, deren Irreduzibilität damit sicher nachgewiesen werden kann, und alle abzulehnenden (d.h. nicht irreduziblen) Instanzen immer abgelehnt werden, existieren auch anzunehmenden Instanzen, deren Irreduzibilität mit diesem Test “auf keinem erlaubten Rechenweg” bewiesen werden kann.

4.2.3 Der Faktorgraph

Zur Beschreibung von Implikationsklassen auf einem Comparabilitygraphen $G = (V, E)$ haben wir in Abschnitt 1.2.4 den Γ -Graphen $G_\Gamma = (E, \Gamma)$ eingeführt. Dieser bildet ab, zwischen welchen Kanten in G eine Γ -Relation besteht. Die Zusammenhangskomponenten im Γ -Graphen sind gerade die Implikationsklassen von G . Nun wollen wir vor dem Hintergrund der Reduzierbarkeit von Plänen einen ähnlichen Graphen für H-Comparabilitygraphen definieren.

Durch das Entfernen von Kanten aus einem Graphen werden Γ -Relationen nur dann zerstört, wenn eine der beiden beteiligten Kanten entfernt wird. Da im Zusammenhang mit der Reduzierung von Plänen ausschließlich Schrägkanten entfernt werden können, sind Γ -Relationen, die zwischen regulären Kanten bestehen in jedem Fall unzerstörbar. Solche Γ -Relationen sind gerade die Γ_S -Relationen.

Das Entstehen neuer Γ -Relationen

Andererseits können durch das Entfernen von Schrägkanten neue Γ -Relationen entstehen; zwischen Planimplikationsklassen, zwischen Schrägkanten und Planimplikationsklassen, zwischen Schrägkanten untereinander, und sogar innerhalb von Planimplikationsklassen. Während jedoch die Entstehung von neuen Γ -Relationen zwischen verschiedenen Planimplikationsklassen und zwischen Schrägkanten und anderen Kanten eine hilfreiche Information liefern, ist die Entstehung von Γ -Relationen innerhalb einer Planimplikationsklasse irrelevant. Die beteiligten Kanten sind ohnehin in jedem Teilgraphen des gerade betrachteten H-Comparabilitygraphen, der den H-Graphen enthält, Γ -verbunden.

Ebensowenig ist bei der Entstehung einer Γ -Relation zwischen einer Schrägkante und einer Planimplikationsklasse von Belang, mit welcher Kante der Planimplikationsklasse diese Schrägkante in Γ -Relation steht. Für jede Γ -Verbindung von dieser Schrägkante zu irgendeiner Kante dieser Planimplikationsklasse gilt, dass sie ausschließlich durch Entfernen der Schrägkante unterbrochen werden kann.

Eine Variante des Γ -Graphen

Zur Beschreibung von Reduzierungsvorgängen wollen wir statt des Γ -Graphen in Zukunft den folgenden Graphen betrachten.

Definition 4.31 (Faktorgraph) Sei durch $G = (SIJ, E_{reg} + E_{diag})$ ein H-Graph über einer Operationenmenge SIJ mit einer beliebigen Schrägkantenmenge E_{diag} gegeben. Der Faktorgraph $G_{\mathcal{F}}$ ist der schlichte Graph, der aus dem Γ -Graphen $G_\Gamma = (E_{reg} + E_{diag}, \Gamma)$ durch die Kontraktion aller Kanten in G_Γ , die Γ_S -Relationen in G repräsentieren, entsteht. $G_{\mathcal{F}}$ heißt konsistent, wenn G_Γ konsistent ist.

Im Faktorgraphen $G_{\mathcal{F}}$ werden also alle Kanten, die in einer Planimplikationsklasse von G liegen, zu einem Knoten kontrahiert. In Abbildung 6.2 (links) auf Seite 123 (Beispiel 6.6) ist ein Beispiel für einen Faktorgraphen dargestellt.

$G_{\mathcal{F}}$ ist konsistent, wenn keine der Zusammenhangskomponenten einen Knoten enthält, der eine Schrägkante repräsentiert, deren Umkehrung ebenfalls in dieser Komponente liegt, oder wenn keiner der Knoten, die eine Planimplikationsklasse repräsentieren, gleichzeitig auch deren Umkehrung repräsentiert. Der Faktorgraph ist genau dann konsistent, wenn der zugehörige H-Graph mit beliebiger Schrägkantenmenge ein Comparabilitygraph ist (mit Folgerung 1.32).

4.2.4 Trennende Kantenmengen

Zur Beschreibung von Planimplikationsklassen ist auch der Begriff der *trennenden Kantenmenge* zwischen verschiedenen Planimplikationsklassen mitunter sehr hilfreich. Planimplikationsklassen sind $\tilde{\Gamma}$ -Komponenten bezüglich der Kantenmenge $F = E_{diag}$. Für zwei Planimplikationsklassen P_1 und P_2 aus einer gemeinsamen Implikationsklasse I gilt zum einen, dass es mindestens einen Γ -Weg von P_1 nach P_2 gibt. Zum anderen gilt aber, dass jeder dieser Γ -Wege mindestens eine Schrägkante enthält. Die Menge der Schrägkanten *trennt* somit die Kantenmengen P_1 und P_2 gewissermaßen voneinander. Ihre Existenz sorgt dafür, dass P_1 und P_2 bezüglich F in verschiedenen $\tilde{\Gamma}$ -Komponenten liegen.

Isolation einer Planimplikationsklasse gegenüber allen anderen

Um P_1 und P_2 voneinander zu trennen, sind in aller Regel jedoch nicht alle Kanten aus $F = E_{diag}$ notwendig. Da P_1 und P_2 zu einer gemeinsamen Implikationsklasse I gehören, reicht zum Trennen der beiden die Menge der Schrägkanten in I mit Sicherheit aus. D.h. P_1 und P_2 liegen auch bezüglich $F' = E_{diag} \cap I \subseteq F$ in verschiedenen $\tilde{\Gamma}$ -Komponenten. Es ist sogar möglich, dass eine echte Teilmenge von F' ausreicht, um P_1 und P_2 voneinander zu trennen.

Definition 4.32 (trennende Kantenmenge $F(P_i)$) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan, und sei $P_1, \dots, P_l \subseteq I$ die Menge der Planimplikationsklassen in einer Implikationsklasse $I \in \mathcal{I}_{[S^{tr}]}$ in $[S^{tr}] = (SIJ, E_{reg}(S) + E_{diag}(S))$. Eine Kante $e \in E_{diag} \cap I$ gehört zur trennenden Kantenmenge $F(P_i) = F_{P_i} = F_i$ einer Planimplikationsklasse P_i , wenn für ein $j \in \{1, \dots, l\} - i$ ein inklusionsminimaler Γ -Weg von P_i nach P_j existiert, der e enthält.

Die trennende Kantenmenge von P_i trennt P_i von allen anderen P_j aus I . Sie isoliert P_i . Wir nennen sie daher auch die *P_i isolierende Kantenmenge*. Sie ist eindeutig bestimmt. Bezüglich $F(P_i)$ zerfällt I mindestens in zwei $\tilde{\Gamma}$ -Komponenten. Eine dieser $\tilde{\Gamma}$ -Komponenten enthält als reguläre Kanten ausschließlich Kanten aus P_i . Wir bezeichnen die $\tilde{\Gamma}$ -Komponente, die P_i enthält mit \tilde{P}_i . Neben den Kanten aus P_i kann \tilde{P}_i jedoch auch Schrägkanten enthalten. Allerdings nur solche, die ausschließlich zu P_i in direkter oder indirekter Γ -Verbindung stehen.

$$B_{16} = \begin{bmatrix} 2 & 1 & & & & \\ & 2 & 3 & & & \\ & & 2 & 1 & & \\ & & & 2 & 3 & \\ & & & & 4 & \end{bmatrix} \quad B_{17} = \begin{bmatrix} 2 & 1 & & & & \\ & 2 & 3 & & & \\ & & 2 & 1 & & \\ & & & 2 & 3 & \\ & & & & 1 & \end{bmatrix} \quad B_{18} = \begin{bmatrix} 2 & 1 & & & & \\ & 2 & 3 & & & \\ & & 1 & 3 & & \\ & & & 2 & 1 & \\ & & & & 2 & \end{bmatrix}$$

Betrachten wir beispielsweise den Plan B_{16} , und die daraus abgeleiteten reduzierenden Pläne B_{17} und B_{18} . Sein H-Comparabilitygraph besteht aus vier verschiedenen Implikationsklassen und enthält fünf Schrägkanten. Eine der Implikationsklassen enthält drei verschiedenen Planimplikationsklassen (P_1 , P_2 und P_3), die drei anderen bestehen jeweils nur aus einer regulären Kante (P_4 und P_5) bzw. einer Schrägkante (e_5) (Abbildung 4.5). Der Plan B_{17} entsteht aus B_{16} durch Umkehren von P_5 . B_{18} entsteht aus B_{16} durch Umkehren von P_3 und P_4 . In $[B_{16}^{tr}]$ gilt $F_1 = \{e_1\}$, $F_2 = \{e_1, e_2\}$, $F_3 = \{e_2\}$ und $F_4 = F_5 = \emptyset$. Die Schrägkanten e_3 und e_4 liegen bezüglich F_3 in der $\tilde{\Gamma}$ -Komponente \tilde{P}_3 . In $[B_{17}^{tr}]$ gilt $F_1 = \{e_1\}$, $F_2 = \{e_1, e_2\}$, $F_3 = \{e_2, e_3\}$ bzw. $F_4 = \{e_3\}$.

Minimale trennende Kantenmengen

Zu jeder Kante $e \in F_i = F(P_i)$ existiert mindestens ein Γ -Weg von P_i zu einem $P_j \neq P_i$, der e enthält, und der nicht durch einen echten Teilweg ersetzt werden kann. Durch die Forderung, dass die betrachteten Γ -Wege inklusionsminimal sein sollen, wird ausgeschlossen, dass die trennenden Kantenmengen solche Schrägkanten enthalten, die in keinem Fall notwendig für die Abgrenzung von P_i sind. Ein Beispiel hierfür sind die Kanten e_3 und e_4 bezüglich P_3 in B_{16} .

Das bedeutet jedoch nicht, dass die so definierte Kantenmenge $F(P_i)$ eine inklusionsminimale Kantenmenge ist, die P_i von allen anderen P_j trennt. Mit $F^*(P_i) = F_i^*$ wollen wir den Schnitt von $F(P_i)$ mit der Vereinigung der Γ -Nachbarschaften aller Kanten aus P_i bezeichnen. Eine Kante $e \in F(P_i)$ liegt also genau dann in $F^*(P_i)$, wenn $e\Gamma e'$ für ein $e' \in P_i$ gilt. Wir schreiben hierfür auch $e\Gamma P_i$.

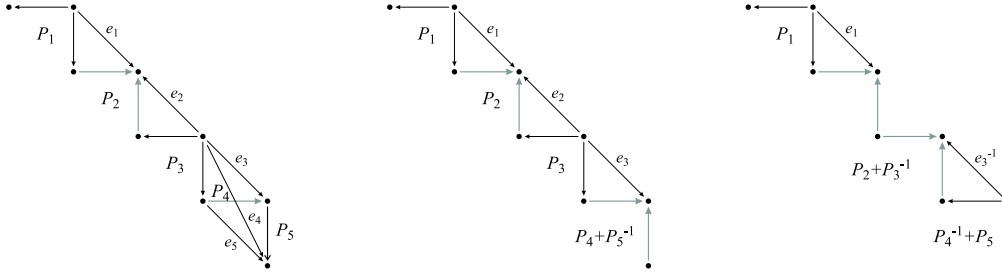


Abbildung 4.5: Die durch B_{16} , B_{17} bzw. B_{18} beschriebenen Planorientierungen von $[B_{16}^{tr}]$, $[B_{17}^{tr}]$ bzw. $[B_{18}^{tr}]$.

Mehrere Planimplikationsklassen

Die Notation der trennenden Kantenmenge zu eine Planimplikationsklasse lässt sich ohne weiteres auf mehrere Planimplikationsklassen ausweiten.

Definition 4.33 (trennende Kantenmenge $F(P_1, \dots, P_r)$) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan, und sei $\mathcal{P}_S = \{P_1, \dots, P_k\}$ die Menge der Planimplikationsklassen von S . Eine Kante $e \in F(P_1) \cup \dots \cup F(P_r)$ gehört zur trennenden Kantenmenge $F(P_1, \dots, P_r) = F_{P_1, \dots, P_r} = F_{1, \dots, r}$ zwischen den Planimplikationsklassen $P_1, \dots, P_r \in \mathcal{P}_S$, wenn ein inklusionsminimaler Γ -Weg von einem P_i zu einem P_j mit $i, j \in \{1, \dots, r\}$ existiert, der e enthält.

Für B_{17} gilt beispielsweise $F_{1,2} = \{e_1\}$, $F_{1,3} = \{e_1, e_2\}$, $F_{1,4} = \{e_1, e_2, e_3\}$ und $F_{1,2,4} = \{e_1, e_2, e_3\}$. Auch diese Mengen sind eindeutig bestimmt, aber in aller Regel nicht inklusionsminimal.

Wird aus $[B_{17}^{tr}]$ die Schrägkante e_2 entfernt, dann hat das unmittelbare Konsequenzen für alle Planimplikationsklassen P_i , für die die Kante e_2 in der isolierenden trennenden Kantenmenge F_i liegt. Durch die Entfernung von e_2 sind die Planimplikationsklassen P_2 und P_3 in ihrer Ausrichtung nicht mehr prinzipiell aneinander gebunden. Sie gehören dann nicht mehr einer gemeinsamen Implikationsklasse an, und können beispielsweise gegeneinander verdreht werden. Das führt zu den Plan B_{18} , der B_{17} streng reduziert.

Eigenschaften trennender Kantenmengen

Aus den Definitionen folgen unmittelbar die folgenden Eigenschaften.

Lemma 4.34 (trennende Kantenmengen) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan, und sei $\mathcal{P}_S = \{P_1, \dots, P_k\}$ die Menge der Planimplikationsklassen von S . Seien $\dot{P}_1, \dots, \dot{P}_k$ die $\dot{\Gamma}$ -Komponenten bezüglich $F = F_1 \cup \dots \cup F_k \subseteq E_{diag}$. Dann gelten

- (i) $\dot{P}_i \cap \dot{P}_j = \emptyset$ für alle $i \neq j$
- (ii) $E = \dot{P}_1 + \dots + \dot{P}_k + F$
- (iii) $F_{1, \dots, l} \subseteq F_1 \cup \dots \cup F_l$ für $\{1, \dots, l\} \subseteq \{1, \dots, k\}$, und $F_{1, \dots, k} = F_1 \cup \dots \cup F_k$

Beweis. Teil (i) und Teil (ii) folgen aus der Definition der $\dot{\Gamma}$ -Komponenten als Äquivalenzklassen der $\dot{\Gamma}^*$ -Relation. (iii) $F_{1, \dots, l} \subseteq F_1 \cup \dots \cup F_l$ ist trivial, da $F_{1, \dots, l}$ keine Schrägkante enthalten kann, die nicht in irgendeinem F_i ($i \in \{1, \dots, l\}$) liegt (Definition F_i). Jedoch trennt $F_{1, \dots, l}$ nur P_1, \dots, P_l untereinander; die übrigen Planimplikationsklassen werden unter Umständen nicht voneinander getrennt. Es sind die Fälle $P_i \subseteq \dot{P}_j$ und $P_i \subseteq \dot{P}_j^*$ mit $\dot{P}_j^* \neq \dot{P}_j$ möglich (mit $i \in \{l+1, \dots, k\}$ und $j \in \{1, \dots, l\}$). Für $l = k$ sind beide Fälle nicht möglich; hier muss Gleichheit gelten. ■

Insbesondere liegt jede Schrägkante eines H-Comparabilitygraphen entweder in der trennenden Kantenmenge $F = F_{1, \dots, k}$, oder sie gehört zu einer $\dot{\Gamma}$ -Komponente \dot{P}_i für ein $P_i \in \mathcal{P}_S$.

4.2.5 Stabile Schrägkanten

Zu einem Plan $S \in \mathcal{S}_{SIJ}$ können mehrere irreduzible Pläne existieren. Der Plan A aus Beispiel 4.8 besitzt mindestens fünf verschiedene ihn reduzierende irreduzible Pläne. Betrachten wir die Menge der Schrägkanten der bekannten irreduziblen Pläne von A (Abbildung 4.3 auf Seite 72), dann stellen wir fest, dass jeder dieser fünf Pläne die Kante $\widehat{1.4}$ enthält. Auf der anderen Seite enthält keiner dieser Pläne die Kante $\widehat{1.7}$, $\widehat{2.7}$, $\widehat{3.4}$, $\widehat{3.6}$, $\widehat{3.8}$ oder $\widehat{3.9}$.

Trivial-stabile Schrägkanten

WILLENUS [66] prägte für Schrägkanten zu einem Plan $S \in \mathcal{S}_{SIJ}$, die im Durchschnitt aller irreduziblen Pläne dieses Plans liegen, den Begriff der stabilen Schrägkante.

Definition 4.35 (stabile Schrägkante) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan, und sei $[S^{tr}]$ sein H -Comparabilitygraph. Eine Schrägkante, die im H -Comparabilitygraphen zu jedem irreduziblen Plan von S liegt, heißt stabil.

Für reguläre Kanten ist der Begriff der Stabilität irrelevant. Da jeder irreduzible Plan eine Orientierung auf der Menge der regulären Kanten ist, ist jede reguläre Kante stabil. Eine unmittelbare Folgerung aus den Überlegungen des vorangegangenen Abschnitts ist die folgende Aussage.

Lemma 4.36 (Kanten aus P^{tr}) [66] Sei $P \in \mathcal{P}_S$ eine Planimplikationsklasse von einem Plan $S \in \mathcal{S}_{SIJ}$. Dann ist jede Kante aus P^{tr} stabil.

Besitzt ein Plan $S \in \mathcal{S}_{SIJ}$ nur eine Planimplikationsklasse P , dann gilt $G(S) = P$ und $[S^{tr}] = P^{tr} + (P^{tr})^{-1}$. Jede der Schrägkanten ist dann stabil, und aus der Definition folgt unmittelbar die Irreduzibilität von S . Betrachten wir die Pläne B_{19} und B_{20} (WILLENUS [66]). Der H -Comparabilitygraph $[B_{19}^{tr}]$ besteht aus zwei verschiedenen Implikationsklassen, von denen eine aus nur einer regulären Kante besteht ($O_{11}O_{21}$). Das Umkehren dieser Kante führt zu dem Plan B_{20} . Die andere Implikationsklasse enthält lediglich eine einzige Planimplikationsklasse, in deren transitiver Hülle sämtliche Schrägkanten von $[B_{19}^{tr}]$ liegen. Die Pläne B_{19} und B_{20} sind damit sowohl ähnlich, als auch irreduzibel. Für jeden der beiden gilt dabei, dass seine Irreduzibilität nicht durch Satz 4.30 nachgewiesen werden kann.

$$B_{19} = \begin{bmatrix} 1 & 6 & 7 & 8 \\ 2 & 5 & 6 & 7 \\ 3 & 4 & 5 & 1 \\ 4 & 7 & 1 & 2 \end{bmatrix} \quad B_{20} = \begin{bmatrix} 2 & 6 & 7 & 8 \\ 1 & 5 & 6 & 7 \\ 3 & 4 & 5 & 1 \\ 4 & 7 & 1 & 2 \end{bmatrix}$$

Die transitiven Hüllen der Planimplikationsklassen eines Plans können sehr schnell bestimmt werden. Daher ist die Bedingung aus Lemma 4.36 ähnlich schnell überprüfbar, wie die Bedingung aus Satz 4.30. Insbesondere sind beide Bedingungen in polynomieller Zeit überprüfbar.

Es gibt jedoch auch Pläne, die stabile Schrägkanten enthalten, für die die Überprüfung der Stabilität nicht so einfach ist. Um diese qualitativ verschiedenen stabilen Schrägkanten besser unterscheiden zu können, wollen wir die leicht erkennbaren stabilen Schrägkanten in Zukunft auch trivial-stabile Schrägkanten nennen.

Definition 4.37 (trivial-stabile Schrägkante) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan, und sei $[S^{tr}]$ sein H -Comparabilitygraph. Eine Schrägkante, die in der transitiven Hülle einer Planimplikationsklasse liegt, heißt trivial-stabil.

Folgerung 4.38 (alle Schrägkanten trivial-stabil) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan. Sind alle Schrägkanten in $[S^{tr}]$ trivial-stabil, dann ist S irreduzibel.

Nachweis der Stabilität

Der Begriff der Stabilität von Schrägkanten liefert uns jedoch einen möglichen Ansatz zum Nachweis der Irreduzibilität für irreduzible Pläne, die weder nach Satz 4.30 noch nach Lemma 4.36 irreduzibel sind. Gelänge uns in polynomieller Zeit der Nachweis, dass eine gegebene Schrägkante stabil ist,

dann würde das einen polynomiellen Test für IRRED nach sich ziehen. Könnten wir die Stabilität einer gegebenen Schrägkante mit einem randomisierten Algorithmus wenigstens auf mindestens einem erlaubten Rechenweg nachweisen, so würde dies immerhin einen NP-Test für IRRED liefern, der $\text{IRRED} \in \text{NP} \cap \text{co-NP}$ nach sich ziehen würde.

Um zu zeigen, dass eine Schrägkante $\hat{e} \in E_{\text{reg}}(S)$ zu einem Plan $S \in \mathcal{S}_{SIJ}$ stabil ist, muss gezeigt werden, dass kein Comparabilitygraph G mit $G_{SIJ} \subseteq G \subseteq [S^{\text{tr}}] - \hat{e}$ existiert. Das führt uns zu einer weiteren Problemformulierung für IRRED.

Problem IRREDUCIBILITY (IRRED) (#3): Gegeben sei ein Plan $S \in \mathcal{S}_{SIJ}$ für das Open-Shop Problem $O \parallel C_{\text{max}}$. Gilt für jede Schrägkante $e \in E_{\text{diag}}(S)$ in $[S^{\text{tr}}] = (SIJ, E_{\text{reg}}(S) + E_{\text{diag}}(S))$, dass keiner der Graphen G mit $G_{SIJ} \subseteq G \subseteq [S^{\text{tr}}] - \hat{e}$ ein Comparabilitygraph ist?

Die Antwort auf die Frage “Existiert in $[S^{\text{tr}}]$ eine Schrägkante $e \in E_{\text{diag}}(S)$, für die ein Comparabilitygraph G mit $G_{SIJ} \subseteq G \subseteq [S^{\text{tr}}] - \hat{e}$ existiert?” liefert eine Antwort für co-IRRED.

4.2.6 Planimplikationsklassen als Bausteine von Plänen

In Abschnitt 1.2.7 haben wir Implikationsklassen als fundamentale Bausteine von transitiven Orientierungen beschrieben. Ist $\mathcal{I}_G = \{I_1, \dots, I_l, I_1^{-1}, \dots, I_l^{-1}\}$ die Menge der Implikationsklassen eines Comparabilitygraphen $G = (V, E)$, dann ist jede transitive Orientierung $T \in \mathcal{T}_G$ eine Kombination von Orientierungen auf den Farbklassen \hat{I}_j ($j = 1, \dots, l$), $T = J_1 + \dots + J_l$ mit $J_j \in \{I_j, I_j^{-1}\}$ ($j = 1, \dots, l$). Umgekehrt liefert jede solche Rekombination der Implikationsklassen eine potentiell transitive Orientierung, die genau dann transitiv ist, wenn sie kreisfrei ist.

Zur Beschreibung von Beispielen wollen wir die folgende Notation vereinbaren (vgl. Folgerung 1.20).

Definition 4.39 (Rekombination der Planimplikationsklassen) Sei $\mathcal{P}_S = \{P_1, \dots, P_k\}$ die Menge der Planimplikationsklassen von einem Plan $S \in \mathcal{S}_{SIJ}$, und sei $M = \{i_1, \dots, i_m\} \subseteq \{1, \dots, k\}$. Wir nennen die Orientierung $S_M = S - P_{i_1} - \dots - P_{i_m} + P_{i_1}^{-1} + \dots + P_{i_m}^{-1}$, die durch Umkehren der Orientierung auf allen \hat{P}_{i_j} , $i_j \in M$, entsteht, eine Rekombination der Planimplikationsklassen von S . Ist $S_M \in \mathcal{S}_{SIJ}$ ein Plan, dann heißt er der resultierende Plan aus der durch M beschriebenen Rekombination.

Für Planimplikationsklassen und Pläne gelten ähnliche Aussagen, wie für Implikationsklassen. Jeder reduzierende Plan ist eine Kombination von Orientierungen auf den Klassen \hat{P}_i .

Satz 4.40 (reduzierende Pläne sind Rekombinationen) Sei $\mathcal{P}_S = \{P_1, \dots, P_k\}$ die Menge der Planimplikationsklassen von einem Plan $S \in \mathcal{S}_{SIJ}$. Ist S^* ein Plan, der S reduziert, $S^* \preceq S$, dann ist S^* eine Rekombination der Planimplikationsklassen von S , $S^* = Q_1 + \dots + Q_k$ mit $Q_i \in \{P_i, P_i^{-1}\}$ für alle $i = 1, \dots, k$.

Beweis. Sei $S = P_1 + \dots + P_k$. Sowohl S , als auch $S^* = P_1^* + \dots + P_l^*$ sind Orientierungen des H-Graphen $G_{SIJ} = (SIJ, E_{\text{reg}}(SIJ))$, $[S] = [S^*] = E_{\text{reg}}(SIJ)$, und $S \cap S^{-1} = \emptyset$ bzw. $S^* \cap (S^*)^{-1} = \emptyset$. Nach Lemma 4.29 gilt in S^* für jede Planimplikationsklasse P_i entweder $P_i \subseteq P_j^*$ oder $P_i^{-1} \subseteq P_j^*$. Für jede Orientierung $S^* \subseteq E_{\text{reg}}(SIJ)$ von G_{SIJ} , die diese Eigenschaft besitzt, muss $S^* = Q_1 + \dots + Q_k$ mit $Q_i \in \{P_i, P_i^{-1}\}$ für alle $i = 1, \dots, k$ gelten. Jede Planimplikationsklasse P_j^* von S^* besteht aus einer Kombination von Planimplikationsklassen aus einer Teilmenge aller Planimplikationsklassen von S , $P_j^* = Q_{j_1} + \dots + Q_{j_s}$ mit $Q_{j_s} \in \{Q_1, \dots, Q_k\}$. ■

Eine unmittelbare Folgerung ist, dass die Anzahl der Planimplikationsklassen eines reduzierenden Plans nicht größer als die des Ausgangsplans sein kann.

Folgerung 4.41 (Anzahl der Planimplikationsklassen) Für die Anzahl der Planimplikationsklassen zweier Pläne $S, S^* \in \mathcal{S}_{SIJ}$ mit $S^* \preceq S$ gilt $|\mathcal{P}_{S^*}| \leq |\mathcal{P}_S|$.

Eine weitere unmittelbare Folgerung ist, dass bei der Reduktion über eine Kante nach Satz 4.14 die betreffende Kante $e = ab$ zwingend eine eigene Planimplikationsklasse bilden muss. Daraus wiederum

folgt, dass nicht nur die in Satz 4.14 genannten Schrägkanten existieren, sondern a zu *jedem* Knoten aus der Spalte von b , und b zu *jedem* Knoten aus der Spalte von a adjazent ist.

Umgekehrt gilt, ähnlich wie bei Implikationsklassen, dass jede kreisfreie Rekombination der Planimplikationsklassen einen Plan beschreibt.

Satz 4.42 (Rekombinationen sind Pläne) Sei $\mathcal{P}_S = \{P_1, \dots, P_k\}$ die Menge der Planimplikationsklassen von einem Plan $S \in \mathcal{S}_{SIJ}$. Eine beliebige Rekombination $S^* = Q_1 + \dots + Q_k$ mit $Q_i \in \{P_i, P_i^{-1}\}$ ($i = 1, \dots, k$) liefert genau dann einen Plan $S^* \in \mathcal{S}_{SIJ}$, wenn sie kreisfrei ist.

Beweis. Jede beliebige Orientierung der regulären Kanten zu einer Operationenmenge liefert genau dann einen Plan, wenn sie kreisfrei ist. Eine Rekombination $S^* = Q_1 + \dots + Q_k$ (mit $Q_i \in \{P_i, P_i^{-1}\}$) von Planimplikationsklassen P_1, \dots, P_k eines Ausgangsplans $S \in \mathcal{S}_{SIJ}$ beschreibt eine Orientierung der regulären Kanten über SIJ . Also ist S^* genau dann ein Plan (zulässige Kombination von Organisation und Technologie), wenn $Q_1 + \dots + Q_k$ kreisfrei ist. ■

Rekombinationen von Planimplikationsklassen und reduzierende Pläne

Jede beliebige kreisfreie Rekombination der Planimplikationsklassen eines Ausgangsplans liefert also einen Plan. Nicht jede solche Rekombination liefert jedoch auch einen Plan, der den Ausgangsplan reduziert.

Satz 4.43 (Rekombinationen und reduzierende Pläne) Sei $\mathcal{P}_S = \{P_1, \dots, P_k\}$ die Menge der Planimplikationsklassen von einem Plan $S \in \mathcal{S}_{SIJ}$. Eine kreisfreie Rekombination $S^* = Q_1 + \dots + Q_k$ mit $Q_i \in \{P_i, P_i^{-1}\}$ ($i = 1, \dots, k$) liefert genau dann einen reduzierenden Plan $S^* \in \mathcal{S}_{SIJ}$ ($S^* \preceq S$), wenn jede Schrägkante $e \in (Q_1 + \dots + Q_k)^{tr}$ auch eine Schrägkante in S^{tr} oder $(S^{-1})^{tr}$ ist.

Beweis. Eine kreisfreie Rekombination $S^* = Q_1 + \dots + Q_k$ mit $Q_i \in \{P_i, P_i^{-1}\}$ für alle $i = 1, \dots, k$ beschreibt nach Satz 4.42 einen Plan $S^* \in \mathcal{S}_{SIJ}$. Gilt darüber hinaus $[(S^*)^{tr}] \subseteq [S^{tr}]$, dann reduziert S^* den Ausgangsplan S nach Satz 4.2. ■

Durch das Umkehren einzelner Planimplikationsklassen in einem Plan $S \in \mathcal{S}_{SIJ}$ können u.U. tatsächlich Schrägkanten induziert werden, die in $[S^{tr}]$ nicht enthalten sind (Beispiel 4.44). Wir wollen solche Schrägkanten in Zukunft als *neue Schrägkanten* bezeichnen. Jede Rekombination der Planimplikationsklassen von $S \in \mathcal{S}_{SIJ}$, die kreisfrei ist, und keine neuen Schrägkanten induziert, nennen wir *zulässig*.

Mit diesen Überlegungen ergibt sich eine weitere Problemformulierung für IRRED.

Problem IRREDUCIBILITY (IRRED) (#4): Gegeben sei ein Plan $S \in \mathcal{S}_{SIJ}$ für das Open-Shop Problem $O \parallel C_{\max}$. Liefern alle zulässigen Rekombinationen der Planimplikationsklassen von S Pläne, die ähnlich zu S sind?

Die Antwort auf die Frage ‐Existiert eine zulässige Rekombination der Planimplikationsklassen von S , bei der mindestens eine Schrägkante entfernt wird?‑ liefert eine Antwort für co-IRRED.

Betrachten wir zur Veranschaulichung der Interpretation der Planimplikationsklassen als fundamentaler Bausteine der reduzierenden Pläne zu einem Ausgangsplan noch einmal den Plan A aus Beispiel 4.8. Zu Beginn des vorangegangenen Abschnitts haben wir den Eindruck vermittelt, dass die Kante $\widehat{1.4}$ im Plan A möglicherweise stabil sei. Dies ist keineswegs der Fall. Der Plan A besitzt neben den fünf bereits gezeigten noch weitere irreduzible Pläne.

Beispiel 4.44 (Reduktion durch Umkehren von Planimplikationsklassen) In Beispiel 4.8 haben wir gezeigt, dass der Plan $A \in \mathcal{S}_{SIJ}$ acht Implikationsklassen I_1, \dots, I_8 (Abbildung 4.2) und 14 Schrägkanten besitzt (Abbildung 4.6).

$$A = \begin{bmatrix} 1 & 2 & & & 6 \\ & 3 & 4 & 1 & 5 \\ 2 & & 3 & & \end{bmatrix}.$$

Drei der Implikationsklassen zerfallen in jeweils zwei Planimplikationsklassen. Eine besteht aus einer einzelnen Schrägkante. Es sei $P_1 \subseteq I_1$, $P_2 + P_3 \subseteq I_2$, $P_4 \subseteq I_3$, $P_5 + P_6 \subseteq I_4$, $P_7 \subseteq I_5$, $P_8 \subseteq I_6$,

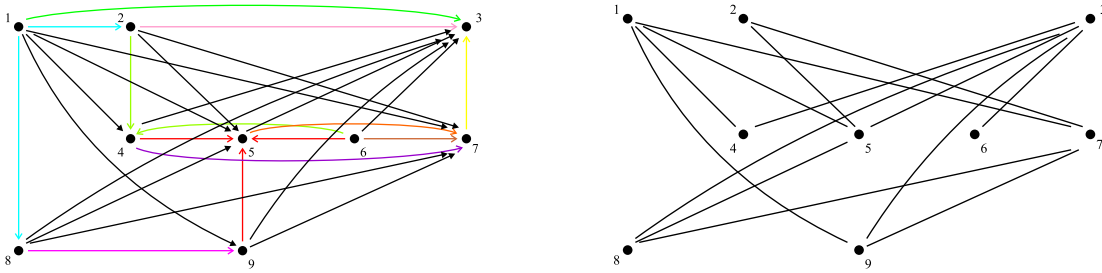


Abbildung 4.6: Der Plangraph $G(A)$ und die Menge der Schrägkanten von $[A^{tr}]$.

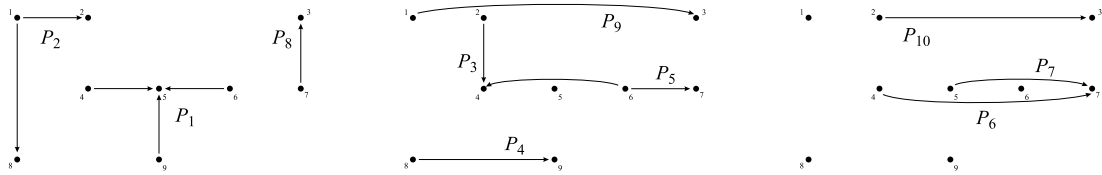


Abbildung 4.7: Die Zerlegung der Kantenmenge des Plangraphen in Planimplikationsklassen $A = P_1 + \dots + P_{10}$.

$P_9 + P_{10} \subseteq I_7$ und $I_8 = \widehat{8.5}$ (Abbildung 4.7).

Wir wissen bereits, dass es 36 verschiedene reduzierende Pläne gibt, die als Rekombinationen der Implikationsklassen von A dargestellt werden können. Ferner wissen wir bereits, dass darunter auch 5 irreduzible Pläne sind (Abbildung 4.3).

Es stellt sich nun die Frage, ob es noch weitere reduzierende Pläne zu A gibt. Und insbesondere, ob es noch weitere irreduzible Pläne gibt. Nach Satz 4.40 ist jeder reduzierte Plan zu A als eine Rekombination der Planimplikationsklassen darstellbar. Und nach Satz 4.42 beschreibt jede kreisfreie Rekombination der Planimplikationsklassen einen Plan. Von den $2^{10} = 1024$ möglichen Rekombinationen der Planimplikationsklassen P_1, \dots, P_{10} von A , sind genau 344 kreisfrei. Damit gibt es 172 verschiedene Pläne über SIJ , die durch die Planimplikationsklassen von A darstellbar sind. Nicht jeder dieser Pläne reduziert jedoch A . Kehren wir beispielsweise gegenüber A die Planimplikationsklassen P_2 und P_9 um, dann gelangen wir zu dem Plan $A_{29} = A_{1345678.10}^{-1}$. In $[A_{29}^{tr}]$ existiert die transitive Schrägkante $\widehat{6.1}$, $6.1 \in (P_5 + P_6 + P_9^{-1})^{tr}$. Diese Schrägkante gehört nicht zur Schrägkantenmenge von $[A^{tr}]$. Daher kann unabhängig von allen anderen Schrägkanten nicht $[A_{29}^{tr}] \subseteq [A^{tr}]$ gelten. Also reduziert A_{29} den Plan A nicht, $A_{29} \not\leq A$. (In diesem Fall gilt sogar, dass A umgekehrt A_{29} reduziert.) Von den acht Nichtkanten in $[A^{tr}]$ können immerhin sechs als neue Schrägkanten auftauchen. Lediglich zu den Nichtkanten $\widehat{4.8}$ und $\widehat{4.9}$ gibt es keine kreisfreien Rekombinationen, die sie als transitive Kanten induzieren.

Von den 172 verschiedenen Plänen, die durch Rekombination der Planimplikationsklassen entstehen können, enthalten 44 mindestens eine neue Schrägkante. Die verbleibenden 128 Pläne erfüllen die Bedingungen von Satz 4.43, und reduzieren demnach A . Darunter befinden sich auch die 36 bereits bekannten reduzierenden Pläne. Das bedeutet, es existieren $128 - 36 = 92$ reduzierende Pläne, die nicht durch Umkehren von Implikationsklassen generiert werden können. Jeder dieser Pläne reduziert A streng. Neben den 5 bereits entdeckten gibt es noch 10 weitere irreduzible Pläne. Diese 15 verschiedenen irreduziblen Pläne sind in Abbildung 4.8 abgebildet. Die Irreduzibilität kann für jeden dieser Pläne leicht nachgeprüft werden; jeder besteht nur aus einer Planimplikationsklasse. Alternativ ergeben sich bei Vorliegen der Menge aller reduzierenden Pläne zu einem Ausgangsplan die irreduziblen Pläne auch als die minimalen Elemente in der durch " \subseteq " bestimmten Halbordnung auf der Menge der H -Comparabilitygraphen. Allerdings ist hierfür natürlich die Betrachtung aller Rekombination notwendig. Das ist nur mit exponentiellem Aufwand realisierbar.

Die in Beispiel 4.8 gefundenen 5 irreduziblen Pläne sind genau jene, bei denen keine der drei Implikationsklassen, die mehrere Planimplikationsklassen enthalten, auseinandergerissen ist. D.h., bei denen

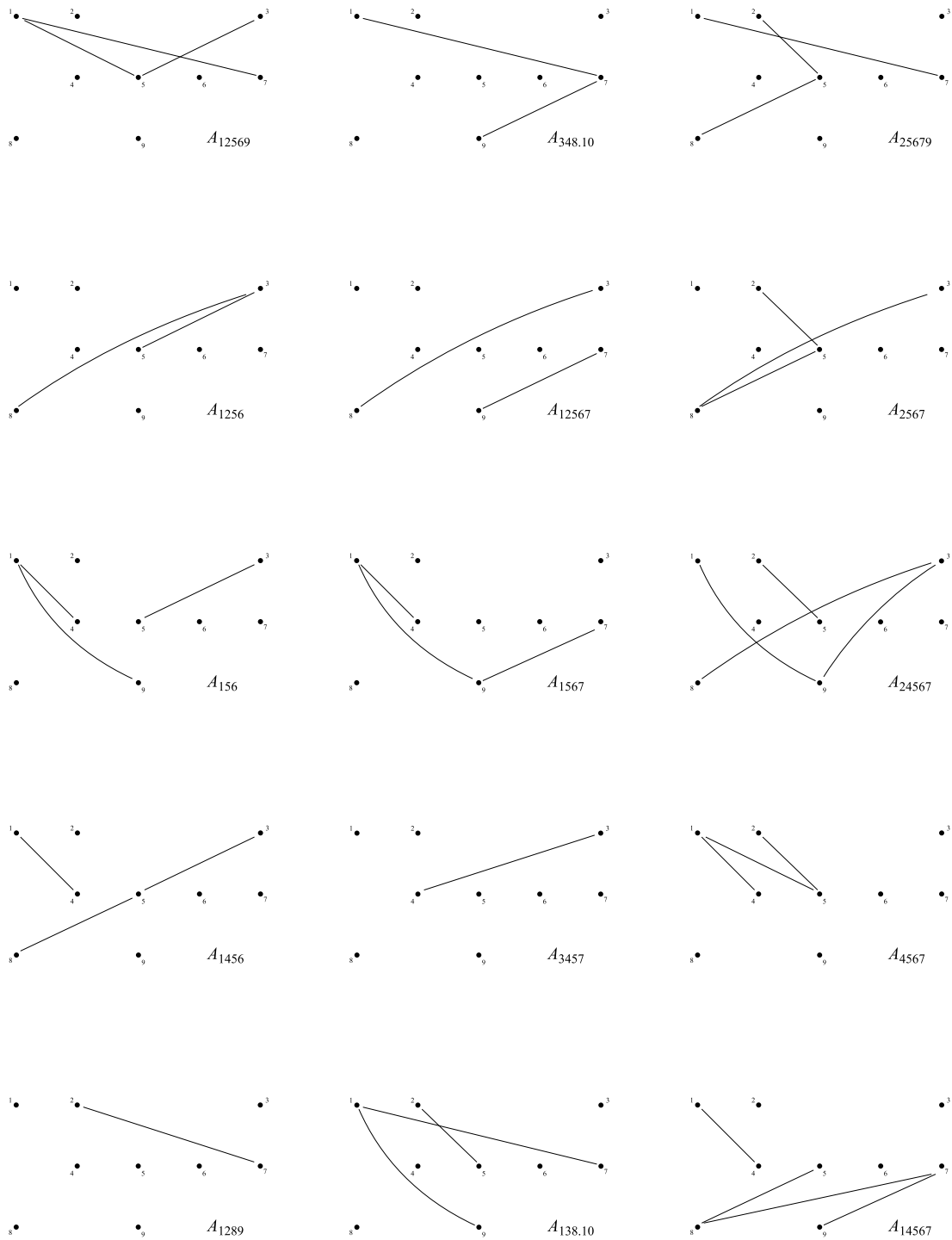


Abbildung 4.8: Die Schrägkantenmengen aller 15 irreduziblen Pläne zu dem Plan A aus Beispiel 4.44.

weder P_2 gegen P_3 , P_5 gegen P_6 , noch P_9 gegen P_{10} verdreht wurde.

Da der Durchschnitt der Schrägkanten aller irreduziblen Pläne von A leer ist, ist keine der Schrägkanten von A stabil. Umgekehrt ergibt die Vereinigung der Schrägkanten aller irreduzibler Pläne nicht die Schrägkantenmenge von A . (Die Kante $6.3 \in A$ ist in keinem irreduziblen Plan zu A enthalten.)

In Beispiel 4.8 sind wir durch Entfernen der Schrägkante $\widehat{1.5}$ aus $[A^{tr}]$ zu einem Plan S gelangt (Abbildung 4.4).

Der Plan $S = A_{234}$ entsteht aus A durch Umkehren von P_2 , P_3 und P_4 . Neben den bereits bekannten irreduziblen Plänen zu A_{234} (A_{156} und A_{1567}) gibt es noch drei weitere: A_{1256} , A_{3457} und A_{12567} . Der Durchschnitt der Schrägkanten dieser fünf Pläne ist ebenfalls leer. Also ist auch in A_{234} keine der acht Schrägkanten stabil. Entfernen wir nun aus $[A_{234}^{tr}]$ genau wie in Beispiel 4.8 die Kante $\widehat{3.4}$, und zusätzlich auch die Kante $\widehat{3.8}$, dann kommen von den fünf genannten irreduziblen Plänen von A_{234} nur noch die Pläne A_{156} und A_{1567} in Frage. Beide Pläne enthalten sowohl die Kante $\widehat{1.4}$, als auch die Kante $\widehat{1.9}$. Diese Kanten sind in $[A_{234}^{tr}] - \widehat{3.4} - \widehat{3.8}$ trivial-stabile Kanten. (Sie liegen beide in $(P_1 + P_2^{-1} + P_3^{-1} + P_4^{-1} + P_6 + P_8^{-1} + P_9^{-1} + P_{10}^{-1})^{tr}$.) ■

4.2.7 Erweiterte Planimplikationsklassen

Nachdem wir nun wissen, dass die Planimplikationsklassen die Bausteine sind, aus denen die reduzierenden Pläne eines Ausgangsplans zusammengesetzt sind, wollen wir noch auf eine Verfeinerung der Resultate des vorangegangenen Abschnitts eingehen.

Der H-Comparabilitygraph zu dem Plan B_{21} ist eindeutig transitiv orientierbar, und enthält genau zwei Schrägkanten. Beide Schrägkanten sind trivial-stabil. D.h. B_{21} ist irreduzibel. Die regulären Kanten zerfallen in zwei Planimplikationsklassen P_1 und P_2 , mit $P_2 = \{O_{22}O_{24}\}$, und $P_1 = B_{21} - P_2$. Die einzige alternative Rekombination der Planimplikationsklassen $P_1 + P_2^{-1}$ ist nicht kreisfrei. D.h. B_{21} ist irreduzibel, und außer B_{21}^{-1} gibt es keinen weiteren ähnlichen Plan.

$$B_{21} = \begin{bmatrix} 1 & & 3 & & \\ & 1 & 2 & 3 & \\ & & 1 & & 2 \end{bmatrix}$$

Der Grund dafür, dass $P_1 + P_2^{-1}$ einen Kreis enthält, ist, dass die Planimplikationsklasse P_2 in der transitiven Hülle von P_1 liegt, $P_2 \subseteq P_1^{tr}$. Es gibt also prinzipiell keine Möglichkeit, die beiden betrachteten Planimplikationsklassen gegeneinander zu verdrehen.

Definition und Berechnung erweiterter Planimplikationsklassen

In einer solchen Situation ist es algorithmisch sinnvoll, diese beiden Planimplikationsklassen nicht als zwei unabhängig voneinander orientierbare Bausteine von Plänen zu betrachten, sondern als einen gemeinsamen Baustein. WILLENUS [66] hat solche Konstellationen durch den Begriff der *erweiterten Planimplikationsklassen* beschrieben.

Definition 4.45 (erweiterte Planimplikationsklasse) Sei $P \in \mathcal{P}_S$ eine Planimplikationsklasse von einem Plan $S \in \mathcal{S}_{SIJ}$. Die erweiterte Planimplikationsklasse P^* von P ist die kleinste Kantenmenge mit $P \subseteq P^*$, für die $(P^*)^{tr} = P^*$, und entweder $Q \cap P^* = Q$ oder $Q \cap \mathcal{P}_S = \emptyset$ für alle $Q \in \mathcal{P}_S$ gilt.

Die erweiterte Planimplikationsklasse P^* kann Schrägkanten enthalten. Sie kann, wie im Fall von B_{21} , aus mehreren Planimplikationsklassen und ihren induzierten *trivial-stabilen* Schrägkanten bestehen. Sie ist jedoch in jedem Fall eine Teilmenge der Implikationsklasse $I(P) \in \mathcal{I}_{[S^{tr}]}$ von $P \in \mathcal{P}_S$.

Die erweiterten Planimplikationsklassen zu allen Planimplikationsklassen P_1, \dots, P_k eines Plans $S \in \mathcal{S}_{SIJ}$ mit n Aufträgen auf m Maschinen können folgendermaßen in polynomieller Laufzeit ($\mathcal{O}(n^6 m^6)$) bestimmt werden (WILLENUS [66]). Ausgehend von einem $P_i \in \mathcal{P}_S$ wird zunächst $P_i^{tr} = Q'$ bestimmt. Gilt $P_i^{tr} \cap P_j \neq \emptyset$ für ein $P_j \in \mathcal{P}_S$, so wird P_j vollständig zu Q' hinzugefügt. Zu diesem Q' mit $P_i^{tr} + P_j \subseteq Q'$ wird anschließend erneut die transitive Hülle $(Q')^{tr} = Q''$ bestimmt. Dieser Vorgang wiederholt sich solange, bis keine neue Kante mehr hinzugefügt wird.

Bemerkung 4.46 (erweiterte Planimplikationsklassen und Pläne) Die Aussagen in Lemma 4.24, Lemma 4.29, Satz 4.30, Lemma 4.36, Folgerung 4.38, Satz 4.40, Satz 4.42 und Satz 4.43 behalten ihre Gültigkeit, wenn man jeweils “Planimplikationsklasse” durch “erweiterte Planimplikationsklasse” ersetzt. Das gleiche gilt für die Problemformulierung #4 für IRRED.

Die Bedeutung erweiterter Planimplikationsklassen für die Reduzierung

Trotz der scheinbaren Bedeutung dieses Begriffs liefert die Betrachtung von erweiterten Planimplikationsklassen keine weiterführenden Erkenntnisse. Die Eigenschaften der Planimplikationsklassen sind relevant. Die erweiterten Planimplikationsklassen reduzieren allenfalls die Kombinationsmöglichkeiten für die Anzahl der reduzierenden Pläne geringfügig, oder erhöhen die Anzahl der als trivial-stabil erkannten Schrägkanten. Beides erleichtert den quantitativen Umgang bei der Betrachtung konkreter Instanzen, hat jedoch keinen Einfluss auf die qualitativen Schwierigkeiten von IRRED. Darüber hinaus ist der Fall, dass in einem Plan auch nur eine erweiterte Planimplikationsklasse P^* zu einer Planimplikationsklasse P mehr Kanten als P^{tr} enthält, eine eher seltene Ausnahme.

Der zentrale Begriff bleibt der der Planimplikationsklasse. Wir werden uns daher darauf beschränken, mit Planimplikationsklassen zu argumentieren, und allenfalls bei der Formulierung von Theoremen und Algorithmen auf den Begriff der erweiterten Planimplikationsklassen zurückgreifen.

4.3 Immer-transitive Kanten und stabile Schrägkanten

In Abschnitt 1.4.1 haben wir immer-transitive Kanten beschrieben. Das sind Kanten in einem Comparabilitygraphen, die in jeder transitiven Orientierung transitiv sind. Für immer-transitive Kanten gilt, dass sie eine der beiden Konstellationen aus Abbildung 1.8 auf Seite 28 erfüllen.

Stabile Schrägkanten sind immer-transitiv

Stabile Schrägkanten sind Schrägkanten, die in jeder kreisfreien und reduzierenden Rekombination der Planimplikationsklassen eines H-Comparabilitygraphen transitiv sind. Das bedeutet insbesondere, dass sie in jeder kreisfreien Rekombination der Implikationsklassen transitiv sind. Damit sind stabile Schrägkanten immer-transitiv, und erfüllen eine der beiden Konstellationen aus Abbildung 1.8.

Ferner gilt für stabile Schrägkanten, dass sie nicht aus dem H-Comparabilitygraph entfernt werden können. D.h. sie erfüllen nicht nur eine der beiden angesprochenen Konstellationen, sondern darüber hinaus auch die entsprechende Bedingung aus Satz 1.44—mindestens zwei an einer solchen Konstellation beteiligte Kanten aus der Implikationsklasse der betrachteten Schrägkante liegen in einer gemeinsamen $\bar{\Gamma}$ -Komponente bezüglich dieser Schrägkante.

Umgekehrt ist jedoch nicht jede immer-transitive Schrägkante, die diese Bedingungen erfüllt, auch stabil. Die Kante 4.1 aus der Schrägkantenmenge zu dem Plan A_{234} aus Beispiel 4.44 (Abbildung 4.4 auf Seite 72) ist beispielsweise immer-transitiv innerhalb ihrer Implikationsklasse I mit $(P_1 + P_3^{-1} + P_4^{-1} + P_6) + P_2^{-1} + P_5 \subseteq I$. Zwischen den beiden transitivierenden Kanten $4.2 \in P_3^{-1}$ und $2.1 \in P_2^{-1}$ existiert ein Γ -Weg, der die Schrägkante 4.1 nicht enthält (4.2 Γ 4.5 Γ 9.5 Γ 9.1 Γ 2.1). D.h. beide Kanten liegen bezüglich 4.1 in einer gemeinsamen $\bar{\Gamma}$ -Komponente. Von den 5 irreduziblen Plänen zu A_{234} (A_{1256} , A_{156} , A_{3457} , A_{12567} , A_{1567}) enthalten jedoch nur 2 auch die Schrägkante $\widehat{1.4}$ (Abbildung 4.8).

4.3.1 Implikationsklassen und Planimplikationsklassen

Es gibt jedoch noch einige weitere Zusammenhänge zwischen immer-transitiven Kanten und stabilen Schrägkanten.

Stabile Schrägkanten und immer-transitive Kanten sind ähnlich

Implikationsklassen sind die fundamentalen Bausteine für transitive Orientierungen eines Comparabilitygraphen. Eine immer-transitive Kante ist in jeder zulässigen Anordnung dieser Bausteine eine transitive Kante. Planimplikationsklassen, auf der anderen Seite, sind die fundamentalen Bausteine

für die reduzierenden Pläne zu einem H-Comparabilitygraphen. Eine Schrägkante ist stabil, wenn sie in jeder zulässigen Anordnung dieser Bausteine transitiv ist.

Damit verhalten sich stabile Schrägkanten bezüglich Planimplikationsklassen genauso, wie sich immer-transitive Kanten bezüglich Implikationsklassen verhalten.

Implikationsklassen und Planimplikationsklassen sind ähnlich

Darüber hinaus haben Planimplikationsklassen in manchen Zusammenhängen fast die gleichen lokalen Eigenschaften wie Implikationsklassen. Implikationsklassen haben die folgende Eigenschaft: Wann immer zwei Kanten $ab \in I_1$ und $ac \in I_2$ aus Implikationsklassen I_1 und $I_2 \neq I_1$ einen gemeinsamen Knoten haben, dann gehören beide Kanten zu einem gemeinsamen Dreieck—die Kante \widehat{bc} existiert. Andernfalls würde aus der Definition der Γ -Relation $I_1 = I_2$ folgen. Wir haben diese Eigenschaft im Beweis des Dreieckslemmas (Satz 1.15) von Golumbic benutzt. Man beachte bitte, dass dies auch im Fall $I_2 = I_1^{-1}$ gilt. Dann folgt die Existenz von cb , und es gilt $cb \in I_1$ (mit Satz 1.16).

Für $\dot{\Gamma}_F$ -Komponenten zu einer beliebigen ungerichteten Teilmenge $F \subseteq E$ der Kanten zu einem Graphen $G = (V, E)$ gilt diese Eigenschaft ebenfalls. Liegen zwei Kanten $ab \in \dot{I}_1$ und $ac \in \dot{I}_2$ mit einem gemeinsamen Knoten in verschiedenen $\dot{\Gamma}$ -Komponenten \dot{I}_1 und $\dot{I}_2 \neq \dot{I}_1$. Dann existiert die Kante $\widehat{bc} \in E$.

Da Planimplikationsklassen $\dot{\Gamma}_F$ -Komponenten bezüglich der Kantenmenge $F = E_{diag}$ der Schrägkanten eines H-Comparabilitygraphen sind, verhalten sich Planimplikationsklassen bezüglich der Existenz der dritten Kante in einem Dreieck mit zwei Kanten aus unterschiedlichen Planimplikationsklassen, genau wie Implikationsklassen.

4.3.2 Das erweiterte Dreieckslemma

Das Dreieckslemma von Golumbic besagt, dass in einem Graphen zwei Dreiecke, die bezüglich der Zugehörigkeit der Kanten zu Implikationsklassen in zwei Kanten übereinstimmen, bei Vorliegen einiger Voraussetzungen auch in der dritten Kante übereinstimmen (Satz 1.15). Für den Beweis der Charakterisierung immer-transitiver Kanten (Satz 1.44) ist das Dreieckslemma von entscheidender Bedeutung. Für Teilstrukturen von Implikationsklassen gilt die Ähnlichkeit der Dreiecke nicht. Da sich jedoch $\dot{\Gamma}_F$ -Komponenten und Implikationsklassen in vielerlei Hinsicht gleich verhalten, können wir hoffen, die Aussagen des Dreieckslemmas so auf Planimplikationsklassen verallgemeinern zu können, dass wir dennoch zu einer Beschreibung von stabilen Schrägkanten gelangen.

Satz 4.47 (erweitertes Dreieckslemma) *Sei $G = (V, E)$ ein Graph, und seien \dot{A} , \dot{B} und \dot{C} $\dot{\Gamma}$ -Komponenten bezüglich einer Teilmenge $F \subseteq E \in \mathcal{I}_G$ einer Implikationsklasse mit $\dot{A} \neq \dot{B}^{-1}$ und $\dot{A} \neq \dot{C}$. Ferner seien $ab \in \dot{B}$, $ac \in \dot{C}$ und $bc \in \dot{A}$. Gilt außerdem $\dot{B}, \dot{C} \not\subseteq I$ oder $a \notin V(F)$, dann folgt aus der Existenz von $b'c' \in \dot{A}$ die Existenz von $ab' \in \dot{B}$ und $ac' \in \dot{C}$, und keine Kante aus \dot{A} ist adjazent zu Knoten a .*

Beweis. Der Beweis erfolgt völlig analog zu dem Beweis von Satz 1.15. Sei $b'c' \in \dot{A}$. Dann existiert eine kanonische $\dot{\Gamma}$ -Kette $bc = b_0c_0\dot{\Gamma}b_1c_1\dot{\Gamma}\dots\dot{\Gamma}b_kc_k = b'c'$ in \dot{A} . Wir zeigen nun per Induktion die Existenz von $ab_i \in \dot{B}$ und $ac_i \in \dot{C}$ für alle $i = 0, \dots, k$. Für $i = 0$ ist nichts zu tun, da nach Voraussetzung $ab_0 = ab \in \dot{B}$ und $ac_0 = ac \in \dot{C}$ gelten. Sei nun $i \geq 1$. Nach Induktionsvoraussetzung existieren $ab_{i-1} \in \dot{B}$ und $ac_{i-1} \in \dot{C}$. Aus der Existenz von $ac_{i-1} \in \dot{C}$ (Induktionsvoraussetzung) und $b_i c_{i-1} \in \dot{A}$ (aus $\dot{\Gamma}$ -Kette) folgt nun wegen $\dot{A} \neq \dot{C}$ die Existenz der ungerichteten Kante $\widehat{ab_i} \in E$. Andererseits folgt aus der direkten Γ -Nachbarschaft von $b_i c_{i-1}$ und $b_{i-1} c_{i-1}$ in der $\dot{\Gamma}$ -Kette $\widehat{b_{i-1} c_{i-1}} \notin E$. Damit liegen also ab_i und $ab_{i-1} \in \dot{B}$ in Γ -Relation zueinander, und es gilt entweder $ab_i \in \dot{B}$ oder $ab_i \in F$. Gilt nun $\dot{B} \not\subseteq I$, dann kann ab_i nicht in $F \subseteq I$ liegen. Ebenso wenig für $a \notin V(F)$. Also folgt in beiden Fällen $ab_i \in \dot{B}$. Aus $ab_i \in \dot{B}$ folgt nun mit $b_i c_i \in \dot{A}$ ($\dot{\Gamma}$ -Kette) und $\dot{A} \neq \dot{B}^{-1}$ die Existenz der ungerichteten Kante $\widehat{ac_i} \in E$. Da nun aber die Knoten c_i und c_{i-1} nicht benachbart sind ($b_i c_i \dot{\Gamma} b_i c_{i-1}$), liegen ac_i und $ac_{i-1} \in \dot{C}$ in Γ -Relation zueinander. Hier folgt dann analog entweder $ac_i \in \dot{C}$ oder $ac_i \in F$. Der Fall $ac_i \in F \subseteq I$ kann jedoch für $\dot{C} \not\subseteq I$ oder $a \notin V(F)$ nicht eintreten. Damit folgt $ac_i \in \dot{C}$. $a \notin V(\dot{A})$ folgt unmittelbar aus der Anwendung des ersten Teils der Aussage. ■

Bei ANDRESEN [4] findet sich diese Aussage für den Fall, dass $F = \{\hat{e}\}$ aus einer einzigen Kante besteht. Dort wird auch gezeigt, dass in diesem Fall $\dot{I} + \hat{e}$ transitiv für jede $\dot{\Gamma}$ -Komponente \dot{I} ist.

Anwendung auf Planimplikationsklassen

Wenden wir Satz 4.47 auf die uns interessierende speziellen $\dot{\Gamma}$ -Komponenten an, gelangen wir zu den folgenden Aussagen.

Folgerung 4.48 (Anwendung auf Planimplikationsklassen #1) *Seien $A^*, B^*, C^* \subseteq I \in \mathcal{I}_{[S^{tr}]}$ verschiedene erweiterte Planimplikationsklassen eines Plans $S \in \mathcal{S}_{SIJ}$, mit $ab \in B^*$, $ac \in C^*$ und $bc \in A^*$. Sei durch $F = F_{A^*}^* \cup F_{B^*}^* \cup F_{C^*}^*$ eine trennende Kantenmenge gegeben, die A^* , B^* und C^* von allen anderen Planimplikationsklassen trennt. Gilt $a \notin V(F)$, dann folgt aus der Existenz von $b'c' \in A^*$ die Existenz von $ab' \in B^*$ und $ac' \in C^*$, sowie $a \notin V(A^*)$.*

Folgerung 4.49 (Anwendung auf Planimplikationsklassen #2) *Seien A und B verschiedene Planimplikationsklassen eines Plans $S \in \mathcal{S}_{SIJ}$, mit $ab \in B$ und $bc \in A$. Seien \dot{A} und \dot{B} die $\dot{\Gamma}$ -Komponenten von A und B bezüglich $F = F_A^* \cup F_B^*$, und sei $ac \in E_{diag}(S)$ mit $ac \notin \dot{A}$. Gilt $a \notin V(F)$ und $a \notin V(\dot{A})$, dann impliziert $b'c' \in \dot{A}$ die Existenz von $ab' \in \dot{B}$ und $ac' \in \dot{C}$ für eine $\dot{\Gamma}$ -Komponente $\dot{C} \neq \dot{A}$. Gilt zusätzlich auch $a \notin V(F_C^*)$, dann folgt $ac \in \dot{C}$.*

Folgerung 4.50 (Umkehrung) *Sei A eine Planimplikationsklasse eines Plans $S \in \mathcal{S}_{SIJ}$, und seien $ab \in \dot{B}$, $ac \in \dot{C}$ und $bc \in \dot{A}$ Kanten in $[S^{tr}]$, wobei \dot{A} , \dot{B} und \dot{C} $\dot{\Gamma}$ -Komponenten mit $\dot{A} \neq \dot{B}^{-1}$ und $\dot{A} \neq \dot{C}$ bezüglich F_A (F_A^*) sind. Existiert in $[S^{tr}]$ eine Kante $b'c' \in \dot{A}$ mit $ab' \notin \dot{B}$ oder $ac' \notin \dot{C}$, dann gilt $a \in V(F_A)$ ($a \in V(F_A^*)$), und $\dot{B} \subseteq I(A)$ oder $\dot{C} \subseteq I(A)$.*

Für die Umkehrung der Aussage des erweiterten Dreieckslemmas ist eine entscheidende Beobachtung, dass der Fall $B = C$ bzw. $\dot{B} = \dot{C}$ genau wie im ursprünglichen Dreieckslemma zulässig ist. Daher muss von der Trennungsmenge nicht gefordert werden, dass sie auch zwischen \dot{B} und \dot{C} trennt.

Anwendung auf Kanten der Trennungsmenge

Der Beweis von Satz 4.47 funktioniert prinzipiell sogar, wenn die zentrale Bezugskante des alten oder des neuen Dreiecks nicht in einer $\dot{\Gamma}$ -Komponente liegen, sondern zu der Trennungsmenge F gehört.

Lemma 4.51 ($b'c' \in F$) *Sei $G = (V, E)$ ein Graph, und seien \dot{A} , \dot{B} und \dot{C} $\dot{\Gamma}$ -Komponenten mit $\dot{A} \neq \dot{B}^{-1}$ und $\dot{A} \neq \dot{C}$ bezüglich einer Kantenmenge $F \subseteq E$ mit $F \cap I(\dot{A}) \neq \emptyset$ und $F \cap I(\dot{C}) = \emptyset$ ($I(\dot{A}), I(\dot{C}) \in \mathcal{I}_G$). Ferner seien $ab \in \dot{B}$, $ac \in \dot{C}$ und $bc \in \dot{A}$. Gilt $a \notin V(F)$, dann folgt aus der Existenz von $b'c' \in F$ die Existenz von $ab' \in \dot{B}$ und $ac' \in \dot{C}$, und keine Kante aus \dot{A} ist adjazent zu Knoten a .*

Beweis. Wegen $F \cap I(\dot{A}) \neq \emptyset$, existiert eine (kanonische) Γ -Kette zwischen bc und $b'c'$ in $\dot{A} + F$, $bc = b_0c_0\Gamma\widehat{b_1c_0}\Gamma b_1c_1\Gamma \dots \Gamma b_kc_k = b'c'$. Alternierend gilt nun folgendes: Aus jeder Γ -Relation $b_{i-1}c_{i-1}\Gamma b_i c_{i-1}$ folgt $\widehat{b_{i-1}b_i} \notin E$. Wegen $\dot{A} \neq \dot{C}$ (falls $b_i c_{i-1} \in \dot{A}$) bzw. $\dot{C} \cap I = \emptyset$ (falls $b_i c_{i-1} \in F$) und $ac_{i-1} \notin F$ folgt die Existenz von $\widehat{ab_i} \in E$. Mit $ab_{i-1} \in \dot{B}$ und mit $ab_{i-1} \notin F$ folgt $ab_i \in \dot{B}$ ($ab_i \Gamma \widehat{ab_{i-1}}$).

Andererseits impliziert jede Γ -Relation $b_i c_{i-1}\Gamma b_i c_i$ die Nichtexistenz von $\widehat{c_{i-1}c_i}$, $\widehat{c_{i-1}c_i} \notin E$. Wegen $ab_i \in \dot{B}$, $b_i c_i \in \dot{A} + F$ und $I(\dot{A}) \neq I(\dot{B}^{-1})$ (wegen $\dot{A} \neq \dot{B}^{-1}$) folgt die Existenz von $\widehat{ac_i} \in E$ mit $ac_i \in \dot{C}$ ($ac_i \Gamma \widehat{ac_{i-1}}$, und $ac_i \notin F$ nach Voraussetzung). Induktion nach i liefert $ab_i \in \dot{B}$ und $ac_i \in \dot{C}$ für alle $i = 0, \dots, k$. ■

Lemma 4.52 ($bc \in F$) *Sei $G = (V, E)$ ein Graph, und seien \dot{B} und \dot{C} $\dot{\Gamma}$ -Komponenten bezüglich einer Kantenmenge $F \subseteq E$. Ferner seien $ab \in \dot{B}$, $ac \in \dot{C}$ und $bc \in F^*$ mit $F^* = F \cap I^*$ für eine Implikationsklasse $I^* \in \mathcal{I}_G$ mit $I^* \neq I(\dot{B})^{-1}$. Gilt $\dot{C} \cap F^* = \emptyset$ und $a \notin V(F)$, dann folgt aus der Existenz von $b'c' \in \dot{A}$ für eine $\dot{\Gamma}$ -Komponente \dot{A} in I^* mit $\dot{A} \neq \dot{B}^{-1}$ und $\dot{A} \neq \dot{C}$ die Existenz von $ab' \in \dot{B}$ und $ac' \in \dot{C}$, und keine Kante aus \dot{A} ist adjazent zu Knoten a .*

Beweis. In $F^* + \dot{A}$ existiert eine (kanonische) Γ -Kette zwischen $bc \in F^* \subseteq I^*$ und $b'c' \in \dot{A} \subseteq I^*$, $bc = b_0c_0\Gamma b_1c_0\Gamma b_1c_1\Gamma \dots \Gamma b_kc_k = b'c'$. Alternierend gilt nun folgendes: Aus jeder Γ -Relation $b_{i-1}c_{i-1}\Gamma b_i c_{i-1}$

folgt $\widehat{b_{i-1}b_i} \notin E$. Wegen $\dot{C} \neq \dot{A}$ (falls $b_i c_{i-1} \in \dot{A}$) bzw. $\dot{C} \cap F^* = \emptyset$ (falls $b_i c_{i-1} \in F^*$) und $ac_{i-1} \notin F^*$ folgt die Existenz von $\widehat{ab_i} \in E$. Mit $ab_{i-1} \in \dot{B}$ und mit $ab_{i-1} \notin F$ folgt $ab_i \in \dot{B}$ ($ab_i \in \widehat{ab_i \dot{\Gamma} ab_{i-1}}$).

Andererseits impliziert jede Γ -Relation $b_i c_{i-1} \Gamma b_i c_i$ die Nichtexistenz von $\widehat{c_{i-1}c_i}, \widehat{c_{i-1}c_i} \notin E$. Wegen $I^* \neq I(\dot{B}^{-1})$ folgt aus $ab_i \in \dot{B}$ und $b_i c_i \in F^* + \dot{A} \subseteq I^*$ die Existenz von $\widehat{ac_i} \in E$ mit $ac_i \in \dot{C}$ ($ac_i \in \widehat{ac_i \dot{\Gamma} ac_{i-1}}$, und $ac_i \notin F$ nach Voraussetzung). Induktion nach i liefert $ab_i \in \dot{B}$ und $ac_i \in \dot{C}$ für alle $i = 0, \dots, k$. ■

Vor- und Nachteile von großen und kleinen Trennungsmengen

Der Anwendung der aus dem erweiterten Dreieckslemma (Satz 4.47) abgeleiteten Aussagen sind durch die jeweiligen Voraussetzungen leider enge Grenzen gesetzt. Es wird gefordert, dass entweder Teile des betrachteten Dreiecks zu verschiedenen Implikationsklassen gehören, oder, dass der Bezugsknoten a mit keiner Kante aus der trennenden Kantenmenge inzidiert. Wie wir noch sehen werden, wird die Frage nach der etwaigen Stabilität einer Schrägkante, typischerweise erst aufkommen, wenn der betrachtete H-Comparabilitygraph nur noch aus einer oder allenfalls sehr wenigen verschiedenen Implikationsklassen besteht. Für eine ausreichende Beschreibung von stabilen Schrägkanten kann die erste Forderung damit keine sinnvolle Voraussetzung sein. Somit bleibt die zweite; keine Kante aus der Trennungsmenge F darf mit a inzidieren.

Wird jedoch beispielsweise die Menge aller Schrägkanten als trennende Kantenmenge gewählt, $F = E_{diag}$, dann ist Satz 4.47 ausschließlich auf Dreiecke anwendbar, deren Bezugsknoten a mit *keiner* Schrägkante inzidiert. Damit können dann jedoch allenfalls Aussagen über Dreiecke gemacht werden, die vollständig aus regulären Kanten bestehen. Und solche Dreiecke liegen in jedem Fall in einer gemeinsamen Spalte bzw. Zeile. Für die Beschreibung von stabilen Schrägkanten ist auch diese Voraussetzung daher nicht hilfreich. Nebenbei führt die Anwendung von Satz 4.47 basierend auf dieser Voraussetzung nicht nur zu wenig hilfreichen Aussagen. Sie ist darüber hinaus auch nur selten überhaupt möglich, da selbst irreduzible Pläne sehr viele Schrägkanten induzieren können, so dass man Schwierigkeiten haben kann, überhaupt Knoten zu finden, die mit keiner Schrägkante inzidieren (Abschnitt 4.5).

Verkleinern wir dagegen die trennende Kantenmenge $F \subseteq E_{diag}$, so dass sie nur aus ein paar Schrägkanten, beispielsweise nur aus $F^*(A)$ für eine Planimplikationsklasse A , besteht, dann ist die Anzahl der Knoten, die nicht mit einer Kante aus F inzidieren, viel größer. Leider sind dann aber auch die resultierenden $\dot{\Gamma}$ -Komponenten weniger stark differenziert, so dass der Wert der Aussage, die vielleicht getroffen werden kann, deutlich niedriger ist. Bei der Wahl der Trennungsmenge $F \subseteq E_{diag}$ gilt es demnach abzuwägen zwischen einer starken Differenzierung der $\dot{\Gamma}$ -Komponenten durch eine große Trennungsmenge, verbunden mit einer hohen Aussagekraft, aber praktisch keiner Anwendbarkeit, und einer eher möglichen Anwendbarkeit durch eine kleine Trennungsmenge, verbunden mit einer sehr geringen Aussagekraft, bedingt durch eine sehr schwache Differenzierung der $\dot{\Gamma}$ -Komponenten.

Die beiden Grenzfälle sind $F = E_{diag}$ und $F = \emptyset$. Im ersten Fall sind die $\dot{\Gamma}$ -Komponenten gerade die Planimplikationsklassen, die als fundamentale Bausteine der reduzierenden Pläne hochgradig relevant sind. Leider ist in diesem Fall die Anwendbarkeit nur bei ausgewählten Beispielen gegeben. Im zweiten Fall sind die $\dot{\Gamma}$ -Komponenten gerade die Implikationsklassen. Das erweiterte Dreieckslemma wird auf den Spezialfall des ursprünglichen Dreieckslemmas (Satz 1.15) reduziert, und ist somit in jedem Graphen bedingungslos anwendbar. Leider können in diesem Fall keinerlei relevante Teilstrukturen der Implikationsklassen unterschieden werden. Damit können wir zwar immer-transitive Kanten beschreiben, aber wir können nicht zwischen stabilen Schrägkanten und immer-transitiven Schrägkanten, die nicht stabil sind, differenzieren.

4.3.3 Nichttrivial-stabile Schrägkanten

Immer-transitive Kanten sind Kanten, die in jeder zulässigen Kombination von Implikationsklassen transitiv sind. Stabile Schrägkanten sind Schrägkanten, die in jeder zulässigen Kombination von Planimplikationsklassen transitiv sind. Immer-transitive Kanten können durch eine endliche Menge von Konstellationen charakterisiert werden, die lokale Eigenschaften beschreiben. Sie sind entweder transitiv innerhalb ihrer Implikationsklasse, oder sie gehören zu einem vollständigen Teilgraphen auf vier

Knoten, der Kanten aus genau zwei verschiedenen Implikationsklassen enthält (Satz 1.44 und Abbildung 1.8 auf Seite 28). Können wir stabile Schrägkanten auf analoge Weise beschreiben?

Konstellationen mit zwei beteiligten Planimplikationsklassen

Beispiele für stabile Schrägkanten, deren Transitivität durch eine einzige Planimplikationsklasse induziert wird, haben wir bereits häufig gesehen. Wir haben solche Kanten als trivial-stabil bezeichnet. Ob wir hierbei von Planimplikationsklassen oder von erweiterten Planimplikationsklassen sprechen, ist von untergeordneter Bedeutung. Es sei jedoch noch einmal erwähnt, dass wir für gewöhnlich der Einfachheit halber “Planimplikationsklassen” schreiben, aber “erweiterte Planimplikationsklassen” meinen. Vor diesem Hintergrund werden wir trivial-stabile Schrägkanten, ebenfalls der Einfachheit halber, auch als transitiv *innerhalb* ihrer Planimplikationsklasse beschreiben.

In der Tat sind die stabilen Schrägkanten in *sämtlichen* bisher betrachteten Beispielen trivial-stabil gewesen. Es gibt jedoch durchaus auch Pläne, die nichttrivial-stabile Schrägkanten enthalten.

Definition 4.53 (nichttrivial-stabile Schrägkante) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan, und sei $[S^{tr}]$ sein H-Comparabilitygraph. Eine stabile Schrägkante, die nicht in der transitiven Hülle einer erweiterten Planimplikationsklasse liegt, heißt nichttrivial-stabil.

Aus der Konstellation mit zwei Implikationsklassen in Abbildung 1.8 lässt sich leicht ein Beispiel mit zwei Planimplikationsklassen erzeugen, das eine Schrägkante enthält, die in jeder Kombination dieser beiden Planimplikationsklassen transitiv ist.

$$B_{22} = \begin{bmatrix} 2 & & & & \\ 1 & 2 & 3 & & \\ & & 1 & 4 & 2 \\ & & & & 1 \\ & & & & & 1 \end{bmatrix} \quad B_{23} = \begin{bmatrix} 2 & & & & \\ 1 & 3 & 4 & & \\ & & 2 & 1 & 3 \\ & & & & 2 \\ & & & & & 2 \end{bmatrix}$$

Der H-Comparabilitygraph zum Plan B_{22} ist eindeutig transitiv orientierbar und enthält drei Schrägkanten. Die Menge der regulären Kanten besteht aus zwei verschiedenen Planimplikationsklassen P_1 und P_2 . Die vier inneren Knoten bilden einen K_4 , der eine trivial-stabile ($O_{32}O_{23}$) und eine nichttrivial-stabile Schrägkante ($O_{22}O_{33}$) induziert—analog zu der Konstellation in Abbildung 1.8. Wird eine der beiden Planimplikationsklassen umgekehrt, dann entfällt die dritte Schrägkante, und die beiden Planimplikationsklassen P_1 und P_2^{-1} verschmelzen miteinander. Der resultierende Plan (B_{23} oder B_{23}^{-1}) ist irreduzibel. Durch das Verschmelzen von P_1 mit P_2^{-1} durch das Entfernen der dritten Schrägkante $O_{21}O_{33}$ wird auch die vorher nichttrivial-stabile Schrägkante trivial-stabil.

Lemma 4.54 (nichttrivial-stabile Schrägkante) Sei $e = ab \in E_{diag}(S)$ eine nichttrivial-stabile Schrägkante in einem Plan $S \in \mathcal{S}_{SIJ}$. Dann existieren in G_{SIJ} mindestens zwei verschiedene minimale reguläre Wege von a nach b , die durch verschiedene Kombinationen von jeweils mindestens zwei verschiedenen erweiterten Planimplikationsklassen realisiert werden können.

Beweis. Existiert in G_{SIJ} nur ein Weg von a nach b , dann ist e entweder nicht stabil, oder trivial-stabil. Existieren mehrere verschiedene Wege, aber insgesamt nur eine Kombination der beteiligten Planimplikationsklassen, die simultan jeden dieser Wege realisiert, dann ist e ebenso entweder nicht stabil (mindestens eine Rekombination der beteiligten Planimplikationsklasse ist zulässig) oder trivial-stabil (keine Rekombination ist zulässig, d.h. beteiligte Planimplikationsklassen liegen in einer gemeinsamen erweiterten Planimplikationsklasse). Existiert ein regulärer Weg mit Kanten aus nur einer erweiterten Planimplikationsklasse, dann ist e trivial-stabil. ■

Konstellationen mit drei oder mehr beteiligten Planimplikationsklassen

Im Zusammenhang mit Implikationsklassen genügt es, nach Konstellationen mit genau einer oder genau zwei verschiedenen beteiligten Äquivalenzklassen zu suchen. Sind mehr als zwei beteiligt, dann erfüllt eine Teilkonstellation dennoch die Bedingungen aus Satz 1.44 bzw. Abbildung 1.8. Der Grund hierfür ist die Gültigkeit des Dreieckslemmas und die daraus abgeleiteten strukturellen Eigenschaften von Implikationsklassen (Struktursatz von Golumbic, Satz 1.21).

Für Planimplikationsklassen gelten diese Eigenschaften nicht. Hier ist es denkbar, dass eine Schrägkante unter Beteiligung von drei oder mehr verschiedenen Planimplikationsklassen stabil ist, ohne dass diese Konstellation auf zwei Planimplikationsklassen reduzierbar wäre. Ist die Schrägkante $e = ab \in E_{diag}(S)$ zu einem Plan $S \in \mathcal{S}_{SIJ}$, beispielsweise in S transitiv durch die Planimplikationsklassen $P_1, P_2, P_3 \in \mathcal{P}_S$, d.h. $e \in (P_1 + P_2 + P_3)^{tr}$. Dann kann es theoretisch in jeder Rekombination dieser drei Planimplikationsklassen einen regulären Weg von a nach b geben, der jeweils nur Kanten aus $\hat{P}_1 + \hat{P}_2 + \hat{P}_3$ enthält. Eine mögliche Konstellation für solche Wege ist in Abbildung 4.9 dargestellt.

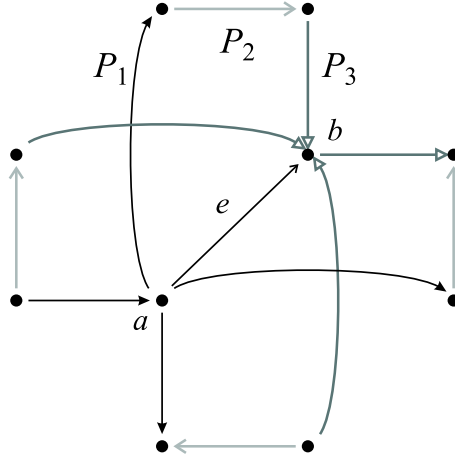


Abbildung 4.9: Eine denkbare Konstellation von regulären Kanten aus drei verschiedenen Planimplikationsklassen, die die Schrägkante e in jeder Rekombination als transitive Schrägkante enthalten.

Man beachte, dass in diesem Beispiel jeder Weg von a nach b in einer Rekombination der Planimplikationsklassen zuerst eine Kante aus P_1 enthält, dann eine Kante aus P_2 , und schließlich eine Kante aus P_3 . Da für Planimplikationsklassen jedoch kein so starkes Theorem wie für Implikationsklassen gilt, das viele zunächst denkbare Konstellationen sofort ausschließen kann, ist die hier getroffene Festlegung der Reihenfolgen absolut willkürlich, und nur eine von vielen denkbaren. Da alle drei Planimplikationsklassen in einer gemeinsamen Implikationsklasse liegen dürfen, und jeder der Knoten zu fast allen anderen adjazent ist, kann das erweiterte Dreieckslemma in keiner Formulierung Anwendung finden. Der Anzahl der verschiedenen denkbaren Konstellationen sind daher kaum Grenzen gesetzt.

Darüber hinaus gibt es keinen Grund, warum jeder der regulären Wege, die e transitiv in einer Rekombination der Planimplikationsklassen machen, nur aus genau drei Kanten bestehen soll. Solche Wege können theoretisch beliebig lang sein—begrenzt nur durch die Endlichkeit des Graphen. Ferner ist es für die Stabilität von e nicht einmal notwendig, dass in *jeder* beliebigen Kombination der drei Planimplikationsklassen ein Weg von a nach b oder von b nach a existiert. Es genügt, dass ein solcher in jeder kreisfreien Kombination existiert. Noch genauer: es genügt sogar, dass ein solcher Weg in jeder kreisfreien Kombination existiert, die gegenüber S keine neuen Schrägkanten induziert.

Die Anzahl der denkbaren Konstellationen, die eine Schrägkante unter Beteiligung von genau drei Planimplikationsklassen zur stabilen Schrägkante machen, ist damit praktisch nicht begrenzt. Zu jeder dieser denkbaren Konstellationen ist die Existenz eines H-Comparabilitygraphen denkbar, der sie enthält.

Darüber hinaus gibt es keinen Grund für die Annahme, dass jede Konstellation, die eine stabile Schrägkante induziert, nur drei Planimplikationsklassen oder weniger beteiligt. Auch für die Anzahl der relevanten Planimplikationsklassen gibt es keine obere Schranke.

Bemerkung 4.55 (Anzahl Konstellationen) *Im Fall, dass die trennende Kantenmenge leer ist, $F = \emptyset$, ist die Anzahl der verschiedenen Konstellationen, die eine gegebene Kante in jeder zulässigen Rekombination der $\tilde{\Gamma}$ -Komponenten (Implikationsklassen) transitiv sein lassen, begrenzt (Satz 1.44). Es sind entweder nur eine oder genau zwei verschiedene $\tilde{\Gamma}$ -Komponenten beteiligt, und es ergibt sich jeweils genau eine mögliche Konstellation (Abbildung 1.8). Ist die trennende Kantenmenge gleich der Schrägkantenmenge eines Plans $S \in \mathcal{S}_{SIJ}$, $F = E_{diag}(S)$, dann ist die Anzahl der denkbaren*

Konstellationen unbegrenzt—sowohl im Hinblick auf die Anzahl der beteiligten $\dot{\Gamma}$ -Komponenten (Planimplikationsklassen), als auch im Hinblick auf die jeweilige Ausgestaltung der Konstellation bei einer festen Zahl von beteiligten $\dot{\Gamma}$ -Komponenten.

Bemerkung 4.56 (keine Beispiele bekannt) *Trotz der überwältigenden Anzahl an denkbaren Möglichkeiten für Konstellationen, die stabile Schrägkanten induzieren, ist kein einziges Beispiel für einen Plan bekannt, der eine stabile Schrägkante enthält, für deren Stabilität drei oder mehr verschiedene Planimplikationsklassen notwendig sind.*

Konstruktion von Beispielen

Die Tatsache, dass wir uns eine denkbare Konstellation wie in Abbildung 4.9 überlegen können, bedeutet nicht, dass es auch einen Plan gibt, der diese Konstellation enthält. Im Falle von Implikationsklassen können wir eine beliebig komplizierte Konstellation entweder elementar auf eine der beiden in Satz 1.44 beschriebenen reduzieren, oder wir können mit Hilfe des Dreieckslemmas die Nichtexistenz eines Comparabilitygraphen mit einer solchen Konstellation beweisen. Dieser Schritt funktioniert für Planimplikationsklassen nicht. Es gibt keine Möglichkeit, von den lokalen Eigenschaften in einer gegebenen Konstellation auf die lokalen Eigenschaften an einer anderen Stelle zu schließen. Insbesondere kann die Nichtexistenz eines H-Comparabilitygraphen mit dieser Konstellation auf diese Weise nicht bewiesen werden.

Bemerkung 4.57 (Konstruktion von Beispielen) *Versucht man, ausgehend von einer gegebenen Konstellation (z.B. Abbildung 4.9), eine Planorientierung in einem H-Comparabilitygraphen zu konstruieren, dann muss man die folgenden Eigenschaften sicherstellen.*

- (i) *Alle Kanten, die zu einer gemeinsamen Planimplikationsklasse gehören sollen, müssen durch Γ -Ketten mit regulären Kanten (Γ_S -Ketten) untereinander verbunden sein.*
- (ii) *Jede Zeile und jede Spalte muss einen vollständigen Graphen bilden. Jede reguläre Kante gehört zu einer Planimplikationsklasse.*
- (iii) *Keine der Planimplikationsklassen darf in der Erweiterung einer anderen relevanten Planimplikationsklasse liegen.*
- (iv) *Es darf keine Γ -Relation zwischen zwei regulären Kanten aus verschiedenen Planimplikationsklassen geben.*
- (v) *Jede Schrägkante in der Ausgangsorientierung muss transitiv sein. Insbesondere auch solche Schrägkanten, die eingefügt werden, um eine Γ -Relation zwischen Kanten aus verschiedenen Planimplikationsklassen zu verhindern.*
- (vi) *Es dürfen keine neuen Wege entstehen, die die gegebene Konstellation “außer Kraft” setzen.*
- (vii) *Die Ausgangsorientierung der Konstellation muss kreisfrei sein.*

Man beachte, dass die verschiedenen Anforderungen bei einer solchen Konstruktion untereinander in Beziehung stehen. Die Lösung eines Problems bringt häufig neue Probleme mit sich. Das Hinzufügen von Knoten geht einher mit neuen regulären Kanten in den betreffenden Zeilen und Spalten. Deren Orientierung induziert unter Umständen Schrägkanten, die bestehende Γ_S -Relationen zerstören. Oder sie verknüpft verschiedene Planimplikationsklassen miteinander, die nicht verknüpft werden sollen, woraufhin neue Schrägkanten eingefügt werden müssen, die unerwünschte Γ_S -Relationen wieder zerstören. Diese neuen Schrägkanten müssen jedoch transitiv sein, d.h. es muss einen regulären Weg von ihrem Anfangsknoten zu ihrem Endknoten geben. Dieser reguläre Weg ist möglicherweise nur durch das Einfügen weiterer Knoten zu realisieren, usw..

Ein Ergebnis einer solchen Konstruktion ist beispielsweise der Plan B_{22} auf Seite 91. Ausgangspunkt hierfür war die Konstellation der inneren vier Knoten. Jeder zusätzliche Knoten und jede zusätzliche Kante erfüllt mindestens eine der genannten Aufgaben. Wir werden noch einige solcher konstruierter Beispiele betrachten. Ihnen gemeinsam ist, dass sie für gewöhnlich sehr unübersichtlich

und kompliziert wirken. Der Grund hierfür ist, dass die Anzahl der Knoten in einer Zeile oder Spalte so gering wie möglich gehalten ist. Tatsächlich sind diese Beispiele somit bedeutend kleiner und übersichtlicher als minimale Beispiele mit stark besetzter Operationenmenge.

Geht man bei der Frage nach der Existenz von Plänen mit bestimmten Konstellationen über die Frage der prinzipiellen Existenz hinaus, und möchte man beispielsweise Pläne über Operationenmengen mit bestimmten Eigenschaften, wie beispielsweise der Vollständigkeit, konstruieren, wird die Aufgabe ungleich schwieriger.

4.3.4 Zwei relevante Planimplikationsklassen

Im Fall von nur zwei für die Stabilität einer Schrägkante verantwortlichen Planimplikationsklassen können wir eine Eigenschaft ableiten, die eine Erkennung der Stabilität ermöglichen kann.

Satz 4.58 (nicht reduzierbare Konstellation) *Seien $P_1, P_2 \in \mathcal{P}_S$ zwei verschiedene (erweiterte) Planimplikationsklassen aus einer gemeinsamen Implikationsklasse $P_1, P_2 \subseteq I \in \mathcal{I}_{[S^{tr}]}$ zu einem Plan $S \in \mathcal{S}_{SIJ}$. Existiert eine Schrägkante $e \in E_{diag}(S)$, die sowohl mit P_1 als auch zu P_2 direkt Γ -verbunden ist, und bezüglich jeder Kombination von P_1 und P_2 transitiv ist, dann gilt für jeden reduzierenden Plan S^* von S , $S^* \preceq S$, entweder $P_1 + P_2 \subseteq S^*$ oder $P_1^{-1} + P_2^{-1} \subseteq S^*$.*

Beweis. Sei $e \in F_{1,2} = F(P_1, P_2) \subseteq I \cap E_{diag}(S)$ immer-transitiv bzgl. P_1 und P_2 , d.h. $e \in (P_1 + P_2)^{tr}$ und $e \in (P_1 + P_2^{-1})^{tr}$, bzw. $e^{-1} \in (P_1 + P_2^{-1})^{tr}$. Seien $e_1 \in P_1$ und $e_2 \in P_2$ die Kanten mit $e_1 \Gamma e \Gamma e_2$. Wir zeigen, dass jeder Plan $S^* \in \mathcal{S}_{SIJ}$ mit (o.B.d.A.) $P_1 + P_2^{-1} \subseteq S^*$ mindestens eine Schrägkante induziert, die in $[S^{tr}]$ nicht enthalten ist. Wegen $e_1, e_2^{-1} \in S^*$ und $e_1^{-1}, e_2 \notin S^*$ muss in $[(S^*)^{tr}]$ entweder $e_1 \Gamma e \not\Gamma e_2$, oder $e_1 \not\Gamma e^{-1} \Gamma e_2^{-1}$ gelten (abhängig davon, ob e oder e^{-1} in S_2^* liegt). Damit muss $[(S^*)^{tr}]$ jedoch entweder die Schrägkante enthalten, die $e \Gamma e_2$ verhindert, oder die, die $e_1 \Gamma e$ verhindert. Nach Voraussetzung liegt keine der beiden in $[S^{tr}]$. Mit Satz 4.43 folgt dann $S_2^* \not\subseteq S$. ■

Zerstörung von Γ -Ketten zwischen Planimplikationsklassen

Die Annahme, dass die Schrägkante e nicht nur in der trennenden Kantenmenge $F_{1,2}$ zwischen P_1 und P_2 liegt, sondern mit beiden Klassen in direkter Γ -Relation, ist dabei eine notwendige Voraussetzung für die Schlussfolgerung in Satz 4.58. Ist eine bezüglich P_1 und P_2 immer-transitive Schrägkante $e \in F_{1,2}$ nur mit einer der beiden Planimplikationsklassen direkt Γ -verbunden—oder sogar zu gar keiner, dann ist es durchaus denkbar, dass P_1 und P_2 dennoch gegeneinander verdreht werden können. Eine notwendige Bedingung hierfür ist, dass jede Γ -Kette zwischen P_1 und P_2 zerstört wird. Die Kette, die e enthält, kann in diesem Fall jedoch ohne weiteres zerstört werden, ohne dass die stabile Schrägkante e entfernt werden muss.

Ein Beispiel für einen Plan mit einer immer-transitiven Schrägkante aus der trennenden Kantenmenge zwischen zwei Planimplikationsklassen, die nicht direkt mit beiden Klassen Γ -verbunden ist, ist nicht bekannt.

Satz 4.59 (Zerstörung von Γ -Ketten) *Seien $P_1, P_2 \in \mathcal{P}_S$ zwei verschiedene (erweiterte) Planimplikationsklassen aus einer gemeinsamen Implikationsklasse $P_1, P_2 \subseteq I \in \mathcal{I}_{[S^{tr}]}$ zu einem Plan $S \in \mathcal{S}_{SIJ}$, und sei $F_{1,2} = F(P_1, P_2)$ die trennende Kantenmenge zwischen P_1 und P_2 in $[S^{tr}]$. Existiert ein Plan $S^* \in \mathcal{S}_{SIJ}$ mit $P_1 + P_2^{-1} \subseteq S^*$, der S reduziert, $S^* \preceq S$, dann gilt für jede Γ -Kette $K \subseteq F_{1,2}$ in $[S^{tr}]$ von einer Kante aus P_1 zu einer Kante aus P_2 , dass mindestens eine der Kanten aus \hat{K} nicht in $[(S^*)^{tr}]$ enthalten ist. Insbesondere reduziert S^* den Plan S streng, $S^* \prec S$.*

Beweis. Die Aussage folgt direkt aus den Überlegungen aus dem Beweis von Satz 4.58. Damit durch das Verdrehen von P_2 gegenüber P_1 keine neuen Schrägkanten entstehen, müssen notwendigerweise alle Γ -Ketten zwischen P_1 und P_2 in $[S^{tr}]$ zerstört werden. Das Zerstören einer Γ -Relation ist nur durch das Hinzufügen der Kante, die die Γ -Relation induziert, oder durch das Entfernen einer der beiden Kanten der Relation möglich. Da S^* nach Voraussetzung S reduziert, kann keine Kante hinzugefügt worden sein (Satz 4.2). Also muss aus jeder Γ -Kette mindestens eine Kante entfernt worden sein. ■

Allerdings enthält $G(B_{25})$ nach wie vor drei Kanten aus der trennenden Kantenmenge zwischen P_1 und P_2 in B_{24} . Diese Kanten liegen nach wie vor in direkter Γ -Relation sowohl zu P_1 , als auch zu P_2 . Jede dieser Kanten induziert damit in B_{25} eine Schrägkante, die nicht in $G(B_{24})$ enthalten ist. In diesem Beispiel induzieren alle drei Kanten die gleiche neue Schrägkante 5.11.

Der Plan $B_{25} = P_1 + P_2^{-1}$ reduziert damit nicht den Plan $B_{24} = P_1 + P_2$ (Satz 4.43). Da es nach Satz 4.40 jedoch keinen weiteren Kandidaten für einen nichtähnlichen reduzierenden Plan von B_{24} gibt, folgt, dass B_{24} irreduzibel ist. Insbesondere folgt damit auch, dass alle Schrägkanten in $G(B_{24})$ stabil sind. D.h. der Plan B_{24} ist ein irreduzibler Plan, bei dem 4 von 9 Schrägkanten nichttrivial-stabil sind.

Irreduzible Pläne mit beliebig vielen Planimplikationsklassen

Aus dem Plan B_{24} lässt sich ganz leicht ein irreduzibler Plan mit beliebig vielen verschiedenen Planimplikationsklassen konstruieren. Erstellt man von B_{24} eine Kopie und fügt man diese an den Plan B_{24} an, indem man den Knoten 1 der Kopie mit dem Knoten 12 des Originals identifiziert, erhält man einen Plan, der aus nunmehr 3 verschiedenen Planimplikationsklassen besteht. Die Planimplikationsklasse P_1 der Kopie verschmilzt mit P_1 des Originals. Und sowohl die Kopie, als auch das Original enthalten jeweils eine weitere Planimplikationsklasse. Da sämtliche Eigenschaften von B_{24} auch in der Kopie auftreten, ist der so erzeugte Plan genau wie B_{24} irreduzibel.

An diesen Plan kann man nun eine weitere Kopie von B_{24} anfügen. Dadurch entsteht ein Plan, der ebenfalls irreduzibel ist, und aus 4 verschiedenen Planimplikationsklassen in insgesamt 3 Kopien von B_{24} besteht. Da die Eigenschaften der einzelnen Teilpläne unabhängig voneinander sind, lässt sich auf diese Art und Weise ein irreduzibler Plan mit beliebig großer Anzahl von Planimplikationsklassen konstruieren.

Bemerkung 4.61 (Anzahl Planimplikationsklassen in irreduziblen Plänen) Die Anzahl der Planimplikationsklassen $|\mathcal{P}_S|$ eines Plans $S \in \mathcal{S}_{SIJ}$ zu einer gegebenen Operationenmenge SIJ ist beschränkt durch die Anzahl der regulären Kanten über SIJ . Andererseits gibt es zu jeder natürlichen Zahl k einen irreduziblen Plan $S^* \in \mathcal{S}_{SIJ^*}$ über einer Operationenmenge SIJ^* , mit $|\mathcal{P}_{S^*}| = k$.

4.4 Zur Komplexität verwandter Probleme

Nachdem wir uns in diesem Kapitel bisher mit den Eigenschaften und der Bedeutung der Planimplikationsklassen für das Problem IRRED befasst haben, wollen wir in diesem Abschnitt die Komplexität einiger zu IRRED verwandter Probleme näher betrachten.

Eine der Problemformulierungen für IRRED fragt nach der Existenz eines H-Comparabilitygraphen G mit $G_{SIJ} \subseteq G \subsetneq [S^{tr}]$ (Seite 68). Von besonderer Bedeutung in diesem Zusammenhang ist die Betrachtung des sogenannten SEQUENCE-SUBGRAPH Problems in Abschnitt 4.4.3.

4.4.1 GRAPH SANDWICH, SUBGRAPH und COMP-DEL

In Abschnitt 2.3 haben wir die NP-vollständigen Probleme SUBGRAPH und COMP-ED (Comparability-Editing) kennen gelernt. Bei SUBGRAPH lautet die Frage, ob von zwei gegebenen Graphen G_1 und G_2 einer isomorph zu einem Teilgraphen des anderen ist. Bei COMP-ED lautet die Frage, ob es zu einem gegebenem Graphen $G = (V, E)$ eine Teilmenge $F \subseteq V \times V$ mit $|F| \leq k$ gibt, so dass die symmetrische Differenz $E \Delta F$ eine transitiv orientierbare Kantenmenge über V ist. Als Variante können wir hier auch $F \subseteq E$ fordern. Dann lautet die Frage: Existiert eine Kantenmenge mit höchstens k Kanten, deren Entfernung aus G zu einem Comparabilitygraphen führt (Comparability-Deletion, COMP-DEL)? Genau wie COMP-ED ist auch COMP-DEL NP-vollständig (YANNAKAKIS (1981) [68]).

Sowohl das Problem SUBGRAPH, als auch das Problem COMP-DEL haben große Gemeinsamkeiten mit IRRED. Bei IRRED suchen wir einen transitiv orientierbaren Teilgraphen eines gegebenen Graphen, der durch das Entfernen von Kanten entstanden ist. Dennoch hat die Komplexität keines dieser beiden Probleme Auswirkungen auf die Komplexität von IRRED.

SUBGRAPH und IRRED

Bei SUBGRAPH sind die beiden Graphen G_1 und G_2 völlig unabhängig voneinander gegeben. Die Isomorphie zweier Graphen bedeutet die Existenz einer Abbildung der Knoten des einen Graphen auf die Knotenmenge des anderen Graphen, so dass die Kanten des einen in die des anderen überführt werden. Bei IRRED ist die Knotenmenge durch die Operationenmenge SIJ fest beschrieben. Isomorphie spielt keine Rolle. Von jedem Graphen G , der als Teilgraph von $[S^{tr}]$ in Frage kommt, kann schnell entschieden werden, ob er tatsächlich ein Teilgraph ist, oder nicht. Als Abbildung zwischen den Knotenmengen von G und $[S^{tr}]$ kommt ausschließlich die Identität in Frage.

Bemerkung 4.62 (SUBGRAPH) *Im Gegensatz zu SUBGRAPH sind bei IRRED alle möglichen Teilgraphen über einer gegebenen und unveränderlichen Knotenmenge beschrieben. Für einen reduzierenden Plan S^* eines Ausgangsplans S stellt sich nicht die Frage nach Isomorphie von $[(S^*)^{tr}]$ zu einem Teilgraphen von $[S^{tr}]$, sondern lediglich die Frage nach der Identität. Die Komplexität von SUBGRAPH hat somit keine Konsequenzen für die Komplexität von IRRED.*

COMPARABILITY-DELETION und IRRED

Die Probleme IRRED und COMP-DEL liegen deutlich näher zusammen. Der Unterschied zwischen beiden ist der folgende. Während bei beiden nach der Existenz einer Entfernungsmenge $F \subseteq E$ gefragt ist, die zu einem Comparabilitygraphen führt, ist der Ausgangsgraph bei COMP-DEL ein beliebiger Graph, wohingegen der Ausgangsgraph bei IRRED ein Comparabilitygraph ist. IRRED ist also ein Spezialfall von COMP-DEL, und könnte daher sehr wohl einfacher zu lösen sein. Auf der anderen Seite ist die Menge der entfernbaren Kanten bei COMP-DEL nicht eingeschränkt. Jede beliebige Kante könnte entfernt werden. Bei IRRED kommt nur die Entfernung von bestimmten Kanten in Frage—den Schrägkanten. Die Auswirkung, die diese Einschränkung auf die Komplexität von IRRED hat, ist unklar. Sie könnte das Problem IRRED gegenüber COMP-DEL mit einem Comparabilitygraphen als Ausgangsgraphen sowohl leichter, als auch schwerer machen.

Bemerkung 4.63 (COMP-DEL) *Im Gegensatz zu COMP-DEL ist der Ausgangsgraph bei IRRED ein Comparabilitygraph. Darüber hinaus ist die Kantenmenge, die entfernt werden darf, eine echte Teilmenge aller Kanten. Die Probleme COMP-DEL und IRRED sind somit komplexitätstheoretisch nicht vergleichbar.*

COMPARABILITY GRAPH SANDWICH und IRRED

Neben den Problemen SUBGRAPH und COMP-DEL ist auch das sogenannte COMPARABILITY GRAPH SANDWICH Problem eng mit IRRED verwandt.

Problem COMPARABILITY GRAPH SANDWICH (COMP-GS): Gegeben seien zwei Graphen $G_1 = (V_1, E_1)$ und $G_2 = (V_2, E_2)$ mit $V_1 \subseteq V_2$ und $E_1 \subseteq E_2$. Existiert ein Comparabilitygraph G mit $G_1 \subseteq G \subseteq G_2$?

Genau wie die beiden bereits betrachteten Probleme ist auch das COMP-GS Problem NP-vollständig (GOLUMBIC, KAPLAN UND SHAMIR (1995) [35]). Und genau wie im Falle von SUBGRAPH und COMP-DEL ist auch hier die Nähe zu IRRED nicht so groß, wie es zunächst scheint.

Während das COMP-GS Problem für beliebige Ausgangsgraphen G_1 und G_2 also schwierig ist, ist es für die spezielle Wahl $G_1 = G_{SIJ}$ und $G_2 = [S^{tr}]$ trivial lösbar. Der H-Comparabilitygraph $[S^{tr}]$ zu dem Ausgangsplan $S \in \mathcal{S}_{SIJ}$ ist bereits ein Comparabilitygraph. Bei IRRED fragen wir nach der (Nicht-) Existenz eines echten Teilgraphen von G_2 .

IRRED ist äquivalent zu der Frage nach der Nichtexistenz eines Comparability-Sandwich-Graphen zwischen $G_1 = G_{SIJ}$ und jedem der Graphen $G_2 = [S^{tr}] - \hat{e}$ für ein $\hat{e} \in E_{diag}(S)$. Dieses spezielle Menge von COMP-GS Problemen kann bei Ausnutzung der Eigenschaften des H-Comparabilitygraphen $[S^{tr}]$ jedoch durchaus einfacher zu lösen sein, als das allgemeine COMP-GS Problem.

Bemerkung 4.64 (COMP-GS) *Im Gegensatz zu COMP-GS ist der Ausgangsgraph bei IRRED bereits ein Comparabilitygraph, und es wird nach einem echten Teilgraphen gesucht. Die Probleme COMP-GS und IRRED sind somit komplexitätstheoretisch nicht vergleichbar.*

4.4.2 Spezielle Operationenmengen

Eine andere Gruppe verwandter Probleme sind Spezialfälle von IRRED, die deutlich einfacher zu lösen sind, als das allgemeine Problem.

Zwei Maschinen

Ein solcher Spezialfall ist, dass die Operationenmenge nur zwei Maschinen oder, mit Symmetrie, nur zwei Aufträge enthält. In diesem Fall ist nicht nur das Schedulingproblem polynomiell lösbar ($O2 \parallel C_{\max} \in P$, Abschnitt 3.4). Auch das Problem IRRED kann polynomiell gelöst werden. BRÄSEL UND KLEINAU (1996) [16] haben gezeigt, dass jeder irreduzible Plan $S \in \mathcal{S}_{SIJ}$ über der vollständigen Operationenmenge SIJ mit (o.B.d.A.) zwei Aufträgen und m Maschinen durch Vertauschen von Zeilen und Spalten in die folgenden Form gebracht werden kann,

$$S = \begin{bmatrix} 1 & 2 & \dots & k-1 & k & \dots & m \\ m-k+2 & m-k+3 & \dots & m & 1 & \dots & m-k+1 \end{bmatrix}.$$

Ob ein gegebener Plan $S \in \mathcal{S}_{SIJ}$ dieser Form entspricht, kann in polynomieller Laufzeit überprüft werden (BRÄSEL UND KLEINAU [16]).

Tatsächlich lässt sich leicht zeigen, dass jeder Plan, der diese Form erfüllt, nur aus einer Planimplikationsklasse besteht. Damit ist ein Plan über einer solchen Operationenmenge genau dann irreduzibel, wenn er nur aus einer Planimplikationsklasse besteht. Auch auf diesem Wege ist eine Erkennung der irreduziblen Pläne in polynomieller Zeit möglich.

Ist die Operationenmenge unvollständig, dann existieren zusätzliche Operationen in den beiden Zeilen, die jeweils die einzige Operation in ihrer Spalte sind. Eine reguläre Kante, die eine dieser Operationen mit der entsprechenden Operation aus einer Spalte mit zwei Operationen verbindet, kann immer und ohne Einschränkung so orientiert werden, dass sie in der gleichen Planimplikationsklasse liegt, wie die senkrechte Kante der betreffenden Spalte. D.h. die angegebene Beschreibung der irreduziblen Pläne für vollständige Operationenmenge des Formats $2 \times m$ ist auf kanonische Weise auf unvollständige Mengen erweiterbar.

Satz 4.65 (2 Maschinen) [16] *Für Operationenmengen mit höchstens zwei Maschinen (oder höchstens zwei Aufträgen) liegt IRRED in P.*

Operationenmenge tree-like

Das Problem IRRED wird ebenfalls einfach, wenn man sich auf sogenannte *baumartige* Operationenmengen beschränkt. Jede Operationenmenge $SIJ \subseteq I \times J$ lässt sich durch den folgenden bipartiten Graphen vollständig beschreiben: Zwischen den Knoten $i \in I$ und $j \in J$ existiert genau dann eine Kante, wenn die Operation O_{ij} zu SIJ gehört. Ist dieser bipartite Graph ein Baum, dann heißt die durch ihn beschriebene Operationenmenge $SIJ \subseteq I \times J$ *baumartig* oder *tree-like*.

Man kann sich leicht überlegen, dass eine Operationenmengen genau dann tree-like ist, wenn zwei beliebige Operationen in G_{SIJ} nur durch genau einen ungerichteten Weg verbunden sind. D.h. G_{SIJ} enthält keine Kreise, die nicht Kreise in einer Zeile oder Spalte sind.

TAUTENHAHN (2000) [64] konnte zeigen, dass für Operationenmengen SIJ mit dieser Eigenschaft jede Ecke in einem irreduziblen Plan $S \in \mathcal{S}_{SIJ}$ zwei polynomiell überprüfbare Bedingungen erfüllt.

Satz 4.66 (SIJ tree-like) [64] *Für Operationenmengen SIJ , die tree-like sind, liegt IRRED in P.*

Bemerkenswerterweise ist das Open-Shop Problem für diese Einschränkung der Operationenmenge, im Gegensatz zu der Einschränkung auf nur zwei Maschinen oder Aufträge, nach wie vor NP-vollständig (GONZALES UND SAHNI (1976) [36]).

Aus den bei TAUTENHAHN [64] angegebenen Bedingungen lässt sich schließen, dass irreduzible Pläne über baumartige Operationenmengen nur trivial-stabile Schrägkanten induzieren können. Da die Erweiterungen der Planimplikationsklassen mit polynomiellem Aufwand bestimmt werden können, ergibt sich hieraus eine alternative Begründung für die Zugehörigkeit von IRRED zu P (mit Problemformulierung #3 auf Seite 82), unabhängig von der polynomiellen Überprüfbarkeit der bei TAUTENHAHN angegebenen Bedingungen.

Die erste der beiden bei TAUTENHAHN [64] angegebenen Bedingungen lautet, dass jede *Ecke*—d.h. eine mittlere Operation auf einem minimalen regulären Weg der Kantenlänge 2 (induziert eine Eckkante)—in S entweder in einem Teilplan isomorph zu Y , oder einen Teilplan isomorph zu Y^{-1} enthalten ist.

$$Y = \begin{bmatrix} \mathbf{2} & \mathbf{3} & 1 \\ \mathbf{1} & & \\ & 1 & \\ & & 2 \end{bmatrix} \quad Y^{-1} = \begin{bmatrix} \mathbf{2} & \mathbf{1} & \mathbf{3} \\ \mathbf{3} & & \\ & 2 & \\ & & 1 \end{bmatrix}$$

Der Teilplan Y enthält genau eine Schrägkante, die trivial-stabil in der einzigen Planimplikationsklasse ist. Während diese Bedingung für *Ecken* zwar notwendig, aber nicht hinreichend für die Irreduzibilität von S ist, wird sie hinreichend und, abgesehen von einer kleinen Ausnahme, auch notwendig, wenn sie für jede *Eckkante* gefordert wird. Der Unterschied zwischen diesen beiden Formulierungen besteht darin, dass eine Operation, die eine Ecke ist, gleichzeitig mehrere verschiedene Eckkanten induzieren kann. In dem Plan B_{26} ist die Eck-Operation O_{11} bezüglich der Eckkante $O_{21}O_{12}$ in einem Teilplan isomorph zu Y enthalten, nicht jedoch bezüglich der Eckkante $O_{21}O_{14}$.

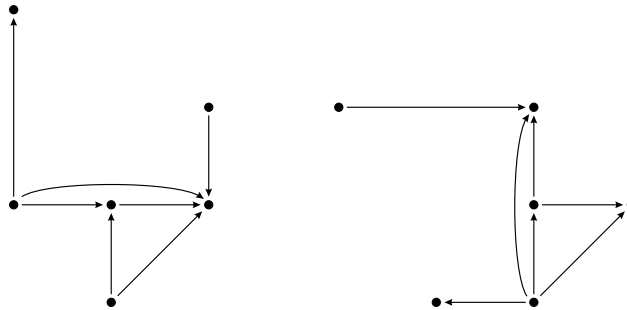


Abbildung 4.11: Die beiden einzigen Möglichkeiten bei baumartiger Operationenmenge für eine Γ_S -Kette zwischen zwei transitivierenden regulären Kanten einer Eckkante (isomorph zu Y bzw. Y^{-1}).

Lemma 4.67 (Eckkante) *Eine Eckkante in einem Plan $S \in \mathcal{S}_{SIJ}$ über einer Operationenmenge, die tree-like ist, liegt genau dann in der transitiven Hülle einer Planimplikationsklasse, wenn sie in einem Teilplan enthalten ist, der isomorph zu Y oder Y^{-1} ist.*

Beweis. Die Teilpläne Y und Y^{-1} beschreiben tatsächlich die einzigen beiden Möglichkeiten, wie unter den gegebenen Einschränkungen der Operationenmenge zwei transitivierende reguläre Kanten einer Eckkante in einer gemeinsamen Planimplikationsklassen liegen können. Ist die erste Kante der Ecke eine vertikale Kante, muss sie in Γ_S -Relation zu einer horizontalen Kante liegen. Diese kann entweder mit dem Anfangs- oder dem Endknoten der vertikalen ersten Kante inzidieren. In jedem der beiden Fälle sind die jeweils folgenden Γ_S -Relationen eindeutig bestimmt, da alle Alternativen entweder zu isomorphen Teilstrukturen führen, oder zu Γ_S -Ketten, die unter der Einschränkung der Operationenmenge nicht mehr die andere transitivierende reguläre Kante erreichen können. Von den beiden Möglichkeiten führt eine letztendlich zu Y und die andere zu Y^{-1} (Abbildung 4.11). ■

Bemerkung 4.68 (Konstellationen für Eckkante) *Für beliebige Operationenmengen ist die Menge der Konstellationen, die jeweils zu einer Γ_S -Verbindung zwischen zwei transitivierenden regulären Kanten zu einer Eckkante führen, nicht beschränkt.*

Neben Eckkanten, die transitiv innerhalb einer Planimplikationsklasse sind, kann es jedoch auch solche geben, die zwar transitiv in einer erweiterten Planimplikationsklasse sind, deren transitivierende Kanten jedoch zu verschiedenen Planimplikationsklassen gehören.

$$B_{26} = \begin{bmatrix} \mathbf{2} & \mathbf{4} & 1 & \mathbf{3} \\ \mathbf{1} & & & \\ & 1 & & \\ & & 2 & \end{bmatrix} \quad B_{27} = \begin{bmatrix} 2 & & & \\ 1 & \mathbf{2} & \mathbf{3} & \mathbf{4} \\ & \mathbf{1} & & \\ & & 4 & 1 \end{bmatrix}$$

Der Plan B_{27} enthält fünf Schrägkanten, von denen vier in der transitiven Hülle einer Planimplikationsklasse liegen, die alle regulären Kanten außer $O_{32}O_{34}$ enthält. Jede dieser vier Schrägkanten ist eine Eckkante, und in einem Teilplan isomorph zu Y oder Y^{-1} enthalten. Die Eckkante $O_{42}O_{34}$ liegt jedoch nicht in einem solchen Teilplan. Die beiden transitivierenden regulären Kanten gehören damit zu verschiedenen Planimplikationsklassen. Allerdings liegt die Planimplikationsklasse $O_{32}O_{34}$ in der transitiven Hülle der anderen. Die Eckkante $O_{42}O_{34}$ ist damit ebenso trivial-stabil.

Satz 4.69 (irreduzible Pläne für SIJ tree-like) *Ein Plan $S \in \mathcal{S}_{SIJ}$ über einer Operationenmenge, die tree-like ist, ist genau dann irreduzibel, wenn alle Schrägkanten in einer gemeinsamen erweiterten Planimplikationsklasse liegen.*

Beweis. Die Rückrichtung folgt aus Folgerung 4.38 (mit Bemerkung 4.46). Sei also S irreduzibel. Da zwei Knoten in G_{SIJ} nach Voraussetzung (SIJ tree-like) nur durch genau einen minimalen regulären ungerichteten Weg verbunden sind, kann S nach Lemma 4.54 keine nichttrivial-stabilen Schrägkanten enthalten. Folglich sind alle Schrägkanten trivial-stabil, d.h. sie liegen in der Erweiterung einer Planimplikationsklassen von S .

Angenommen, S enthält zwei verschiedene erweiterte Planimplikationsklassen, die jeweils Schrägkanten enthalten. Da wir SIJ o.B.d.A. als zusammenhängend voraussetzen können, existiert dann ein Knoten, der mit regulären Kanten aus beiden Klassen inzidiert. Ist von diesen beiden regulären Kanten eine horizontal und eine vertikal, dann folgt die Existenz einer Schrägkante, deren transitivierende Kanten nicht in einer gemeinsamen Planimplikationsklasse liegen (Widerspruch). Sind beide Kanten horizontal (oder beide vertikal), dann folgt die Existenz einer solchen Schrägkante ebenso. Jede der beiden Kanten steht in Γ_S -Relation zu einer vertikalen (bzw. horizontalen) Kante, von denen eine eine solche Schrägkante induziert. ■

4.4.3 Das SEQUENCE-SUBGRAPH Problem

Ein spezielles COMP-GS Problem ist das von WILLENUS (2000) [66] betrachtete SEQUENCE-SUBGRAPH Problem.

Problem SEQUENCE-SUBGRAPH (SEQ-SG): Gegeben sei eine Operationenmenge SIJ eines Open-Shop Problems, und ein Graph G^* mit $G_{SIJ} \subseteq G^*$ und $V(G^*) = SIJ$. Existiert ein Plan $S \in \mathcal{S}_{SIJ}$ mit $[S^{tr}] \subseteq G^*$?

Der Ausgangsgraph G^* besteht somit aus dem H-Graphen G_{SIJ} und ein beliebiger Menge von Schrägkanten. Da ein Plan $S \in \mathcal{S}_{SIJ}$ mit $[S^{tr}] \subseteq G^*$ genau dann existiert, wenn ein Comparabilitygraph G mit $G_{SIJ} \subseteq G \subseteq G^*$ existiert, ist der einzige Unterschied zwischen SEQ-SG und COMP-GS, dass die begrenzenden Graphen bei SEQ-SG einer fest gewählten Struktur entsprechen. WILLENUS [66] konnte zeigen, dass auch dieser Spezialfall das COMP-GS Problem nicht einfacher macht.

Satz 4.70 (SEQ-SG) [66] *Das Problem SEQ-SG liegt in NPC.*

Genau wie im Fall des allgemeinen COMP-GS Problems hat jedoch auch dieses Resultat keine direkten Auswirkungen auf die Schwierigkeit von IRRED, da der Ausgangsgraph G^* dort bereits ein Comparabilitygraph ist, und nach einem echten Sandwich-Graphen $G \subsetneq G^*$ gesucht wird.

Bemerkung 4.71 (SEQ-SG) *SEQ-SG ist ein Spezialfall von COMP-GS. Daher sind auch SEQ-SG und IRRED komplexitätstheoretisch nicht vergleichbar.*

Bemerkenswert ist dieses Resultat vor allem aus einem anderen Grund. Genau wie bei COMP-GS erfolgt der Nachweis der NP-Vollständigkeit von SEQ-SG durch die Reduktion von NOT-ALL-EQUAL-3-SATISFIABILITY (NAE-3-SAT). Im Reduktionsschritt wird jede Instanz von NAE-3-SAT in einen Graphen G^* überführt, der den H-Graphen $K_2 \times K_m$ enthält. D.h. jede zulässige Instanz von NAE-3-SAT wird in eine Instanz von SEQ-SG auf nur zwei Maschinen überführt. Der Beweis zeigt damit nicht nur die NP-Vollständigkeit von SEQ-SG, sondern darüber hinaus auch, dass sogar die Einschränkung von SEQ-SG auf nur zwei Maschinen NP-vollständig ist.

Folgerung 4.72 (SEQ-SG) Auch die Einschränkung des Problems SEQ-SG auf Operationenmengen mit nur zwei Maschinen (oder zwei Aufträgen) liegt in NPC.

Bemerkung 4.73 (IRRED und SEQ-SG) Die Tatsache, dass SEQ-SG selbst bei Einschränkung auf zwei Maschinen NP-vollständig ist, während IRRED in diesem Fall polynomiell lösbar ist, kann als Hinweis darauf interpretiert werden, dass IRRED möglicherweise auch im allgemeinen Fall einfacher als SEQ-SG ist.

4.5 Schranken für die Anzahl von Schrägkanten

Zum Abschluss dieses Kapitels wollen wir noch kurz eine untere und eine obere Schranke für die Anzahl der Schrägkanten in Planorientierungen von H-Comparabilitygraphen diskutieren. Wir werden sehen, dass für genügend große vollständige Operationenmengen jeder Plan mindestens 25 Prozent aller möglichen Schrägkanten enthält. Auf der anderen Seite können selbst irreduzible Pläne einen Großteil aller potentiellen Schrägkanten enthalten.

4.5.1 Untere Schranke

Untere Schranken für die Anzahl der Schrägkanten lassen sich durch Überlegungen wie die folgende ermitteln. Der $K_2 \times K_3$ ist nicht transitiv orientierbar. Die einzige Implikationsklasse ist nicht echt. Um ihn zu einem H-Comparabilitygraphen zu machen, genügt es jedoch, eine beliebige Schrägkante hinzuzufügen. In einem beliebigen H-Comparabilitygraphen muss damit jedoch gelten, dass jeder induzierte Teilgraph, der den $K_2 \times K_3$ enthält, ebenfalls mindestens eine Schrägkante enthalten muss. Bei vollständiger Operationenmenge lässt sich daraus leicht eine untere Schranke für die Anzahl der Schrägkanten ermitteln.

Eine bekannte untere Schranke

Eine bessere Schranke erhalten wir, wenn wir größere Teilstrukturen als den $K_2 \times K_3$ als Maßstab benutzen. BRÄSEL UND KLEINAU (1996) [16] haben gezeigt, dass irreduzible Pläne $S \in \mathcal{S}_{SIJ}$ vom Format $2 \times m$ die folgende Form haben,

$$S = \begin{bmatrix} 1 & 2 & \dots & k-1 & k & \dots & m \\ m-k+2 & m-k+3 & \dots & m & 1 & \dots & m-k+1 \end{bmatrix}.$$

Die Anzahl der Schrägkanten in einem solchen Plan ist zum einen von m , zum anderen aber auch von k abhängig. Es ist $f_m(k) = k^2 - k(2+m) + 1 + \frac{1}{2}(m^2 + m)$ (BRÄSEL ET AL. [14]). Diese Anzahl wird minimal für $k = \lfloor \frac{1}{2}m + 1 \rfloor$, d.h. mit dem durch den Parameter k beschriebenen induzierten $K_2 \times K_2$ möglichst "in der Mitte". Es folgt, dass ein beliebiger Plan vom Format $2 \times m$ mindestens $f_m(\frac{1}{2}m + 1) = \frac{1}{4}m(m-2)$ Schrägkanten haben muss ([14]). In einem beliebigen Plan vom Format $n \times m$ muss damit jeder der $\binom{n}{2}$ Teilpläne vom Format $2 \times m$ jeweils mindestens $\frac{1}{4}m(m-2)$ Schrägkanten haben.

Satz 4.74 (untere Schranke) [14] Sei $S \in \mathcal{S}_{SIJ}$ ein Plan mit vollständiger Operationenmenge vom Format $n \times m$ ($m \geq n \geq 2$). Dann hat $[S^{tr}]$ mindestens $\frac{1}{4}m(m-2)\binom{n}{2}$ (ungerichtete) Schrägkanten.

Diese Abschätzung gilt auch im Falle von $n < m$, ist aber für $m \geq n$ besser. Da sich für vollständige Operationenmengen auch die Anzahl der potentiell möglichen Schrägkanten leicht bestimmen lässt, können wir leicht eine untere Schranke für den minimalen Anteil der tatsächlichen Schrägkanten an den potentiell möglichen angeben.

Bemerkung 4.75 (Anteil an potentiell möglichen Schrägkanten) Für Pläne mit vollständiger Operationenmenge ist die Anzahl der potentiell möglichen Schrägkanten gegeben durch $\frac{nm}{2}(n-1)(m-1)$. Es gilt $\lim_{n,m \rightarrow \infty} \frac{\frac{1}{4}m(m-2)\binom{n}{2}}{\frac{nm}{2}(n-1)(m-1)} = \frac{1}{4}$ ($m \geq n$). D.h. für große Pläne gilt, dass jeder Plan, und damit insbesondere auch jeder irreduzible Plan, mindestens $\frac{1}{4}$ aller denkbaren Schrägkanten enthält.

Pläne mit minimaler Anzahl von Schrägkanten

Nach der Konstruktion der Schranke folgt für einen Plan, der sie erreicht, dass in jedem enthaltenen $K_2 \times K_m$ der durch k beschriebene induzierte $K_2 \times K_2$ sehr zentral liegt. Die Enumeration aller Pläne für kleine Formate (bis 4×5) zeigt jedoch, dass in den Plänen mit minimaler Anzahl von Schrägkanten die durch den Parameter k beschriebenen induzierten $K_2 \times K_2$ gleichmäßig über die Gegendiagonale verteilt sind.

$$\begin{aligned}
 B_{28} &= \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{bmatrix} &
 B_{29} &= \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \\ 3 & 4 & 1 & 2 \end{bmatrix} &
 B_{30} &= \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 4 & 1 & 2 \\ 4 & 5 & 1 & 2 & 3 \end{bmatrix} \\
 B_{31} &= \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \\ 3 & 4 & 1 & 2 \\ 4 & 1 & 2 & 3 \end{bmatrix} &
 B_{32} &= \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 4 & 5 & 1 \\ 3 & 4 & 5 & 1 & 2 \\ 4 & 5 & 1 & 2 & 3 \end{bmatrix}
 \end{aligned}$$

Die Pläne B_{28} bis B_{32} beschreiben Pläne mit minimaler Anzahl an Schrägkanten für die Formate 3×3 , 3×4 , 3×5 , 4×4 und 4×5 . Sie suggerieren, dass wir uns bei der Suche nach einer Beschreibung von solchen Plänen mit minimaler Kantenzahl auf zyklische Pläne konzentrieren können. Da zyklische Pläne irreduzibel sind (Satz 4.17), ist das keine große Überraschung.

Beschränken wir uns auf zyklische Pläne, die wir durch die Angabe der ersten Spalte (o.B.d.A.) eindeutig beschreiben können, ist es leicht, auch für die Formate 3×6 , 3×7 und 3×8 die Pläne mit geringster Kantenzahl herauszufinden.

1	1	1	1	1	1	1	1	1	1
2	2	2	2	3	3	3	4	4	5
3	4	5	6	4	5	6	5	6	6
27	23	23	27	23	21	23	23	23	27

Tabelle 4.3: Alle nichtisomorphen zyklischen Pläne (jeweils beschrieben durch die erste Spalte der Rangmatrix) vom Format 3×6 , und ihre Anzahl an Schrägkanten

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	2	2	2	2	3	3	3	3	4	4	4	5	5	6
3	4	5	6	7	4	5	6	7	5	6	7	6	7	7
41	35	33	35	41	35	31	31	35	33	31	33	35	35	41

Tabelle 4.4: Alle nichtisomorphen zyklischen Pläne (jeweils beschrieben durch die erste Spalte der Rangmatrix) vom Format 3×7 , und ihre Anzahl an Schrägkanten

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
2	2	2	2	2	2	3	3	3	3	3	4	4	4	4	5	5	5	5	6	6	7
3	4	5	6	7	8	4	5	6	7	8	5	6	7	8	6	7	8	7	8	8	8
58	50	46	46	50	58	50	44	42	44	50	46	42	42	46	46	44	46	50	50	58	58

Tabelle 4.5: Alle nichtisomorphen zyklischen Pläne (jeweils beschrieben durch die erste Spalte der Rangmatrix) vom Format 3×8 , und ihre Anzahl an Schrägkanten

Basierend auf diesen empirischen Beobachtungen können wir die folgende Vermutung aufstellen.

Vermutung 4.76 (Plan mit minimaler Kantenzahl) Sei $SIJ = I \times J$ eine vollständige Operationenmenge mit n Aufträgen und m Maschinen, und seien r_1, \dots, r_n die Ränge der ersten Spalte eines

(eindeutig bestimmten) zyklischen Plans $S \in \mathcal{S}_{SIJ}$. Gilt $r_i = \left\lfloor 1 + (i-1) \frac{m-2}{n-1} \right\rfloor$ für alle $i = 1, \dots, n$, dann existiert kein Plan $S^* \in \mathcal{S}_{SIJ}$ mit weniger Schrägkanten, $|E_{diag}(S^*)| < |E_{diag}(S)|$.

Begründung. Durch diese Festsetzung der r_i wird ein zyklischer Plan S mit $r_1 = 1$ und $r_n = m - 1$ beschrieben, bei dem die r_i gleichmäßig verteilt sind.

$$S = \begin{bmatrix} \mathbf{1} & 2 & & \cdots & & & m \\ \mathbf{r}_2 & r_2 + 1 & & \cdots & & \mathbf{1} & \cdots & r_2 - 1 \\ \vdots & \vdots & & & & \cdots & & \vdots \\ \mathbf{r}_i & r_i + 1 & & \cdots & \mathbf{1} & & \cdots & r_i - 1 \\ \vdots & \vdots & & & \cdots & & & \vdots \\ \mathbf{r}_{n-1} & r_{n-1} + 1 & \cdots & \mathbf{1} & & \cdots & & r_{n-1} - 1 \\ \mathbf{m} - \mathbf{1} & m & \mathbf{1} & & \cdots & & & r_n - 1 \end{bmatrix}$$

Für kleine Formate kann durch Enumeration bewiesen werden, dass keine Pläne mit weniger Schrägkanten existieren. ■

Ein Beweis dieser Vermutung gestaltet sich schwierig. Wie wir gesehen haben, wird das Minimum der Schrägkanten über einem $K_2 \times K_m$ angenommen, wenn die beiden Quellen bei entsprechender Sortierung der Spalten einen Abstand von etwa $\frac{1}{2}m$ haben. Jede Abweichung davon erhöht die Anzahl der Schrägkanten in diesem Teilplan. Auf der anderen Seite sind die einzelnen Teilpläne der Form $K_2 \times K_m$ nicht unabhängig voneinander.

Ob neben den durch die Vermutung 4.76 beschriebenen Plänen, und den jeweils unter Umständen existierenden engen Verwandten, die durch eine minimal abweichende Anordnung der Quellen entlang der Gegendiagonalen entstehen, und allen jeweils durch Vertauschen von Zeilen oder Spalten entstehenden Plänen, weitere (nichtzyklische) Pläne mit gleicher Anzahl von Schrägkanten existieren, ist nicht klar. Anzeichen gibt es dafür jedoch keine.

Eine vermutete exakte untere Schranke

Basierend auf Vermutung 4.76 können wir eine alternative untere Schranke angeben, die bei Richtigkeit der Vermutung auch für jedes Format angenommen wird. Leider ist keine geschlossene Darstellung der Anzahl der Schrägkanten für diese Beispiele bekannt.

Satz 4.77 (exakte untere Schranke) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan mit vollständiger Operationenmenge vom Format $n \times m$ ($m \geq n \geq 2$). Sei $f_{n,m}(S)$ die Anzahl der (ungerichteten) Schrägkanten von $[S^{tr}]$. Ferner seien

$$\begin{aligned} s_1(i, j, k) &= \max\{0, (m+1) - j - r_k\} = s_4(i, j, k) \\ s_2(i, j, k) &= m - j \\ s_3(i, j, k) &= (m+1) - r_k, \end{aligned}$$

wobei $r_k = \left\lfloor 1 + (k-1) \frac{m-2}{n-1} \right\rfloor$ sei. Bei Richtigkeit von Vermutung 4.76 gilt

$$f_{n,m}(S) \geq \sum_{i=1}^{n-1} \sum_{j=1}^{m-r_i-1} \sum_{k=i+1}^n s_1 + \sum_{i=2}^{n-1} \sum_{j=m-r_i+2}^{m-1} \sum_{k=i+1}^n s_2 + \sum_{i=2}^{n-1} \sum_{j=m-r_i+2}^m \sum_{k=i+1}^n s_3 + \sum_{i=2}^n \sum_{j=m-r_i+2}^{m-1} \sum_{k=1}^{i-1} s_4.$$

und es gibt Pläne, die diese Schranke annehmen.

Beweis. Die Menge der Schrägkanten in $[S^{tr}]$ für den in Vermutung 4.76 beschriebenen Plan kann in vier Teile zerlegt werden. Es gibt Schrägkanten nach “unten rechts” (Summanden s_1 und s_2), solche nach “unten links” (Summand s_3), und solche nach “oben rechts” (Summand s_4). Die erste der vier Teilsommen zählt die Anzahl der Schrägkanten von einer Operation O_{ij} in eine Zeile k (Schrägkanten nach “unten rechts”, oberhalb der Gegendiagonalen). Wenn eine Operation O_{ij} mit $i \in \{1, \dots, n-1\}$ und $j \in \{1, \dots, m-r_i-1\}$ Nachfolger in Zeile k ($k \in \{i+1, \dots, n\}$) hat (nach “unten rechts”, oberhalb der Gegendiagonalen), dann sind es $(m-j+1) - r_k = s_1 > 0$ viele außerhalb

von Spalte j . Operationen O_{ij} mit $i \in \{2, \dots, n-1\}$ und $j \in \{m-r_i+2, \dots, m-1\}$ haben jeweils $m-j = s_2$ Nachfolger (außerhalb von Spalte j) in Zeile $k \in \{i+1, \dots, n\}$ (nach “unten rechts”, unterhalb der Gegendiagonalen). Operationen O_{ij} mit $i \in \{2, \dots, n-1\}$ und $j \in \{m-r_i+2, \dots, m\}$ haben $m-(r_k-1) = s_3$ Nachfolger (außerhalb von Spalte j) in Zeile $k \in \{i+1, \dots, n\}$ (nach “unten links”). Haben schließlich Operationen O_{ij} mit $i \in \{2, \dots, n\}$ und $j \in \{m-r_i+2, \dots, m-1\}$ zusätzlich auch noch Nachfolger (außerhalb von Spalte j) in Zeile $k \in \{1, \dots, i-1\}$ (nach “oben rechts”), dann sind es $(m-j+1) - r_k = s_4 > 0$ viele. ■

m	3	4	5	6	10	100	1000
n							
3	3	8	14	21	71	9.656	996.506
4		16	30	47	147	18.387	1.883.787
5			50	80	250	30.514	3.117.514
6				120	387	45.990	4.689.720
10					1.200	140.994	14.326.344
100						16.170.000	1.639.329.099

Tabelle 4.6: Minimale Anzahl der Schrägkanten in einem Plan mit n Aufträgen und m Maschinen nach Satz 4.77.

n	m (%)	3	4	5	6	10	100	1000
3		16,7	22,2	23,3	23,3	26,3	32,5	33,3
4			22,2	25,0	26,1	27,2	31,0	31,4
5				25,0	26,7	27,8	30,8	31,2
6					26,7	28,7	31,0	31,3
10						29,6	61,7	31,9
100							33,0	33,2

Tabelle 4.7: Minimaler Anteil (in %) der tatsächlichen Schrägkanten in einem Plan $S \in \mathcal{S}_{SIJ}$ an den potentiell möglichen Schrägkanten (nach Satz 4.77).

Bemerkung 4.78 (Anteil an potentiell möglichen Schrägkanten) Bei Richtigkeit von Vermutung 4.76 enthält ein beliebiger Plan über einer vollständigen Operationenmenge für genügend große Werte von n und m sogar mindestens $\frac{1}{3}$ aller jeweils denkbaren Schrägkanten (Tabelle 4.7).

Unvollständige Operationenmengen

Im Falle von unvollständigen Operationenmengen ist die Herleitung von unteren Schranken für die Anzahl der Schrägkanten allenfalls für Spezialfälle möglich. Generell gibt es keine untere Schranke. Es gibt Operationenmengen (mit sogenannter *Caterpillar*-Struktur) mit beliebig großer Anzahl von Aufträgen und Maschinen, auf denen Pläne existieren, die keinerlei Schrägkanten enthalten.

4.5.2 Obere Schranke

Bestimmen wir in G_{SIJ} eine totale Ordnung entlang der regulären Kanten, dann erhalten wir einen Plan $S \in \mathcal{S}_{SIJ}$, dessen H-Comparabilitygraph alle denkbaren Schrägkanten enthält. Grundsätzlich ist also die Anzahl der Schrägkanten in einem H-Comparabilitygraph lediglich durch die Größe des Graphen beschränkt. Anders sieht es aus, wenn wir Pläne mit bekannter oder beschränkter Anzahl der Implikationsklassen oder Planimplikationsklassen betrachten. Da die Existenz von Γ -Ketten oder Γ_S -Ketten die Nichtexistenz von Schrägkanten voraussetzt, können wir bei gegebener Anzahl der Klassen eine obere Schranke für die Anzahl der Schrägkanten ermitteln.

Eine obere Schranke in Abhängigkeit von der Anzahl der Planimplikationsklassen

Wir konzentrieren uns auf den Fall, dass die Anzahl der Planimplikationsklassen bekannt ist. Analog zum Γ -Graphen $G_\Gamma = (E, \Gamma)$ zu einem Graphen $G = (V, E)$ können wir den Γ_S -Graphen $G_{\Gamma_S}(S) = (E_{reg}(S), \Gamma_S)$ zu einem H-Comparabilitygraphen $[S^{tr}] = (SIJ, E_{reg}(S) + E_{diag}(S))$ definieren. G_{Γ_S} ist ein induzierter Teilgraph des Γ -Graphen von $[S^{tr}]$. Seine Zusammenhangskomponenten entsprechen den Planimplikationsklassen von $[S^{tr}]$, d.h. den Planimplikationsklassen von S , und deren Umkehrungen.

Satz 4.79 (obere Schranke, SIJ beliebig) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan mit genau p Planimplikationsklassen über einer beliebigen Operationenmenge $SIJ \subseteq I \times J$. Ferner sei d_{\max} die Anzahl der potentiell möglichen Schrägkanten über SIJ , und r die Anzahl der regulären Kanten über SIJ . Dann hat $[S^{tr}]$ höchstens $d_{\max} - \left\lceil \frac{r-2p}{2} \right\rceil$ (ungerichtete) Schrägkanten.

Beweis. Der Γ_S -Graph $G_{\Gamma_S}(S)$ hat genau r Knoten und $2p$ Zusammenhangskomponenten—die Planimplikationsklassen von S und die Planimplikationsklassen von S^{-1} . Er hat damit mindestens $(r - 2p)$ Kanten. Jede Kante in G_{Γ_S} repräsentiert genau eine Γ_S -Relation in $[S^{tr}]$, und jede Γ_S -Relation geht mit der Existenz einer Nichtkante in $[S^{tr}]$ einher. Umgekehrt kann jede Nichtkante in $[S^{tr}]$ maximal zwei verschiedene Γ_S -Relationen induzieren. Damit muss $[S^{tr}]$ also mindestens $\left\lceil \frac{r-2p}{2} \right\rceil$ verschiedene Nichtkanten haben. D.h. $|E_{diag}(S)| \leq d_{\max} - \left\lceil \frac{r-2p}{2} \right\rceil$. ■

Folgerung 4.80 (obere Schranke, SIJ vollständig) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan mit genau p Planimplikationsklassen über einer vollständigen Operationenmenge. Dann hat $[S^{tr}]$ höchstens $\left\lfloor \frac{nm}{4} (2nm - 3n - 3m + 4) + p \right\rfloor$ (ungerichtete) Schrägkanten.

Beweis. Bei vollständiger Operationenmenge gelten $d_{\max} = \frac{nm}{2} (n - 1)(m - 1)$ und $r = n \binom{m}{2} + m \binom{n}{2}$. Ferner gilt, dass jede Nichtkante in $[S^{tr}]$ genau zwei verschiedene Γ_S -Relationen induziert. Damit folgt aus Satz 4.79

$$\begin{aligned} |E_{diag}(S)| &\leq \frac{nm}{2} (n - 1)(m - 1) - \left\lceil \frac{n \binom{m}{2} + m \binom{n}{2} - 2p}{2} \right\rceil \\ &= \frac{nm}{2} (n - 1)(m - 1) - \left\lceil \frac{nm(m-1)}{4} + \frac{mn(n-1)}{4} - p \right\rceil \\ &= \left\lfloor \frac{nm}{2} (n - 1)(m - 1) - \frac{nm(m-1)}{4} - \frac{mn(n-1)}{4} + p \right\rfloor \\ &= \left\lfloor \frac{nm}{4} [2(n-1)(m-1) - (m-1) - (n-1)] + p \right\rfloor \\ &= \left\lfloor \frac{nm}{4} [2nm - 3n - 3m + 4] + p \right\rfloor. \end{aligned}$$

■

Irreduzible Pläne mit maximaler Anzahl von Schrägkanten

Wenden wir die Schranke aus Folgerung 4.80 auf Pläne mit genau einer Planimplikationsklasse an, dann erhalten wir einen Hinweis darauf, wie viele Schrägkanten irreduzible Pläne möglicherweise maximal haben können. Diese Vorgehensweise liefert jedoch keine obere Schranke für die Anzahl der Schrägkanten in irreduziblen Plänen, da wir nicht sicher sein können, dass keine irreduziblen Pläne mit mehreren Planimplikationsklassen existieren, die noch mehr Schrägkanten enthalten.

Da es jedoch generell keine Hinweise darauf gibt, dass irreduzible Pläne mit mehreren Planimplikationsklassen etwas anderes als seltene Ausnahmen sind, und die Schranke aus Folgerung 4.80 darüber hinaus nur sehr langsam in p wächst, liefert sie uns eine sehr gute Näherung für eine obere Schranke der Anzahl der Schrägkanten in irreduziblen Plänen.

Folgerung 4.81 (obere Schranke, 1 Planimplikationsklasse) Sei $S \in \mathcal{S}_{SIJ}$ ein Plan über einer vollständigen Operationenmenge mit genau einer Planimplikationsklasse. Dann hat $[S^{tr}]$ höchstens $\left\lfloor \frac{nm}{4} (2nm - 3n - 3m + 4) + 1 \right\rfloor$ (ungerichtete) Schrägkanten.

Bemerkung 4.82 (Anteil an potentiell möglichen Schrägkanten) Es gilt $\lim_{n,m \rightarrow \infty} \frac{\lfloor \frac{nm}{4}(2nm-3n-3m+4)+1 \rfloor}{\frac{nm}{2}(n-1)(m-1)} = 1$. D.h. irreduzible Pläne können theoretisch beliebig groß werden, und fast alle möglichen Schrägkanten enthalten.

m	3	4	5	6
n				
2	1 (2)	3 (5)	6 (8)	10 (13)
3	7 (10)	18 (22)	33 (38)	52 (59)
4		45 (49)	82 (86)	≥ 125 (133)

Tabelle 4.8: Maximale Anzahl von Schrägkanten in $[S^{tr}]$ zu Plan $S \in \mathcal{S}_{SIJ}$ mit genau einer Planimplikationsklasse, und obere Schranke nach Folgerung 4.81 (in Klammern).

Für kleine Formate lässt sich die tatsächliche maximale Anzahl der Schrägkanten in Plänen mit genau einer Planimplikationsklasse noch enumerativ ermitteln (Tabelle 4.8).

Obwohl viele Schrägkanten in einem Plan in der Regel zu vielen verschiedenen Planimplikationsklassen führen, können also unter besonderen Umständen auch Pläne mit wenigen oder sogar nur einer Planimplikationsklasse sehr viele Schrägkanten enthalten.

Damit beschließen wir das Kapitel zum Problem IRRED und den ersten Teil dieser Arbeit. Nachdem wir nun alle grundlegenden Aspekte und Eigenschaften der für dieses Problem relevanten Objekte kennen, diskutieren wir im zweiten Teil dieser Arbeit die Reduzierbarkeit von Plänen, bzw. die Erkennung von irreduziblen Plänen.

Teil II

Zur Reduzierbarkeit von Plänen

Kapitel 5

Ansätze zur Reduktion von Plänen

Der zweite Teil dieser Arbeit dient der Analyse der Komplexität der Erkennung irreduzibler Pläne. Durch den Vergleich der Schrägkantenmengen zweier gegebener H-Comparabilitygraphen lässt sich leicht ermitteln, ob einer der beiden einen Plan beschreibt, der alle durch den anderen Graphen beschriebenen Pläne streng reduziert (Satz 4.2). Dies liefert einen co-NP-Test für das Problem IRRED (Satz 4.3).

Wir haben damit also einen nichtdeterministischen Algorithmus mit polynomiell beschränkter Laufzeit, der uns zu *jedem* reduzierbaren Plan mit *positiver* Wahrscheinlichkeit einen ihn streng reduzierenden Plan liefern kann. Offen ist die Frage, ob auch ein NP-Algorithmus existiert, der die Irreduzibilität eines Plans immer mit *positiver* Wahrscheinlichkeit erkennen kann. In Kapitel 6 werden wir einen Algorithmus vorstellen, der fast alle irreduziblen Pläne in polynomiell beschränkter Laufzeit erkennt.

In diesem Kapitel wollen wir einige nahe liegende Ansätze für einen polynomiellen Reduktionsalgorithmus diskutieren. Diese Ansätze umfassen das Umkehren von Planimplikationsklassen (Abschnitt 5.2), die polynomielle Erkennung von stabilen Schrägkanten (Abschnitt 5.3), die Anwendung des erweiterten Dreieckslemmas auf Planimplikationsklassen (Abschnitt 5.4) und die Ausnutzung von speziellen Eigenschaften von Comparabilitygraphen (Abschnitt 5.5). Zuerst wollen wir uns jedoch noch auf einen geeigneten Startpunkt für die Suche nach reduzierenden Plänen verständigen (Abschnitt 5.1).

5.1 Normale Pläne als Ausgangspunkt

Als Ausgangspunkt für sämtliche Ansätze zur Reduktion von Plänen wollen wir Pläne betrachten, die nicht mehr durch Umkehren von Implikationsklassen reduziert werden können.

Definition 5.1 (normaler Plan) *Ein Plan $S \in \mathcal{S}_{SIJ}$ heißt normal, wenn sein H-Comparabilitygraph $[S^{tr}]$ keine Implikationsklasse enthält, die ausschließlich aus Schrägkanten besteht, und er durch Umkehren von Implikationsklassen nicht weiter streng reduziert werden kann.*

In Abschnitt 4.1.4 haben wir beschrieben, wie ein Plan in polynomieller Zeit auf Reduzierbarkeit durch Umkehren von Implikationsklassen getestet werden kann.

Lemma 5.2 (normaler Plan) *Jeder Plan $S \in \mathcal{S}_{SIJ}$ kann in polynomieller Laufzeit in einen normalen Plan $S^* \in \mathcal{S}_{SIJ}$ mit $S^* \preceq S$ überführt werden.*

Beweis. Implikationsklassen, die ausschließlich aus Schrägkanten bestehen, können nach Satz 1.45 entfernt werden. Die Reduktion durch das Umkehren von Implikationsklassen ist nach Bemerkung 4.7 in polynomieller Laufzeit möglich. ■

Bemerkung 5.3 (Eigenschaften normaler Pläne) *Sei $S \in \mathcal{S}_{SIJ}$ ein normaler Plan, \mathcal{P}_S die Menge der Planimplikationsklassen von S , und $\mathcal{F}_{[S^{tr}]}$ die Menge der Farbklassen von $[S^{tr}]$. Dann gelten*

- (i) Zu jeder Farbklassse von $[S^{tr}]$ gehört mindestens eine Planimplikationsklasse von S . Insbesondere gilt $|\mathcal{F}_{[S^{tr}]}| \leq |\mathcal{P}_S|$.
- (ii) Jede transitive Orientierung von $[S^{tr}]$ ist eine Planorientierung.

Eine unmittelbare Folgerung aus diesen Eigenschaften ist, dass ein Plan, bei dem jede Planimplikationsklasse zu einer anderen Implikationsklasse gehört, nicht weiter streng reduziert werden kann.

Satz 5.4 (hinreichende Bedingung für IRRED) Sei $S \in \mathcal{S}_{SIJ}$ ein normaler Plan. Gilt $|\mathcal{F}_{[S^{tr}]}| = |\mathcal{P}_S|$, dann ist S irreduzibel.

Beweis. Nach Satz 4.40 ist jeder reduzierende Plan von S eine Rekombination der Planimplikationsklassen S . Nach Voraussetzung ist jede Rekombination der Planimplikationsklassen jedoch eine Rekombination der Implikationsklassen. Da S jedoch normal ist, kann S nicht durch Rekombination der Implikationsklassen reduziert werden. Also ist S irreduzibel. ■

5.2 Naives Umkehren von Planimplikationsklassen

Nach Satz 4.43 beschreibt eine Rekombination der Planimplikationsklassen eines Ausgangsplans genau dann einen reduzierenden Plan, wenn sie kreisfrei ist und keine neue Schrägkante induziert. Durch die Enumeration aller Rekombinationen ließe sich selbstverständlich zweifelsfrei ermitteln, ob streng reduzierende Pläne existieren, oder nicht (Problemformulierung #4 auf Seite 83). Jedoch ist die Anzahl der dabei zu betrachtenden Rekombinationen exponentiell in der Anzahl der Planimplikationsklassen des Ausgangsplans. Und da auch irreduzible Pläne sehr viele Planimplikationsklassen enthalten können (Bemerkung 4.61), ist auf diesem Weg kein polynomiell beschränkter Algorithmus zur Erkennung von Irreduzibilität denkbar.

Möglichkeit zur polynomiellen Beschränkung

Um die Laufzeit polynomiell beschränken zu können, müssen wir den exponentiellen Suchbaum der Enumeration aller Rekombinationen geeignet einschränken. Eine Möglichkeit hierzu besteht darin, jeweils immer nur genau eine Planimplikationsklasse umzukehren. Wir starten mit einem (normalen) Ausgangsplan $S \in \mathcal{S}_{SIJ}$ und bestimmen die Planimplikationsklassen von S , $S = P_1 + \dots + P_k$. Anschließend kehren wir ein beliebiges P_i um. Um die Chancen zu erhöhen, ein geeignetes P_i zu finden, können wir uns erlauben, alle P_i einzeln zu testen, und anschließend eines auszuwählen. Wir bestimmen also in einer Stufe maximal k verschiedene Rekombinationen der Form $S_i = S - P_i + P_i^{-1}$, und prüfen für jedes S_i , ob $S_i \prec S$ gilt. Haben wir ein S_i mit $S_i \prec S$ gefunden, dann fahren wir mit S_i fort.

Algorithmus NAIVES UMKEHREN

Input: $S \in \mathcal{S}_{SIJ}$ normaler Plan.

Output: $S^* \in \mathcal{S}_{SIJ}$ mit $S^* \preceq S$.

```

recombine( $S^*$ ) {
    Bestimme Planimplikationsklassen  $P_1, \dots, P_k$  von  $S^*$ ;
     $i := 1$ ;
    while ( $i \leq k$ ) do {
        Bestimme  $S^{**} := S^* - P_i + P_i^{-1}$ ;
        if ( $S^{**}$  kreisfrei and  $S^{**} \prec S^*$ ) then {
            recombine( $S^{**}$ );
        };
        else  $i := i + 1$ ;
    };
}

```

```

    return  $S^*$ ;
  };
BEGIN {
   $S^* := S$ ;
  recombine( $S^*$ );
};
END.

```

Lemma 5.5 (Naives Umkehren) *Sei $S \in \mathcal{S}_{SIJ}$ ein normaler Plan über einer Operationenmenge mit n Aufträgen und m Maschinen. Der Algorithmus **Naives Umkehren** ist korrekt, und hat einen durch $\mathcal{O}(n^6 m^6)$ beschränkten Aufwand.*

Beweis. Die Prozedur **recombine** wird mit der Eingabe $S^{**} = S^* - P_i + P_i^{-1}$ für ein $P_i \in \mathcal{P}_{S^*}$ nur aufgerufen, wenn S^{**} den Plan S^* streng reduziert. Eine Rückgabe erfolgt erst, wenn der Plan S^* entweder nur noch aus einer Planimplikationsklasse besteht, oder nicht weiter durch Umkehren von genau einer Planimplikationsklasse streng reduziert werden kann. Damit folgt die Korrektheit. Die Bestimmung der Planimplikationsklassen ist mit dem Aufwand $\mathcal{O}(|V_{[S^{tr}]}| \cdot |E_{[S^{tr}]}|) = \mathcal{O}(n^3 m^3)$ realisierbar. Alle Prüfungen und die Bestimmung von S^{**} sind mit dem Aufwand $\mathcal{O}(|E_{[S^{tr}]}|) = \mathcal{O}(n^2 m^2)$ realisierbar. Da der Plan S^{**} bei Aufruf von **recombine** mindestens eine Schrägkante weniger enthält, als der Plan S^* wird die Funktion **recombine** maximal $|E_{diag}(S)|$ oft aufgerufen. Bei jedem Aufruf von **recombine** wird die Schleife maximal k -mal durchlaufen, wobei k durch $|E_{reg}(S)| = \mathcal{O}(n^2 m^2)$ beschränkt ist. Damit ist die Gesamtlaufzeit beschränkt durch $\mathcal{O}(n^2 m^2 \cdot (n^2 m^2 \cdot n^2 m^2)) = \mathcal{O}(n^6 m^6)$. ■

Der Nachteil dieses Verfahrens besteht darin, dass nicht sichergestellt werden kann, dass der zurückgegebene Plan S^* irreduzibel ist. Während sich für Implikationsklassen mit Hilfe von Golumbic's Struktursatz (Satz 1.21) beweisen lässt, dass zu jeder transitiven Orientierung $T = I_1 + \dots + I_k$ auf einem Comparabilitygraphen mindestens eine der Orientierungen $T_i = T - I_i + I_i^{-1}$ kreisfrei, und damit ebenfalls transitiv ist, ist das für Planimplikationsklassen nicht möglich. Weder lässt sich einfach zeigen, dass mindestens eine der Rekombinationen $S^* = S_i = S - P_i + P_i^*$ kreisfrei ist, noch würde daraus folgen, dass S_i den Plan S reduziert oder sogar streng reduziert. Ein Plan S könnte beispielsweise ausschließlich durch Umkehren von genau zwei von mindestens vier verschiedenen Planimplikationsklassen streng reduzierbar sein. In diesem Fall würde der Algorithmus $S^* = S$ mit $S^* \preceq S$ zurückgeben.

Andererseits sind keine Beispiele für Pläne bekannt, bei denen dieses Verfahren *nicht* zu einem irreduziblen Plan führt. Könnte man also zeigen, dass solche Beispiele nicht existieren, dann hätte man ein polynomielles Verfahren zur Bestimmung irgendeines irreduziblen Plans zu einem gegebenen Ausgangsplan.

IRRED in ZPP(1/2)?

Betrachten wir die folgende Erweiterung der Schleife in der Prozedur **recombine**. Kehren wir nicht nur genau eine, sondern mit $l = 1$ beginnend, solange jeweils genau l viele Planimplikationsklassen um, bis wir einen streng reduzierenden Plan finden, dann finden wir bei jedem Aufruf von **recombine** einen streng reduzierenden Plan $S^{**} \prec S^*$, sofern S^* reduzierbar ist. Allerdings verlieren wir dadurch die polynomielle Beschränktheit innerhalb eines Aufrufs, da wir unter Umständen vollständig enumerieren müssen.

Abhängig von der relativen Häufigkeit solcher Instanzen, bei denen das ursprüngliche Verfahren keinen irreduziblen Plan liefert, könnte dieses variierte Verfahren theoretisch dennoch eine polynomielle maximale durchschnittliche Laufzeit haben. Der Algorithmus **Naives Umkehren** könnte damit mit der Behauptung, dass der zurückgegebene Plan S^* irreduzibel ist, zu einem ZPP($\frac{1}{2}$)-Verfahren umgerüstet werden, bei dem jede Eingabe der Länge nm (Anzahl der Operationen) eine Versagenswahrscheinlichkeit $\varepsilon(nm) < \frac{1}{2}$ hat (mit Satz 2.16).

Bemerkung 5.6 (Naives Umkehren) *Durch das Umkehren von jeweils genau einer Planimplikationsklasse kann eine polynomielle Beschränkung des Suchbaums der vollständigen Enumeration aller*

Rekombinationen der Planimplikationsklassen des Ausgangsplans erreicht werden (Problemformulierung #4). Andererseits lässt sich hiermit die Irreduzibilität nicht beweisen. Während nicht bekannt ist, ob Instanzen für IRRED existieren, bei denen das Verfahren nicht zu einem irreduziblen Plan führt, könnte IRRED selbst bei Existenz solcher Instanzen noch in $ZPP(\frac{1}{2}) \subseteq NP \cap co-NP$ liegen.

5.3 Erkennung stabiler Schrägkanten

Ein anderer Lösungsansatz zur Reduktion von Plänen besteht darin, alle stabilen Schrägkanten, die nicht entfernt werden können, von den nicht-stabilen Schrägkanten, unterscheiden zu können (mit Problemformulierung #3 auf Seite 82).

Charakterisierung durch endliche Menge von Konstellationen?

In Abschnitt 4.3 haben wir die Ähnlichkeiten zwischen den stabilen Schrägkanten im Zusammenhang mit Planimplikationsklassen und den immer-transitiven Kanten im Zusammenhang mit Implikationsklassen betont. Da immer-transitive Kanten durch eine Menge von nur zwei möglichen und polynomiell überprüfbaren Konstellationen charakterisiert werden können (Satz 1.44), ist die Frage naheliegend, ob sich stabile Schrägkanten nicht ebenso durch eine endliche Menge von Konstellationen charakterisieren lassen. In Abschnitt 4.3.3 haben wir diese Frage bereits untersucht (Bemerkung 4.55). Sofern man nicht mit anderen Mitteln zeigen kann, dass nichttrivial-stabile Schrägkanten, deren Stabilität durch mehr als zwei verschiedene Planimplikationsklassen induziert wird, nicht existieren, ist die Menge der denkbaren Konstellationen unbegrenzt.

Transitivierende Kombinationen der Planimplikationsklassen

Wenn eine vollständige Charakterisierung nicht möglich ist, dann lassen sich möglicherweise wenigstens einige polynomiell überprüfbare notwendige Bedingungen ableiten. Schrägkanten, die eine solche für nichttrivial-stabile Schrägkanten notwendige Bedingung erfüllen, können dann als potentielle stabile Schrägkante behandelt werden, die nicht entfernt werden dürfen, solange sie diese Bedingungen erfüllen.

Eine leicht überprüfbare Bedingung ist die folgende. In einer gegebenen azyklischen Orientierung von G_{SIJ} lässt sich zu jeder Schrägkante in einem Plan $S \in \mathcal{S}_{SIJ}$ leicht ein regulärer Weg finden, der für die Transitivität, und damit für die Existenz dieser Schrägkante verantwortlich ist (Bestimmung eines kürzesten Weges in einem gerichteten Graphen). Zu diesem Weg kann ebenfalls sehr leicht die Anzahl der beteiligten Planimplikationsklassen bestimmt werden. Ist nur eine Planimplikationsklasse beteiligt, dann ist die Schrägkante trivial-stabil. Sind es zwei oder mehr, dann kann keine Aussage getroffen werden. Im Falle von genau zwei Planimplikationsklassen P_1 und P_2 können wir uns jedoch erlauben, in $(P_1 + P_2^{-1})^{tr}$ nach der Existenz der gewählten Schrägkante zu suchen. Die Gesamtlaufzeit bleibt dann immernoch polynomiell beschränkt. Enthält $(P_1 + P_2^{-1})^{tr}$ die gewählte Schrägkante, dann ist diese stabil durch P_1 und P_2 . Enthält $(P_1 + P_2^{-1})^{tr}$ die Schrägkante nicht, dann kann S durch Verdrehen von P_1 gegenüber P_2 streng reduziert werden, sofern die Schrägkante im Ausgangsplan nicht noch auf einem anderen Weg durch andere Planimplikationsklassen transitiviert wird, und $(P_1 + P_2^{-1})^{tr}$ keine neuen Schrägkanten induziert. Da wir o.B.d.A. P_1 und P_2 als erweiterte Planimplikationsklassen betrachten können, ist $(P_1 + P_2)^{tr}$ immerhin in jedem Fall kreisfrei.

Enthält der gefundene transitivierende reguläre Weg zu der gewählten Schrägkante drei oder mehr Planimplikationsklassen, dann erfüllt diese Schrägkante eine einfache notwendige Bedingung für nicht-trivial-stabile Schrägkanten. In diesem Fall sollten wir uns bei der Reduktion von S zunächst auf andere Schrägkanten konzentrieren. Diese Strategie führt dazu, dass die Reduktion prinzipiell durch die Entfernung von Eckkanten erfolgt. Die Bedingung kann daher gegebenenfalls auch von vornherein auf Wege der Länge zwei beschränkt werden.

Das Problem mit dieser Strategie besteht darin, dass auf diese Weise in einem reduzierbaren Plan möglicherweise sämtliche Schrägkanten als potentielle stabile Schrägkanten markiert werden. Das Kriterium "es existiert ein transitivierender regulärer Weg mit mindestens drei verschiedenen Planimplikationsklassen" ist wenig geeignet, um nichttrivial-stabile Schrägkanten von anderen zu unterscheiden.

Es wird beispielsweise auch von solchen Schrägkanten erfüllt, zu denen sowohl ein (längerer) transitivierender Weg mit nur zwei Planimplikationsklassen, als auch ein (kürzerer) transitivierender Weg mit drei oder mehr Planimplikationsklassen existiert.

Andererseits ist auch hier kein Beispiel für einen reduzierbaren Plan bekannt, der bei Berücksichtigung dieser Bedingung nicht weiter streng reduziert werden kann. Mit den Überlegungen aus Bemerkung 4.57 (Konstruktion von Beispielen) wird klar, dass es nicht leicht ist, einen Plan zu konstruieren, bei dem keine der Ecken entfernt werden kann.

Bemerkung 5.7 (Erkennung stabiler Schrägkanten) *Eine Charakterisierung von stabilen Schrägkanten durch eine endliche Menge von Konstellationen ist möglich, falls der Nachweis gelingt, dass nichttrivial-stabile Schrägkanten nur durch maximal zwei verschiedene Planimplikationsklassen transitiviert werden können. Andernfalls ist eine solche polynomiell überprüfbare Charakterisierung zwar nicht ausgeschlossen, aber eher unwahrscheinlich. Notwendige Bedingungen für nichttrivial-stabile Schrägkanten sind entweder zu grob (mindestens drei Planimplikationsklassen in einem transitivierenden regulären Weg), oder nicht polynomiell überprüfbar (Bestimmung aller Kombinationen der Planimplikationsklassen, die zur Transitivität einer Schrägkante führen).*

5.4 Erweitertes Dreieckslemma für Planimplikationsklassen

Ein anderer Ansatz zur Erkennung von nichttrivial-stabilen Schrägkanten, und damit zur Reduktion von Plänen, besteht in der Beschreibung von notwendigen oder hinreichenden lokalen Bedingungen für nichttrivial-stabile Schrägkanten auf der einen Seite, und ebensolchen Bedingungen für entfernbare Schrägkanten auf der anderen Seite.

Im Falle von immer-transitiven Kanten und Implikationsklassen ist das Dreieckslemma von Golumbic (Satz 1.15) hierzu ein geeignetes Hilfsmittel. Für Schrägkanten und Planimplikationsklassen muss hierzu die Erweiterung des Dreieckslemmas auf beliebige $\dot{\Gamma}$ -Komponenten herangezogen werden (Satz 4.47), die wir in Abschnitt 4.3.2 beschrieben haben. Dort sind zahlreiche Bedingungen für H-Comparabilitygraphen beschrieben. Leider ist die Anwendbarkeit dieser Bedingungen durch die starken Voraussetzungen so stark eingeschränkt, dass sie keine große Hilfe sind. Wir haben dort auch die Vor- und Nachteile bei der Wahl einer jeweils geeigneten Trennungsmenge diskutiert (Abschnitt 4.3.2).

Während die Konstellation mit drei beteiligten Klassen in Abbildung 4.9 auf Seite 92 im Falle von Implikationsklassen durch mehrmalige Anwendung des Dreieckslemmas als unmöglich realisierbar bewiesen werden kann, ist dies für Planimplikationsklassen nicht möglich. Die einzige definitive Aussage, die sich in diesem Fall für diese lokale Konstellation treffen lässt, ist, dass sowohl a , als auch b mit Kanten aus F_{P_2} inzidieren müssen. Das folgt jeweils aus dem Vergleich verschiedener Dreiecke bei Anwendung von Folgerung 4.50 auf Kanten aus P_2 . Für die Existenz oder Nichtexistenz eines Plans mit dieser Konstellation liefert diese Information jedoch keinerlei Anhaltspunkte. Auch für die Konstruktion eines Beispiels ist diese Information in keiner Weise hilfreich.

Bemerkung 5.8 (erweitertes Dreieckslemma) *Die Erweiterung des Dreieckslemmas auf $\dot{\Gamma}$ -Komponenten zu beliebigen Trennungsmengen $F \subseteq E$ eines Graphen $G = (V, E)$ ist für Planimplikationsklassen in H-Comparabilitygraphen prinzipiell anwendbar. Wegen der starken Voraussetzung ist jedoch die tatsächliche Anwendbarkeit so stark eingeschränkt, dass für konkrete Konstellationen kaum relevante Aussagen getroffen werden können. Daher lassen sich weder für nichttrivial-stabile Schrägkanten, noch für entfernbare Schrägkanten notwendige oder hinreichende Bedingungen ableiten.*

5.5 Eigenschaften von Comparabilitygraphen

Ein letzter naheliegender Ansatz zur Reduktion von Plänen ist die Ausnutzung der Eigenschaften von Comparabilitygraphen. Wir suchen schließlich einen echten Teilgraphen eines H-Comparabilitygraphen, der selber transitiv orientierbar ist (Problemformulierung #2). Nach der Diskussion der Komplexität von verwandten Problemen zu IRRED in Abschnitt 4.4 muss jeder polynomielle Reduktionsalgorithmus zwingend ausnutzen, dass durch den H-Comparabilitygraphen des Ausgangsplans bereits ein transitiv orientierbarer Graph gegeben ist.

5.5.1 Bestimmung von Mengen entfernbarer Schrägkanten?

Wir wissen, dass wir alle Schrägkanten eines H-Comparabilitygraphen in polynomieller Zeit auf ihre jeweilige Entfernbareit überprüfen können (Abschnitt 4.1.4). Sei es durch die jeweilige Überprüfung der Bedingung aus Satz 1.43, oder durch die Überprüfung von $G - \hat{e}$ für $e \in E_{diag}$ auf Existenz einer transitiven Orientierung. Andererseits haben wir auch gesehen, dass die Information über die einzelne Entfernbareit der Schrägkanten lediglich Aufschluss über die Reduzierbarkeit eines Ausgangsplans durch Umkehren von Implikationsklassen liefert. Durch die Reduzierung über die Ermittlung von Schrägkanten $e \in E_{diag}$, für die $G - \hat{e}$ transitiv orientierbar ist, gelangen wir zu normalen Plänen (Definition 5.1). Ein normaler Plan ist jedoch nicht unbedingt irreduzibel.

$$B_{33} = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 4 & 3 & 1 \end{bmatrix} \quad B_{34} = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 2 & 3 & 1 \end{bmatrix}$$

Der H-Comparabilitygraph $[B_{33}^{tr}]$ zu dem Plan B_{33} beispielsweise ist prim. Er kann also nicht weiter durch Umkehren von Implikationsklassen reduziert werden. Keine der sechs Schrägkanten (Abbildung 5.1) kann einzeln entfernt werden. Dennoch ist B_{33} nicht irreduzibel. Er besteht aus zwei verschiedenen Planimplikationsklassen, von denen eine drei der Schrägkanten als trivial-stabile Schrägkanten enthält (graue Schrägkanten ohne gemeinsamen Knoten). Werden von den verbleibenden drei Schrägkanten die beiden Schrägkanten aus der trennenden Kantenmenge zwischen den beiden Planimplikationsklassen (die beiden Kanten, die mit Operationen mit Rang 2 inzidieren) gleichzeitig entfernt, dann kann B_{33} streng reduziert werden. Der resultierende irreduzible Plan ist der Plan B_{34} .

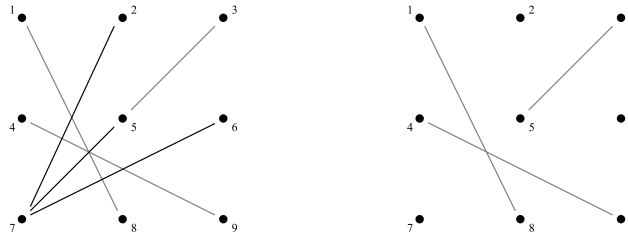


Abbildung 5.1: Die Schrägkantenmengen von $[B_{33}^{tr}]$ (links) und $[B_{34}^{tr}]$ (rechts). Aus $[B_{33}^{tr}]$ müssen simultan mindestens zwei Schrägkanten entfernt werden, um wieder zu einem Comparabilitygraphen zu gelangen.

Lassen sich aus den Informationen über die individuelle Entfernbareit aller Schrägkanten Rückschlüsse auf mögliche Mengen simultan entfernbarer Schrägkanten ziehen? Betrachten wir hierzu ein weiteres Mal ein bereits bekanntes Beispiel. Der Plan A_{234} aus Beispiel 4.44 bzw. Beispiel 4.8 (Plan S , Abbildung 4.4) besteht aus 3 Implikationsklassen und enthält 8 Schrägkanten. Zu A_{234} gibt es insgesamt genau 5 verschiedene irreduzible Pläne: A_{156} , A_{1567} , A_{1256} , A_{12567} und A_{3457} (Abbildung 4.8). Während jede der 8 Schrägkanten in $[A_{234}^{tr}]$ einzeln entfernt werden kann—keine der Schrägkanten ist stabil—, kann nicht jede beliebige Teilmenge der einzeln entfernbarer Schrägkanten simultan entfernt werden. Die Menge der Schrägkanten $\widehat{1.4}$, $\widehat{3.4}$ und $\widehat{3.8}$ beispielsweise enthält aus jedem der 5 irreduziblen Pläne zu A_{234} eine Kante. Die Entfernung dieser Menge aus $[A_{234}^{tr}]$ kann damit, unabhängig von allen anderen Schrägkanten, auf keinen Fall zu einem Comparabilitygraphen führen.

Bemerkung 5.9 (individuelle Entfernbareit von Schrägkanten) *Die individuelle Entfernbareit jeder einzelnen Kante einer Teilmenge der Schrägkanten in einem H-Comparabilitygraph ist weder eine notwendige, noch eine hinreichende Bedingung für die Entfernbareit der gesamten Teilmenge.*

Im folgenden Kapitel werden wir einen fortgeschritteneren Ansatz zu Bestimmung einer zulässigen Entfernungsmenge unter der Menge der Schrägkanten eines H-Comparabilitygraphen beschreiben.

5.5.2 Verbotene Teilgraphen

Eine andere Eigenschaft von Comparabilitygraphen ist, dass sie durch das Nichtenthaltensein einer Menge von Teilgraphen charakterisiert werden können. Ein Graph ist genau dann ein Comparability-

graph, wenn er keinen der Graphen aus einer Liste \mathcal{C} als induzierten Teilgraphen enthält (Satz 1.36). Diese Liste $\mathcal{C} = \mathcal{C}_1 + \mathcal{C}_2$ enthält dabei die Graphenklassen und Graphen, die durch die Abbildungen 1.5, 1.6 und 1.7 (Seite 25-27) beschrieben sind. Das besondere an dieser Charakterisierung ist, dass diese Liste \mathcal{C} der verbotenen Teilgraphen bei beschränkter Anzahl der Knoten endlich ist.

Die H-Comparabilitygraphen zu irreduziblen Plänen sind die minimalen Elemente der durch die Reduzierbarkeits-Relation beschriebenen Halbordnung auf der Menge der H-Graphen mit beliebiger Schrägkantenmenge (Abschnitt 4.1.1). Für sie gilt, dass es keine Menge der Schrägkanten gibt, deren Entfernung wieder zu einem Comparabilitygraphen führt. Nach Satz 1.36 (verbotene Teilgraphen) gilt also für jede Schrägkante $e \in E_{diag}(S)$ in einem irreduziblen Plan $S \in \mathcal{S}_{SIJ}$, dass der Graph $[S^{tr}] - \hat{e}$ einen Graphen aus der Liste \mathcal{C} als induzierten Teilgraphen enthält. Eine naheliegende Frage ist nun: Können stabile Schrägkanten auf diese Weise charakterisiert werden? Gilt beispielsweise, dass eine Schrägkante e in einem beliebigen Plan $S \in \mathcal{S}_{SIJ}$ genau dann stabil ist, wenn der Graph $[S^{tr}] - \hat{e}$ einen Graphen aus \mathcal{C} als induzierten Teilgraphen enthält?

Reguläre H-Einbettungen von Graphen aus \mathcal{C}

Während für eine stabile Schrägkante natürlich gilt, dass $[S^{tr}] - \hat{e}$ einen Graphen aus \mathcal{C} enthält, ist die Umkehrung jedoch nicht richtig, wie wir bereits an mehreren Beispielen gesehen haben (zuletzt an dem Plan B_{33}). Enthält $[S^{tr}] - \hat{e}$ den Graphen $G^* \in \mathcal{C}$ als induzierten Teilgraphen, dann kann das Entfernen einer oder mehrerer Kanten von G^* den verbleibenden Graphen durchaus wieder transitiv orientierbar machen. Würde G^* jedoch ausschließlich aus regulären Kanten bestehen, dann könnte keine dieser Kanten mehr entfernt werden. Jeder Teilgraph von $[S^{tr}] - \hat{e}$, der den H-Graphen G_{SIJ} enthält, würde den Graphen G^* als induzierten Teilgraphen enthalten. In diesem Fall wäre die entfernte Kante $e \in E_{diag}(S)$ tatsächlich stabil.

Wenn wir eine Darstellung eines Graphen $G^* \in \mathcal{C}$ als induzierten Teilgraphen eines H-Graphen G_{SIJ} zu einer Operationenmenge SIJ eine *H-Einbettung* von G^* nennen, dann können wir die folgende Frage stellen: Kann man stabile Schrägkanten $e \in E_{diag}(S)$ daran erkennen, dass der Graph $[S^{tr}] - \hat{e}$ eine *reguläre H-Einbettung* eines Graphen $G^* \in \mathcal{C}$ enthält, d.h. eine H-Einbettung, die ausschließlich aus regulären Kanten besteht?

Für den Plan B_{34} auf Seite 114 gilt beispielweise, dass das Entfernen einer beliebigen der drei Schrägkanten zu einem induzierten $K_2 \times K_3$ führt. Der Graph $[B_{34}^{tr}] - \hat{e}$ enthält damit für jede Schrägkante $e \in E_{diag}(B_{34})$ eine reguläre H-Einbettung des Komplementgraphen von C_6 (Abbildung 1.6 auf Seite 26).

Leider ist der Komplementgraph von C_n für gerade n , $\bar{C}_n = K_2 \times K_{\frac{n}{2}}$, offenbar einer von nur sehr wenigen Graphen in \mathcal{C} , die eine solche reguläre H-Einbettung besitzen. Neben \bar{C}_n für gerade n besitzen offenbar nur noch \bar{L}_1 und \bar{L}'_1 überhaupt eine reguläre H-Einbettung.

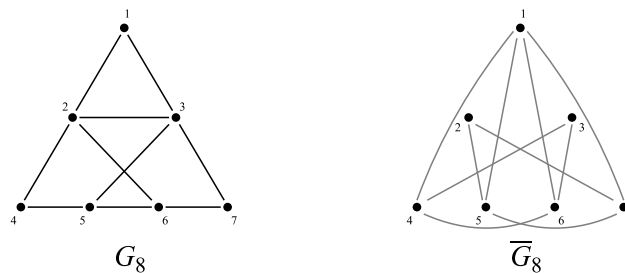


Abbildung 5.2: Der Graph G_8 und sein Komplement $\bar{G}_8 \in \mathcal{C}$ (vgl. Abbildung 1.7).

Hinzufügen einer Kante zu den Graphen aus \mathcal{C}

In einem alternativen Ansatz zur Ausnutzung dieser Eigenschaften von Comparabilitygraphen können wir versuchen, irreduzible Pläne über einer gegebenen Operationenmenge durch Einfügen von einzelnen Schrägkanten in H-Einbettungen von Graphen aus \mathcal{C} zu konstruieren. Wir können uns fragen, welche Eigenschaften ein Graph $[S^{tr}] - \hat{e}$ für einen irreduziblen Plan $S \in \mathcal{S}_{SIJ}$ und eine Schrägkante

$e \in E_{diag}(S)$ hat. Möglicherweise kann man von solchen Graphen Rückschlüsse auf irreduzible Pläne ziehen, deren H-Comparabilitygraphen diese Graphen enthalten. Die einfachsten Beispiele für solche Graphen liefern scheinbar die H-Einbettungen von Graphen $G^* \in \mathcal{C}$ selber.

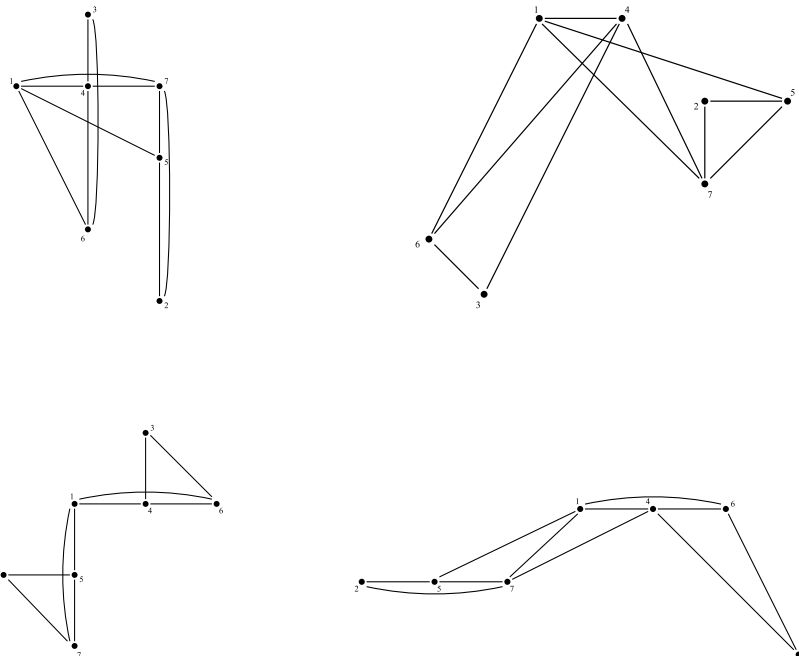


Abbildung 5.3: 4 von 928 verschiedenen nichtisomorphen H-Einbettungen von $\bar{G}_8 \in \mathcal{C}$.

In Abbildung 5.3 sind einige ausgewählte H-Einbettungen des Graphen $\bar{G}_8 \in \mathcal{C}$ (vgl. Abbildung 5.2) dargestellt. Während jedoch zu jeder dieser Einbettungen Schrägkanten so hinzugefügt werden können, dass daraus (ggf. nach Einführung weiterer Operationen) H-Comparabilitygraphen entstehen, können andererseits jeweils auch Schrägkanten entfernt werden, so dass die Pläne, die durch Hinzufügen einer Kante entstanden sind, nicht unbedingt irreduzibel sein müssen. Bei den beiden H-Einbettungen von \bar{G}_8 mit nur zwei Schrägkanten können beispielsweise jeweils beide Schrägkanten entfernt werden.

Andererseits existiert zu jeder Operationenmenge $SIJ \subseteq I \times J$ eine Menge von Graphen, die jeweils durch Hinzufügen von genau einer Kante zu Plangraphen von irreduziblen Plänen werden. Diese Graphen bestehen jeweils aus mehreren miteinander kombinierten Graphen aus \mathcal{C} . Die große Zahl der möglichen Kombinationen zu einer gegebenen Operationenmenge, verbunden mit der großen Anzahl möglicher H-Einbettungen jedes Einzelnen dieser Graphen—allein der Graph \bar{G}_8 besitzt 928 verschiedene nichtisomorphe H-Einbettungen—lässt eine Enumeration der irreduziblen Pläne auf diesem Weg allerdings wenig aussichtsreich erscheinen. Einen Ansatz für eine etwas zielgerichteteren Methode zur Konstruktion von irreduziblen Plänen durch das Hinzufügen von Kanten werden wir im Abschnitt 6.5 betrachten.

Bemerkung 5.10 (verbotene Teilgraphen) *Die Charakterisierung von Comparabilitygraphen durch eine endliche Liste (bei beschränkter Anzahl der Knoten) von verbotenen Teilgraphen ist weder bei der Erkennung von stabilen Schrägkanten, noch bei der Konstruktion von irreduziblen Plänen von großem Nutzen.*

5.5.3 Eigenschaften von Modulen

Bei der Beschreibung von Comparabilitygraphen beschränken wir uns in dieser Arbeit auf die “kantenorientierte” Darstellung durch Implikationsklassen, während bei der algorithmischen Bestimmung von transitiven Orientierungen die “knotenorientierte” Beschreibung durch die modulare Dekomposition (Abschnitt 1.4.2) eine bedeutende Rolle spielt. Es stellt sich daher die Frage: Existiert eine

Formulierung von IRRED mit Hilfe der modularen Dekomposition, die die Erkennung von stabilen Schrägkanten oder von irreduziblen Plänen in polynomiell beschränkter Zeit ermöglicht?

In Abschnitt 1.4.2 sind wir auf die enge Verzahnung zwischen der Beschreibung von Comparabilitygraphen durch Module und der Beschreibung durch Implikationsklassen eingegangen. Der Unterschied zwischen diesen beiden Varianten ist in erster Linie quantitativ. Mit Hilfe der modularen Dekomposition lässt sich eine transitive Orientierung schneller finden, als durch Bestimmung der Implikationsklassen anhand der Γ -Relation. Einen qualitativen Unterschied zwischen diesen beiden Beschreibungen gibt es nicht.

Als fundamentale Bausteine der Pläne zu Open-Shop Schedulingproblemen haben wir die Planimplikationsklassen beschrieben. Es ist grundsätzlich denkbar, dass sich diese Teilstrukturen der Implikationsklassen in Teilstrukturen von Modulen—z.B. in *Planmodule*—übersetzen lassen. Wir können jedoch nicht erwarten, dass eine solche alternative Beschreibung gegenüber der Beschreibung durch Planimplikationsklassen tiefere Einsichten liefert.

Bemerkung 5.11 (modulare Dekomposition) *Transitive Orientierungen können durch Implikationsklassen oder Module beschrieben werden. Da jeweils die gleiche Teilstruktur eines Comparabilitygraphen beschrieben wird, hat jede Eigenschaft in einer der beiden Beschreibungen ein Äquivalent in der jeweiligen anderen Sprache. Man kann nicht erwarten, dass eine Betrachtung von IRRED in einer Formulierung mit Hilfe von Modulen zu Einsichten führt, die durch eine Formulierung mit Hilfe von Planimplikationsklassen unzugänglich bleiben.*

Wir beschließen damit dieses Kapitel über wenig aussichtsreiche Ansätze zur Reduktion von Plänen.

Kapitel 6

Ein Reduktionsalgorithmus

Einer der Ansätze aus dem vorangegangenen Kapitel zur Reduktion von Plänen zielte darauf ab, in einem H-Comparabilitygraphen eine Menge entfernbarer Schrägkanten zu bestimmen (Abschnitt 5.5.1). Diese Idee wollen wir in diesem Kapitel weiter verfolgen.

Wenn ein Ausgangsplan $S \in \mathcal{S}_{SIJ}$ durch Verdrehen von einer Planimplikationsklasse $P_1 \in \mathcal{P}_S$ gegenüber einer Planimplikationsklasse $P_2 \in \mathcal{P}_S$ reduziert werden soll, dann müssen sämtliche Γ -Ketten, durch die die beiden in $[S^{tr}]$ möglicherweise verbunden sind, getrennt werden (Satz 4.59). Das liefert einen ersten Hinweis auf eine Menge von Schrägkanten, die entfernt werden muss, um durch Umkehren von P_1 oder P_2 zu einem reduzierenden Plan zu gelangen. Da die Entfernung einer auf diese Art und Weise bestimmten Kantenmenge aus $[S^{tr}]$ typischerweise nicht zu einem Comparabilitygraphen führt, bedarf es noch der Betrachtung einiger weiterer Begleitumstände, um zu einer *zulässigen Entfernungsmenge* zu gelangen. Der Beschreibung eines auf dieser Grundidee basierenden Verfahrens zur konstruktiven Reduktion eines Ausgangsplans dient dieses Kapitel.

Zur Beschreibung der angesprochenen Begleitumstände greifen wir den Begriff des bereits bekannten *Faktorgraphen* auf, und stellen ihm einem weiteren abgeleiteten Hilfsgraphen gegenüber—den *Konsequenzgraphen* (Abschnitt 6.1). Aus den Informationen, die diese beide Graphen liefern, leiten wir anschließend einen sogenannten *Reduktionsgraphen* ab (Abschnitt 6.2). Mit Hilfe dieser Tools entwickeln wir eine Strategie zur Bestimmung einer zulässigen Entfernungsmenge (Abschnitt 6.3), die wir in zwei Ausprägungen eines Reduktionsverfahrens einfließen lassen (Abschnitt 6.4).

Zuletzt diskutieren wir noch, inwiefern man auch durch sukzessives *Hinzufügen* von Schrägkanten zum leeren H-Graphen G_{SIJ} irreduzible Pläne erzeugen kann (Abschnitt 6.5), und einen daraus abgeleiteten Ansatz für eine Nachbarschaft zwischen irreduziblen Plänen (Abschnitt 6.6)

6.1 Faktorgraph und Konsequenzgraph

Das Ziel des in diesem Kapitel beschriebenen Verfahrens ist die Bestimmung einer Teilmenge M der Schrägkanten $E_{diag}(S)$ eines Plans $S \in \mathcal{S}_{SIJ}$, mit der Eigenschaft, dass $[S^{tr}] - M$ ein Comparabilitygraph ist. Bevor wir uns den aus einem H-Comparabilitygraphen $[S^{tr}]$ abgeleiteten Hilfsgraphen widmen, wollen wir jedoch kurz einige Sprechweisen im Zusammenhang mit solchen *Entfernungsmengen* festhalten.

6.1.1 Zulässige und zulässig erweiterbare Entfernungsmengen

Definition 6.1 (Entfernungsmenge) *Eine Entfernungsmenge eines H-Comparabilitygraphen $[S^{tr}]$ zu einem Plan $S \in \mathcal{S}_{SIJ}$ ist eine ungerichtete Teilmenge $M = \bar{M} \subseteq E_{diag}(S)$ der Schrägkantenmenge. Wir nennen M zulässig, wenn $[S^{tr}] - M$ ein H-Comparabilitygraph ist, und zulässig erweiterbar, wenn eine zulässige Entfernungsmenge M^* mit $M \subsetneq M^*$ existiert. Eine Entfernungsmenge, die nicht zulässig erweiterbar ist, heißt unzulässig.*

Man beachte, dass eine *nicht* zulässige Entfernungsmenge nicht unbedingt *unzulässig* ist. Andererseits kann eine Entfernungsmenge auch zulässig sein, und dennoch zulässig erweiterbar. Mit dem

Begriff der Entfernungsmenge folgt unmittelbar eine weitere Möglichkeit zur Formulierung des Problems IRRED.

Problem IRREDUCIBILITY (IRRED) (#5): Gegeben sei ein Plan $S \in \mathcal{S}_{SIJ}$ für das Open-Shop Problem $O \parallel C_{\max}$. Ist jede Entfernungsmenge $M \subseteq E_{diag}(S)$ in $[S^{tr}] = (SIJ, E_{reg}(S) + E_{diag}(S))$ unzulässig?

Die Antworten auf die Fragen “Existiert eine zulässige Entfernungsmenge in $[S^{tr}]$?” bzw. “Existiert eine zulässig erweiterbare Entfernungsmenge in $[S^{tr}]$?” liefern jeweils Antworten für co-IRRED.

6.1.2 Der Faktorgraph und seine Bedeutung

Zur Beschreibung der Zulässigkeit einer Entfernungsmenge ziehen wir die in $[S^{tr}]$ enthaltenen Informationen über die Rolle der einzelnen Schrägkanten heran. Vor dem Hintergrund der Zerlegung der regulären Kanten eines Plans in Planimplikationsklassen haben Schrägkanten zwei verschiedene Funktionen. Zum Einen führt ihre Existenz zu Γ -Ketten zwischen verschiedenen Planimplikationsklassen. Und zum Anderen verhindert ihre Existenz das Entstehen von Γ_S -Relationen zwischen regulären Kanten aus verschiedenen Planimplikationsklassen.

Der Faktorgraph $G_{\mathcal{F}}$

Zur Beschreibung der ersten dieser beiden Funktionen haben wir in Abschnitt 4.2.3 bereits den Faktorgraphen $G_{\mathcal{F}} = G_{\mathcal{F}}(S)$ zu einem Plan $S \in \mathcal{S}_{SIJ}$ kennen gelernt. Als Knoten enthält $G_{\mathcal{F}}$ die gerichteten Schrägkanten von $[S^{tr}]$, sowie die erweiterten Planimplikationsklassen aus \mathcal{P}_S und $\mathcal{P}_{S^{-1}}$. Eine (ungerichtete) Kante zwischen zwei Knoten in $G_{\mathcal{F}}$ existiert genau dann, wenn zwischen den jeweils repräsentierten Kanten oder Kantenmengen in $[S^{tr}]$ eine Γ -Relation besteht. Der Faktorgraph ist insbesondere symmetrisch bezüglich der Umkehrung der betrachteten Kanten oder Kantenmengen aus $[S^{tr}]$.

Lemma 6.2 (Anzahl der Knoten und Kanten in $G_{\mathcal{F}}$) Sei $G_{\mathcal{F}} = (V_{\mathcal{F}}, E_{\mathcal{F}})$ der Faktorgraph zu einem Plan $S \in \mathcal{S}_{SIJ}$ über einer Operationenmenge SIJ mit n Aufträgen und m Maschinen. Dann gelten $|V_{\mathcal{F}}| = \mathcal{O}(n^2m^2)$ und $|E_{\mathcal{F}}| = \mathcal{O}(n^3m^3)$.

Beweis. Die Anzahl der Knoten in $G_{\mathcal{F}}$ ist beschränkt durch die Anzahl der Kanten in $[S^{tr}]$, $|V_{\mathcal{F}}| \leq |E_{[S^{tr}]}| = |E_{reg}(S)| + |E_{diag}(S)| \leq [n\binom{m}{2} + m\binom{n}{2}] + [\frac{nm}{2}(n-1)(m-1)] = \mathcal{O}(n^2m^2)$. Die Anzahl der Kanten in $G_{\mathcal{F}}$ ist bestimmt durch die Anzahl der Γ -Relationen in $[S^{tr}]$. Da zwei Knoten in $[S^{tr}]$ maximal $nm - 2$ gemeinsame Nachbarn haben können, kann jede Nichtkante in $[S^{tr}]$ maximal $nm - 2$ verschiedene Γ -Relationen induzieren. Die Anzahl der möglichen Nichtkanten ist durch die Anzahl der möglichen Schrägkanten, $\frac{nm}{2}(n-1)(m-1)$, beschränkt. Damit gilt $|E_{\mathcal{F}}| \leq [nm - 2] \cdot [\frac{nm}{2}(n-1)(m-1)] = \mathcal{O}(n^3m^3)$. ■

In der Abschätzung der Anzahl der Γ -Relationen in $[S^{tr}]$ können nicht beide Faktoren gleichzeitig maximal bzw. minimal sein, da jede mögliche Schrägkante entweder eine Nichtkante, oder aber eine Schrägkante in $[S^{tr}]$ ist, und daher nur zu genau einem der beiden Faktoren einen Beitrag leistet. Dennoch ist eine quadratische Abschätzung der Anzahl der Γ -Relationen, und damit der Anzahl der Kanten in $G_{\mathcal{F}}$, nicht möglich.

Der Faktorgraph und Entfernungsmengen

Eine zulässige Entfernungsmenge, die einen Ausgangsplan $S \in \mathcal{S}_{SIJ}$ durch das Umkehren einer Planimplikationsklasse P_1 gegenüber einer anderen Planimplikationsklasse P_2 aus der gleichen Implikationsklasse reduzieren soll, muss aus jeder Γ -Kette in $[S^{tr}]$ zwischen P_1 und P_2 mindestens eine Schrägkante enthalten. Jeder Γ -Weg zwischen P_1 und P_2 muss unterbrochen werden (Satz 4.59). Haben wir nun eine Entfernungsmenge $M \subseteq E_{diag}(S)$ gegeben, die dazu führt, dass P_1 gegenüber P_2 verdreht wird, dann können wir aus dem Faktorgraphen ablesen, ob P_1 und P_2 in $[S^{tr}] - M$ nicht nach wie vor durch einen Γ -Weg verbunden sind. Solange P_1 und P_2 in $G_{\mathcal{F}} - M$ in einer Zusammenhangskomponente liegen, sind nicht alle Γ -Wege zerstört. Mit $G_{\mathcal{F}} - M$ bezeichnen wir dabei den Faktorgraphen,

aus dem alle Knoten entfernt wurden, die die durch $M \subseteq E_{diag}(S)$ beschriebenen Schrägkanten repräsentieren. In Anlehnung an Γ -Graphen können wir diesen Zusammenhang mit dem Begriff der *Konsistenz* beschreiben (Abschnitt 1.2.4): Der Graph $G_{\mathcal{F}} - M$ heißt *konsistent*, wenn $K \cap K^{-1} = \emptyset$ für jede Zusammenhangskomponente K gilt, wobei mit K^{-1} die Menge der Knoten beschrieben sei, die die Umkehrkanten in $[S^{tr}]$ der Knoten aus K repräsentieren (Abschnitt 4.2.3).

Konstruktion und Darstellung von $G_{\mathcal{F}}$

Ein Beispiel für einen Faktorgraphen und seine Darstellung ist auf Seite 123 angeführt (Beispiel 6.6, Abbildung 6.2). Bei der Darstellung von $G_{\mathcal{F}}$ werden wir typischerweise auf die Darstellung des gesamten Graphen verzichten, und uns auf die Darstellung aller nichtsymmetrischen Knoten und Kanten beschränken. Ferner werden wir die Umkehrmenge einer Kantenmenge P aus Gründen der besseren Darstellung häufig auch mit \bar{P} statt P^{-1} bezeichnen.

Bemerkung 6.3 (Konstruktion von $G_{\mathcal{F}}$) *Der Faktorgraph $G_{\mathcal{F}} = G_{\mathcal{F}}(S)$ zu einem Plan $S \in \mathcal{S}_{SIJ}$ enthält Informationen über die Γ -Relationen in einem Comparabilitygraphen. Er kann daher genau wie der Γ -Graph in polynomieller Laufzeit bestimmt werden. Auch die Überprüfung eines Teilgraphen $G_{\mathcal{F}} - M$ auf Konsistenz ist in polynomieller Zeit durchführbar.*

6.1.3 Der Konsequenzgraph und seine Bedeutung

Während der Faktorgraph $G_{\mathcal{F}}$ Informationen über Γ -Wege in einem H-Comparabilitygraphen $[S^{tr}]$ liefert, können wir die Informationen über die durch die Existenz der einzelnen Schrägkanten verhinderten Γ_S -Relationen durch dem sogenannten Konsequenzgraphen darstellen.

Konsequenzen aus dem Entfernen von Schrägkanten

Durch das Entfernen von Kanten entstehen in aller Regel neue Γ -Relationen. Der Konsequenzgraph gibt Auskunft darüber, zwischen welchen Schrägkanten oder Planimplikationsklassen neue Γ -Relationen durch die Entfernung einer Schrägkante $e_i \in E_{diag}(S)$ entstehen.

Definition 6.4 (Konsequenzgraph) *Sei $[S^{tr}] = (SIJ, E_{reg}(S) + E_{diag}(S))$, mit $|E_{diag}(S)| = d$, der H-Comparabilitygraph zu einem Plan $S \in \mathcal{S}_{SIJ}$, und sei $\mathcal{P}_S = \{P_1, \dots, P_k\}$ die Menge der Planimplikationsklassen von S , $E_{reg}(S) = P_1 + \dots + P_k + P_1^{-1} + \dots + P_k^{-1}$. Der Konsequenzgraph $G_{\mathcal{K}} = G_{\mathcal{K}}(S) = (V_{\mathcal{K}}, E_{\mathcal{K}})$ ist ein gefärbter Multigraph, dessen Knotenmenge aus der Menge der erweiterten Planimplikationsklassen von S und S^{-1} und aus der Menge der verbleibenden gerichteten Schrägkanten $E_{diag}(S)$ besteht, $V_{\mathcal{K}} \subseteq E_{diag}(S) + \mathcal{P}_S + \mathcal{P}_{S^{-1}}$. Zwei Knoten e' und e'' aus $V_{\mathcal{K}}$ sind in $G_{\mathcal{K}}$ genau dann durch eine ungerichtete Kante der Farbe $i \in \{1, \dots, d\}$ verbunden, wenn die Entfernung der Schrägkante $e_i \in E_{diag}(S)$ aus $[S^{tr}]$ eine Γ -Relation zwischen den durch e' bzw. e'' repräsentierten Schrägkanten oder Planimplikationsklassen induziert,*

$$E_{\mathcal{K}} = \left\{ \widehat{e'e''} \text{ der Farbe } i: e'\Gamma e'' \text{ in } [S^{tr}] - e_i, e_i \in E_{diag}(S) \right\}.$$

Mit $G_{\mathcal{K}_i}$ bezeichnen wir den Teilgraphen von $G_{\mathcal{K}}$, der alle mit i gefärbten Kanten enthält. Es gilt $G_{\mathcal{K}} = G_{\mathcal{K}_1} + \dots + G_{\mathcal{K}_d}$.

Der Konsequenzgraph gibt damit insbesondere darüber Auskunft, welche Planimplikationsklassen durch die Entfernung einer Schrägkante $\hat{e}_i \in E_{diag}(S)$ unter Umständen miteinander *verschmelzen*, d.h. welche *Konsequenzen* die Entfernung von \hat{e}_i nach sich zieht. Ein Beispiel für einen Konsequenzgraphen und seine Darstellung ist im Beispiel 6.6 auf Seite 123 angeführt (Abbildung 6.2).

Das Verschmelzen zweier Planimplikationsklassen durch die Entfernung von \hat{e}_i nennen wir auch eine *direkte* Konsequenz. Trivial-stabile Schrägkanten werden als Kanten ihrer jeweiligen (erweiterten) Planimplikationsklassen behandelt. Da sie in keinem Fall entfernt werden können, unterscheiden sie sich nicht von regulären Kanten. Der Konsequenzgraph ist genau wie der Faktorgraph symmetrisch bezüglich der Umkehrung der betrachteten Kanten oder Kantenmengen aus $[S^{tr}]$.

Ferner gelten für die Anzahl der Knoten und Kanten von $G_{\mathcal{K}}$ jeweils dieselben Abschätzungen wie beim Faktorgraphen.

Lemma 6.5 (Anzahl der Knoten und Kanten in $G_{\mathcal{K}}$) Sei $G_{\mathcal{K}} = (V_{\mathcal{K}}, E_{\mathcal{K}})$ der Konsequenzgraph zu einem Plan $S \in \mathcal{S}_{SIJ}$ über einer Operationenmenge SIJ mit n Aufträgen und m Maschinen. Dann gelten $|V_{\mathcal{K}}| = \mathcal{O}(n^2m^2)$ und $|E_{\mathcal{K}}| = \mathcal{O}(n^3m^3)$. Für die Teilgraphen $G_{\mathcal{K}_e}$ mit $e \in E_{diag}(S)$ gelten $|V_{\mathcal{K}_e}| = |V_{\mathcal{K}}|$ und $|E_{\mathcal{K}_e}| = \mathcal{O}(nm)$.

Beweis. Die Anzahl der Knoten in $G_{\mathcal{K}}$ ist beschränkt durch die Anzahl der Kanten in $[S^{tr}]$, $|V_{\mathcal{K}}| \leq |E_{[S^{tr}]}| = |E_{reg}(S)| + |E_{diag}(S)| \leq [n \binom{m}{2} + m \binom{n}{2}] + [\frac{nm}{2}(n-1)(m-1)] = \mathcal{O}(n^2m^2)$. Die Anzahl der Kanten in $G_{\mathcal{K}}$ ist bestimmt durch die Anzahl der Dreiecke in $[S^{tr}]$, die mindestens eine Schrägkante enthalten. Es gilt $G_{\mathcal{K}} = \bigcup_{e \in E_{diag}(S)} G_{\mathcal{K}_e}$. Da zwei Knoten in $[S^{tr}]$ maximal $nm - 2$ gemeinsame Nachbarn haben können, kann jede Schrägkante in $[S^{tr}]$ in maximal $nm - 2$ verschiedenen Dreiecken liegen. Also gilt $|E_{\mathcal{K}_e}| = \mathcal{O}(nm)$. Mit $|E_{diag}(S)| \leq \frac{nm}{2}(n-1)(m-1) = \mathcal{O}(n^2m^2)$ folgt dann $|E_{\mathcal{K}}| \leq \mathcal{O}(n^2m^2) \cdot |E_{\mathcal{K}_e}| = \mathcal{O}(n^3m^3)$. ■

Kontraktion der Planimplikationsklassen und Multikanten

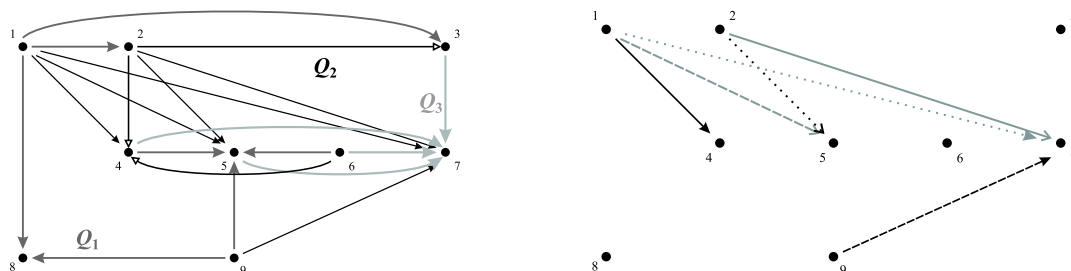
Eine neue Γ -Relation zwischen zwei beliebigen Kanten aus $[S^{tr}]$ kann jeweils höchstens durch die Entfernung einer einzigen Kante aus $[S^{tr}]$ induziert werden. Würden wir die Knotenmenge von $G_{\mathcal{K}}$ also abweichend als die Menge aller Kanten aus $[S^{tr}]$ annehmen, dann wäre $G_{\mathcal{K}}$ kein Multigraph. Multikanten in $G_{\mathcal{K}}$ entstehen dadurch, dass unter Umständen die Entfernung mehrerer verschiedener Schrägkanten aus $[S^{tr}]$ jeweils unabhängig voneinander eine Γ -Relation zwischen einer Planimplikationsklasse und einer Schrägkante oder einer anderen Planimplikationsklasse induziert.

Durch die Kontraktion aller Kanten einer erweiterten Planimplikationsklasse zu einem einzelnen Knoten in $G_{\mathcal{K}}$ verlieren wir die Übersicht, welche Kanten dieser erweiterten Planimplikationsklasse mit welchen Schrägkanten oder mit welchen Kanten aus anderen Planimplikationsklassen in Γ -Relation liegen. Da (erweiterte) Planimplikationsklassen durch das Entfernen von Schrägkanten jedoch nicht zerstört werden können— Γ_S -Relationen können dadurch nicht unterbrochen werden (Lemma 4.24 bzw. Satz 4.40)—sind diese differenzierten Informationen für uns jedoch nicht weiter relevant. Uns interessiert lediglich, ob beispielsweise eine Schrägkante, mit *irgendeiner* Kante aus einer bestimmten Planimplikationsklasse in Γ -Relation steht, oder nicht.

Konstruktion und Darstellung von $G_{\mathcal{K}}$

Betrachten wir zur Veranschaulichung der Begriffe und Darstellungsmöglichkeiten ein Beispiel. Bei der Darstellung von $G_{\mathcal{K}}$ verzichten wir genau wie bei der Darstellung von $G_{\mathcal{F}}$ aus Gründen der Übersichtlichkeit auf die Darstellung von symmetrischen Kanten, und bezeichnen Umkehrmengen abweichend durch \bar{P} statt P^{-1} .

Da die Darstellung des Konsequenzgraphen für größere Beispiele sehr schnell unübersichtlich wird, werden wir diesen Graphen häufig durch eine ‘Farbliste’ darstellen, bei der wir jeder Farbe (d.h. jeder nichttrivial-stabilen Schrägkante des Ausgangsplans) die entsprechend gefärbten Kanten zuordnen. Diese Kanten in $G_{\mathcal{K}}$ beschreiben wir durch die jeweils beschriebenen Γ -Relationen. Die Verschmelzung von Planimplikationsklassen und Schrägkanten machen wir hierbei jedoch nicht durch $P_i \Gamma P_j$, sondern der Übersicht halber durch $P_i + P_j$ deutlich. Der Vorzug dieser Notation wird sich im Zusammenhang mit dem Reduktionsgraphen erschließen.



Abbildungung 6.1: Der Plan $A_{48} \prec A$ zu dem Plan A aus Beispiel 4.44 und seine Schrägkantenmenge.

Beispiel 6.6 (Faktorgraph und Konsequenzgraph) Wir betrachten erneut den bekannte Plan $A \in \mathcal{S}_{SIJ}$ aus Beispiel 4.8 bzw. 4.44 und seine reduzierende Pläne. Hier wollen wir den Plan A_{48} betrachten, der aus A durch Umkehren von P_4 und P_8 entsteht (Abbildung 6.1).

$$A_{48} = \begin{bmatrix} 1 & 2 & & 3 \\ & 3 & 4 & 1 & 5 \\ 2 & & 1 & & \end{bmatrix}$$

A_{48} besteht aus drei verschiedenen Planimplikationsklassen $Q_1 = P_1 + P_2 + \bar{P}_4 + P_9$, $Q_2 = P_3 + P_{10}$ und $Q_3 = P_5 + P_6 + P_7 + \bar{P}_8$. Der H -Comparabilitygraph $[A_{48}^{tr}]$ enthält die sechs Schrägkanten 1.4, 1.5, 1.7, 2.5, 2.7 und 9.7. Da alle Planimplikationsklassen von A_{48} im Faktorgraphen in einer gemeinsamen Komponente liegen (Abbildung 6.2), ist $[A_{48}^{tr}]$ prim. Alle drei Planimplikationsklassen liegen in einer gemeinsamen Implikationsklasse. A_{48} ist damit insbesondere auch ein normaler Plan. Von den 15 irreduziblen Plänen zu A sind nur die Pläne $A_{348.10}$, A_{4567} und A_{1289} auch reduzierende Pläne von A_{48} (Abbildung 4.8). Will man nun beispielsweise Q_1 gegenüber Q_2 umkehren, dann folgt aus $G_{\mathcal{F}}$

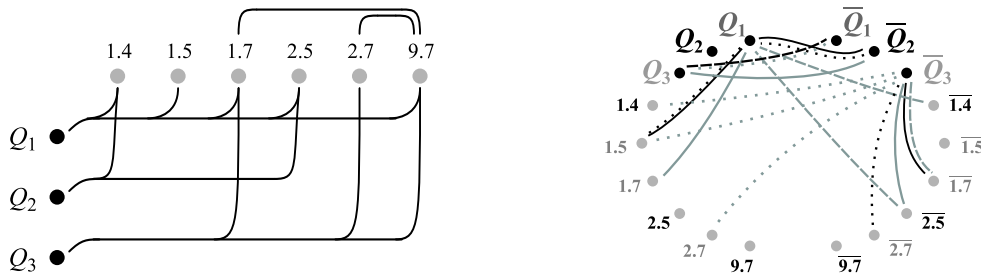


Abbildung 6.2: Der Faktorgraph $G_{\mathcal{F}}$ (links) und der Konsequenzgraph $G_{\mathcal{K}}$ zu dem Plan A_{48} . Wegen der Symmetrie zwischen A_{48} und A_{48}^{-1} sind bei beiden Graphen aus Gründen der Übersichtlichkeit nur die Hälfte aller Kanten dargestellt.

sofort, dass die Schrägkanten 1.4 und 2.5, die beide in der trennenden Kantenmenge $F_{1,2}$ zwischen Q_1 und Q_2 liegen, zwingend zur Entfernungsmenge gehören müssen. Aus $G_{\mathcal{K}}$ können wir dann ablesen, dass die Entfernung von $\bar{1.4}$ die Verschmelzung von Q_1 mit \bar{Q}_2 , und Γ -Relationen zwischen Q_1 und 1.5, sowie zwischen \bar{Q}_3 und $\bar{1.7}$ (bzw. Q_3 und 1.7) nach sich zieht (Abbildung 6.2 oder Tabelle 6.1). Die beiden letzteren sind dabei keine neue Informationen. Bereits in $[A_{48}^{tr}]$ liegen Q_1 und 1.5, bzw. Q_3 und 1.7 jeweils in Γ -Relation (vergleiche Faktorgraph $G_{\mathcal{F}}$). ■

1.4	$Q_1 + \bar{Q}_2$	$Q_1 + 1.5$	$Q_3 + 1.7$	
1.5		$\bar{Q}_1 + 1.4$	$\bar{Q}_1 + 2.5$	
1.7	$Q_1 + \bar{Q}_3$	$\bar{Q}_3 + 1.4$	$\bar{Q}_3 + 1.5$	$Q_1 + 2.7$
2.5	$Q_1 + \bar{Q}_2$	$Q_1 + 1.5$	$Q_3 + 2.7$	
2.7	$Q_2 + \bar{Q}_3$	$Q_1 + 1.7$	$\bar{Q}_3 + 2.5$	
9.7	$Q_1 + \bar{Q}_3$			

Tabelle 6.1: Der Konsequenzgraph $G_{\mathcal{K}}$ zu dem Plan A_{48} in tabellarischer Darstellung. Angegeben sind die Konsequenzen (rechts), die die Entfernung der jeweiligen Kanten (links) aus $[A_{48}^{tr}]$ nach sich ziehen. Wegen der Symmetrie der Γ -Relation ist jeweils nur eine induzierte Γ -Relation angegeben.

Bemerkung 6.7 (Konstruktion von $G_{\mathcal{K}}$) Der Konsequenzgraph $G_{\mathcal{K}} = G_{\mathcal{K}}(S)$ zu einem Plan $S \in \mathcal{S}_{SIJ}$ enthält Informationen über die Kanten zu allen gemeinsamen Nachbarknoten der jeweiligen Endknoten einer gegebenen Kante aus $[S^{tr}]$. Diese Informationen können bei der Überprüfung der Bedingung der Γ -Relation ohne zusätzlichen Aufwand mit ausgelesen werden. Der Konsequenzgraph kann damit insbesondere in polynomieller Zeit bestimmt werden. Auch die Überprüfung der Konsequenzen aus dem Entfernen einer gegebenen Schrägkante aus $[S^{tr}]$ ist in polynomieller Zeit durchführbar, da die Anzahl der möglichen Konsequenzen durch die Anzahl der Kanten in $[S^{tr}]$ begrenzt ist.

6.2 Der Reduktionsgraph

Der Faktorgraph $G_{\mathcal{F}}(S)$ und der Konsequenzgraph $G_{\mathcal{K}}(S)$ zu einem Plan $S \in \mathcal{S}_{SIJ}$ vermitteln uns ein genaues Bild von den für uns wichtigen Eigenschaften jeder einzelnen Schrägkante. Wir können ablesen, ob eine Schrägkante in einer trennenden Kantenmenge zwischen zwei gegebenen Planimplikationsklassen liegt, oder ob ihre Existenz die Verschmelzung von einer Planimplikationsklasse mit der Umkehrung einer anderen verhindert. Im Falle von nichtnormalen Plänen können wir sogar erkennen, ob in $[S^{tr}]$ Implikationsklassen existieren, die ausschließlich aus Schrägkanten bestehen.

6.2.1 Der Begriff des Reduktionsgraphen

Die beiden Graphen $G_{\mathcal{F}}$ und $G_{\mathcal{K}}$ liefern uns damit alle Informationen, um eine gegebene Entfernungsmenge $M \subseteq E_{diag}(S)$ auf ihre Zulässigkeit zu überprüfen. Dies liefert uns andererseits jedoch auch ein Test auf Existenz einer transitiven Orientierung auf $[S^{tr}] - M$. Und das sogar sehr viel schneller. Was uns ein solcher Test jedoch nicht liefert, ist zum Einen ein Hinweis darauf, wie eine nicht zulässige, aber zulässig erweiterbare Entfernungsmenge M erweitert werden kann bzw. muss. Und zum Anderen kann uns ein reiner Nachweis, dass $[S^{tr}] - M$ kein Comparabilitygraph ist, keinen Beweis dafür liefern, dass M nicht nur nicht zulässig, sondern möglicherweise sogar unzulässig, d.h. auch nicht mehr zulässig erweiterbar ist.

Um auch diese Fragen beantworten zu können, konstruieren wir einen weiteren Hilfsgraphen, aus dem sich solche Informationen zu einer gegebenen Entfernungsmenge ablesen lassen.

Definition 6.8 (Reduktionsgraph) Sei $[S^{tr}] = (SIJ, E_{reg}(S) + E_{diag}(S))$ der H -Comparabilitygraph zu einem Plan $S \in \mathcal{S}_{SIJ}$. Sei $G_{\mathcal{F}}$ der Faktorgraph, und $G_{\mathcal{K}}$ der Konsequenzgraph zu S . Für eine Entfernungsmenge $M \subseteq E_{diag}(S)$ entsteht der Reduktionsgraph $G_{\mathcal{R}_M} = G_{\mathcal{R}_M}(S) = (V_{\mathcal{R}_M}, E_{\mathcal{R}_M})$ aus $G_{\mathcal{F}}$ durch Hinzufügen aller Kanten aus $G_{\mathcal{K}}$, die mit Farben aus M gefärbt sind, und anschließendem Entfernen sämtlicher Knoten, die Kanten aus M repräsentieren,

$$G_{\mathcal{R}_M} = \left[G_{\mathcal{F}} + \bigcup_{e \in M} G_{\mathcal{K}_e} \right] - M,$$

wobei mit $G_{\mathcal{K}_e}$ der Teilgraph von $G_{\mathcal{K}}$ bezeichnet sein soll, der die mit $e \in E_{diag}(S)$ gefärbten Kanten enthält.

Als Ausgangspunkt für die Beschreibung von $[S^{tr}] - M$ benutzen wir also den Faktorgraphen $G_{\mathcal{F}}(S) = G_{\mathcal{F}}([S^{tr}])$. In diesen Graphen übertragen wir anschließend sämtliche Konsequenzen, die durch die Entfernung von M aus $[S^{tr}]$ ausgelöst werden. Diese Konsequenzen bestehen zum Einen in der Entfernung von Kanten aus $[S^{tr}]$ (Löschen von Knoten aus $G_{\mathcal{F}}$), und zum Anderen in der dadurch induzierten Entstehung neuer Γ -Relationen (Hinzufügen von Kanten zwischen den verbleibenden Knoten in $G_{\mathcal{F}}$). Entstehen dabei neue Γ -Relationen zwischen Planimplikationsklassen, d.h. führt die Entfernung einiger Kanten aus M zu der Verschmelzung von Planimplikationsklassen, dann können wir die entsprechenden Knoten in $G_{\mathcal{F}}$, die diese Planimplikationsklassen repräsentieren zweckmäßigerweise jeweils zu einem gemeinsamen Knoten kontrahieren.

Die Anzahl der Knoten von $G_{\mathcal{R}_M}$ nimmt nicht zuletzt durch solche Operationen für wachsende M ab. Dennoch gelten für die Anzahl der Knoten und Kanten des Reduktionsgraphen die gleichen Abschätzungen wie für den Faktorgraphen und den Konsequenzgraphen.

Lemma 6.9 (Anzahl der Knoten und Kanten in $G_{\mathcal{R}_M}$) Sei $G_{\mathcal{R}_M} = (V_{\mathcal{R}_M}, E_{\mathcal{R}_M})$ der Reduktionsgraph zu einem Plan $S \in \mathcal{S}_{SIJ}$ über einer Operationenmenge SIJ mit n Aufträgen und m Maschinen. Dann gelten $|V_{\mathcal{R}_M}| = \mathcal{O}(n^2m^2)$ und $|E_{\mathcal{R}_M}| = \mathcal{O}(n^3m^3)$.

Beweis. Nach Konstruktion ist die Anzahl der Knoten des Reduktionsgraphen beschränkt durch die Anzahl der Knoten des Faktorgraphen bzw. des Konsequenzgraphen, $|V_{\mathcal{R}_M}| \leq |V_{\mathcal{F}}| = |V_{\mathcal{K}}|$. Mit Lemma 6.2 bzw. 6.5 gilt also $|V_{\mathcal{R}_M}| = \mathcal{O}(n^2m^2)$. Die Anzahl der Kanten von $G_{\mathcal{R}_M}$ ist beschränkt durch die Summe der Anzahl der Kanten von $G_{\mathcal{F}}$ und $G_{\mathcal{K}}$, $|E_{\mathcal{R}_M}| \leq |E_{\mathcal{F}}| + |E_{\mathcal{K}}| = \mathcal{O}(n^3m^3) + \mathcal{O}(n^3m^3) = \mathcal{O}(n^3m^3)$. ■

Reduktionsgraphen und Faktorgraphen

Der Reduktionsgraph enthält damit sämtliche Informationen über die Planimplikationsklassen in $[S^{tr}] - M$ und die Γ -Relationen zwischen den verbleibenden Schrägkanten untereinander und zwischen ihnen und den verbleibenden Planimplikationsklassen. Er ist damit identisch zu dem Faktorgraphen $G_{\mathcal{F}}$ von $[S^{tr}] - M$.

Lemma 6.10 (Reduktionsgraph und Faktorgraph) *Sei $G_{\mathcal{R}_M}$ der Reduktionsgraph zu einer Entfernungsmenge $M \subseteq E_{diag}(S)$ aus einem H-Comparabilitygraph $[S^{tr}] = (SIJ, E_{reg}(S) + E_{diag}(S))$ zu einem Plan $S \in \mathcal{S}_{SIJ}$. Dann gilt $G_{\mathcal{R}_M} = G_{\mathcal{F}}([S^{tr}] - M)$.*

Beweis. Der Graph $[S^{tr}] - M$ ist ein H-Graph mit einer beliebigen Schrägkantenmenge. Nach Konstruktion enthält $G_{\mathcal{R}_M}$ genau die Informationen über $[S^{tr}] - M$, die der Faktorgraph $G_{\mathcal{F}}$ abbildet (Definition 4.31). ■

Konsistente Reduktionsgraphen und zulässige Entfernungsmengen

Wir können damit auch den Begriff der *Konsistenz* des Faktorgraphen auf den Reduktionsgraphen übertragen. Der Reduktionsgraph ist *konsistent*, wenn keine Zusammenhangskomponente in $G_{\mathcal{R}_M}$ zwei Knoten enthält, die jeweilige Umkehrmengen in $[S^{tr}] - M$ repräsentieren, d.h. wenn $K \cap K^{-1} = \emptyset$ für alle Komponenten K gilt.

Satz 6.11 (Reduktionsgraph) *Sei $[S^{tr}] = (SIJ, E_{reg}(S) + E_{diag}(S))$ der H-Comparabilitygraph zu einem Plan $S \in \mathcal{S}_{SIJ}$, sei $M \subseteq E_{diag}(S)$ eine Entfernungsmenge, und sei $G_{\mathcal{R}_M}$ der Reduktionsgraph von S bezüglich M . Dann sind die folgenden Aussagen äquivalent:*

- (i) M ist zulässig
- (ii) $[S^{tr}] - M$ ist ein Comparabilitygraph
- (iii) $G_{\mathcal{R}_M}$ ist konsistent.

Beweis. Nach Definition 6.1 ist eine Entfernungsmenge M von $[S^{tr}]$ genau dann zulässig, wenn der Graph $[S^{tr}] - M$ ein H-Comparabilitygraph ist. Der Reduktionsgraph $G_{\mathcal{R}_M}$ ist der Faktorgraph von $[S^{tr}] - M$ (Lemma 6.10). Der Faktorgraph ist genau dann konsistent, wenn der Γ -Graph, aus dem er abgeleitet wurde, konsistent ist (Definition 4.31). Der Γ -Graph von $[S^{tr}] - M$ ist schließlich genau dann konsistent, wenn $[S^{tr}] - M$ ein Comparabilitygraph ist (Folgerung 1.32) ■

Bemerkung 6.12 (Konstruktion von $G_{\mathcal{R}_M}$) *Da der Reduktionsgraph $G_{\mathcal{R}_M}$ von einem Plan $S \in \mathcal{S}_{SIJ}$ bezüglich einer Entfernungsmenge M nichts anderes als der Faktorgraph von $[S^{tr}] - M$ ist, ist er insbesondere auch polynomiell bestimmbar (mit Bemerkung 6.3). Auch die Überprüfung von $G_{\mathcal{R}_M}$ auf Konsistenz ist in polynomieller Zeit durchführbar, da es hierzu genügt, für jeden Knoten $v \in V_{\mathcal{R}_M}$ mittels Breiten- oder Tiefensuche zu überprüfen, ob der Knoten v^{-1} in derselben Komponente wie v liegt.*

6.2.2 Konflikte im Reduktionsgraphen

Wir wissen nun also, woran wir eine zulässige Entfernungsmenge erkennen können. Aber wie entscheiden wir für eine nicht zulässige, aber zulässig erweiterbare Entfernungsmenge, welche weiteren Schrägkanten wir zu dieser Menge hinzufügen müssen, um zu einer zulässigen Entfernungsmenge zu gelangen? Hierzu wollen wir den Begriff eines Konflikts im Reduktionsgraphen einführen. Es sei daran erinnert, dass wir im Zusammenhang mit dem Faktorgraphen die Knoten mit den jeweils repräsentierten Kanten oder Kantenmengen aus $[S^{tr}]$ identifizieren können. Das gleiche gilt analog auch für den Konfliktgraphen und den Reduktionsgraphen.

Definition 6.13 (Konflikt l-ter Ordnung) *Sei $G_{\mathcal{R}_M} = (V_{\mathcal{R}_M}, E_{\mathcal{R}_M})$ der Reduktionsgraph zu einem Plan $S \in \mathcal{S}_{SIJ}$ bezüglich einer Entfernungsmenge $M \subseteq E_{diag}(S)$. Existiert in $G_{\mathcal{R}_M}$ ein Weg $W \subseteq V_{\mathcal{R}_M}$ von einer erweiterten Planimplikationsklasse $P_i \in V_{\mathcal{R}_M}$ zu ihrer Umkehrmenge $P_i^{-1} \in V_{\mathcal{R}_M}$, dann beschreibt W einen Konflikt ist $G_{\mathcal{R}_M}$. Ist W ein inklusions-minimaler Weg, dann nennen wir die Anzahl $l \geq 0$ der enthaltenen Schrägkanten die Ordnung dieses Konflikts. Konflikte der Ordnung $l = 0$ nennen wir direkte Konflikte. Ein Reduktionsgraph, der keine Konflikte enthält, heißt konfliktfrei.*

Die Auflösung von Konflikten

Jeder Konflikt in $G_{\mathcal{R}_M}$ spiegelt einen Γ -Weg in $[S^{tr}] - M$ von einer erweiterten Planimplikationsklasse P_i zu ihrer Umkehrung P_i^{-1} wider. Soll M zu einer zulässigen Entfernungsmenge erweitert werden, dann muss jeder einzelne dieser Konflikte ‐aufgelöst‐ werden. D.h. jeder dieser Γ -Wege muss unterbrochen werden. Wir haben bereits mehrfach festgestellt, dass eine Γ -Relation zwischen zwei Kanten in einem Graphen durch das Entfernen von Kanten aus diesem Graphen nur genau dann zerstört werden kann, wenn mindestens eine der beiden Kanten zu den entfernten Kanten gehört. Da reguläre Kanten nicht aus $[S^{tr}] - M$ entfernt werden dürfen, und die Planimplikationsklassen hinsichtlich der relativen Orientierungen ihrer Kanten unveränderlich sind, können Konflikte in Reduktionsgraphen somit ausschließlich durch das Entfernen von beteiligten Schrägkanten aufgelöst werden.

Satz 6.14 (Auflösen von Konflikten) *Sei durch $W \subseteq V_{\mathcal{R}_M}$ ein Konflikt l -ter Ordnung in einem Reduktionsgraphen $G_{\mathcal{R}_M}$ zu einem Plan $S \in \mathcal{S}_{SIJ}$ beschrieben, und sei $\{e_1, \dots, e_l\} \subseteq W$ die Menge der Schrägkanten $e_i \in E_{diag}(S)$ in W . Ist M eine zulässig erweiterbare Entfernungsmenge, und $M^* \subseteq E_{diag}(S)$ eine zulässige Erweiterung von M , $M \subseteq M^*$, dann gilt $e_i \in M^*$ für mindestens ein $i \in \{1, \dots, l\}$.*

Beweis. Da M einen Konflikt enthält, ist M unzulässig. Ist M jedoch zulässig erweiterbar, dann muss der durch $W \subseteq V_{\mathcal{R}_M}$ beschriebene Konflikt in jeder zulässigen Erweiterung M^* , mit $M \subseteq M^*$, aufgelöst sein. Da ein Γ -Weg in $[S^{tr}] - M$ von einer erweiterten Planimplikationsklasse P_i zu ihrer Umkehrung P_i^{-1} ausschließlich durch das Entfernen einer der beteiligten Schrägkanten unterbrochen werden kann, muss M^* mindestens eine der Schrägkanten e_1, \dots, e_l enthalten. ■

Lemma 6.15 (Konfliktfreiheit und Konsistenz) *Ein Reduktionsgraph $G_{\mathcal{R}_M}$ ist genau dann konfliktfrei, wenn er konsistent ist, d.h. genau dann wenn M eine zulässige Entfernungsmenge ist.*

Beweis. In einem konfliktfreien Reduktionsgraphen gilt $K \cap K^{-1} = \emptyset$ für jede Komponente $K \subseteq V_{\mathcal{R}_M}$. Er erfüllt damit die Definition der Konsistenz (Definition 4.31 und Lemma 6.10). Der Rest der Behauptung folgt aus Satz 6.11. ■

Satz 6.14 liefert uns einen Hinweis darauf, welche der in $[S^{tr}] - M$ verbleibenden Schrägkanten für eine Erweiterung von M in Frage kommen. Das Problem der Bestimmung einer zulässigen Erweiterung von M ist äquivalent zu dem Finden einer Teilmenge der verbleibenden Schrägkanten von $[S^{tr}] - M$, deren Entfernung aus $[S^{tr}] - M$, d.h. deren Hinzufügen zu M , simultan sämtliche Konflikte in $G_{\mathcal{R}_M}$ auflöst.

Folgerung 6.16 (Auflösen von Konflikten) *Sei $G_{\mathcal{R}_M}$ ein Reduktionsgraph zu einem Plan $S \in \mathcal{S}_{SIJ}$ bezüglich einer nicht zulässigen Entfernungsmenge $M \subseteq E_{diag}(S)$, und seien durch $W_1, \dots, W_s \subseteq V_{\mathcal{R}_M}$ alle Konflikte in $G_{\mathcal{R}_M}$ beschrieben. Ist M zulässig erweiterbar, dann gilt für jede zulässige Erweiterung $M^* \subseteq E_{diag}(S)$ von $M \subseteq M^*$ und für alle W_i ($i = 1, \dots, s$) $W_i \cap M^* \neq \emptyset$. D.h. die Kantenmenge $M^* - M$ löst simultan sämtliche Konflikte in $G_{\mathcal{R}_M}$ auf.*

Bemerkung 6.17 (zulässige Erweiterbarkeit und IRRED) *Eine nicht zulässige Entfernungsmenge M ist genau dann zulässig erweiterbar, wenn jeder einzelne Konflikt im Reduktionsgraphen $G_{\mathcal{R}_M}$ zu dem Plan $S \in \mathcal{S}_{SIJ}$ auflösbar ist. Können wir für jede Schrägkante $e_i \in E_{diag}(S)$ in polynomieller Zeit feststellen, ob die Entfernungsmenge $M = \{\hat{e}_i\}$ zulässig erweiterbar ist, oder nicht, dann liegt IRRED in P. Existiert für jeden irreduziblen Plan S und jede Entfernungsmenge $M = \{\hat{e}_i\}$ mindestens ein erlaubter Rechenweg, der in polynomiell begrenzter Laufzeit beweisen kann, dass M nicht zulässig erweiterbar ist, dann liegt IRRED in NP. Das würde $IRRED \in NP \cap co-NP = ZPP^*$ nach sich ziehen.*

Direkte Konflikte und Konflikte erster Ordnung

Eine besondere Bedeutung bei der Bestimmung von Schrägkanten, um die eine nicht zulässige Entfernungsmenge M erweitert werden muss, kommt dabei den Konflikten der Ordnungen 0 und 1 zu.

Satz 6.18 (direkte Konflikte) *Sei $G_{\mathcal{R}_M}$ ein Reduktionsgraph zu einem Plan $S \in \mathcal{S}_{SIJ}$ bezüglich einer Entfernungsmenge $M \subseteq E_{diag}(S)$. Enthält $G_{\mathcal{R}_M}$ einen direkten Konflikt (Konflikt 0-ter Ordnung), dann ist M nicht zulässig erweiterbar.*

Beweis. Ein direkter Konflikt in $G_{\mathcal{R}_M}$ ist eine Γ -Relation zwischen einer erweiterten Planimplikationsklasse P_i und ihrer Umkehrung P_i^{-1} . Da alle beteiligten Kanten regulär, und damit nicht entfernbar sind, ist dieser Konflikt durch keine mögliche Erweiterung von M mehr auflösbar. Für jede Entfernungsmenge M^* , die M enthält, $M \subseteq M^*$, gilt, dass der Graph $[S^{tr}] - M^*$ kein Comparabilitygraph ist. ■

Mit Satz 6.18 steht uns ein sehr schlagkräftiges Kriterium zur Verfügung, um zwischen nicht zulässigen, aber zulässig erweiterbaren Entfernungsmengen auf der einen Seite, und unzulässigen Entfernungsmengen auf der anderen Seite, unterscheiden zu können. Das ist ein wichtiger Schritt auf dem Weg zu der Möglichkeit einer polynomiellen Nachweisbarkeit der Irreduzibilität eines Plans.

Folgerung 6.19 (direkte Konflikte und IRR) *Sei $G_{\mathcal{R}_M}$ ein Reduktionsgraph zu einem Plan $S \in \mathcal{S}_{SIJ}$ bezüglich einer Entfernungsmenge $M \subseteq E_{diag}(S)$. Enthält $G_{\mathcal{R}_M}$ einen direkten Konflikt, dann existiert kein H -Comparabilitygraph G^* mit $G_{SIJ} \subseteq G^* \subseteq [S^{tr}] - M$.*

Während direkte Konflikte also unter keinen Umständen mehr auflösbar sind, liefern Konflikte erster Ordnung eine notwendige Bedingung für die Existenz einer zulässigen Erweiterung von M .

Satz 6.20 (Konflikte erster Ordnung) *Sei $G_{\mathcal{R}_M}$ ein Reduktionsgraph zu einem Plan $S \in \mathcal{S}_{SIJ}$ bezüglich einer Entfernungsmenge $M \subseteq E_{diag}(S)$, und sei $e \in E_{diag}([S^{tr}] - M)$ eine Schrägkante aus $[S^{tr}] - M$, die in $G_{\mathcal{R}_M}$ einen Konflikt erster Ordnung beschreibt. Ist M zulässig erweiterbar, dann gilt $e \in M^*$ für jede zulässige Erweiterung $M^* \subseteq E_{diag}(S)$ von M , $M \subseteq M^*$.*

Beweis. Nach Satz 6.14 besteht die einzige Möglichkeit, einen Konflikt erster Ordnung aufzulösen darin, die einzige beteiligte Schrägkante aus $[S^{tr}] - M$ zu entfernen, d.h. zu der Entfernungsmenge M hinzuzufügen. ■

Darstellung von $G_{\mathcal{R}_M}$

Im kommenden Abschnitt widmen wir uns der Ausgestaltung eines konkreten Verfahrens zur Reduzierung eines Ausgangsplans. Vorher wollen wir uns jedoch noch die in diesem Abschnitt diskutierten Eigenschaften von Reduktionsgraphen an einem Beispiel veranschaulichen. Genau wie bei der Darstellung des Faktorgraphen und des Konsequenzgraphen beschränken wir uns auch hier der Übersicht halber auf die Darstellung der nicht redundanten Informationen.

Beispiel 6.21 (Reduktionsgraph) *Wir wollen weiter den uns mittlerweile vertrauten Plan $A \in \mathcal{S}_{SIJ}$ aus Beispiel 4.8 betrachten. In Beispiel 6.6 haben wir den Faktorgraphen $G_{\mathcal{F}}$ und den Konsequenzgraphen $G_{\mathcal{K}}$ des aus A abgeleiteten Plans A_{48} bestimmt (Abbildung 6.2 und Tabelle 6.1). Nun wollen wir uns der Darstellung und Interpretation des Reduktionsgraphen $G_{\mathcal{R}_M}$ von A_{48} für verschiedene Entfernungsmengen widmen.*

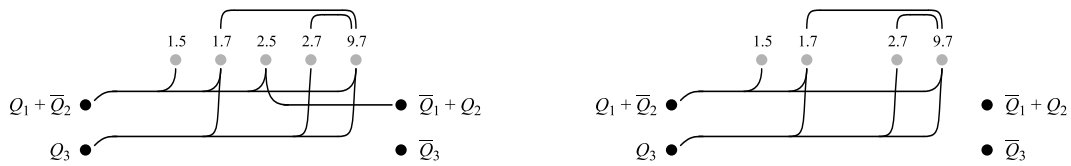


Abbildung 6.3: Die Reduktionsgraphen $G_{\mathcal{R}_{M_i}}$ von A_{48} bezüglich der Entfernungsmengen $M_1 = \{\widehat{1.4}\}$ (links) bzw. $M_2 = \{\widehat{1.4}, \widehat{2.5}\}$ (rechts).

Betrachten wir zunächst beispielsweise die Entfernungsmenge $M_1 = \{\widehat{1.4}\}$. Aus dem Konsequenzgraphen $G_{\mathcal{K}}$ (Tabelle 6.1 auf Seite 123) geht hervor, dass die Entfernung der Schrägkante 1.4 die bereits "bekannt" Γ -Relationen zwischen Q_1 und 1.5, sowie zwischen Q_3 und 1.7, aber auch eine neue Γ -Relation zwischen Q_1 und $Q_2^{-1} = \bar{Q}_2$ induziert. Der Reduktionsgraph $G_{\mathcal{R}_{M_1}}$ ist in Abbildung 6.3 (links) dargestellt. Der einzige Konflikt in $G_{\mathcal{R}_{M_1}}$ ist die Schrägkante 2.5, die sowohl zu der Planimplikationsklasse $(Q_1 + \bar{Q}_2)$, als auch zu $(Q_1 + \bar{Q}_2)^{-1} = (\bar{Q}_1 + Q_2)$ in Γ -Relation liegt. Sie bildet

also einen Konflikt erster Ordnung.

Fügen wir die Kante 2.5 zu M_1 hinzu, $M_2 = M_1 + \widehat{2.5} = \{\widehat{1.4}, \widehat{2.5}\}$, dann lösen wir damit diesen Konflikt auf. Das Hinzufügen von 2.5 zu der Entfernungsmenge induziert Γ -Relationen zwischen Q_1 und \bar{Q}_2 , zwischen Q_1 und 1.5, sowie zwischen Q_3 und 2.7 (Konsequenzgraph). Keine dieser Relationen liefert eine neue Information. Der Reduktionsgraph $G_{\mathcal{R}_{M_2}}$ ist somit konfliktfrei (Abbildung 6.3, rechts). Der Graph $[A_{48}^{tr}] - M_2$ ist ein H-Comparabilitygraph, der aus den beiden Planimplikationsklassen $(Q_1 + Q_2) = (P_1 + P_2 + \bar{P}_3 + \bar{P}_4 + P_9 + \bar{P}_{10})$ und $Q_3 = (P_5 + P_6 + P_7 + \bar{P}_8)$ besteht, die, zusammen mit den vier Schrägkanten 1.5, 1.7, 2.7 und 9.7 in einer gemeinsamen Implikationsklasse liegen. Die beiden Schrägkanten 1.5 und 2.7 sind in dieser transitiven Orientierung nicht transitiv, und können daher ebenfalls entfernt werden. Dadurch verschmelzen die beiden Planimplikationsklassen miteinander. Der resultierende Plan ist der irreduzible Plan $A_{348.10}$ (vgl. Abbildung 4.8 auf Seite 85).

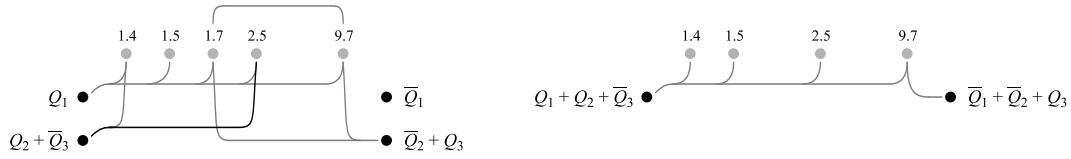


Abbildung 6.4: Die Reduktionsgraphen $G_{\mathcal{R}_{M_3}}$ von A_{48} bezüglich der Entfernungsmengen $M_3 = \{\widehat{2.7}\}$ (links) bzw. $M_4 = \{\widehat{2.7}, \widehat{1.7}\}$ (rechts).

Betrachten wir nun alternativ die Entfernungsmenge $M_3 = \{\widehat{2.7}\}$. Neben der Verschmelzung von Q_2 mit \bar{Q}_3 induziert die Entfernung von 2.7 eine Γ -Relation zwischen 2.5 mit \bar{Q}_3 (hervorgehoben in Abbildung 6.4 (links)). Der Reduktionsgraph $G_{\mathcal{R}_{M_3}}$ zu M_3 enthält weder einen direkten Konflikt, noch einen Konflikt erster Ordnung. Er enthält jedoch Konflikte höherer Ordnung. Beispielsweise ist die Planimplikationsklasse $(Q_2 + \bar{Q}_3)$ über die Schrägkanten 1.4 und 1.7 mit ihrer Umkehrung Γ -verbunden (Konflikt 2-ter Ordnung). Etwas genauer, liefert jede Kombination einer Kante aus $\{1.4, 2.5\}$ mit einer Kante aus $\{1.7, 9.7\}$ einen Konflikt 2-ter Ordnung.

Konzentrieren wir uns zunächst nur auf den Konflikt mit den Kanten 1.4 und 1.7, und betrachten wir das Hinzufügen von 1.7 zu der Entfernungsmenge. Sei also $M_4 = M_3 + \{\widehat{1.7}\}$. Dadurch verschmelzen die beiden Planimplikationsklassen Q_1 und $(Q_2 + \bar{Q}_3)$ miteinander. Darüber hinaus entstehen keine Γ -Relationen, die nicht bereits durch die Verschmelzung der beiden Planimplikationsklassen entstanden sind. Die Induktion einer Γ -Relation zwischen \bar{Q}_1 und 2.7 bleibt offensichtlich ohne Auswirkungen, da 2.7 zur Entfernungsmenge gehört. Der Reduktionsgraph $G_{\mathcal{R}_{M_4}}$ zu M_4 (Abbildung 6.4, rechts) enthält nun nur noch einen Konflikt erster Ordnung—die Schrägkante 9.7. Von den ursprünglich $2^2 = 4$ Konflikten 2-ter Ordnung haben sich die beiden Konflikte, die die Kanten 1.7 enthielten, aufgelöst. Die beiden verbleibenden Konflikte $\{1.4, 9.7\}$ und $\{2.5, 9.7\}$ enthalten beide die Kante 9.7. Sie sind zu einem Konflikt niedrigerer Ordnung “fusioniert”. Entfernen wir nun noch die Schrägkante 9.7, dann ist der resultierende Reduktionsgraph konfliktfrei (die Entfernung von 9.7 induziert keine neuen Γ -Relationen). Der entstehende H-Comparabilitygraph $[A_{48}^{tr}] - \{\widehat{2.7}, \widehat{1.7}, \widehat{9.7}\}$ besteht aus der Planimplikationsklasse $(Q_1 + Q_2 + \bar{Q}_3) = P_1 + P_2 + P_3 + \bar{P}_4 + \bar{P}_5 + \bar{P}_6 + \bar{P}_7 + P_8 + P_9 + P_{10}$ und den Schrägkanten 1.4, 1.5 und 2.5, die alle drei transitiv in der durch die Planimplikationsklasse induzierten transitiven Orientierung auf $[A_{48}^{tr}] - \{\widehat{2.7}, \widehat{1.7}, \widehat{9.7}\}$ sind. Der dadurch beschriebene irreduzible Plan ist der Plan A_{4567} .

Hätten wir uns zur Auflösung der Konflikte 2-ter Ordnung im Reduktionsgraphen zu M_3 beispielsweise für die Schrägkante 1.4 als nächster zu entfernenden Kante entschieden, dann wären wir analog zu unserem Vorgehen zu einem Reduktionsgraphen mit einem einzigen Konflikt erster Ordnung (der Schrägkante 2.5) gelangt. Dieser hätte uns im nächsten Schritt zu dem irreduziblen Plan $A_{348.10}$ geführt. ■

6.3 Ein Reduktionsschritt

Das Ziel dieses Kapitels besteht in der Entwicklung eines Verfahrens zur Reduzierung eines Ausgangsplans. Im Abschnitt 6.4 werden wir zwei Ausprägungen dieses Verfahrens diskutieren. Beiden Varianten gemeinsam ist der eigentliche Reduktionsschritt, bei dem eine Menge von Schrägkanten aus $[S^{tr}] - M$ identifiziert wird, die jeweils einen Konflikt im Reduktionsgraphen $G_{\mathcal{R}_M}$ auflösen, und daher jeweils zur Erweiterung einer gegebenen Entfernungsmenge $M \subseteq E_{diag}(S)$ benutzt werden können.

6.3.1 Die Erweiterung von zulässigen Entfernungsmengen

Als Ausgangspunkt betrachten wir einen Plan $S \in \mathcal{S}_{SIJ}$, und eine gegebene nicht zulässige Entfernungsmenge $\emptyset \neq M \subseteq E_{diag}(S)$. Da M nicht zulässig ist, $[S^{tr}] - M$ also kein H-Comparabilitygraph ist, der zu einem reduzierenden Plan von S gehört, ist der Reduktionsgraph $G_{\mathcal{R}_M}$ nicht konfliktfrei. Nach Satz 6.14 müssen wir aus jedem Konflikt mindestens eine Schrägkante zu M hinzufügen, um zu einer zulässigen Entfernungsmenge zu gelangen. Das Hinzufügen einer Schrägkante aus einem Konflikt in $G_{\mathcal{R}_M}$ zu der Entfernungsmenge M induziert zunächst in jedem Fall die Auflösung dieses Konflikts. Die entsprechende Γ -Kette in $[S^{tr}] - M$ zwischen einer Planimplikationsklasse P_i und ihrer Umkehrung P_i^{-1} wird unterbrochen. Andererseits kann die Entfernung einer Schrägkante aus $[S^{tr}] - M$ auch neue Γ -Relationen induzieren, und neue Konflikte hervorrufen. Letztendlich suchen wir also eine Teilmenge der in $[S^{tr}] - M$ noch verbliebenen Schrägkanten, die simultan sämtliche Konflikte in $G_{\mathcal{R}_M}$ löst, und dabei keine zusätzlichen Konflikte induziert.

Reihenfolge der Erweiterung

Bei der Bestimmung einer solchen Teilmenge müssen wir jedoch keineswegs sämtliche Variationen der möglichen Mengen mit je einer Schrägkante aus jedem Konflikt überprüfen. Da jeder einzelne der Konflikte unabhängig von allen anderen gelöst werden muss, genügt es, die Entfernungsmenge M in jedem Schritt nur um genau eine Schrägkante zu erweitern.

Lemma 6.22 (zulässige Erweiterungen) *Sei M^* eine zulässige Entfernungsmenge zu einem Plan $S \in \mathcal{S}_{SIJ}$. Dann ist jede nichtleere Teilmenge $M \subsetneq M^*$ zulässig erweiterbar.*

Beweis. Die Aussage folgt unmittelbar aus Definition 6.1. Insbesondere existiert keine unzulässige Entfernungsmenge M^{**} , die Teilmenge einer zulässig erweiterbaren Menge M , oder einer zulässigen Entfernungsmenge M^* ist, $M^{**} \subseteq M \subsetneq M^*$ (mit Satz 6.18). ■

Satz 6.23 (zulässige Erweiterungen) *Sei M eine zulässig erweiterbare Entfernungsmenge, und seien M_1, \dots, M_s die zulässigen Erweiterungen von M . Dann ist $M + \hat{e}$ für $e \notin M$ genau dann zulässig erweiterbar, wenn $e \in (\bigcup_{i=1}^s M_i) - M$ gilt.*

Beweis. Ist $M + \hat{e}$ zulässig erweiterbar, dann gilt nach Voraussetzung $M + \hat{e} \subseteq M_i$ für ein $i \in \{1, \dots, s\}$, und damit $e \in M_i - M$. Gilt umgekehrt $e \in M_i - M$ für ein $i \in \{1, \dots, s\}$, dann folgt aus $M \subseteq M_i$ (Voraussetzung) $M + \hat{e} \subseteq M_i$. ■

Die Idee, wie wir eine nicht zulässige Entfernungsmenge M erweitern, besteht also darin, in jedem Schritt einen beliebigen Konflikt im Reduktionsgraphen $G_{\mathcal{R}_M}$ zu identifizieren, und eine der beteiligten Schrägkanten zur Entfernungsmenge M hinzuzufügen.

Auswirkungen der Erweiterung auf den Reduktionsgraphen

Bevor wir jedoch den Algorithmus zum Erweitern der Entfernungsmenge M um eine Schrägkante aufschreiben, wollen wir uns überlegen, welche Auswirkungen die Aktualisierung des Reduktionsgraphen haben kann, wenn wir uns für eine Erweiterungskante $\hat{e}^* \in E_{diag}(S) - M$ entschieden haben. Zur näheren Beschreibung der zwei verschiedenen Typen von Knoten in $G_{\mathcal{R}_M}$ vereinbaren wir die folgenden Bezeichnungen.

Definition 6.24 (V* und V)** Sei $G_{\mathcal{R}_M} = (V_{\mathcal{R}_M}, E_{\mathcal{R}_M})$ der Reduktionsgraph zu einem Plan $S \in \mathcal{S}_{SIJ}$ bezüglich der Entfernungsmenge $M \subseteq E_{diag}(S)$. Die Knotenmenge $V_{\mathcal{R}_M}$ zerfällt in zwei disjunkte Knotenmengen $V^* = V_M^*$ und $V^{**} = V_M^{**}$, $V_{\mathcal{R}_M} = V_M^* + V_M^{**}$, mit

$$\begin{aligned} V_M^* &= \{v \in V_{\mathcal{R}_M} : v \text{ ist erweiterte Planimplikationsklasse in } [S^{tr}] - M\} \\ V_M^{**} &= \{w \in V_{\mathcal{R}_M} : w \text{ ist keine trivial-stabile Schrägkante in } [S^{tr}] - M\}. \end{aligned}$$

Lemma 6.25 (Anzahl der Knoten in V* und V)** Sei $G_{\mathcal{R}_M} = (V_M^* + V_M^{**}, E_{\mathcal{R}_M})$ der Reduktionsgraph zu einem Plan $S \in \mathcal{S}_{SIJ}$ über einer Operationenmenge SIJ mit n Aufträgen und m Maschinen. Dann gelten $|V_M^*| = \mathcal{O}(n^2m^2)$ und $|V_M^{**}| = \mathcal{O}(n^2m^2)$.

Beweis. Die Anzahl der Planimplikationsklassen in $[S^{tr}] - M$ ist beschränkt durch die Anzahl der regulären Kanten in $[S^{tr}]$, $|V_M^*| \leq |E_{reg}(S)| = n \binom{m}{2} + m \binom{n}{2} = \mathcal{O}(n^2m^2)$. Die Anzahl der Knoten in V_M^{**} ist beschränkt durch die Anzahl der möglichen Schrägkanten in $[S^{tr}]$, $|V_M^{**}| \leq |E_{diag}(S)| \leq \frac{nm}{2}(n-1)(m-1) = \mathcal{O}(n^2m^2)$. ■

Aktualisierung des Reduktionsgraphen

Haben wir eine Erweiterungskante \hat{e}^* gefunden, dann müssen wir den Reduktionsgraphen $G_{\mathcal{R}_{M^*}}$ zu der neuen Entfernungsmenge $M^* = M + \hat{e}^*$ bestimmen. Es ist jedoch nicht notwendig $G_{\mathcal{R}_M}$ für jede Entfernungsmenge M neu aus dem Faktorgraphen $G_{\mathcal{F}}$ und dem Konsequenzgraphen $G_{\mathcal{K}}$ zu bestimmen. Wir können $G_{\mathcal{R}_{M^*}}$ auch bestimmen, indem wir den Reduktionsgraphen $G_{\mathcal{R}_M}$ zu der alten Entfernungsmenge M aktualisieren (Algorithmus `updateRedGraph`).

Bei dieser Aktualisierung werden die beiden Knoten aus V^{**} entfernt, die die Schrägkanten e^* und $(e^*)^{-1}$ repräsentieren, und die Konsequenzen des Entfernens von \hat{e}^* aus $[S^{tr}] - M$, die aus dem Konsequenzgraphen $G_{\mathcal{K}}$ abgelesen werden, übertragen. Das Verschmelzen von Planimplikationsklassen kann durch die Kontraktion der entsprechenden Knoten $v \in V^*$ abgebildet werden. Im Zuge dieser Aktualisierung ist es darüber hinaus sinnvoll, die transitiven Hüllen der Planimplikationsklassen in $[S^{tr}] - M^*$ zu bestimmen, da es durch das Verschmelzen von Planimplikationsklassen vorkommen kann, dass einzelne Schrägkanten $w \in V^{**}$ in diesen transitiven Hüllen liegen. Solche Schrägkanten sind dann trivial-stabil, und können in $G_{\mathcal{R}_{M^*}}$ mit der entsprechenden Planimplikationsklasse $v \in V^*$ kontrahiert werden. Da sich dadurch unter Umständen erneut Verschmelzungen zwischen Planimplikationsklassen ergeben, muss dieser Vorgang solange iteriert werden, bis nach Abschluss aller Kontraktionen keine Kante in $G_{\mathcal{R}_{M^*}}$ zwischen zwei Knoten aus V^* existiert.

Algorithmus `UPDATEREDGRAPH`

Input: Entfernungsmenge $M^* = M + \hat{e}^*$, $G_{\mathcal{R}_M} = (V_M^* + V_M^{**}, E_{\mathcal{R}_M})$, $G_{\mathcal{K}}$,
letzte Erweiterungskante $\hat{e}^* \notin M$.

Output: Reduktionsgraph $G_{\mathcal{R}_{M^*}}$ für M^* .

BEGIN {

$G_{\mathcal{R}_{M^*}} := (G_{\mathcal{R}_M} - e^* - \bar{e}^*) + G_{\mathcal{K}_{e^*}};$

while $(\exists \widehat{vv'} \in E_{\mathcal{R}_{M^*}}$ mit $v, v' \in V_{M^*}^*$) **do** {

Kontrahiere ggf. Knoten aus $V_{M^*}^*$;

Bestimme die transitiven Hüllen der Planimplikationsklassen $v \in V_{M^*}^*$;

Kontrahiere ggf. trivial-stabile Schrägkanten $w \in V_{M^*}^{**}$ mit
zugehörigen Planimplikationsklassen $v \in V_{M^*}^{**}$;

};

return $G_{\mathcal{R}_{M^*}}$;

};

END.

Lemma 6.26 (Algorithmus `updateRedGraph`) Der Algorithmus `updateRedGraph` ist korrekt und hat einen Aufwand von $\mathcal{O}(n^5m^5)$.

Beweis. Der Schritt $G_{\mathcal{R}_{M^*}} := (G_{\mathcal{R}_M} - e^* - \bar{e}^*) + G_{\mathcal{K}_{e^*}}$ entspricht der Bestimmung von $G_{\mathcal{R}_{M^*}}$ aus $G_{\mathcal{R}_M}$ nach der Definition des Reduktionsgraphen (Definition 6.8). Alle nachfolgenden Schritte

verändern $G_{\mathcal{R}_{M^*}}$, sind aber für die Existenz oder Nichtexistenz einzelner Konflikte irrelevant. Sie führen lediglich zu einer übersichtlicheren Darstellung. Der Algorithmus ist damit korrekt.

Der Aufwand der Übertragung der Konsequenzen von \hat{e} aus dem Konsequenzgraphen ist bestimmt durch $|E_{\mathcal{K}_{\hat{e}^*}}| = \mathcal{O}(nm)$ (Lemma 6.5). Der Aufwand der Kontrahierung von Knoten aus V^* ist durch die Anzahl der höchstens zu betrachtenden Kanten bestimmt, $|E_{\mathcal{R}_{M^*}}| = \mathcal{O}(n^3m^3)$ (Lemma 6.9). Die Bestimmung der transitiven Hüllen der veränderten Planimplikationsklassen $v \in V^*$ ist mit dem Aufwand $\mathcal{O}(|V_{[S^{tr}]}|^3) = \mathcal{O}(n^3m^3)$ möglich (z.B. mit dem Algorithmus von WARSHALL). Der Aufwand der Kontrahierung von trivial-stabilen Schrägkanten $w \in V^{**}$ mit ihren jeweiligen Planimplikationsklassen $v \in V^*$ kann ebenfalls durch die Anzahl der dafür höchstens zu betrachtenden Kanten in $G_{\mathcal{R}_{M^*}}$ abgeschätzt werden, $|E_{\mathcal{R}_{M^*}}| = \mathcal{O}(n^3m^3)$. Der Aufwand der Schleife ist damit beschränkt durch $\mathcal{O}(n^3m^3)$. Da in jeder Iteration mindestens eine Schrägkante $w \in V^{**}$ mit einem $v \in V^*$ kontrahiert wird, wird die Schleife maximal $|V^{**}| = \mathcal{O}(n^2m^2)$ oft durchlaufen. Der Gesamtaufwand von `updateRedGraph` kann damit durch $\mathcal{O}(n^2m^2 \cdot n^3m^3) = \mathcal{O}(n^5m^5)$ abgeschätzt werden. ■

6.3.2 Die Bestimmung von Konflikten

Der Ansatz, eine Entfernungskante, d.h. die Schrägkante, die im aktuellen Reduktionsschritt zu der Entfernungsmenge M hinzugefügt wird, zu bestimmen, besteht darin, im Reduktionsgraphen $G_{\mathcal{R}_M}$ zunächst einen Konflikt zu identifizieren, und diesen anschließend aufzulösen, indem eine der beteiligten Schrägkanten als neue Entfernungskante zu M hinzugefügt wird. Dabei stellen sich mehrere Fragen: Wie können wir Konflikte in $G_{\mathcal{R}_M}$ möglichst schnell erkennen? Ist es notwendig bzw. überhaupt möglich, alle Konflikte in $G_{\mathcal{R}_M}$ in polynomieller Zeit zu erkennen? Und nach welchen Kriterien wählen wir den in diesem Schritt aufzulösenden Konflikt aus, wenn mehrere zur Auswahl stehen?

Um den Reduktionsgraphen auf Konsistenz, d.h. auf Konfliktfreiheit zu prüfen, genügt es, die Zusammenhangskomponenten von $G_{\mathcal{R}_M}$ zu bestimmen, und anschließend für jeden Knoten $v \in V^*$ zu prüfen, ob der Knoten $v^{-1} \in V^*$ in derselben Komponente wie v liegt. Die Bestimmung der Komponenten von $G_{\mathcal{R}_M}$ ist mit Hilfe einer Breiten- oder Tiefensuche in $\mathcal{O}(|E_{\mathcal{R}_M}|) = \mathcal{O}(n^3m^3)$ möglich. Sind die Komponenten von $G_{\mathcal{R}_M}$ gegeben, dann ist der Test auf Konsistenz somit in der Zeit $\mathcal{O}(|V_{\mathcal{R}_M}|) = \mathcal{O}(n^2m^2)$ durchführbar.

Bestimmung aller Konflikte

Wollen wir hingegen nicht nur wissen, ob $G_{\mathcal{R}_M}$ einen Konflikt enthält, sondern wollen wir auch wissen, welche Knoten aus $V_{\mathcal{R}_M}$ daran beteiligt sind, dann müssen wir für ein $v \in V^*$, das zu einem Konflikt gehört, einen Weg von v zu v^{-1} finden. Es gibt eine ganze Reihe von Algorithmen, die kürzeste Wege in ungerichteten Graphen bestimmen. Für unsere Zwecke bietet sich beispielsweise der Algorithmus von DIJKSTRA an. Dieser bestimmt in der Zeit $\mathcal{O}(|V|^2)$ den Abstand von einem Startknoten $v \in V$ zu allen anderen Knoten eines Graphen $G = (V, E)$.

Lemma 6.27 (Bestimmung aller Konflikte) *Sei $G_{\mathcal{R}_M} = (V_{\mathcal{R}_M}, E_{\mathcal{R}_M})$ der Reduktionsgraph bezüglich einer Entfernungsmenge $M \subseteq E_{diag}(S)$ zu einem Plan $S \in \mathcal{S}_{SIJ}$ über einer Operationenmenge mit n Aufträgen und m Maschinen. Dann lassen sich in der Zeit $\mathcal{O}(|V_{\mathcal{R}_M}|^3) = \mathcal{O}(n^6m^6)$ sämtliche Konflikte und ihre beteiligten Schrägkanten ermitteln.*

Beweis. Mit Hilfe des Verfahrens von DIJKSTRA lassen sich für jeden Startknoten $v \in V_{\mathcal{R}_M}$ in der Zeit $\mathcal{O}(|V_{\mathcal{R}_M}|^2)$ die Längen der jeweils kürzesten Wege zu jedem Zielknoten $w \in V_{\mathcal{R}_M}$ bestimmen. Bei geeigneter Speicherung der Zwischenschritte kann im Anschluss in der Zeit $\mathcal{O}(|V_{\mathcal{R}_M}|)$ der tatsächliche Weg von v zu w bestimmt werden. Wird jeder Knoten $v \in V_{\mathcal{R}_M}$ als Startknoten betrachtet, dann kann also in der Zeit $\mathcal{O}(|V_{\mathcal{R}_M}| \cdot (|V_{\mathcal{R}_M}|^2 + |V_{\mathcal{R}_M}|)) = \mathcal{O}(|V_{\mathcal{R}_M}|^3)$ ein kürzester Weg von jedem Startknoten $v \in V_{\mathcal{R}_M}$ zu jedem Zielknoten $w \in V_{\mathcal{R}_M}$ bestimmt werden. Die beteiligten Schrägkanten $w \in V^{**}$ entlang eines kürzesten Weges $W \subseteq V_{\mathcal{R}_M}$ von einem $v \in V^*$ zu v^{-1} , $W \cap V^{**}$, können mit konstantem Aufwand ausgelesen werden. Damit können also alle Konflikte in $G_{\mathcal{R}_M}$ in der Zeit $\mathcal{O}(|V_{\mathcal{R}_M}|^3)$ bestimmt werden. Mit Lemma 6.9 folgt $\mathcal{O}(|V_{\mathcal{R}_M}|^3) = \mathcal{O}(n^6m^6)$. ■

Bestimmung irgendeines Konflikts

Konzentrieren wir uns darauf, lediglich *irgendeinen* Konflikt in $G_{\mathcal{R}_M}$ zu bestimmen, dann ist es nicht notwendig, für jeden Startknoten $v \in V^*$ nach einem Weg zu v^{-1} zu suchen. Es genügt vielmehr, für einen einzigen Knoten $v \in V^*$, dessen Pendant v^{-1} in der gleichen Komponente wie v liegt, einen kürzesten Weg zu bestimmen. Dadurch reduziert sich der Aufwand von $\mathcal{O}\left((n^2m^2)^3\right)$ auf $\mathcal{O}\left((n^2m^2)^2\right)$. Die Bestimmung der Zusammenhangskomponenten ($\mathcal{O}(|E_{\mathcal{R}_M}|) = \mathcal{O}(n^3m^3)$) und die einmalige Bestimmung der Knoten des kürzesten Weges ($\mathcal{O}(|V_{\mathcal{R}_M}|) = \mathcal{O}(n^2m^2)$) aus der Bestimmung der Länge eines kürzesten Weges, sind nach wie vor mit geringerem Aufwand realisierbar.

Auswahl eines geeigneten Konflikts

Nach den Sätzen 6.18 und 6.20 kommt den direkten Konflikten und den Konflikten erster Ordnung eine besondere Bedeutung bei der Bestimmung einer Erweiterungskante zu. Bei Konflikten höherer Ordnung stehen jeweils mehrere Schrägkanten als neue Erweiterungskante zur Auswahl. Da keineswegs klar ist, ob jede an einem Konflikt beteiligte Schrägkante als Erweiterungskante geeignet ist, ergibt sich daraus die Gefahr, Fehlentscheidungen zu treffen. Entscheiden wir uns beispielsweise für die Erweiterung einer Entfernungsmenge M um eine an einem Konflikt beteiligte Schrägkante e , von der wir nicht wissen, dass sie nichttrivial-stabil ist, dann ist $M + \hat{e}$ keine zulässig erweiterbare Entfernungsmenge mehr. Unter Umständen müssen wir also bei der Auswahl von mehreren zur Verfügung stehenden Erweiterungskanten verzweigen.

Im Gegensatz dazu sind die Informationen, die uns direkte Konflikte oder Konflikte erster Ordnung liefern, definitiv. Bei der Existenz eines direkten Konflikts wissen wir, dass M nicht mehr zulässig erweiterbar ist. Und zwar unabhängig von allen anderen Konflikten. Bei der Existenz eines Konflikts erster Ordnung wissen wir, dass M nur genau dann zulässig erweiterbar sein kann, wenn die beteiligte Schrägkante zu M hinzugefügt wird. Liegen mehrere Konflikte erster Ordnung vor, dann ist die Reihenfolge, in der M um die entsprechenden Schrägkanten erweitert wird, irrelevant. Für eine bessere Anschauung ist auch die Vorstellung, dass sämtliche Schrägkanten aus Konflikten erster Ordnung gleichzeitig zu der Entfernungsmenge M hinzugefügt werden, geeignet.

Bemerkung 6.28 (mehrere Konflikte erster Ordnung) Sei $G_{\mathcal{R}_M}$ der Reduktionsgraph zu einem Plan $S \in \mathcal{S}_{SIJ}$ bezüglich einer Entfernungsmenge $M \subseteq E_{diag}(S)$, und sei durch $N \subseteq E_{diag}(S) - M$ die Menge der Schrägkanten in $[S^{tr}] - M$ gegeben, die jeweils einen Konflikt erster Ordnung in $G_{\mathcal{R}_M}$ beschreiben. Dann ist M genau dann zulässig erweiterbar, wenn $M + \hat{N}$ zulässig erweiterbar ist.

Während bei der Bestimmung aller Konflikte in $G_{\mathcal{R}_M}$ nach Lemma 6.27 auch die direkten Konflikte und die erster Ordnung entdeckt werden, und sinnvollerweise als erste betrachtet werden, ist es bei der Bestimmung irgendeines Konflikts daher angebracht, zunächst direkt nach Konflikten der Ordnungen 0 und 1 zu suchen, und erst anschließend ggf. einen Konflikt höherer Ordnung zu finden.

Der Algorithmus `findNextEdge` besteht daher aus drei Abschnitten. Zunächst suchen wir nach der Existenz von direkten Konflikten (Schritt (1)). Enthält $G_{\mathcal{R}_M}$ keine direkten Konflikte, suchen wir nach einem Konflikt erster Ordnung. Und nur wenn $G_{\mathcal{R}_M}$ weder direkte Konflikte noch solche erster Ordnung besitzt, bestimmen wir einen beliebigen Konflikt höherer Ordnung, und wählen eine beliebige der beteiligten Schrägkanten. Bevor wir diese gewählte Schrägkante e^* jedoch als neue Erweiterungskante akzeptieren, prüfen wir, ob die Konsequenzen, die e^* induziert, nicht im Widerspruch zu $G_{\mathcal{R}_M}$ stehen. Dieser Test zielt insbesondere darauf ab, sicherzustellen, dass das Verdrehen der beiden Planimplikationsklassen, die durch e^* voneinander getrennt werden, nicht die Transitivität von e^* induziert. Ist das der Fall, dann ist e^* bezüglich der Entfernungsmenge M stabil, und wir müssen aus den verbleibenden Schrägkanten des ermittelten Konflikts eine neue Kante als Erweiterungskante bestimmen. Das gilt natürlich genauso für den Fall, dass die Kante e^* bereits als (nichttrivial-) stabile Schrägkante erkannt worden ist. Führen *alle* Schrägkanten eines Konflikts zu direkten Konflikten, dann haben wir dadurch ein Zertifikat dafür, dass M nicht zulässig erweiterbar ist.

Lemma 6.29 (direkte Konflikte als Konsequenz) Sei $G_{\mathcal{R}_M}$ der Reduktionsgraph zu einem Plan $S \in \mathcal{S}_{SIJ}$ bezüglich einer Entfernungsmenge $M \subseteq E_{diag}(S)$, und sei durch $X \subseteq E_{diag}(S) - M$ die Menge der Schrägkanten eines Konflikts der Ordnung l ($l \geq 1$) in $G_{\mathcal{R}_M}$ gegeben. Enthält der

Reduktionsgraph $G_{\mathcal{R}_{M^*}}$ bezüglich $M^* = M + \hat{e}$ für jede Schrägkante $e \in X$ einen direkten Konflikt, dann ist M nicht zulässig erweiterbar.

Beweis. Nach Satz 6.14 ist M genau dann zulässig erweiterbar, wenn $M + \hat{e}$ für ein $e \in X$ zulässig erweiterbar ist. Mit Satz 6.18 folgt jedoch aus den Voraussetzungen, dass $M + \hat{e}$ für jedes $e \in X$ nicht zulässig erweiterbar ist. ■

Auswahl einer Entfernungskante

Algorithmus FINDNEXTEDGE

Input: Entfernungsmenge M , $G_{\mathcal{R}_M} = (V_M^* + V_M^{**}, E_{\mathcal{R}_M})$ nicht konfliktfrei, $G_{\mathcal{K}}$.

Output: nicht trivial-stabile Schrägkante $\hat{e}^* \notin M$, oder “ M nicht zulässig erweiterbar”.

```

BEGIN {
  // (1) Teste auf Existenz direkter Konflikte
  forall  $v \in V_M^*$  do {
    if  $(\widehat{vv^{-1}} \in E_{\mathcal{R}_M})$  then return “ $M$  nicht zulässig erweiterbar”;
  };
  // (2) Teste auf Existenz Konflikter 1. Ordnung
  forall  $v \in V_M^*$  do {
    forall  $\widehat{vw} \in E_{\mathcal{R}_M}$  do{
      if  $(\widehat{wv^{-1}} \in E_{\mathcal{R}_M})$  then {
         $e^* := w \in V_M^{**} \subseteq E_{diag}(S)$ ;
        if (Konsequenzen von  $e^*$  führen zu direktem Konflikt in  $G_{\mathcal{R}_M}$  or
           $\hat{e}^*$  “stabil”) then {
          return “ $M$  nicht zulässig erweiterbar”;
        };
        else return  $\hat{e}^*$ ;
      };
    };
  };
  // (3) Finde irgendeinen Konflikt höherer Ordnung
  Bestimme Zusammenhangskomponenten von  $G_{\mathcal{R}_M}$ ;
  forall  $v \in V_M^*$  do {
    if ( $v$  und  $v^{-1}$  liegt in gemeinsamer Komponente) then {
      Finde kürzesten Weg  $W \subseteq V_{\mathcal{R}_M}$  zwischen  $v$  und  $v^{-1}$ ;
       $X := W \cap V_M^{**}$ ;
      while ( $X \neq \emptyset$ ) do {
        Wähle eine Schrägkante  $e^* \in X$ ;
        if (Konsequenzen von  $e^*$  führen zu direktem Konflikt in  $G_{\mathcal{R}_M}$  or
           $\hat{e}^*$  “stabil”) then {
           $X := X - \hat{e}^*$ ;
        };
        else return  $\hat{e}^*$ ;
      };
      return “ $M$  nicht zulässig erweiterbar”;
    };
  };
};
END.

```

Lemma 6.30 (Algorithmus findNextEdge) *Der Algorithmus findNextEdge ist korrekt und hat einen Aufwand von $\mathcal{O}(|V_{\mathcal{R}_M}|^2) = \mathcal{O}(n^4m^4)$.*

Beweis. In Schritt (1) wird ein direkter Konflikt entdeckt, sofern mindestens einer existiert. Dabei wird für jeden Knoten $v \in V^*$ die Existenz der Kante $\widehat{vv'} \in E_{\mathcal{R}_M}$ überprüft. Die Laufzeit von

Schritt (1) kann damit durch $\mathcal{O}(|V^*|) = \mathcal{O}(n^2m^2)$ (Lemma 6.25) abgeschätzt werden. In Schritt (2) wird zu jeder Kante zwischen einem $v \in V^*$ und einem $w \in V^{**}$ die Existenz der Kante zwischen $w \in V^{**}$ und $v^{-1} \in V^*$ überprüft. Dadurch wird ein Konflikt erster Ordnung in diesem Schritt erkannt, sofern mindestens einer existiert. Die Laufzeit ist dabei beschränkt durch $\mathcal{O}(|V^*| \cdot |V^{**}|) = \mathcal{O}(n^2m^2 \cdot n^2m^2) = \mathcal{O}(n^4m^4)$ (Lemma 6.25). Andererseits wird jede Kante in $G_{\mathcal{R}_M}$ höchstens einmal betrachtet. Daher ist der Aufwand von Schritt (2) auch beschränkt durch $\mathcal{O}(|E_{\mathcal{R}_M}|) = \mathcal{O}(n^3m^3)$ (Lemma 6.25).

Erreicht der Algorithmus den Schritt (3), dann enthält $G_{\mathcal{R}_M}$ ausschließlich Konflikte höherer Ordnung. (Die Existenz von Konflikten ist durch die Voraussetzung gesichert.) In $\mathcal{O}(|E_{\mathcal{R}_M}|) = \mathcal{O}(n^3m^3)$ lassen sich nun die Zusammenhangskomponenten von $G_{\mathcal{R}_M}$ bestimmen. Anschließend kann mit der Laufzeit $\mathcal{O}(|V^*|)$ ein Knoten $v \in V^*$ ermittelt werden, der mit $v^{-1} \in V^*$ in einer gemeinsamen Komponente von $G_{\mathcal{R}_M}$ liegt, und somit einen Konflikt in $G_{\mathcal{R}_M}$ induziert. Für diesen Knoten $v \in V^*$ kann dann in der Zeit $\mathcal{O}(|V_{\mathcal{R}_M}|^2)$ (beispielsweise mittels des Verfahrens von DIJKSTRA) die Länge eines kürzesten Weges zu jedem beliebigen Knoten, insbesondere zu v^{-1} der Komponente bestimmt werden. Anschließend kann ein kürzester Weg $W \subseteq V_{\mathcal{R}_M}$ zwischen v und v^{-1} aus den Informationen über die Abstände zwischen den Knoten mit dem Aufwand $\mathcal{O}(|V_{\mathcal{R}_M}|)$ extrahiert werden (Auslesen des Wegbaums). Im Anschluss daran erfolgt für maximal $|X| \leq |V^{**}| = \mathcal{O}(n^2m^2)$ Schrägkanten e^* jeweils die Überprüfung, ob die Entfernung von \hat{e}^* aus $[S^{tr}] - M$ zu direkten Konflikten führt. Dazu müssen jeweils maximal $|E_{\mathcal{K}_{e^*}}| = \mathcal{O}(nm)$ (Lemma 6.5) Kanten in $G_{\mathcal{K}}$ überprüft werden. Da alle genannten Elemente in Schritt (3) nacheinander und jeweils nur einmal ausgeführt werden, ergibt sich für den Gesamtaufwand von Schritt (3)

$$\begin{aligned} & \mathcal{O}(|E_{\mathcal{R}_M}|) + \left[\mathcal{O}(|V^*|) + \mathcal{O}(|V_{\mathcal{R}_M}|^2) + \mathcal{O}(|V_{\mathcal{R}_M}|) + [\mathcal{O}(|V^{**}|) \cdot \mathcal{O}(|E_{\mathcal{K}_{e^*}}|)] \right] \\ = & \mathcal{O}(n^3m^3) + \left[\mathcal{O}(n^2m^2) + \mathcal{O}((n^2m^2)^2) + \mathcal{O}(n^2m^2) + [\mathcal{O}(n^2m^2) \cdot \mathcal{O}(nm)] \right] \\ = & \mathcal{O}(n^3m^3) + [\mathcal{O}(n^4m^4) + [\mathcal{O}(n^3m^3)]] \\ = & \mathcal{O}(n^4m^4). \end{aligned}$$

Der Aufwand des gesamten Verfahrens beträgt damit $\mathcal{O}(n^2m^2) + \mathcal{O}(n^3m^3) + \mathcal{O}(n^4m^4) = \mathcal{O}(n^4m^4)$. ■

Für die zweifelsfreie Erkennung der Stabilität einer Kante ist es notwendig, sich zu merken, ob zu einer ausgewählten Erweiterungskante e^* Alternativen bestanden, oder nicht. Diese Information kann bei der Bestimmung der Kante e^* erhoben, und zusammen mit der Kante übergeben werden.

Definition 6.31 (alternativlose Erweiterungskante) Sei $M \subseteq E_{diag}(S)$ eine Entfernungsmenge zu einem Plan $S \in \mathcal{S}_{SIJ}$. Sei $e^* \notin M$ eine mit dem Algorithmus `findNextEdge` ermittelte Erweiterungskante, und sei $W \subseteq V_{\mathcal{R}_M}$ der Konflikt, der e^* enthält. Wir nennen e^* alternativlos bezüglich M , wenn sie die einzige Schrägkante in $W \cap V_M^{**}$ ist, die den Konflikt W auflöst.

Eine Erweiterungskante ist damit genau dann alternativlos, wenn sie entweder zu einem Konflikt erster Ordnung gehört, oder wenn alle anderen Kanten eines höheren Konflikts zu einer nicht mehr zulässig erweiterbaren Entfernungsmenge führen würden. D.h. wenn alle anderen Kanten aus $X = W \cap V^{**}$ stabil bezüglich M sind.

Nichttrivial-stabile Schrägkanten und nicht trivial-stabile Schrägkanten

In Abschnitt 4.3.3 haben wir den Begriff der nichttrivial-stabilen Schrägkante geprägt. Das sind stabile Schrägkanten, die nicht trivial-stabil sind, also nicht in einer erweiterten Planimplikationsklasse liegen. Wir müssen in diesem Zusammenhang die Begriffe “nichttrivial-stabile Schrägkante” und “nicht trivial-stabile Schrägkante” sorgfältig auseinanderhalten. Während es sich bei einer Kante der ersten Form definitiv um eine stabile Schrägkante handelt, ist eine Schrägkante, die nicht trivial-stabil ist, möglicherweise überhaupt nicht stabil.

Der Algorithmus `findNextEdge` liefert als Ergebnis entweder ein Zertifikat, das die Entfernungsmenge M nicht zulässig erweiterbar ist, oder eine Erweiterungskante \hat{e}^* , von der wir wissen, dass sie keine trivial-stabile Schrägkante bezüglich der durch M induzierten Planimplikationsklassen ist. Wir

können uns jedoch nicht sicher sein, dass \hat{e}^* nicht dennoch stabil ist. Gleichwohl ist der enthaltene Test auf etwaige inkonsistente Konsequenzen in der Lage, auch manche nichttrivial-stabile Schrägkante als stabile Schrägkante zu erkennen.

Betrachten wir noch einmal den Plan B_{24} auf Seite 95 (Abbildung 4.10). Bei der Untersuchung von Konstellationen für nichttrivial-stabile Schrägkanten mit zwei beteiligten Planimplikationsklassen haben wir festgestellt, dass die Kanten 3.5 und 7.5 in B_{24} in der trennenden Kantenmenge zwischen den beiden Planimplikationsklassen P_1 und P_2 liegen. Betrachten wir nun die Konsequenzen einer Entfernung dieser beiden Kanten, dann stellen wir fest, dass beide Schrägkanten sowohl die Verschmelzung von P_1 mit P_2 , als auch die Verschmelzung von P_1 mit P_2^{-1} induzieren. Die Konsequenzen jeder dieser beiden Kanten sind also bereits inkonsistent, und die Kanten damit stabil. Der Algorithmus `findNextEdge` erkennt diese Form von nichttrivial-stabilen Schrägkanten. (Entweder in Schritt (3) oder, im Falle eines Konfliktes erster Ordnung, in Schritt (1) beim nächsten Durchlauf.) Für die nichttrivial-stabile Schrägkante in dem anderen Beispiel für einen solchen Plan, den Plan B_{22} auf Seite 91, gilt ein identisches Resultat: die Entfernung der Schrägkante $O_{22}O_{33}$ zieht sowohl $P_1 + P_2$, als auch $P_1 + P_2^{-1}$ nach sich.

6.3.3 Getarnte stabile Schrägkanten

Wir unterteilen die Menge der nichttrivial-stabilen Schrägkante daher in die Menge der Kanten, deren Stabilität wir durch den Algorithmus `findNextEdge` ohne die Notwendigkeit von Verzweigungen in der Laufzeit $\mathcal{O}(n^6m^6)$ beweisen können, und solche stabilen Schrägkanten, deren Stabilität wir auf diese Weise nicht erkennen können. Schrägkanten aus der zweiten Gruppe nennen wir *getarnte* stabile Schrägkanten.

Definition 6.32 (getarnte stabile Schrägkante) Sei $[S^{tr}] = (SIJ, E_{reg}(S) + E_{diag}(S))$ der *H-Comparabilitygraph* zu einem Plan $S \in \mathcal{S}_{SIJ}$, und sei $G_{\mathcal{R}_M}$ der Reduktionsgraph von S bezüglich einer Entfernungsmenge $M \subseteq E_{diag}(S)$. Eine bezüglich M stabile Schrägkante $e \in E_{diag}(S)$, die nicht durch eine Folge von Konflikten erster Ordnung zu einem direkten Konflikt in $G_{\mathcal{R}_{M^*}}$, mit $M + e \subseteq M^*$, führt, heißt *getarnt stabil*.

Eine stabile Schrägkante ist beispielsweise höchstens dann getarnt stabil, wenn ihre Entfernung aus $[S^{tr}]$ nicht gleichzeitig $P_i + P_j$ und $P_i + P_j^{-1}$ impliziert. Auch wenn Konstellationen für getarnte stabile Schrägkanten mit nur zwei beteiligten Planimplikationsklassen denkbar sind, ist vor dem Hintergrund der Diskussion zu den Eigenschaften nichttrivial-stabiler Schrägkanten (Abschnitt 4.3.3) eine Schrägkante mutmaßlich vor allem dann getarnt stabil, wenn ihre Stabilität nur durch mehr als zwei verschiedene Planimplikationsklassen induziert wird.

Bemerkung 6.33 (keine Beispiele bekannt) Es sind bisher keine Beispiele für Pläne bekannt, die getarnte stabile Schrägkanten enthalten. Diese Beobachtung korrespondiert mit der Beobachtung, dass auch keine Beispiele für Pläne mit nichttrivial-stabilen Schrägkanten und mehr als zwei beteiligten Planimplikationsklassen bekannt sind (Bemerkung 4.56).

Die Frage der Existenz oder Nichtexistenz von getarnten stabilen Schrägkanten ist eng mit der Frage der Existenz eines polynomiellen Verfahrens zur Reduzierbarkeit von Plänen verbunden. Der Begriff der getarnten stabilen Schrägkante beschreibt genau solche stabilen Kanten, bei denen der Nachweis der Stabilität unter Umständen nicht mit polynomiell beschränktem Aufwand möglich ist.

Satz 6.34 (getarnte stabile Schrägkante) Sei $G_{\mathcal{R}_M} = (V_M^* + V_M^{**}, E_{\mathcal{R}_M})$ der Reduktionsgraph zu einem normalen Plan $S \in \mathcal{S}_{SIJ}$ bezüglich einer zulässig erweiterbaren Entfernungsmenge $M \subseteq E_{diag}(S)$. Ferner sei $W \subseteq V_{\mathcal{R}_M}$ ein Konflikt in $G_{\mathcal{R}_M}$, und $e^* \in X = W \cap V_M^{**}$ eine beteiligte Schrägkante. Enthält S keine getarnten stabilen Schrägkanten, dann kann e^* mit dem Aufwand $\mathcal{O}(n^5m^5)$ auf Stabilität überprüft werden.

Beweis. Ist e^* stabil bezüglich M , dann induziert die Entfernung von \hat{e}^* aus $[S^{tr}] - M$ auf direktem Wege und ohne weitere Verzweigungsmöglichkeiten (Folge von Konflikten erster Ordnung) einen direkten Konflikt. Ist e^* nicht stabil bezüglich M , dann ist die Entfernungsmenge $M^* = M + \hat{e}^*$

entweder zulässig, oder zulässig erweiterbar. Die Länge einer solchen Folge von Konflikten erster Ordnung ist durch $|E_{diag}(S)| = \mathcal{O}(n^2m^2)$ beschränkt. Der Aufwand, in jedem Schritt einen Konflikt der Ordnung 0 oder 1 zu finden, ist durch $\mathcal{O}(n^3m^3)$ beschränkt (Lemma 6.30). Allerdings muss in jedem Schritt auch der Reduktionsgraph aktualisiert werden. Der Aufwand hierfür ist gegeben durch $\mathcal{O}(|E_{diag}(S)| \cdot n^3m^3)$ (Lemma 6.26). Andererseits nimmt mit jeder Iteration in `updateRedGraph` die Anzahl der verbleibenden Schrägkanten, die zur Erweiterung der Entfernungsmenge zur Verfügung stehen, echt ab. Jede Schrägkante aus $E_{diag}(S) - M$ wird damit im Zuge der Überprüfung von M auf die Möglichkeit einer Erweiterung durch eine Folge von Konflikten erster Ordnung höchstens einmal aus V^{**} gestrichen (entweder als Erweiterungskante, oder bei der Kontraktion als trivial-stabile Schrägkante). Daher ist der Gesamtaufwand dieser Überprüfung durch $\mathcal{O}(n^2m^2 \cdot (n^3m^3 + n^3m^3)) = \mathcal{O}(n^5m^5)$ beschränkt. ■

Bemerkung 6.35 (getarnte stabile Schrägkanten und IRRED) *Kann gezeigt werden, dass getarnte stabile Schrägkanten nicht existieren, dann ist mit Hilfe von `findNextEdge` ein polynomieller Test auf Irreduzibilität möglich. Wäre die Nachweisbarkeit der Stabilität einer nichttrivial-stabilen Schrägkante zwar von der betrachteten Entfernungsmenge M abhängig, aber auf mindestens einem erlaubten Rechenweg möglich, dann wäre immernoch ein NP-Test für Irreduzibilität realisierbar.*

Wie ein solcher NP-Test für die Irreduzibilität aussehen kann, wollen wir im kommenden Abschnitt diskutieren. Vorher wollen wir uns jedoch noch einmal den Grund dafür vergegenwärtigen, dass bei der Abwesenheit von getarnten stabilen Schrägkanten auch ein polynomieller Teil des exponentiell großen Suchbaums ausreicht, um bei irreduziblen Plänen die Irreduzibilität auch zu beweisen: Enthält ein Plan keine solchen Schrägkanten, dann kann man bei der Auswahl einer Erweiterungskante praktisch keine Fehler machen. Wählt man eine stabile Schrägkante als Erweiterungskante, dann ist nicht nur schnell klar, dass die so erzeugte Entfernungsmenge nicht mehr zulässig erweiterbar ist. Insbesondere ist auch sofort klar, in welchem Schritt der Fehler gemacht wurde.

6.4 REDUCING1 und REDUCING2

Nachdem wir nun eine Vorstellung davon haben, wie wir eine Schrägkante aus $[S^{tr}] - M$ finden können, um die wir eine gegebene Entfernungsmenge $M \subseteq E_{diag}(S)$ erweitern können, können wir alle Elemente unseres Reduktionsverfahrens zusammensetzen. Vor dem Hintergrund der Fragen nach der Existenz oder Nichtexistenz von getarnten stabilen Schrägkanten bzw. der Erkennbarkeit solcher Kanten, wollen wir den Reduktionsalgorithmus in zwei verschiedenen Ausprägungen diskutieren. Bei der Variante `Reducing1` (Abschnitt 6.4.1) starten wir mit einer beliebigen Schrägkante und versuchen anschließend, diese Entfernungsmenge solange zu erweitern, bis sie zulässig ist. Gelangen wir dabei zu einem Widerspruch, dann verwerfen wir die gesamte Entfernungsmenge, und beginnen mit einer anderen Startkante erneut. Bei der alternativen Variante `Reducing2` (Abschnitt 6.4.2) verzweigen wir dagegen mit einer Tiefensuchstrategie solange bei jeder Wahlmöglichkeit in einem Schritt, bis wir eine zulässige Entfernungsmenge gefunden haben.

6.4.1 Der Algorithmus REDUCING1

Das Ziel beider vorgestellten Varianten unseres Reduktionsverfahrens besteht darin, für einen normalen Ausgangsplan $S \in \mathcal{S}_{SIJ}$ entweder einen reduzierenden Plan $S^* \in \mathcal{S}_{SIJ}$ mit $S^* \prec S$ zu finden, oder nachzuweisen, dass S irreduzibel ist. Für die Lösung von IRRED ist es nicht notwendig, zu einem Ausgangsplan S einen irreduziblen Plan zu finden. Es genügt, einen beliebigen reduzierenden Plan zu finden. Für die Suche nach einem irreduziblen Plan kann dieser gefundene Plan anschließend seinerseits auf Irreduzibilität überprüft werden. Da jeder streng reduzierende Plan S^* mindestens eine Schrägkante weniger als S enthält (Satz 4.2), ist die Anzahl der Iterationen, die durchlaufen werden müssen, um zu einem reduzierbaren Ausgangsplan S einen irreduziblen Plan zu finden, beschränkt durch die Anzahl der Schrägkanten in $[S^{tr}]$, d.h. durch $E_{diag}(S) = \mathcal{O}(n^2m^2)$. Existiert also ein polynomieller Algorithmus, der zu einem Ausgangsplan S entweder einen beliebigen streng reduzierenden Plan S^* finden, oder die Irreduzibilität von S beweisen kann, dann ist auch die Bestimmung eines irreduziblen Plans mit polynomiellem Aufwand realisierbar.

Mit der Variante `Reducing1` wollen wir uns zunächst Gedanken über eine polynomielle Laufzeit machen. Dabei müssen wir möglicherweise eine Fehlerquote in Kauf nehmen. Im Gegensatz dazu konzentrieren wir uns bei der Variante `Reducing2` auf eine exakte Antwort, und müssen unter Umständen Abstriche bei der Laufzeit hinnehmen.

Die Grundidee einer polynomiellen Beschränkung

Der relevante Suchraum besteht aus der Menge aller Teilmengen $M^* \subseteq E_{diag}(S)$ der Schrägkanten des H-Comparabilitygraphen $[S^{tr}]$, die aus einer echt kleineren Entfernungsmenge M durch die Erweiterung einer durch den Algorithmus `findNextEdge` bestimmten Erweiterungskante \hat{e}^* hervorgehen, $M^* = M + \hat{e}^*$. Dieser Suchraum enthält zwar typischerweise nicht alle Teilmengen der Schrägkanten—beispielsweise ist die Entfernungsmenge $M = E_{diag}(S)$ in aller Regel unzulässig—, hat aber dennoch eine exponentielle Größe.

Um die Laufzeit unseres Reduktionsverfahrens polynomiell begrenzen zu können, betrachten wir daher im Wesentlichen zu jeder nichttrivial-stabilen Startkante $\hat{e} \in E_{diag}(S)$ genau einen Ast des Suchbaums. In jedem Reduktionsschritt, bei dem für die Erweiterung der Entfernungsmenge mehrere Schrägkanten zur Auswahl stehen, betrachten wir nur die Auswahl einer einzigen dieser Kanten. Finden wir auf diesem Weg eine zulässige Entfernungsmenge M^* , dann ist der aus $[S^{tr}] - M^*$ resultierende Plan S^* ein streng reduzierender Plan von S , und wir sind fertig. Ist S jedoch reduzierbar, und wir gelangen auf diesem Weg zu einer nicht mehr zulässig erweiterbaren Entfernungsmenge, dann haben wir bei mindestens einer Wahlmöglichkeit einen Fehler gemacht. Um die polynomielle Beschränktheit nicht zu gefährden, verzichten wir in diesem Fall jedoch auf die Suche nach diesem Fehler. Wir verwerfen die gesamte Entfernungsmenge, und beginnen mit einer anderen Startkante erneut.

Gab es bei der Bestimmung der unzulässigen Entfernungsmenge keine Wahlmöglichkeiten, d.h. führte die Startkante \hat{e} alternativlos zu einem direkten Konflikt, dann können wir auf die Stabilität von \hat{e} schließen. Finden wir mit keiner Startkante eine zulässige Entfernungsmenge, und konnten wir nicht die Stabilität aller Schrägkanten nachweisen, dann ist nicht klar, ob der Ausgangsplan S reduzierbar ist, oder nicht.

Die Laufzeit dieser Vorgehensweise bestimmt sich folgendermaßen: Zu jeder Startkante $\hat{e} \in E_{diag}(S)$ betrachten wir maximal $|E_{diag}(S)| = \mathcal{O}(n^2m^2)$ viele Reduktionsschritte. Jeder Reduktionsschritt hat einen Aufwand von $\mathcal{O}(n^5m^5)$ (Algorithmus `updateRedGraph`). Da wir maximal $|E_{diag}(S)| = \mathcal{O}(n^2m^2)$ viele Startkanten untersuchen, ist die Gesamtlaufzeit beschränkt durch $\mathcal{O}(n^9m^9)$.

Verbesserung der Aussagekraft bei gleichbleibendem Aufwand

Um sowohl die Trefferchance für eine zulässige Entfernungsmenge, als auch die Nachweischance für eine stabile Schrägkante zu erhöhen, können wir uns bei gleichbleibendem Aufwand erlauben, bei der Feststellung der Unzulässigkeit einer Entfernungsmenge zum jeweils letzten Entscheidungsknoten im Suchbaum zurückzugehen. Ermitteln wir also in einem Reduktionsschritt zu der Entfernungsmenge M einen Konflikt höherer Ordnung, der aus den Kanten X besteht, dann wählen wir eine Schrägkante $\hat{e}^* \in X$ als Erweiterungskante, und speichern $M' = M$, $X' = X$ und $\hat{e}' = \hat{e}^*$ ab.

Führt die Erweiterung von M um die Kante \hat{e}^* in einer Folge von Konflikten erster Ordnung, d.h. alternativlos, zu einem direkten Konflikt, dann verwerfen wir nicht sofort die gesamte Entfernungsmenge, sondern betrachten eine andere Kante $\hat{e}^{**} \in X' - \hat{e}'$ als Erweiterungskante von M' . Führt die Erweiterung von M um die Kante \hat{e}^* in einem späteren Schritt zu einer erneuten Verzweigung, dann werden die Daten dieses Entscheidungsknotens gespeichert. In diesem Fall kommen wir auf keinen Fall mehr zu der Entfernungsmenge M zurück. Wir verwerfen den Ast im Suchbaum zu einer Startkante \hat{e} erst, wenn wir einen Konflikt X' gefunden haben, bei dem alle Schrägkanten stabil bezüglich M' sind.

Dadurch, dass wir jeweils nur bis zum letzten Entscheidungsknoten zurückgehen, also Verzweigungen nur betrachten, wenn sich ein Ast der Verzweigung ohne weitere Verzweigungsmöglichkeiten als Sackgasse erweist, bleibt das Verfahren insgesamt nach wie vor polynomiell beschränkt.

Wie wir noch sehen werden, ermöglicht diese Vorgehensweise bei Ausgangsplänen, die keine getarnten stabilen Schrägkanten enthalten, sogar die sichere Entscheidung der Reduzierbarkeit.

Der Algorithmus REDUCING1

Die eben diskutierte Grundidee eines polynomiell beschränkten Verfahrens zur Reduzierung von Plänen spiegelt sich genau wie die angesprochene Verfeinerung im Algorithmus `Reducing1` wider. Wir präsentieren das Verfahren hier in einer vereinfachten und übersichtlicheren Darstellungsweise. Eine detailliertere Darstellung des gesamten Verfahrens und aller verwendeten Teilprozeduren (in Klammern) ist im Anhang A.1 und A.2 enthalten. Die detailliertere Darstellung dort dient insbesondere der Abschätzung der Laufzeit. Der Algorithmus `Reducing1` besteht aus zwei ineinandergeschachtelten Schleifen. Die äußere Schleife wird maximal $|E_{diag}(S)| = \mathcal{O}(n^2m^2)$ oft durchlaufen. Die innere Schleife enthält als zentrale Punkte die Bestimmung eine Menge $X \in V_M^{**}$ der Schrägkanten zu einem Konflikt im Reduktionsgraphen $G_{\mathcal{R}_M}$ (Prozedur `findX`, enthält `findNextEdge`), und die Aktualisierung von $G_{\mathcal{R}_M}$ (Prozedur `updateRedGraph`). Sie wird aufgrund der möglichen Rückschritte zum jeweils letzten Entscheidungsknoten unter Umständen nicht nur $|E_{diag}(S)| = \mathcal{O}(n^2m^2)$ oft durchlaufen. Die Erkennung einer Alternative in einem Entscheidungsknoten als Sackgasse kann jedoch vom logische Standpunkt aus, auch als Bestandteil der inneren Schleife betrachtet werden.

Algorithmus REDUCING1

Input: $[S^{tr}]$ zu normalem Plan $S \in \mathcal{S}_{SIJ}$.

Output: zulässige Entfernungsmenge M , “ S ist irreduzibel”, oder “?” (Versagen).

BEGIN {

Bestimme erweiterte Planimplikationsklassen $\mathcal{P}_S = \{P_1, \dots, P_k\}$,

Faktorgraph $G_{\mathcal{F}}$ und Konsequenzgraph $G_{\mathcal{K}}$;

Setze $M := \emptyset$, $G_{\mathcal{R}_M} = (V_M^* + V_M^{**}, E_{\mathcal{R}_M}) := G_{\mathcal{F}}$ und $Y := V_M^{**}$;

while ($Y \neq \emptyset$) **do** {

Wähle ein $\hat{e} \in Y$ (nicht stabil) und setze $M := \{\hat{e}\}$;

Aktualisiere $G_{\mathcal{R}_M}$ (`updateRedGraph`);

while ($V_M^{**} - M \neq \emptyset$) **do** {

Finde Menge $X \subseteq V_M^{**}$ möglicher Erweiterungskanten (`findX`);

if ($X = \{\hat{e}^*\}$) **then** {

Unterbrich ggf. Schleife, und setze $Y := Y - \hat{e}$,

markiere dabei Startkante \hat{e} ggf. als stabil (`dontBranch`);

};

else Wähle ein $\hat{e}^* \in X$ (`branch`);

$M := M + \hat{e}^*$;

Aktualisiere $G_{\mathcal{R}_M}$ (`updateRedGraph`);

if ($G_{\mathcal{R}_M}$ ist konfliktfrei) **then return** M ;

};

};

if (alle $e \in Y$ sind stabil) **then return** “ S ist irreduzibel”;

else return “?”;

};

END.

Lemma 6.36 (Algorithmus Reducing1) *Der Algorithmus Reducing1 ist korrekt und hat einen Aufwand von $\mathcal{O}(n^9m^9)$.*

Beweis. Eine Entfernungsmenge $M \subseteq E_{diag}(S)$ wird nur zurückgegeben, wenn der zugehörige Reduktionsgraph $G_{\mathcal{R}_M}$ konfliktfrei ist. In diesem Fall ist M zulässig, und $[S^{tr}] - M$ ein H-Comparabilitygraph, bei dem jede transitive Orientierung auf $[S^{tr}] - M$ einen Plan $S^* \in \mathcal{S}_{SIJ}$ beschreibt, der S streng reduziert, $S^* \prec S$. Die Ausgabe “ S ist irreduzibel” erfolgt nur dann, wenn jede Schrägkante $e \in E_{diag}(S)$ entweder trivial-stabil ist (und damit nicht in V^{**} repräsentiert ist), oder als stabil erkannt wurde. Dabei wird eine Schrägkante genau dann als stabil erkannt, wenn ihre Entfernung aus $[S^{tr}]$ alternativlos zu unauflösbaren Konflikten führt. Das geschieht durch eine Reihe von Konflikten erster Ordnung, die letztendlich entweder zu einem direkten Konflikt führen, oder zu einem Konflikt höherer Ordnung, bei dem sämtliche beteiligten Schrägkanten unmittelbar (im nächsten Schritt) einen direkten Konflikt nach sich ziehen (vgl. detailliertere Darstellungen der Teilprozeduren im Anhang A.1). Sind alle Schrägkanten stabil, dann ist S irreduzibel (Definition 4.35). Im Falle der Ausgabe

“?” sind entweder alle Schrägkanten von $[S^{tr}]$ stabil, und die Stabilität mindestens einer Schrägkante konnte durch den gewählten Rechenweg nicht zweifelsfrei (alternativlos) nachgewiesen werden. Oder zu $[S^{tr}]$ existieren zulässige Entfernungsmengen, von denen keine durch den gewählten Rechenweg entdeckt wurde. Der Algorithmus ist damit korrekt.

Die Bestimmung der erweiterten Planimplikationsklassen ist mit dem Aufwand $\mathcal{O}(n^6m^6)$ möglich (Abschnitt 4.2.7). Die Bestimmung des Faktorgraphen (Lemma 6.2) und des Konsequenzgraphen (Lemma 6.5) ist genau wie die Überprüfung von $G_{\mathcal{R}_M}$ auf Konfliktfreiheit (beispielsweise mittels Tiefen- oder Breitensuche) mit dem Aufwand $\mathcal{O}(n^3m^3)$ verbunden. Die Aktualisierung des Reduktionsgraphen ist mit dem Aufwand $\mathcal{O}(n^5m^5)$ möglich (Lemma 6.9). Die Teilprozeduren `dontBranch` und `branch` haben jeweils einen Aufwand von $\mathcal{O}(|E_{\mathcal{K}_e}|) = \mathcal{O}(nm)$ bzw. $\mathcal{O}(|E_{diag}(S)|) = \mathcal{O}(n^2m^2)$. Sowohl die äußere, als auch die innere while-Schleife wird jeweils maximal $|V^{**}| = \mathcal{O}(|E_{diag}(S)|)$ oft durchlaufen, wenn die Erkennung eines Teilasts eines Entscheidungsknoten als Sackgasse logisch als Bestandteil der inneren Schleife aufgefasst wird. Der Aufwand innerhalb der inneren while-Schleife ist dann zum Einen durch den Algorithmus `findNextEdge` ($\mathcal{O}(n^4m^4)$, Lemma 6.30) bzw. den Algorithmus `updateRedGraph` ($\mathcal{O}(n^5m^5)$, Lemma 6.26) bestimmt. Zum Anderen durch den Aufwand zur Erkennung eines Teilasts als Sackgasse. Dieser ist nach Lemma 6.34 durch $\mathcal{O}(n^5m^5)$ beschränkt. Der Gesamtaufwand von `Reducing1` ist damit bestimmt durch $\mathcal{O}(n^6m^6 + |E_{diag}(S)| \cdot (n^5m^5 + |E_{diag}(S)| \cdot (n^4m^4 + n^5m^5 + n^5m^5))) = \mathcal{O}(n^6m^6 + n^2m^2 \cdot (n^2m^2 \cdot (n^5m^5))) = \mathcal{O}(n^9m^9)$. ■

NP-Test für IRRED?

Der Algorithmus `Reducing1` ist ein randomisiertes Verfahren mit zweiseitiger Versagenswahrscheinlichkeit (vgl. Kapitel 2). Ersetzen wir die Ausgabe “?” durch “ S ist irreduzibel”, dann wird aus `Reducing1` ein randomisiertes Verfahren mit einseitigem Fehler. Während die Ausgabe “ S ist reduzierbar” immer korrekt ist, kann die Ausgabe “ S ist irreduzibel” auch bei reduzierbaren Plänen erfolgen. `Reducing1` ist damit ein NP-Test für die Reduzierbarkeit. Ersetzen wir die Ausgabe “?” durch “ S ist reduzierbar”, dann kann zwar lediglich die Ausgabe “ S ist reduzierbar” falsch sein, aber `Reducing1` wird dadurch nicht zu einem NP-Test für IRRED. Während in diesem Fall zwar jede abzulehnende (d.h. reduzierbare) Instanz auf jedem Rechenweg abgelehnt wird, ist nicht klar, ob auch *jede* anzunehmende (d.h. irreduzible) Instanz, auf mindestens einem Rechenweg als irreduzibel erkannt werden kann.

REDUCING1 und getarnte stabile Schrägkanten

Möglicherweise existieren Instanzen, die zwar irreduzibel sind, bei denen die Stabilität der einzelnen Schrägkanten aber auf keinem möglichen Rechenweg (d.h. bei keiner zulässigen Auswahl der jeweiligen Erweiterungskanten) bewiesen werden kann. Klar ist jedoch, dass, wenn solche Instanzen existieren, sie getarnte stabile Schrägkanten enthalten müssen.

Lemma 6.37 (Reducing1 und getarnte stabile Schrägkanten) *Sei $S \in \mathcal{S}_{SIJ}$ ein normaler Plan. Enthält S keine getarnten stabilen Schrägkanten, dann versagt der Algorithmus `Reducing1` nicht (Ausgabe “?”).*

Beweis. Da die Prozedur `findNextEdge` (Seite 6.3.2) vor der Suche nach Konflikten höherer Ordnung explizit nach solchen der Ordnung 0 bzw. 1 sucht, wird die Stabilität von nichtgetarnten stabilen Schrägkanten bei ihrer Auswahl als Erweiterungskanten durch `Reducing1` erkannt (Satz 6.34 und Definition 6.32). Der Algorithmus fährt dann mit einer alternativen Erweiterungskante fort.

Angenommen, der Algorithmus liefert die Ausgabe “?”. Dann wurde weder eine zulässige Entfernungsmenge entdeckt, noch die Stabilität aller Startkanten aus Y festgestellt. D.h. der Algorithmus hat die Untersuchung mindestens einer Startkante $\hat{e} \in Y$ abgebrochen, ohne diese Kante als stabil zu markieren. Der Abbruch der inneren Schleife in `Reducing1` erfolgt jedoch nur, wenn jede der Schrägkanten einer Menge X von beteiligten Schrägkanten eines Konflikts in $G_{\mathcal{R}_M}$ zu einem direkten Konflikt führt. Nach Voraussetzung folgt jeder dieser direkten Konflikte alternativlos aus der Auswahl der jeweiligen Kante $\hat{e}^* \in X$ als Erweiterungskante. Damit ist jedoch die Entfernungsmenge M , bezüglich der der Reduktionsgraph $G_{\mathcal{R}_M}$ den Konflikt mit den Schrägkanten X enthält, bereits unzulässig (Satz 6.14). D.h. die Schrägkante \hat{e}' , um die die Entfernungsmenge M' in einem früheren Schritt erweitert worden ist, um zu $M = M' + \hat{e}'$ zu gelangen, ist stabil. Nach Voraussetzung

ist \hat{e}' jedoch keine getarnte stabile Schrägkante, und wäre von `Reducing1` erkannt worden (Widerspruch). ■

Folgerung 6.38 (Versagen von `Reducing1`) Sei $S \in \mathcal{S}_{SIJ}$ ein normaler Plan. Versagt der Algorithmus `Reducing1` (Ausgabe “?”), dann enthält S getarnte stabile Schrägkanten. Die Umkehrung dieser Aussage gilt nicht.

IRRED und P

Enthält der Ausgangsplan $S \in \mathcal{S}_{SIJ}$ also keine getarnten stabilen Schrägkanten, dann liefert der randomisierte Algorithmus `Reducing1` also mit einer Versagenswahrscheinlichkeit von $\varepsilon = 0$ eine richtige Antwort auf die Frage, ob S irreduzibel ist, oder nicht. `Reducing1` ist in diesem Fall nicht nur ein NP-Test sowohl für die Reduzierbarkeit, als auch für die Irreduzibilität von S , sondern sogar ein im weiteren Sinne deterministisches Verfahren. Die Richtigkeit der Antwort hängt nicht vom gewählten randomisierten Rechenweg ab. Damit liegt das Problem IRRED in P, sofern keine Pläne mit getarnten stabilen Schrägkanten existieren.

Satz 6.39 (Reducing1 und getarnte stabile Schrägkanten) Sei $S \in \mathcal{S}_{SIJ}$ ein normaler Plan. Enthält S keine getarnten stabilen Schrägkanten, dann kann der Algorithmus `Reducing1` mit dem Aufwand $\mathcal{O}(n^7 m^7)$ entscheiden, ob S irreduzibel ist.

Beweis. Nach Lemma 6.36 und 6.37 findet der Algorithmus `Reducing1` unter dieser Voraussetzung entweder eine zulässige Entfernungsmenge M oder er beweist die Irreduzibilität von S . Nach Satz 6.34 kann die Stabilität jeder stabilen Schrägkante in der Zeit $\mathcal{O}(n^5 m^5)$ erkannt werden. Das gilt insbesondere für jede Startkante $\hat{e} \in Y$. Zu der ersten Startkante $\hat{e} \in Y$ im Verlauf von `Reducing1`, die nicht stabil ist, findet `Reducing1` in der Zeit $\mathcal{O}(n^2 m^2 \cdot n^5 m^5) = \mathcal{O}(n^7 m^7)$ mit Sicherheit (Voraussetzung) eine zulässige Entfernungsmenge. Der Gesamtaufwand ist damit beschränkt durch $|E_{diag}(S)| \cdot \mathcal{O}(n^5 m^5) + \mathcal{O}(n^7 m^7) = \mathcal{O}(n^2 m^2 \cdot n^5 m^5 + n^7 m^7) = \mathcal{O}(n^7 m^7)$. ■

Folgerung 6.40 (IRRED und P) Existieren keine Probleminstanzen von IRRED mit getarnten stabilen Schrägkanten, dann liegt IRRED in P.

Die Umkehrung von Folgerung 6.40 gilt nicht. Darauf werden bei der Diskussion der Variante `Reducing2` im folgenden Abschnitt eingegangen. Zusammenfassend können wir die Eigenschaften von `Reducing1` folgendermaßen formulieren.

Bemerkung 6.41 (Reducing1) Der Algorithmus `Reducing1` durchsucht einen polynomiell beschränkten Teil des exponentiell großen Suchraums derart, dass er zu einem reduzierbaren Ausgangsplan $S \in \mathcal{S}_{SIJ}$ mit sehr hoher Wahrscheinlichkeit einen reduzierenden Plan $S^* \in \mathcal{S}_{SIJ}$ mit $S^* \prec S$ findet. Findet er keinen reduzierenden Plan, dann ist S mit sehr hoher Wahrscheinlichkeit irreduzibel. Für Pläne, die keine getarnten stabilen Schrägkanten enthalten, sinkt die Fehler- bzw. Versagenswahrscheinlichkeit auf 0.

Ein Beispiel

Bevor wir uns jedoch der Variante `Reducing2` zuwenden, wollen wir uns den Algorithmus `Reducing1` sowohl anhand des bekannten Beispielplans A , als auch anhand des Plans B_{24} mit nichttrivial-stabilen Schrägkanten (Seite 95) veranschaulichen.

Beispiel 6.42 (Reducing1) Wir betrachten zunächst erneut den Plan $A \in \mathcal{S}_{SIJ}$ aus den Beispielen 4.8, 4.44, 6.6 bzw. 6.21. In den Beispielen 6.6 und 6.21 haben wir den Faktorgraphen $G_{\mathcal{F}}$ (Abbildung 6.2) und den Konsequenzgraphen $G_{\mathcal{K}}$ (Tabelle 6.1), sowie die Reduktionsgraphen $G_{\mathcal{R}_M}$ zu den Entfernungsmengen $M_1 = \{\widehat{1.4}\}$, $M_2 = \{\widehat{1.4}, \widehat{2.5}\}$ (Abbildung 6.3), $M_3 = \{\widehat{2.7}\}$ und $M_4 = \{\widehat{2.7}, \widehat{1.7}\}$ (Abbildung 6.4) zu dem aus A abgeleiteten Plan $A_{48} \prec A$ betrachtet. Wir wollen diese Entfernungsmengen noch einmal mit der Notation von Algorithmus `Reducing1` diskutieren.

Sei also $A_{48} \in \mathcal{S}_{SIJ}$ der Ausgangsplan. $[A_{48}^{tr}]$ enthält 6 Schrägkanten, von denen keine trivial-stabil ist. D.h. die Menge $Y \subseteq E_{diag}(A_{48})$ enthält alle 6 Schrägkanten, $Y = E_{diag}(A_{48})$. Nehmen wir an, wir

wählen die Kante $\hat{e} = \widehat{1.4}$ als erste Startkante. Dann gilt $M = M_1$. Der Reduktionsgraph $G_{\mathcal{R}_M}$ ist in Abbildung 6.3 (links) auf Seite 127 dargestellt. Er enthält keinen direkten Konflikt, aber einen Konflikt erster Ordnung, die Kante $\widehat{2.5}$. Wir wählen also die Kante $\widehat{2.5}$ als alternativlose Erweiterungskante aus. Der aktualisierte Reduktionsgraph $G_{\mathcal{R}_M}$ zu der erweiterten Entfernungsmenge $M = \{\widehat{1.4}, \widehat{2.5}\} = M_2$ ist in Abbildung 6.3 (rechts) dargestellt. Er ist konfliktfrei, d.h. die Entfernungsmenge M ist zulässig. Der Algorithmus terminiert. Der resultierende Plan ist der Plan $A_{348.10} \prec A_{48}$.

Nehmen wir nun an, wir wählen die Kante $\hat{e} = \widehat{2.7}$ als erste Startkante. Dann gilt $M = M_3$. Der Reduktionsgraph ist in Abbildung 6.4 (links) dargestellt. Er enthält weder einen direkten Konflikt, noch einen Konflikt erster Ordnung. Er enthält jedoch die folgenden 4 Konflikte 2-ter Ordnung: $X_1 = \{\widehat{1.4}, \widehat{1.7}\}$, $X_2 = \{\widehat{1.4}, \widehat{9.7}\}$, $X_3 = \{\widehat{2.5}, \widehat{1.7}\}$ und $X_4 = \{\widehat{2.5}, \widehat{9.7}\}$. Nehmen wir an, der Konflikt, den wir mit Hilfe der Prozedur *findNextEdge* finden, ist der Konflikt $X = X_2$, und nehmen wir an, wir wählen von den beiden zur Auswahl stehenden Kanten die Kante $\hat{e}^* = \widehat{1.7}$ als Erweiterungskante. Der aktualisierte Reduktionsgraph $G_{\mathcal{R}_M}$ zu der erweiterten Entfernungsmenge $M = \{\widehat{2.7}, \widehat{1.7}\} = M_4$ ist in Abbildung 6.4 (rechts) dargestellt. Er enthält einen Konflikt erster Ordnung. Die Auswahl der Kante $\hat{e}^* = \widehat{9.7}$ ist damit alternativlos. Im nächsten Iterationsschritt zeigt sich, dass die Entfernungsmenge $M = \{\widehat{2.5}, \widehat{1.7}, \widehat{9.7}\}$ zulässig ist. Der resultierende Plan ist der Plan $A_{4567} \prec A_{48}$. Jede andere Wahl der Menge X im zweiten Reduktionsschritt, und jede Wahl einer der beiden jeweiligen Kanten führt im dritten Reduktionsschritt zu einer zulässigen Entfernungsmenge, die entweder zu A_{4567} , oder zu $A_{348.10}$ führt.

Betrachten wir nun den Algorithmus *Reducing1* für den irreduziblen Beispielplan B_{24} auf Seite

2.5		$P_1 + \bar{P}_2$	$\bar{P}_1 + 3.5$	$\bar{P}_1 + 7.5$	
3.5	$P_1 + P_2$	$P_1 + \bar{P}_2$	$P_1 + 2.5$	$\bar{P}_1 + 7.5$	
7.5	$P_1 + P_2$	$P_1 + \bar{P}_2$	$P_1 + 2.5$	$P_1 + 3.5$	$P_1 + 9.5$
9.5		$P_1 + \bar{P}_2$	$\bar{P}_1 + 7.5$		

Tabelle 6.2: Der Konsequenzgraph $G_{\mathcal{K}}$ zum Plan B_{24} .

95 (Abbildung 4.10). Der Plangraph B_{24}^{tr} besteht aus 2 verschiedenen Planimplikationsklassen P_1 und P_2 , sowie 9 Schrägkanten, von denen 5 trivial-stabil in P_1 sind. Die anderen 4 Schrägkanten sind nichttrivial-stabil. Die Menge $Y \subseteq E_{diag}(B_{24})$ enthält damit nur die 4 nicht trivial-stabilen Schrägkanten. Aus dem Konsequenzgraphen $G_{\mathcal{K}}$ (Tabelle 6.2) geht sofort hervor, dass die Kanten $\widehat{3.5}$ und $\widehat{7.5}$ stabil sein müssen, da beide sowohl $P_1 + P_2$, als auch $P_1 + \bar{P}_2$ nach sich ziehen. Wir können beide Schrägkanten daher als stabil markieren (fett gedruckt).

Nehmen wir an, wir wählen die Kante $\widehat{2.5}$ als Startkante, $M = \{\widehat{2.5}\}$. Der resultierende Reduktions-

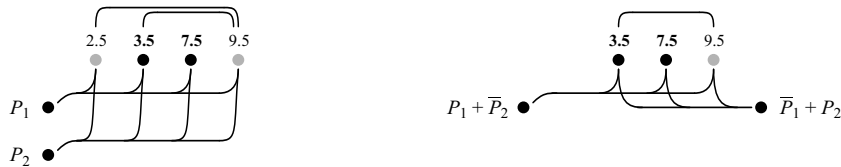


Abbildung 6.5: Der Faktorgraph $G_{\mathcal{F}}$ (links) und der Reduktionsgraph $G_{\mathcal{R}_M}$ bezüglich der Entfernungsmenge $M = \{\widehat{2.5}\}$ zum Plan B_{24} .

graph $G_{\mathcal{R}_M}$ (Abbildung 6.5, rechts) enthält keine direkten Konflikte, aber 3 Konflikte erster Ordnung. Wählen wir nun beispielsweise die (stabile) Kante $\hat{e}^* = \widehat{3.5}$ als neue Erweiterungskante, dann kommen wir zu einer unzulässigen Entfernungsmenge. Da sich die Wahl von $\widehat{3.5}$ alternativlos aus der Wahl der Startkante $\widehat{2.5}$ ergeben hat, können wir die Startkante $\hat{e} = \widehat{2.5}$ als stabil markieren. Als neue Startkante steht nun nur noch die einzige verbleibende noch nicht als stabil erkannte Schrägkante $\widehat{9.5}$ zur

Auswahl. Für diese Startkante können wir genauso schnell die Stabilität nachweisen. Da sämtliche nichttrivial-stabilen Schrägkanten als stabil erkannt worden sind, erkennt der Algorithmus `Reducing1` (auf jedem möglichen Rechenweg) die Irreduzibilität von B_{24} .

Wählen wir im ersten Reduktionsschritt zu der Entfernungsmenge $M = \{\widehat{2.5}\}$ keine der beiden bereits als stabil bekannten Kanten $\widehat{3.5}$ bzw. $\widehat{7.5}$, sondern die Kante $\hat{e}^* = \widehat{9.5}$ als Erweiterungskante, dann erkennen wir die Stabilität der Startkante $\widehat{2.5}$ erst im nächsten Reduktionsschritt. ■

6.4.2 Der Algorithmus REDUCING2

Der Algorithmus `Reducing1` hat also eine polynomiell beschränkte Laufzeit, und liefert mit einer sehr hohen Wahrscheinlichkeit ein richtiges Ergebnis. Legen wir dagegen Wert auf eine unbedingte Sicherheit des Ergebnisses, dann können wir uns nicht unbedingt auf eine beliebige Wahl eines Astes an jedem Entscheidungsknoten verlassen. Insbesondere, um die Irreduzibilität eines Ausgangsplans beweisen zu können, müssen wir alle möglichen Äste eines Entscheidungsknotens ausschließen. Dadurch ist eine polynomielle Beschränkung der Laufzeit grundsätzlich nicht mehr möglich. Wie wir jedoch sehen werden, können wir die Laufzeit in einigen Fällen dennoch polynomiell abschätzen. Vor allem Probleminstanzen, die keine getarnten stabilen Schrägkanten enthalten, eröffnen diese Möglichkeit.

Die Grundidee einer gründlichen, aber dennoch schnellen Suche

Der Algorithmus `Reducing2` unterscheidet sich vom Algorithmus `Reducing1` lediglich dadurch, dass der Teil des Suchbaums zu einer Startkante $\hat{e} \in Y$ nicht gleich bei einem unauflösbaren Konflikt in einem Ast verworfen wird. Vielmehr werden mit einer Tiefensuchstrategie solange alle Verzweigungen betrachtet, die sich bei der Auflösung von Konflikten höherer Ordnung ergeben, bis entweder eine zulässige Entfernungsmenge entdeckt wird, oder die Stabilität der Startkante zweifelsfrei feststeht. Die Stabilität einer Startkante steht dabei unter Umständen erst nach der Betrachtung aller Verzweigungen zu dieser Kante fest. Andererseits kommt bei `Reducing2` die Betrachtung alternativer Startkanten erst in Betracht, wenn von allen vorhergehenden Startkanten die Stabilität nachgewiesen worden ist. Da diese Schrägkanten nicht mehr für die Erweiterung von nicht zulässigen Entfernungsmengen in Frage kommen, reduziert sich die Größe des Suchraums durch jede Schrägkante, deren Stabilität festgestellt wird. Können wir die Reihenfolge, in der die Schrägkanten eines irreduziblen Ausgangsplans als Startkanten betrachtet werden, so wählen, dass zuerst Schrägkanten betrachtet werden, deren Stabilität sich leicht, d.h. ohne großen Suchbaum, nachweisen lässt, dann können wir dadurch den Gesamtaufwand trotz der exponentiellen Abschätzung gering halten.

Bestimmung einer guten Reihenfolge der Kanten aus Y

Wie erkennen wir Schrägkanten, deren mögliche Stabilität sich leichter nachweisen lässt, als die mögliche Stabilität anderer Schrägkanten? Die Stabilität von nicht getarnten stabilen Schrägkanten lässt sich ohne jegliche Verzweigung in polynomieller Laufzeit erkennen (Satz 6.34). In diese Gruppe fallen die trivial-stabilen Schrägkanten, die bei dieser Betrachtung keine Rolle spielen, da sie bereits bei der Bestimmung des Faktorgraphen $G_{\mathcal{F}}$ erkannt werden, und als Kanten ihrer erweiterten Planimplikationsklasse nicht in der Menge $Y = V^{**}$ enthalten sind. In diese Gruppe fallen aber auch beispielsweise die nichttrivial-stabilen Schrägkanten, die inkonsistente Konsequenzen haben. Der Plan B_{24} auf Seite 4.3.4 enthält solche Kanten (Beispiel 6.42).

Getarnte stabile Schrägkanten dagegen sind nichttrivial-stabile Schrägkanten, deren Stabilität sich erst nach der Betrachtung aller Äste einer mindestens zweistufigen Verzweigung erkennen lässt. Dazu dürfen die Entfernungsmengen zu den entsprechenden Entscheidungsknoten weder direkte Konflikte, noch Konflikte erster Ordnung enthalten. Solche Konflikte entstehen in erster Linie dann, wenn die zuletzt zu der Entfernungsmenge hinzugefügte Erweiterungskante Konsequenzen nach sich zieht, die eine oder mehrere Planimplikationsklassen betreffen. Beschränken sich die Konsequenzen einer Erweiterungskante auf neue Γ -Relationen von Schrägkanten untereinander (vgl. Abschnitt 4.3.3, Abbildung 4.9), dann können Konflikte niedriger Ordnung daraus unter Umständen nur entstehen, wenn einige dieser beteiligten Schrägkanten bereits als stabil erkannt worden sind.

Wird eine getarnte stabile Schrägkante in einem irreduziblen Plan als eine der letzten Startkanten untersucht, dann sind fast alle Schrägkanten des Plans bereits als stabil bekannt, und die Wahrschein-

lichkeit ist groß, dass die getarnte stabile Schrägkante nicht mehr “getarnt” ist. Um die Chancen zu erhöhen, sämtliche möglicherweise vorhandenen getarnten stabilen Schrägkanten eines Ausgangsplans im hinteren Teil in der angeordneten Menge $Y = V^{**}$ anzuordnen, können wir die Reihenfolge der Kanten in Y daher nach qualitativen Merkmalen ihrer jeweiligen Konsequenzen bestimmen.

Wir gruppieren die nichttrivial-stabilen Schrägkanten eines Ausgangsplans in zwei Gruppen. Gruppe 1 enthält alle Schrägkanten, die Konsequenzen zwischen jeweils zwei verschiedenen Planimplikationsklassen induzieren. Die Kanten dieser Gruppe werden als erste in Y angeordnet. Die zweite Gruppe enthält alle verbleibenden Schrägkanten. Die Kanten dieser Gruppe werden hinter denen aus der ersten Gruppe in Y angeordnet. Innerhalb dieser Gruppe erscheint eine Sortierung nach der Anzahl der Konsequenzen unter der Beteiligung von Planimplikationsklassen sinnvoll. Getarnte stabile Schrägkanten gehören mit sehr hoher Wahrscheinlichkeit immer zu der zweiten Gruppe, und werden innerhalb dieser Gruppe mutmaßlich weiter hinten angeordnet (vgl. Abschnitt 4.3.3, Abbildung 4.9).

Bemerkung 6.43 (Bedeutung der Reihenfolge in Y) *Werden die nichttrivial-stabilen Schrägkanten in $Y = V^{**}$ so angeordnet, dass Kanten, die viele Konsequenzen zwischen Planimplikationsklassen untereinander, und zwischen Planimplikationsklassen und Schrägkanten induzieren, vor Kanten liegen, die nur wenige Konsequenzen zwischen Planimplikationsklassen und Schrägkanten induzieren, dann steigt dadurch die Wahrscheinlichkeit, dass bei der Betrachtung einer getarnten stabilen Schrägkante als Startkante der Suchbaum durch die Kenntnis der Stabilität aller vorherigen Startkanten soweit eingeschränkt ist, dass ein schneller Nachweis der Stabilität möglich wird.*

Der Algorithmus REDUCING2

Mit dieser Idee zur Sortierung der Schrägkanten in der Menge Y im Hinterkopf, können wir uns nun der konkreten Gestaltung von `Reducing2` widmen. Die Betrachtung aller Äste einer Verzweigung wird durch die sich selbst aufrufende Prozedur `findPossibleEdges` sicher gestellt. Genau wie im Falle von `Reducing1` ist hier lediglich eine etwas vereinfachte Darstellung von `Reducing2` angegeben. Der Grad der Vereinfachung hier ist gegenüber `Reducing1` jedoch bedeutend geringer. Eine kommentierte Darstellung von `Reducing2` und `findPossibleEdges` ist in den Anhängen A.3 bzw. A.1 enthalten.

Algorithmus REDUCING2

Input: $[S^{tr}]$ zu normalem Plan $S \in \mathcal{S}_{SIJ}$.

Output: zulässige Entfernungsmenge M oder “ S ist irreduzibel”.

```

findPossibleEdges( $M, G_{\mathcal{R}_M}, G_{\mathcal{K}}$ ) {
  while ( $V_M^{**} - M \neq \emptyset$ ) do {
    Bestimme alternativloses  $\hat{e}^*$  (nicht stabil) nach Schritt (1), (2) oder (3)
    in findNextEdge( $M, G_{\mathcal{R}_M}, G_{\mathcal{K}}$ ), und setze  $X := \{\hat{e}^*\}$ ,
    oder bestimme  $X := W \cap V_M^{**}$  nach Schritt (3);
    if not (“ $M$  nicht zulässig erweiterbar”) then {
      while ( $X \neq \emptyset$ ) do {
        Wähle ein  $\hat{e}^* \in X$  (nicht stabil);
         $M := M + \hat{e}^*$ ;
        updateRedGraph( $M, G_{\mathcal{R}_M}, G_{\mathcal{K}}, \hat{e}^*$ );
        if ( $G_{\mathcal{R}_M}$  ist konfliktfrei) then return  $M$ ;
        else findPossibleEdges( $M, G_{\mathcal{R}_M}, G_{\mathcal{K}}$ );
      };
    };
  };
}
BEGIN {
  Bestimme erweiterte Planimplikationsklassen  $\mathcal{P}_S = \{P_1, \dots, P_k\}$ ,
  Faktorgraph  $G_{\mathcal{F}}$  und Konsequenzgraph  $G_{\mathcal{K}}$ ;
  Setze  $M := \emptyset$ ,  $G_{\mathcal{R}_M} = (V_M^* + V_M^{**}, E_{\mathcal{R}_M}) := G_{\mathcal{F}}$  und  $Y := V_M^{**}$ ;
  while ( $Y \neq \emptyset$ ) do {
    Wähle ein  $\hat{e} \in Y$  (nicht stabil);
     $M := \{\hat{e}\}$ ;
  };
}

```

```

    updateRedGraph( $M, G_{\mathcal{R}_M}, G_{\mathcal{K}}, \hat{e}$ );
    findPossibleEdges( $M, Y, \hat{e}$ );
    Markiere  $\hat{e}$  als stabil;
     $Y := Y - \hat{e}$ ;
  };
  return "S ist irreduzibel";
};
END.

```

Lemma 6.44 (Algorithmus Reducing2) *Der Algorithmus Reducing2 ist korrekt, und hat eine durch $\mathcal{O}(n^5 m^5 \cdot 2^{|E_{diag}(S)|}) = \mathcal{O}(n^5 m^5 \cdot 2^{n^2 m^2})$ beschränkte maximale Laufzeit.*

Beweis. Der Algorithmus betrachtet zu jeder Startkante $\hat{e} \in Y = V^{**}$ jede sinnvolle Erweiterung der Entfernungsmenge und sämtliche ihrer Verzweigungen, bis entweder eine zulässige Entfernungsmenge M gefunden wird, oder jede Entfernungsmenge als (endgültig) unzulässig bewiesen ist. Da die Konflikte im Reduktionsgraphen $G_{\mathcal{R}_M}$ jeweils unabhängig voneinander gelöst werden müssen, ist eine Verzweigung in der Auswahl des zu lösenden Konflikts im Schritt `findNextEdge` hierzu nicht notwendig. Andererseits folgt aus Lemma 6.22 bzw. Satz 6.23, dass jede zulässige Entfernungsmenge auf diese Weise entdeckt werden kann. Der Algorithmus ist somit korrekt.

Bei Vermeidung von mehrfacher Betrachtung einer Entfernungsmenge M in verschiedenen Teilästen des Suchbaums wird jede mögliche Entfernungsmenge $M \subseteq E_{diag}(S)$ höchstens einmal betrachtet.¹ Bei jeder Betrachtung werden die Kanten eines gewählten Konflikts mit dem Algorithmus `findNextEdge` in $\mathcal{O}(|V_{\mathcal{R}_M}|^2) = \mathcal{O}(n^4 m^4)$ bestimmt (Lemma 6.30). Die Aktualisierung des Reduktionsgraphen nach der Erweiterung von M um eine dieser Kanten erfolgt in $\mathcal{O}(n^5 m^5)$ (Lemma 6.26). Die maximale Anzahl solcher Betrachtungen ist durch $\mathcal{O}(2^{|E_{diag}(S)|}) = \mathcal{O}(2^{n^2 m^2})$ beschränkt. Der Gesamtaufwand ist damit beschränkt durch $\mathcal{O}((n^4 m^4 + n^5 m^5) \cdot 2^{|E_{diag}(S)|}) = \mathcal{O}(n^5 m^5 \cdot 2^{n^2 m^2})$. ■

Für reduzierbare Pläne $S \in \mathcal{S}_{SIJ}$ unterscheidet sich `Reducing2` kaum von `Reducing1`. Während `Reducing1` solche Pläne in polynomieller Laufzeit mit sehr hoher Wahrscheinlichkeit reduzieren kann, findet `Reducing2` mit Sicherheit einen streng reduzierenden Plan, und diesen mit hoher Wahrscheinlichkeit mit derselben Laufzeit wie `Reducing1`. Der Grund hierfür ist, dass der Algorithmus terminiert, sobald eine zulässige Entfernungsmenge entdeckt worden ist.

Suchraum häufig nicht sehr groß

Obwohl die Größe des Suchraums nur exponentiell abgeschätzt werden kann, ist er typischerweise vergleichsweise klein. Insbesondere die Anzahl der irreduziblen Pläne S^* zu einem gegebenen Ausgangsplan S ist für viele Instanzen $S \in \mathcal{S}_{SIJ}$ erstaunlich klein. Die polynomielle Abschätzung der Laufzeit von `Reducing2` ist damit in vielen Fällen sehr grob.

In Tabelle 6.3 ist diese Anzahl für eine kleine Stichprobe von Instanzen vom Format 3×4 bis zum Format 3×12 mit bis zu 34 verschiedenen Planimplikationsklassen angegeben. Die Bestimmung der Anzahl der irreduziblen Pläne erfolgte dabei mit Hilfe des Softwarepakets LiSA [46]. Die Auswahl der Instanzen erfolgte mit Hilfe eines Programms zur Enumeration aller Pläne eines gewählten Formats mit genau k verschiedenen Planimplikationsklassen. Bemerkenswert dabei ist unter anderem, dass von den 20 Instanzen mit 30 oder mehr Planimplikationsklassen, zu denen jeweils über eine Milliarde verschiedener reduzierender Pläne existieren könnten ($2^{|\mathcal{P}^S|}$, Satz 4.40), gerade einmal 3 mehr als 10.000 verschiedene irreduzible Pläne besitzen. Fast die Hälfte (9) dieser Instanzen besitzt deutlich weniger als 1.000 irreduzible Pläne, 4 Instanzen sogar weniger als 100.

¹Eine solche Vermeidung ist möglich, aber in der gewählten Darstellung des Verfahrens nicht explizit berücksichtigt. In der gegebenen Formulierung ist der Aufwand formal nur beschränkt durch $\mathcal{O}(|E_{diag}(S)|^{|E_{diag}(S)|}) = \mathcal{O}((n^2 m^2)^{n^2 m^2})$.

$ \mathcal{P}_S $	$\mathcal{F}_{[S^{tr}]}$	Format	$2^{ \mathcal{P}_S }$	# irred. Pläne	$ \mathcal{P}_S ^3$	$ E_{reg}(S) ^2$
5	1	3×4	32	1	125	900
5	1	3×4	32	1	125	900
6	1	3×4	64	5	216	900
6	2	3×4	64	4	216	900
6	2	3×4	64	4	216	900
7	2	3×4	128	3	343	900
8	2	3×4	256	10	512	900
8	4	3×4	256	6	512	900
8	4	3×4	256	13	512	900
10	4	3×4	1.024	11	1.000	900
12	4	3×4	4.096	68	1.728	900
12	8	3×4	4.096	38	1.728	900
12	8	3×4	4.096	47	1.728	900
14	8	3×4	16.384	107	2.744	900
16	10	3×4	65.536	131	4.096	900
18	12	3×4	262.144	188	5.832	900
18	14	3×4	262.144	426	5.832	900
20	10	3×5	1.048.576	303	8.000	2.025
20	10	3×5	1.048.576	250	8.000	2.025
22	12	3×5	4.194.304	1.052	10.648	2.025
22	12	3×5	4.194.304	732	10.648	2.025
25	15	3×5	33.554.432	1.546	15.625	2.025
25	15	3×5	33.554.432	351	15.625	2.025
25	15	3×7	33.554.432	3.118	15.625	7.056
25	15	3×7	33.554.432	282	15.625	7.056
25	15	3×8	33.554.432	258	15.625	11.664
25	15	3×8	33.554.432	483	15.625	11.664
27	20	3×5	134.217.728	4.223	19.683	2.025
27	20	3×5	134.217.728	3.942	19.683	2.025
30	18	3×5	1.073.741.824	3.311	27.000	2.025
30	20	3×8	1.073.741.824	6.317	27.000	11.664
31	20	3×6	2.147.483.648	5.011	29.791	3.969
32	8	3×9	4.294.967.296	443	32.768	18.225
32	8	3×9	4.294.967.296	263	32.768	18.225
32	8	3×10	4.294.967.296	37	32.768	27.225
32	8	3×10	4.294.967.296	16	32.768	27.225
32	8	3×11	4.294.967.296	1.890	32.768	39.204
32	8	3×11	4.294.967.296	78	32.768	39.204
32	8	3×12	4.294.967.296	164	32.768	54.756
32	8	3×12	4.294.967.296	2.869	32.768	54.756
32	12	3×9	4.294.967.296	40.044	32.768	18.225
32	20	3×9	4.294.967.296	2.074	32.768	18.225
32	20	3×9	4.294.967.296	127	32.768	18.225
32	20	3×9	4.294.967.296	6.937	32.768	18.225
32	20	3×9	4.294.967.296	37.735	32.768	18.225
34	8	3×10	17.179.869.184	28	39.304	27.225
34	8	3×10	17.179.869.184	253	39.304	27.225
34	18	3×9	17.179.869.184	3.674	39.304	18.225
34	18	3×9	17.179.869.184	24.872	39.304	18.225

Tabelle 6.3: Die Anzahl der verschiedenen irreduziblen Pläne $S^* \prec S$ (Spalte 5) zu einer willkürlich gewählten Stichprobe von Probleminstanzen. Neben einigen Kennzahlen zur Beschreibung dieser Instanzen (Spalten 1 bis 3) sind auch einige daraus abgeleitete Kennzahlen angegeben (Spalte 4, und Spalten 6 und 7).

Polynomielle Beschränkung der Laufzeit

Für eine große Gruppe von Instanzen können wir den Aufwand von `Reducing2` tatsächlich sogar polynomiell abschätzen.

Lemma 6.45 (Reducing2 und getarnte stabile Schrägkanten) *Sei $S \in \mathcal{S}_{SIJ}$ ein normaler Plan, der keine getarnten stabilen Schrägkanten enthält. Dann ist die Laufzeit von Algorithmus `Reducing2` für S beschränkt durch $\mathcal{O}(n^7 m^7)$.*

Beweis. Enthält S keine getarnten stabilen Schrägkanten, dann wird jede stabile Schrägkante alternativlos erkannt, und Verzweigungen entstehen nur bei zulässig erweiterbaren Entfernungsmengen M . Jede Erweiterung von M um eine nichtstabile Schrägkante \hat{e}^* führt erneut zu einer zulässig erweiterbaren Entfernungsmenge $M^* = M + \hat{e}^*$. Da der Algorithmus terminiert, sobald eine zulässige Entfernungsmenge gefunden ist, erfolgt ein Rückschritt zu einer bereits besuchten Verzweigung im Suchbaum nur im Falle einer stabilen Schrägkante. Der Ablauf von `Reducing2` ist damit identisch zu dem von `Reducing1`. Insbesondere ist damit auch der Aufwand von `Reducing2`, genau wie der von `Reducing1`, durch $\mathcal{O}(n^7 m^7)$ (Satz 6.39) beschränkt. ■

Folgerung 6.46 (Reducing1 und Reducing2) *Sei $S \in \mathcal{S}_{SIJ}$ ein normaler Plan, der keine getarnten stabilen Schrägkanten enthält. Dann sind die Algorithmen `Reducing1` und `Reducing2` identisch.*

REDUCING2 und Pläne mit getarnten stabilen Schrägkanten

Bei Plänen, die keine getarnten stabilen Schrägkanten enthalten spielt es also keine Rolle, welche der beiden Ausprägungen unseres Reduktionsverfahrens wir benutzen. Wir finden für reduzierbare Pläne immer einen reduzierenden Plan, und erkennen irreduzible Pläne mit polynomiell Aufwand. Wie stehen unsere Chancen bei Plänen mit getarnten stabilen Schrägkanten? Sind sie reduzierbar, dann ist `Reducing1` ein NP-Test für die Bestimmung eines reduzierenden Plans (Abschnitt 6.4.1). Abhängig vom gewählten Rechenweg, besteht die Chance, eine zulässige Entfernungsmenge zu entdecken. Das gleiche gilt für `Reducing2`. Genau wie bei `Reducing1` besteht auch hier die Möglichkeit, in jedem Schritt die richtige Entscheidung zu treffen.

Fangen wir mit einer nicht stabilen Schrägkante an, und finden eine zulässige Entfernungsmenge ohne die Betrachtung von mehreren Ästen einer Verzweigung, dann können wir die Reduzierbarkeit des Ausgangsplans sogar mit dem Aufwand $\mathcal{O}(n^6 m^6)$ entscheiden. Die äußere Schleife wird dann nur genau einmal begonnen, und der Aufwand innerhalb der inneren Schleife ist mit dem Argument zu der Aktualisierung des Reduktionsgraphen innerhalb dieser Schleife (Satz 6.34) nur durch den Aufwand von `findNextEdge` ($\mathcal{O}(n^4 m^4)$) beschränkt.

Untersuchen wir dagegen irreduzible Pläne mit getarnten stabilen Schrägkanten, dann kann der Aufwand unter Umständen dennoch polynomiell beschränkt bleiben. Entscheidend hierfür ist die Wahl eines Rechenweges von `Reducing2`, bei dem die getarnten stabilen Schrägkanten nicht schon früh als mögliche Startkanten einer Entfernungsmenge untersucht werden. Sortieren wir die Schrägkanten aus der Menge Y der jeweiligen Startkanten, wie zu Beginn dieses Abschnitts diskutiert, nach nichtsteigender Anzahl von Konsequenzen, die Planimplikationsklassen betreffen (Bemerkung 6.43), dann haben wir genau dann eine Chance auf den Nachweis der Irreduzibilität eines Plans mit getarnten stabilen Schrägkanten mit polynomiell Aufwand, wenn der Plan neben diesen getarnten stabilen Schrägkanten auch viele nicht-getarnte stabile Schrägkanten enthält.

Bemerkung 6.47 (Anzahl von Schrägkanten) *Sei $S \in \mathcal{S}_{SIJ}$ ein normaler Plan, der getarnte stabile Schrägkanten enthält. Dann enthält der H -Comparabilitygraph $[S^{tr}]$ zu S eine große Anzahl von Schrägkanten, die nicht alle getarnte stabile Schrägkanten sein können.*

Begründung. Eine getarnte stabile Schrägkante gewinnt ihre Tarnung aus der Existenz vieler verschiedener transitivierender Wege (vgl. Abschnitt 4.3.3). Dadurch ergeben sich in $[S^{tr}]$ fast vollständige Teilgraphen als Teilstrukturen (vgl. Abbildung 4.9) mit einer großen Anzahl von Schrägkanten. Diese Konstellationen induzieren von dem Hintergrund der Bemerkung 4.57 (Konstruktion von Beispielen) eine noch viel größere Anzahl von Schrägkanten. Die Funktionen, die diese Schrägkanten dabei erfüllen

müssen (Bemerkung 4.57), vertragen sich dabei sehr schlecht mit den Eigenschaften von nichttrivial-stabilen Schrägkanten. Insbesondere induziert jeder minimale reguläre Weg der Kantenlänge l mindestens $l - 1$ Eckkanten. Die Existenz eines Plans S , der zu einem erheblichen Anteil aus getarnten (nichttrivial-) stabilen Schrägkanten besteht, muss daher als absolut unwahrscheinlich betrachtet werden. ■

Bemerkung 6.48 (polynomielle Laufzeit) Sei $S \in \mathcal{S}_{SIJ}$ ein normaler Plan, der getarnte stabile Schrägkanten enthält. Der Algorithmus `Reducing2` kann unter Umständen dennoch in polynomiell beschränkter Laufzeit entscheiden, ob S irreduzibel ist, oder nicht.

Begründung. Der Algorithmus `Reducing2` ist ein randomisiertes Verfahren. Ist S reduzierbar, dann kann bei geeigneter Wahl der Startkante unter Umständen in der Zeit $\mathcal{O}\left(|E_{diag}(S)| \cdot |V_{\mathcal{R}_M}|^2\right) = \mathcal{O}(n^6 m^6)$ ein reduzierender Plan bestimmt werden (co-IRRED liegt in NP). Sei S also irreduzibel. Dann ist der Nachweis der Stabilität einer getarnten stabilen Schrägkante $\hat{e} \in Y$ möglicherweise erst nach der Betrachtung eines erheblichen Teils des Suchbaums zu dieser Kante möglich. Andererseits ist es jedoch auch möglich, dass der Nachweis der Stabilität *aller anderen* Schrägkanten jeweils in polynomieller Zeit möglich ist. Wird in einem solchen Fall die getarnte stabile Schrägkante \hat{e} als eine der letzten Startkanten betrachtet, dann kann ihre Stabilität sofort erkannt werden, da mindestens einer der Konflikte, die sie induziert, nicht mehr auflösbar ist, da alle beteiligten Schrägkanten stabil sind. ■

NP-Test für IRRED

Existieren also keine Probleminstanzen mit getarnten stabilen Schrägkanten, dann sind die Verfahren `Reducing1` und `Reducing2` identisch, und lösen beide das Problem IRRED für jede Probleminstanz mit polynomiellem Aufwand. Das Problem IRRED liegt damit in P. Existieren andererseits Probleminstanzen mit getarnten stabilen Schrägkanten, dann können wir für diese Instanzen das Problem IRRED dennoch durch einen nichtdeterministischen Algorithmus mit polynomiell beschränkter maximaler Laufzeit lösen, sofern der Anteil der getarnten stabilen Schrägkanten an allen Schrägkanten klein ist. Durch eine geeignete Sortierung der Kanten in der Menge Y haben wir Einfluss auf die Wahrscheinlichkeit, einen günstigen Rechenweg zu finden. Existieren keine Probleminstanzen mit einem hohen Anteil an getarnten stabilen Schrägkanten, dann liegt IRRED nicht nur in co-NP (Folgerung 4.4), sondern auch in NP, und damit in $ZPP^* = NP \cap co-NP$.

Satz 6.49 (NP-Test für IRRED) Existiert zu jedem irreduziblen Plan $S \in \mathcal{S}_{SIJ}$, der getarnte stabile Schrägkanten enthält, mindestens eine Nummerierung der Schrägkanten $\{\hat{e}_1, \dots, \hat{e}_t\} = E_{diag}(S)$, so dass die Stabilität einer Startkante \hat{e}_i unter der Voraussetzung der Stabilität von $\hat{e}_1, \dots, \hat{e}_{i-1}$ durch eine Folge alternativer Erweiterungskanten erkannt werden kann, dann liegt das Problem IRRED in $ZPP^* = NP \cap co-NP$.

Beweis. Der Algorithmus `Reducing1` ist ein co-NP-Test für IRRED. Jeder reduzierbare Plan kann auf mindestens einem erlaubten Rechenweg in polynomieller Zeit streng reduziert werden. Daher muss nur noch gezeigt werden, dass auch ein NP-Test für IRRED existiert. Wir zeigen, dass unter der Voraussetzung `Reducing2` ein solcher NP-Test für IRRED ist. Dazu müssen wir zeigen, dass die Irreduzibilität einer beliebigen Instanz von IRRED auf mindestens einem erlaubten Rechenweg in polynomiell beschränkter Laufzeit nachweisbar ist. Für Pläne, die keine getarnten stabilen Schrägkanten enthalten, löst `Reducing2` das Problem IRRED sogar immer in polynomiell beschränkter Zeit (Lemma 6.45). Es bleibt also zu zeigen, dass `Reducing2` die Stabilität aller Schrägkanten eines beliebigen irreduziblen Plans $S \in \mathcal{S}_{SIJ}$, der getarnte stabile Schrägkanten enthält, auf mindestens einem erlaubten Rechenweg nachweisen kann. Wählen wir die Startkante $\hat{e} \in Y$ beim i -ten Durchlauf der äußeren Schleife von `Reducing2` in der geeigneten Reihenfolge $\hat{e}_1, \dots, \hat{e}_t$, dann können wir nach Voraussetzung die Stabilität von \hat{e} auf einem erlaubten Rechenweg mit polynomiellen Aufwand nachweisen. ■

IRRED und NPI

Führt die diskutierte Sortierung der Kanten aus Y dazu (Bemerkung 6.43), dass dadurch für jede Instanz mit getarnten stabilen Schrägkanten ein erlaubter Rechenweg gefunden wird, der den polynomiellen Nachweis der Reduzierbarkeit bzw. der Irreduzibilität ermöglicht, dann liegt IRRED sogar

trotz der Existenz von Instanzen mit getarnten stabilen Schrägkanten in $P \subseteq NP \cap \text{co-NP}$. Existieren Instanzen, bei denen diese Sortierung nicht zu einem gewünschten Rechenweg führen, dann ist IRRED möglicherweise ein NP-unvollständiges Problem.

Folgerung 6.50 (IRRED und NPI) *Das Problem IRRED liegt genau dann in der Klasse NPI (NP – incomplete), wenn es irreduzible Instanzen mit getarnten stabilen Schrägkanten gibt, deren Schrägkanten sich immer so nummerieren lassen, dass ein sukzessiver polynomieller Nachweis der Stabilität jeder Schrägkante möglich ist, aber diese Nummerierung nicht immer mit polynomiellem Aufwand entdeckt werden kann.*

Die Eigenschaften von `Reducing2` können wir damit folgendermaßen zusammenfassen.

Bemerkung 6.51 (Reducing2) *Der Algorithmus `Reducing2` durchsucht solange den gesamten exponentiell großen Suchraum, bis er einen streng reduzierenden Plan $S^* \in \mathcal{S}_{SIJ}$ zu dem Ausgangsplan $S \in \mathcal{S}_{SIJ}$ findet. Ist S reduzierbar, dann findet der Algorithmus einen reduzierenden Plan S^* mit sehr hoher Wahrscheinlichkeit in polynomiell beschränkter maximaler Laufzeit. Für Instanzen von IRRED, die keine getarnten stabilen Schrägkanten enthalten, stimmt `Reducing2` mit `Reducing1` überein. Für Instanzen von IRRED, die getarnte stabile Schrägkanten enthalten, ist die Wahrscheinlichkeit hoch, dass sie dennoch auf mindestens einem erlaubten Rechenweg mit polynomiell beschränktem Aufwand gelöst werden können. Die Frage, ob Instanzen von IRRED existieren, für die `Reducing2` kein NP-Test ist, bleibt genauso offen, wie die Frage, ob überhaupt Instanzen existieren, für die `Reducing2` keine polynomiell beschränkte Laufzeit hat.*

6.4.3 Enumeration aller reduzierenden Pläne

Der Algorithmus `Reducing2` ist natürlich auch prinzipiell geeignet, alle streng reduzierenden Pläne zu einem Ausgangsplan zu enumerieren. Neben der Enumeration aller Rekombinationen der Planimplikationsklassen und der Enumeration aller Teilmengen der jeweiligen Schrägkantenmenge, stellt er nicht nur eine zielgerichtete Verzweigungsstrategie zur Verfügung. Für Instanzen ohne getarnte stabile Schrägkanten ist diese Verzweigungsstrategie sogar optimal, da *jede* Verzweigung, die nicht sofort als Sackgasse endet, zu einer zulässigen Entfernungsmenge führt. D.h. es werden keine unnötigen Verzweigungen betrachtet.

Eine solche Enumeration liefert insbesondere auch alle irreduziblen Pläne zu einem Ausgangsplan oder einer Operationenmenge. Das Verfahren bietet also auch eine Alternative zu den von BRÄSEL ET AL. [13] angewandten Methoden zur Bestimmung der Anzahl der irreduziblen Pläne für kleine Formate (Tabelle 4.2 auf Seite 67).

Satz 6.52 (Enumeration aller reduzierenden Pläne) *Sei $S \in \mathcal{S}_{SIJ}$ ein Plan. Mit Hilfe von `Reducing2` lassen sich sämtliche streng reduzierenden Pläne $S^* \prec S$ von S enumerieren.*

Beweis. Wird bei der Rückgabe einer zulässigen Entfernungsmenge in `Reducing2` auf die Terminierung des Verfahrens verzichtet, und wird mit jeder zulässigen Entfernungsmenge das Verfahren erneut gestartet, dann enumeriert `Reducing2` sämtliche zulässige Entfernungsmengen $M^* \subseteq E_{diag}(S)$. ■

Wichtig in diesem Zusammenhang ist jedoch, dass bei einem Aufruf des Verfahrens im Falle einer zulässigen Entfernungsmenge $M^* \subseteq E_{diag}(S)$ die Verzweigung mit den Schrägkanten aus $[S^{tr}] - M^*$ erfolgt. Wird zu M^* erst ein resultierender Plan bestimmt, d.h. eine transitive Orientierung auf $[S^{tr}] - M^*$ gesucht, dann werden dadurch unter Umständen Schrägkanten entfernt, die in reduzierbaren Plänen zum Ausgangsplan S enthalten sind. Solche Pläne können möglicherweise nicht auf andere Weise generiert werden.

Bevor wir jedoch das Kapitel zu dem vorgeschlagenen Reduktionsalgorithmus beenden, wollen wir noch kurz auf die Möglichkeiten der Konstruktion irreduzibler Pläne durch das Hinzufügen von Schrägkanten zum H-Graph G_{SIJ} eingehen. Auch dieses Verfahren ist prinzipiell geeignet, alle irreduziblen Pläne zu einem Ausgangsplan zu enumerieren. Direkt abgeleitet werden kann aus dieser Konstruktionsmöglichkeit ein Konzept für eine Nachbarschaft zwischen den irreduziblen Plänen eines Ausgangsplans.

6.5 Konstruktion irreduzibler Pläne durch Hinzufügen von Schrägkanten

Der im vorherigen Abschnitt beschriebene Reduktionsalgorithmus versucht in beiden Ausprägungen (**Reducing1** und **Reducing2**), einen streng reduzierenden Plan zu einem Ausgangsplan zu finden, indem er zielgerichtet Schrägkanten entfernt, um dadurch Konflikte im Reduktionsgraphen aufzulösen. Nun wollen wir kurz den umgekehrten Weg betrachten, und Schrägkanten zum H-Graphen G_{SIJ} hinzufügen, bis wir bei einem irreduziblen Plan angekommen sind. Dabei wollen wir jedoch nicht über alle Schrägkantenmengen mit beispielsweise höchstens r Elementen enumerieren müssen. Stattdessen wollen wir die Informationen, die uns der Faktorgraph und der Konsequenzgraph bieten, dazu nutzen, Schrägkanten zum Einfügen zielgerichtet auswählen zu können. Um einen Bezug für die Definition eines Faktorgraphen bzw. Konsequenzgraphen zu haben, benötigen wir für diesen Ansatz einen Ausgangsplan $S \in \mathcal{S}_{SIJ}$.

Der Reduktionsgraph $G_{\mathcal{R}_H}^*$

Betrachten wir den Reduktionsgraphen $G_{\mathcal{R}_M}$ zu der Entfernungsmenge $M = E_{diag}(S)$, dann ist dieser nicht konfliktfrei, sofern G_{SIJ} nicht selbst ein H-Comparabilitygraph ist. Ohne Einschränkung können wir annehmen, dass $G_{\mathcal{R}_M}$ nur aus einem einzigen Knoten besteht, der sämtliche Planimplikationsklassen von S sowie ihre Umkehrungen enthält. Das Bild ändert sich, wenn wir bei der Bestimmung des Reduktionsgraphen zu der Entfernungsmenge M auf die Kontraktion der Planimplikationsklassen, die in Γ -Relation zueinander stehen, verzichten. Dann enthält $G_{\mathcal{R}_M}$ die Knoten zu den Planimplikationsklassen $P_i \in \mathcal{P}_S$ und die Knoten $\bar{P}_i = P_i^{-1} \in \mathcal{P}_{S^{-1}}$. Als Kanten enthält $G_{\mathcal{R}_M}$ dann genau die jeweiligen Konsequenzen der Schrägkanten zwischen den Planimplikationsklassen in $[S^{tr}]$.

Definition 6.53 (Reduktionsgraph $G_{\mathcal{R}_H}^*$) Sei $[S^{tr}] = (SIJ, E_{reg}(S) + E_{diag}(S))$ der H-Comparabilitygraph zu einem Plan $S \in \mathcal{S}_{SIJ}$, $G_{\mathcal{R}_M} = (V_M^* + V_M^{**}, E_{\mathcal{R}_M})$ der Reduktionsgraph zu einer Entfernungsmenge $M \subseteq E_{diag}(S)$. Dann entsteht der Reduktionsgraph $G_{\mathcal{R}_H}^* = (\mathcal{P}_S + \mathcal{P}_{S^{-1}}, E_{\mathcal{R}_H}^*)$ zu der Schrägkantenmenge $H = E_{diag}(S) - M$ aus $G_{\mathcal{R}_M}$, indem sämtliche Knoten $e \in V_M^*$ zu Schrägkanten $e \in E_{diag}(S) - M$ entfernt werden, und auf die Kontraktion der Knoten $v \in V_M^*$ zu Planimplikationsklassen aus $[S^{tr}]$ verzichtet wird.

In Abbildung 6.6 (links) ist der Reduktionsgraph $G_{\mathcal{R}_H}^*$ für $H = \emptyset$ zu dem Plan $A \in \mathcal{S}_{SIJ}$ aus Beispiel 4.44 (Abbildungen 4.6 und 4.7) dargestellt. Aus Gründen der Übersichtlichkeit ist wie immer von je zwei symmetrischen Γ -Relationen zwischen zwei Planimplikationsklassen aus $[A^{tr}]$ jeweils nur eine abgebildet. Der Konsequenzgraph zu A ist in einer etwas verkürzten Form in Tabelle 6.4 dargestellt.

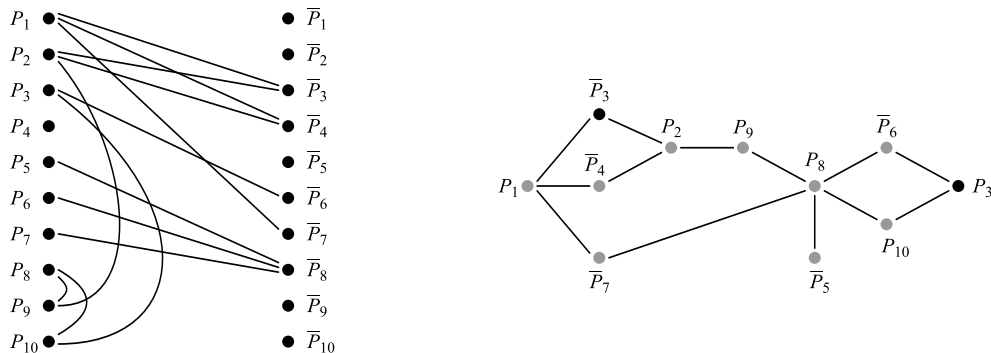


Abbildung 6.6: Der Reduktionsgraph $G_{\mathcal{R}_M} = G_{\mathcal{R}_H}^*$ bezüglich der Entfernungsmenge $M = E_{diag}(A)$ bzw. der Schrägkantenmenge $H = \emptyset$ zu dem Plan $A \in \mathcal{S}_{SIJ}$ aus Beispiel 4.44 (Abbildung 4.7) in einer systematischen Darstellung (links), sowie in einer anschaulicheren Variante (rechts).

Konflikte in $G_{\mathcal{R}_H}^*$

Auf der rechten Seite in Abbildung 6.6 ist eine etwas anschaulichere Darstellung von $G_{\mathcal{R}_H}^*$ abgebildet. Dort kann man erkennen, dass in $[A^{tr}] - E_{diag}(A)$ eine Γ -Kette zwischen P_3 und \bar{P}_3 existiert (schwarze Knoten). Da P_3 seinerseits mit \bar{P}_1 und \bar{P}_2 in Γ -Relation steht, existieren naturlich auch Γ -Ketten zwischen P_1 und \bar{P}_1 , sowie P_2 und \bar{P}_2 , usw.. Der Graph $[A^{tr}] - E_{diag}(A) = G_{SIJ}$ besteht nur aus einer einzigen inkonsistenten Planimplikationsklasse.

1.4	$P_2 + \bar{P}_3$
1.5			
1.7	$P_8 + P_9$		
1.9	$P_2 + \bar{P}_4$		
2.5	$P_1 + \bar{P}_3$		
2.7	$P_3 + \bar{P}_6$	$P_8 + P_{10}$	
4.3	$P_3 + P_{10}$	$P_6 + \bar{P}_8$	
5.3	$P_7 + \bar{P}_8$		
6.3	$P_5 + \bar{P}_8$		
8.3	$P_2 + P_9$		
8.5	$P_1 + \bar{P}_4$		
8.7			
9.3			
9.7	$P_1 + \bar{P}_7$		

Tabelle 6.4: Der Konsequenzgraph $G_{\mathcal{K}}$ zu dem Plan $A \in \mathcal{S}_{SIJ}$ aus Beispiel 4.44.

Analog zu der Vorgehensweise bei den Algorithmen **Reducing1** und **Reducing2** konnen wir nun auch hier versuchen, die Konflikte in $G_{\mathcal{R}_H}^*$ schrittweise aufzulosen. Im Unterschied zu diesen Algorithmen, geschieht das Auflosen in diesem Fall durch das Hinzufugen von Schragkanten. Fugen wir zu $[A^{tr}] - E_{diag}(A)$ beispielsweise die Schragkanten $H = \{\widehat{1.4}\}$ hinzu, dann entfallt in $[A^{tr}] - E_{diag}(A) + H$ die Γ -Relation zwischen P_2 und \bar{P}_3 , und einer der Konflikte in $G_{\mathcal{R}_H}^*$ wird dadurch aufgelost. Der Reduktionsgraph $G_{\mathcal{R}_H}^*$ zu $H = \{\widehat{1.4}\}$ ist in Abbildung 6.7 (links) dargestellt. Er enthalt noch vier Konflikte: zwei Konflikte der Ordnung 4 ($\bar{P}_3 + P_1 + \bar{P}_7 + P_8 + \bar{P}_6 + P_3$ und $\bar{P}_3 + P_1 + \bar{P}_7 + P_8 + P_{10} + P_3$) und zwei Konflikte der Ordnung 6 ($\bar{P}_3 + P_1 + \bar{P}_4 + P_2 + P_9 + P_8 + \bar{P}_6 + P_3$ und $\bar{P}_3 + P_1 + \bar{P}_4 + P_2 + P_9 + P_8 + P_{10} + P_3$).

Die *Ordnung* eines Konflikts in $G_{\mathcal{R}_H}^*$ konnen wir dabei analog zur Ordnung eines Konflikts in $G_{\mathcal{R}_M}$ als Anzahl der Knoten in $G_{\mathcal{R}_H}^*$ auf einem Weg von einem P_i zu \bar{P}_i definieren. Ein *direkter Konflikt* (Ordnung 0) entspricht, genau wie bei $G_{\mathcal{R}_M}$, einer Γ -Relation zwischen P_i und \bar{P}_i . Mittels der Methoden in **findNextEdge** konnen wir auch in $G_{\mathcal{R}_H}^*$ mit polynomiellem Aufwand einen konkreten Konflikt finden. Im Unterschied zu den **Reducing**-Algorithmen suchen wir in diesem Fall jedoch keinen Knoten in $G_{\mathcal{R}_M}$, den wir entfernen wollen, sondern wir suchen nach einer Kante in $G_{\mathcal{R}_H}^*$, die wir durch das Hinzufugen einer oder mehrerer Schragkanten zu $[A^{tr}] - E_{diag}(A) + H$, aus $G_{\mathcal{R}_H}^*$ entfernen wollen.

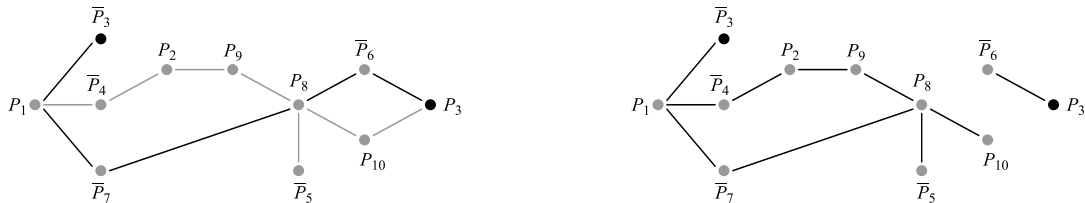


Abbildung 6.7: Die Reduktionsgraphen $G_{\mathcal{R}_{H_i}}^*$ zu $A \in \mathcal{S}_{SIJ}$ bezuglich der Schragkantenmengen $H_1 = \{\widehat{1.4}\}$ (links) und $H_2 = \{\widehat{1.4}, \widehat{4.3}\}$ (rechts). Aus H_2 folgt der irreduzible Plan A_{3457} .

Die Auflösung von Konflikten in $G_{\mathcal{R}_H}^*$

Der Reduktionsgraph $G_{\mathcal{R}_H}^*$ von A zu $H_1 = \{\widehat{1.4}\}$ (Abbildung 6.7, links) enthält also vier Konflikte. Nehmen wir an, wir bestimmen den Konflikt $\bar{P}_3 + P_1 + \bar{P}_7 + P_8 + \bar{P}_6 + P_3$ (schwarze Kanten). Nehmen wir nun an, wir entscheiden uns dafür, die direkte Γ -Verbindung zwischen P_8 und \bar{P}_6 zu zerstören. Aus dem Konsequenzgraphen $G_{\mathcal{K}}$ (Tabelle 6.4) können wir ablesen, dass wir dazu lediglich die Schrägkante $\widehat{4.3}$ zu $[A^{tr}] - E_{diag}(A) + H_1$ hinzufügen müssen.

Das Hinzufügen von $\widehat{4.3}$ zerstört außerdem die direkte Γ -Verbindung zwischen P_3 und P_{10} . Der resultierende Reduktionsgraph $G_{\mathcal{R}_{H_2}}^*$ zu $H_2 = \{\widehat{1.4}, \widehat{4.3}\}$ (Abbildung 6.7, rechts) ist somit konfliktfrei. Der Graph $[A^{tr}] - E_{diag}(A) + H_2 = G_{SIJ} + \{\widehat{1.4}, \widehat{4.3}\}$ besteht aus einer konsistenten Planimplikationsklasse $P_1 + P_2 + \bar{P}_3 + \bar{P}_4 + \bar{P}_5 + P_6 + \bar{P}_7 + P_8 + P_9 + P_{10}$, sowie ihrer Umkehrung. Der resultierende Plan aus dieser Planimplikationsklasse ist der irreduzible Plan A_{3457} (vgl. Abbildung 4.8 auf Seite 85). Er enthält nur die Schrägkante $\widehat{4.3}$. Hätten wir uns also im ersten Schritt gleich für die Zerstörung der Γ -Verbindung zwischen P_8 und \bar{P}_6 entschieden, wären wir sofort zu dem Plan A_{3457} gelangt.

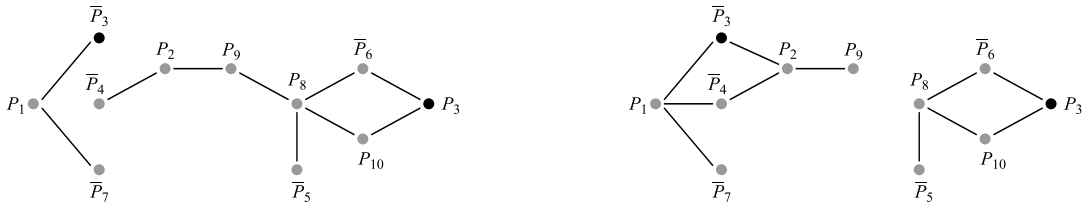


Abbildung 6.8: Die konfliktfreien Reduktionsgraphen $G_{\mathcal{R}_H}^*$ zu den Schrägkantenmengen $H_3 = \{\widehat{1.4}, \widehat{5.3}, \widehat{8.5}\}$ (Plan A_{1456}) (links) bzw. $H_4 = \{\widehat{1.7}, \widehat{5.3}\}$ (Plan A_{12569}).

Hätten wir uns im zweiten Schritt dagegen nicht für die Zerstörung der Γ -Verbindung zwischen P_8 und \bar{P}_6 , sondern stattdessen für die Zerstörung der Γ -Verbindung zwischen \bar{P}_7 und P_8 entschieden, und anschließend für die Zerstörung von $P_1 + \bar{P}_4$, dann wären wir zu der Kantenmenge $H_3 = \{\widehat{1.4}, \widehat{5.3}, \widehat{8.5}\}$ gelangt. Der Reduktionsgraph $G_{\mathcal{R}_{H_3}}^*$ bezüglich H_3 (Abbildung 6.8, links) ist konfliktfrei. Der resultierende irreduzible Plan ist der Plan A_{1456} . Er enthält genau die drei Schrägkanten aus H_3 . Die Unterbrechung von $\bar{P}_7 + P_8$ im ersten Schritt, verbunden mit der Unterbrechung von $P_8 + P_9$ im zweiten Schritt hätte uns zu der Entfernungsmenge $H_4 = \{\widehat{5.3}, \widehat{1.7}\}$ geführt. Der Reduktionsgraph bezüglich H_4 (Abbildung 6.8, rechts) ist konfliktfrei, und beschreibt den irreduziblen Plan A_{12569} . Dieser Plan enthält jedoch neben den transitiven Kanten aus H_4 noch die Schrägkante $\widehat{1.5}$.

Durch die Wahl der Kante $\widehat{1.4}$ im ersten Schritt (Abbildung 6.7, links) können wir neben den Plänen A_{3457} und A_{1456} auch 9 weitere irreduzible Pläne erreichen. Darunter sind alle 5 irreduziblen Pläne, die die Kante $\widehat{1.4}$ enthalten, aber auch die Pläne A_{12569} , $A_{348.10}$, A_{1256} und A_{12567} .

Minimale trennende Kantenmengen in $G_{\mathcal{R}_H}^*$

Durch die Betrachtung verschiedener Verzweigungen aufgrund der Auswahl des jeweils zu lösenden Konflikts, und aufgrund der Art und Weise der Auflösung eines bestimmten Konflikts, lässt sich für dieses Beispiel jeder einzelne der 15 irreduziblen Pläne von A (Abbildung 4.8) erzeugen. Die Bedeutung der einzelnen Schrägkanten ist dabei für einen Schritt nicht ganz so groß, wie bei den beiden Reducing-Algorithmen.

Wir können die verschiedenen irreduziblen Pläne auch finden, indem wir lediglich nach trennenden Kantenmengen zwischen P_i und \bar{P}_i suchen. Ist eine solche trennende Kantenmenge inklusionsminimal, dann ist der resultierende Plan irreduzibel. Die Schrägkanten dieses Plans ergeben sich dann aus den Konsequenzen der Schrägkanten des Ausgangsplans. Es müssen mindestens jene Schrägkanten hinzugefügt werden, die mindestens eine der Kanten aus der trennenden Kantenmenge als Konsequenz nach sich ziehen.

Weitere Kanten können sich dann aus der Bestimmung der transitiven Hülle ergeben. Da die ihrerseits jedoch auch Konsequenzen haben, kann sich die trennende Kantenmenge durch die Hinzunahme weiterer Schrägkanten vergrößern. Außerdem können einzelne Schrägkanten auch mehrere Konsequenzen zwischen verschiedenen Planimplikationsklassen haben. Die Hinzunahme einer solchen Schrägkante \hat{e}^* zu $H^* = H + \hat{e}^*$ führt ebenfalls zu der Entfernung weiterer Kanten aus $G_{\mathcal{R}_H}^*$.

Satz 6.54 (Konstruktion irreduzierbarer Pläne) Sei $\mathcal{P}_S = \{P_1, \dots, P_k\}$ die Menge der Planimplikationsklassen zu einem Plan $S \in \mathcal{S}_{SIJ}$, und $G_{\mathcal{R}_H}^* = (\mathcal{P}_S + \mathcal{P}_{S^{-1}}, E_{\mathcal{R}_H}^*)$ der Reduktionsgraph zu der Schrägkantenmenge $H = \emptyset$. Ferner sei $G_{\mathcal{K}_e} = (V_{\mathcal{K}_e}, E_{\mathcal{K}_e})$ der Teilgraph aller Kanten der Farbe $e \in E_{diag}(S)$ des Konsequenzgraphen von S . Zu einer Kante $c \in E_{\mathcal{R}_H}^*$ beschreiben wir mit $H(c) \subseteq E_{diag}(S)$ die Menge der Schrägkanten $e \in E_{diag}(S)$, die als Konsequenz $c \in E_{\mathcal{K}_e}$ nach sich ziehen. Dann beschreibt eine Kantenmenge $C \subseteq E_{\mathcal{R}_H}^*$, die jeden Konflikt in $G_{\mathcal{R}_H}^*$ zerstört, d.h. jeden Knoten $P_i \in \mathcal{P}_S$ von $P_i^{-1} \in \mathcal{P}_{S^{-1}}$ trennt, einen Plan $S^* = Q_1 + \dots + Q_k \in \mathcal{S}_{SIJ}$, mit $Q_i \in \{P_i, P_i^{-1}\}$. Besteht der Reduktionsgraph $G_{\mathcal{R}_{H^*}}^*$ zu der Menge $H^* = \bigcup_{c \in C} H(c)$ nur aus genau zwei Zusammenhangskomponenten S^* und $(S^*)^{-1}$, dann ist S^* irreduzibel, und die Menge der Schrägkanten von $[(S^*)^{tr}]$ enthält die Kanten aus H^* , $H^* \subseteq E_{diag}(S^*)$.

Beweis. Sei $C \subseteq E_{\mathcal{R}_H}^*$ eine solche trennende Kantenmenge in $G_{\mathcal{R}_H}^*$. Dann ist der Reduktionsgraph $G_{\mathcal{R}_H}^* - C$ konfliktfrei. Jede Zusammenhangskomponente von $G_{\mathcal{R}_H}^* - C$ beschreibt eine Planimplikationsklasse. Zu dieser Menge von Planimplikationsklassen existiert eine kreisfreie Kombination. Andernfalls wäre $G_{\mathcal{R}_H}^* - C$ nicht konfliktfrei (mit Folgerung 1.32). Nach Satz 4.42 beschreibt C damit mindestens einen Plan $S^* \in \mathcal{S}_{SIJ}$. Die Schrägkantenmenge $H^* = \bigcup_{c \in C} H(c)$ ist die Menge der Schrägkanten, die zu G_{SIJ} hinzugefügt werden müssen, um die direkten Γ -Verbindungen $c = Q'Q'' \in C$ zwischen Q' und Q'' zu verhindern. Der H-Comparabilitygraph $[(S^*)^{tr}] = [(Q_1 + \dots + Q_k)^{tr}]$ enthält damit alle Kanten aus H^* . Besteht $G_{\mathcal{R}_{H^*}}^*$ nur aus einer einzigen Planimplikationsklasse $Q_1 + \dots + Q_k$ und ihrer Umkehrung $Q_1^{-1} + \dots + Q_k^{-1}$, dann ist S^* irreduzibel (Satz 4.30). ■

Allen oben genannten Beispiele für konfliktfreie Reduktionsgraphen $G_{\mathcal{R}_H}^*$ enthalten nur eine Planimplikationsklasse und ihre Umkehrung. Durch die Halbierung der Darstellung aufgrund der Symmetrie der Γ -Relation sind jeweils Teile der beiden Zusammenhangskomponenten von $G_{\mathcal{R}_H}^*$ abgebildet. Der dadurch jeweils beschriebene Plan ist bis auf Umkehrung eindeutig.

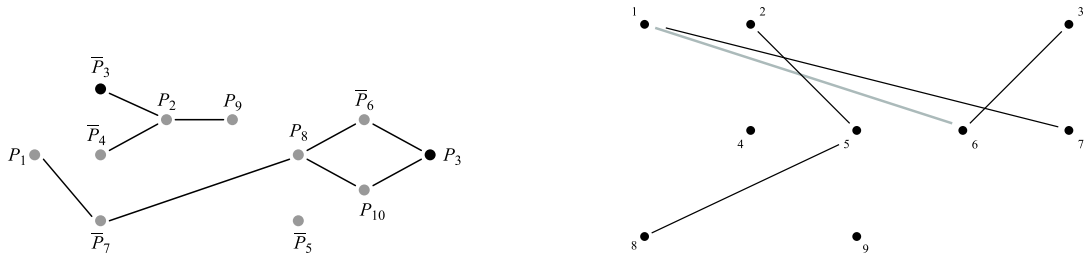


Abbildung 6.9: Der Reduktionsgraph $G_{\mathcal{R}_H}^*$ von $A \in \mathcal{S}_{SIJ}$ bezüglich der (nicht inklusionsminimalen) Schrägkantenmenge $H = \{\widehat{1.7}, \widehat{6.3}, \widehat{8.5}\}$. H führt zu den beiden resultierenden Pläne A_{25679} (irreduzibel) und A_{2679} . Der Plan A_{2679} (rechts) ist kein reduzierender Plan von A .

Irreduzible Pläne und reduzierende Pläne

Um den Unterschied zwischen einer beliebigen trennenden Kantenmenge $C \subseteq E_{\mathcal{R}_H}^*$ und einer inklusionsminimalen trennenden Kantenmenge $C' \subseteq E_{\mathcal{R}_H}^*$ zu verdeutlichen, betrachten wir den Fall, dass wir im Reduktionsgraphen $G_{\mathcal{R}_H}^*$ zu dem Plan $A \in \mathcal{S}_{SIJ}$ die Γ -Verbindungen $P_1 + \bar{P}_4$ ($\widehat{8.5}$), $P_8 + P_9$ ($\widehat{1.7}$) und $\bar{P}_5 + P_8$ ($\widehat{6.3}$) unterbrechen (Abbildung 6.9). Der verbleibende Reduktionsgraph enthält zwei verschiedene Planimplikationsklassen, zum einen die Klasse $P_1 + \bar{P}_2 + P_3 + P_4 + \bar{P}_6 + \bar{P}_7 + P_8 + \bar{P}_9 + P_{10}$, und zum anderen die Klasse P_5 . Daraus ergeben sich zwei verschiedene Pläne, die Pläne A_{25679} und A_{2679} .

Während der Plan $A_{25679} \prec A$ ein irreduzibler reduzierender Plan von A ist (Abbildung 4.8), ist der Plan A_{2679} weder irreduzibel, noch ein reduzierender Plan von A . Er enthält die neue Schrägkante $\widehat{1.6} \notin E_{diag}(A)$. Allerdings ist die trennende Kantenmenge $C = \{P_1 + \bar{P}_4, P_8 + P_9, \bar{P}_5 + P_8\} \subseteq E_{\mathcal{R}_H}^*$ nicht inklusionsminimal gewählt. Begnügen wir uns mit der Unterbrechung der Γ -Verbindungen zwischen $P_1 + \bar{P}_4$ und $P_8 + P_9$, und verzichten wir auf die Unterbrechung von $\bar{P}_5 + P_8$, dann ist der resultierende Plan (der Plan A_{25679}) sowohl eindeutig, als auch irreduzibel.

Bemerkung 6.55 (S* und S) *Der irreduzible Plan $S^* \in \mathcal{S}_{SIJ}$ ist eine kreisfreie Rekombination der Planimplikationsklassen von $S \in \mathcal{S}_{SIJ}$. Nach Satz 4.43 ist er jedoch nicht notwendigerweise auch ein reduzierender Plan von $S \in \mathcal{S}_{SIJ}$, da er zu neuen Schrägkanten führen kann.*

Ein NP-Algorithmus zur Konstruktion eines irreduziblen Plans

Also konzentrieren wir uns darauf, eine minimale trennende Kantenmenge in $G_{\mathcal{R}_H}^*$ zu finden, um eine irreduzible Rekombination der Planimplikationsklasse zu einem Ausgangsplan $S \in \mathcal{S}_{SIJ}$ zu finden.

Lemma 6.56 (NP-Algorithmus) *Sei $\mathcal{P}_S = \{P_1, \dots, P_k\}$ die Menge der Planimplikationsklassen zu einem Plan $S \in \mathcal{S}_{SIJ}$ über einer Operationenmenge mit n Aufträgen und m Maschinen. Aus Satz 6.54 lässt sich ein nichtdeterministischer Algorithmus mit einer durch $\mathcal{O}(n^6 m^6)$ beschränkten maximalen Laufzeit ableiten, der eine irreduzible Rekombination der Planimplikationsklassen von S bestimmt.*

Beweis. Der angedeutete Algorithmus hat die folgende Form: Im Reduktionsgraphen $G_{\mathcal{R}_H}^*$ werden, angefangen mit $H = \emptyset$, sukzessive Konflikte identifiziert, und durch Entfernung jeweils einer Kante (aus $G_{\mathcal{R}_H}^*$) aufgelöst. Der Algorithmus terminiert, wenn der Reduktionsgraph konfliktfrei ist. Die Anzahl der Iterationen dieses Verfahrens ist durch $|E_{diag}(S)| = \mathcal{O}(n^2 m^2)$ beschränkt, da in jedem Schritt mindestens eine Schrägkante zu H hinzugefügt wird. Innerhalb einer Iteration müssen Konflikte bestimmt werden, d.h. es müssen kürzeste Wege von einem Knoten zu einem anderen bestimmt werden. Mit dem Algorithmus von DIJKSTRA ist das mit dem Aufwand $\mathcal{O}(|V(G_{\mathcal{R}_H}^*)|^2) = \mathcal{O}(|\mathcal{P}_S|^2) = \mathcal{O}(n^4 m^4)$ möglich (vgl. Lemma 6.30). Die Aktualisierung des Reduktionsgraphen ist in diesem Fall nicht so aufwendig, da die Schrägkanten in $G_{\mathcal{R}_H}^*$ nicht abgebildet sind. In `updateRedGraph` sind daher keine Iterationen nötig. Der Aufwand der Aktualisierung ist hier sogar beschränkt durch $\mathcal{O}(|E_{\mathcal{R}_H}^*|) = \mathcal{O}(2|V_{\mathcal{K}}|) = \mathcal{O}(n^2 m^2)$. Damit ist der Gesamtaufwand beschränkt durch $\mathcal{O}(n^6 m^6)$. Zu jedem reduzierbaren Ausgangsplan gibt es jeweils mindestens einen Rechenweg, der zu jedem der irreduziblen Pläne führt. Umgekehrt ist jedoch nicht klar, ob man auf jedem erlaubten Rechenweg zu einem irreduziblen Plan gelangt. Daher ist das Verfahren nur nichtdeterministisch. ■

Bemerkung 6.57 (IRRED und P) *Es ist nicht klar, ob reduzierbare Probleminstanzen von IRRED existieren, bei denen der in Lemma 6.56 skizzierte Algorithmus nicht auf jedem erlaubten Rechenweg eine irreduzible Rekombination $S^* \in \mathcal{S}_{SIJ}$ der Planimplikationsklassen von $S \in \mathcal{S}_{SIJ}$ findet. Da eine solche Rekombination S^* jedoch nicht zwangweise ein reduzierender Plan von S ist, hat Lemma 6.56 keine unmittelbare Auswirkung auf die Komplexität von IRRED. Selbst wenn keine reduzierbaren Probleminstanzen von IRRED existieren, bei denen dieses Verfahren versagen kann, d.h. selbst wenn zu jedem reduzierbaren Plan immer mit polynomiellem Aufwand eine irreduzible Rekombination der Planimplikationsklasse gefunden werden kann, folgt daraus nicht die Zugehörigkeit von IRRED zu P.*

Bemerkung 6.58 (getarnte stabile Schrägkanten) *Für Probleminstanzen von IRRED, die keine getarnten stabilen Schrägkanten enthalten, liefert der in Lemma 6.56 skizzierte Algorithmus wahrscheinlich auf jedem erlaubten Rechenweg eine irreduzible Rekombination der Planimplikationsklassen des Ausgangsplans. Ist eine solche Rekombination kein reduzierender Plan des Ausgangsplans, dann liefert das Verfahren keinen Hinweis auf die Reduzierbarkeit des Ausgangsplans.*

Enumeration der irreduziblen Pläne

Auch dieses Verfahren ist geeignet zur Enumeration aller irreduziblen Pläne eines Ausgangsplans. Dazu müssen lediglich sämtliche Verzweigungen verfolgt werden. Verzweigungen ergeben sich bei der

Auswahl des zu lösenden Konflikts, bei der Art und Weise, wie der gewählte Konflikt aufgelöst werden soll, sowie unter Umständen bei der Auswahl eines Plans, falls der konfliktfreie Reduktionsgraph aus mehreren verschiedenen Planimplikationsklassen besteht. Sind wir nur an den irreduziblen Plänen, die den Ausgangsplan reduzieren, dann müssen alle gefunden Pläne anschließend noch darauf getestet werden.

Während das Verfahren, das für einen beliebigen Ausgangsplan $S \in \mathcal{S}_{SIJ}$ beschrieben ist, grundsätzlich zur Enumeration aller irreduziblen Pläne einer Operationenmenge geeignet ist, bleibt offen, ob es dazu besser geeignet ist, als alternative Verfahren. Im Zusammenhang mit dem Algorithmus `Reducing2` haben wir eine Möglichkeit zur Enumeration aller irreduziblen Pläne einer Operationenmenge genannt, bei der die Verzweigung wenig, oder sogar gar keine Redundanzen enthält. Allerdings bestimmen wir mit diesem Verfahren auch sämtliche reduzierenden nicht irreduziblen Pläne des Ausgangsplans. Im Gegensatz dazu liefert das Verfahren aus Lemma 6.56 wahrscheinlich schneller irreduzible Pläne. Allerdings ist die Verzweigung nicht redundanzfrei, wie die Diskussion des Beispielplans A zeigt.

6.6 Eine Nachbarschaft zwischen irreduziblen Plänen

Aus dem Verfahren zur Konstruktion von irreduziblen Plänen aus dem vorangegangenen Abschnitt können wir ganz leicht ein Konzept für eine Nachbarschaft zwischen irreduziblen Plänen ableiten. In Abschnitt 3.1.3 haben wir im Zusammenhang mit Lösungsstrategien für das Open-Shop Problem auf die Bedeutung von Nachbarschaften für iterative Verbesserungsverfahren hingewiesen. Alle gängigen Nachbarschaften enthalten jeweils sämtliche zulässigen Pläne. Um von einem irreduziblen Plan, zu einem anderen irreduziblen Plan mit einem besseren Zielfunktionswert zu gelangen, sind üblicherweise mehrere oder sogar sehr viele Schritte notwendig. Da die Menge der irreduziblen Pläne eine universell-optimale Menge ist (Abschnitt 4.1.1), ist eine direkte Nachbarschaft unter irreduziblen Plänen wünschenswert.

Anwendung der Konstruktion eines irreduziblen Plans

Betrachten wir den Reduktionsgraphen $G_{\mathcal{R}_H}^*$ bezüglich der Kantenmenge $H \subseteq E_{diag}(S)$ eines irreduziblen Plans $S' \in \mathcal{S}_{SIJ}$ zu einem Ausgangsplan $S \in \mathcal{S}_{SIJ}$. $G_{\mathcal{R}_H}^*$ ist konfliktfrei. Entfernen wir nun aus $[(S')^{tr}] = G_{SIJ} + H$ eine Schrägkante $\hat{e} \in H$, dann ist der resultierende Reduktionsgraph nicht mehr konfliktfrei. Andernfalls ist S' nicht irreduzibel gewesen. Mit diesem inkonsistenten Reduktionsgraphen $G_{\mathcal{R}_H}^*$ können wir nun genauso verfahren, wie bei der Konstruktion eines irreduziblen Plans im Abschnitt 6.5. Wir identifizieren in jedem Schritt einen Konflikt, und lösen ihn durch die Hinzunahme von einer oder mehrerer Schrägkanten zu H auf. Dadurch gelangen wir erneut zu einem irreduziblen Plan $S'' \in \mathcal{S}_{SIJ}$. Um zu verhindern, dass wir wieder beim Ausgangsplan $S'' = S'$ landen, müssen wir lediglich vermeiden, dass wir den ausgewählten Konflikt in $G_{\mathcal{R}_H}^*$ durch die Hinzunahme der vorher entfernten Kante \hat{e} lösen.

Die alternative Auflösung eines Konflikts kann dazu führen, dass mehr als eine Schrägkante zu H hinzugefügt wird. Andererseits können sich in $[(S'')^{tr}]$ auch einige der Schrägkanten aus H als nicht transitiv erweisen, insbesondere dann, wenn der irreduzible Plan S' bereits sehr viele Schrägkanten enthält. Daher ist auch ein Übergang von einem irreduziblen Plan S' mit vielen Schrägkanten zu einem irreduziblen Plan S'' mit weniger Schrägkanten denkbar.

Entfernen wir aus dem irreduziblen Plan S' nicht nur genau eine Schrägkante, sondern eine beliebige Menge von Schrägkanten M aus $H = E_{diag}(S')$, dann erweitern wir dadurch das Spektrum der von S' aus erreichbaren irreduziblen Pläne S'' . Einen Einfluss auf den notwendigen Aufwand der Bestimmung von S'' hat diese Maßnahme jedoch nicht, da die Anzahl der Schritte, d.h. die Anzahl der notwendigen Erweiterungen von $H - M$, in jedem Fall von der Wahl der Kanten in $G_{\mathcal{R}_{H-M}}^*$ abhängt, die aus $G_{\mathcal{R}_{H-M}}^*$ entfernt werden sollen, um einen Weg von einem P_i zu $\bar{P}_i = P_i^{-1}$ zu unterbrechen.

Bemerkung 6.59 (Nachbarschaft) *Aus dem durch Satz 6.54 und Lemma 6.56 skizzierten Verfahren zur Konstruktion eines irreduziblen Plans, lässt sich eine Nachbarschaft zwischen irreduziblen Plänen ableiten. Man entfernt aus einem irreduziblen Plan $S' \in \mathcal{S}_{SIJ}$ eine beliebige Menge der*

Schrägkanten, und konstruiert aus dem resultierenden Reduktionsgraphen $G_{\mathcal{R}_H}^*$ einen alternativen Plan $S'' \in \mathcal{S}_{SIJ}$. Abhängig von der tatsächlichen Komplexität dieser Konstruktion (NP oder P), und unter Umständen auch abhängig von dem zugrundeliegenden Ausgangsplan $S \in \mathcal{S}_{SIJ}$, ist der Plan S'' entweder immer irreduzibel, oder mit hoher Wahrscheinlichkeit irreduzibel. Der Aufwand der Bestimmung von S'' aus S' ist durch $\mathcal{O}(n^6 m^6)$ beschränkt (Lemma 6.56).

Anzahl der Nachbarn

Bei der Konstruktion des irreduziblen Plans A_{1456} im vorangegangenen Abschnitt haben wir den Konflikt im Reduktionsgraphen zu der Schrägkantenmenge $H_1 = \{\widehat{1.4}\}$ (Abbildung 6.7, links) durch die Unterbrechung der Verbindungen $P_7 + \bar{P}_8$ und $P_1 + \bar{P}_4$ gelöst. Das führte uns zu der Schrägkantenmenge $H_3 = \{\widehat{1.4}, \widehat{5.3}, \widehat{8.5}\}$. Wollen wir nun, ausgehend von dem Plan A_{1456} , einen alternativen irreduziblen Plan bestimmen, dann müssen wir mindestens eine der drei vorhandenen Schrägkanten aus $[A_{1456}^{tr}]$ entfernen. Entfernen wir beispielsweise die Kante $\widehat{8.5}$, dann hat das die Verschmelzung von P_1 mit P_4 zur Folge.

Der resultierende Reduktionsgraph $G_{\mathcal{R}_H}^*$ entsteht aus dem Reduktionsgraphen bezüglich der Schrägkantenmenge H_3 (Abbildung 6.8, links) durch die Einfügung der Kante zwischen P_1 und \bar{P}_4 . Dieser Reduktionsgraph enthält nun zwei Konflikte der Ordnung 6. Je nach Wahl der Unterbrechungskante in einem dieser Konflikte gelangen wir zu einem der irreduziblen Pläne A_{4567} ($\bar{P}_3 + P_1$), A_{156} ($\bar{P}_4 + P_2$), A_{12569} ($P_2 + P_9$), A_{12568} ($P_9 + P_8$), A_{3457} ($P_8 + \bar{P}_6$ und $P_3 + P_{10}$) oder A_{1289} ($\bar{P}_6 + P_3$ und $P_8 + P_{10}$). D.h. der Plan A_{1456} hat allein bei Entfernung einer einzigen Schrägkante 6 von 14 irreduziblen Plänen als direkte Nachbarn. Lassen wir auch die Entfernung der gesamten Schrägkantenmenge zu, dann ist jeder irreduzible Plan ein Nachbar jedes anderen.

Bemerkung 6.60 (Anzahl der Nachbarn) Die Anzahl der Nachbarn eines irreduziblen Plans $S' \in \mathcal{S}_{SIJ}$ hängt von der Anzahl der Schrägkanten ab, die simultan aus $[(S')^{tr}]$ entfernt werden dürfen. Ist die Entfernung der gesamten Schrägkantenmenge erlaubt, dann sind alle irreduziblen Rekombinationen $S'' \in \mathcal{S}_{SIJ}$ der Planimplikationsklassen des zugrundeliegenden Ausgangsplans $S \in \mathcal{S}_{SIJ}$ direkte Nachbarn von S' .

Die Bedeutung des Ausgangsplans

Zu der Operationenmenge SIJ des Plans $A \in \mathcal{S}_{SIJ}$ existieren neben den 15 uns bekannten irreduziblen Plänen (Abbildung 4.8) noch 11 weitere, die sich nicht als Rekombinationen der Planimplikationsklassen von A darstellen lassen. Für eine beliebig gegebene Problem Instanz des Open-Shop Problems entsteht damit die Frage, ob unter den 15 bekannten irreduziblen Plänen immer ein Plan mit akzeptablem Zielfunktionswert ist. Mit anderen Worten: Wie groß ist der Anteil der Bearbeitungszeitmatrizen P , für die unter den irreduziblen Rekombinationen der Planimplikationsklassen eines Ausgangsplans $S \in \mathcal{S}_{SIJ}$ ein Plan $S' \in \mathcal{S}_{SIJ}$ mit $\frac{C_{\max}(S')}{C_{\max}(S^*)} < \varepsilon$ existiert, an allen Bearbeitungszeitmatrizen, wobei $S^* \in \mathcal{S}_{SIJ}$ einen optimalen Plan für diese Instanz bezeichnet.

Bemerkung 6.61 (Ausgangsplan S) Die Menge der irreduziblen Pläne in der Nachbarschaft ist beschränkt auf die Menge alle irreduziblen Rekombinationen der Planimplikationsklassen des zugrundeliegenden Ausgangsplans $S \in \mathcal{S}_{SIJ}$. Die Anzahl der Planimplikationsklassen von S , $|\mathcal{P}_S| = k$ mit $1 \leq k \leq m \binom{n}{2} + n \binom{m}{2}$, liefert einen Anhaltspunkt für den Anteil der erreichbaren irreduziblen Pläne an allen irreduziblen Plänen. Ob durch die Einschränkung auf eine echte Teilmenge der irreduziblen Pläne die Diversität innerhalb dieser Teilmenge abnimmt, ist nicht klar. Es muss jedoch bezweifelt werden, dass Problem Instanzen für das Open-Shop Problem (Bearbeitungszeiten) existieren, für die gute Näherungslösungen in der Menge der irreduziblen Rekombinationen der Planimplikationsklasse eines Ausgangsplans durch die Wahl dieses Ausgangsplans systematisch ausgeschlossen werden können (sofern $k \gg 1$ gilt).

Bemerkung 6.62 (alle irreduziblen Pläne) Findet als Ausgangsplan ein Plan $S \in \mathcal{S}_{SIJ}$ mit maximalem Rang Anwendung, dann enthält die Nachbarschaft der irreduziblen Rekombination der Planimplikationsklassen von S sämtliche irreduziblen Pläne zu dieser Operationenmenge.

Damit haben wir also einen sehr vielversprechenden Ansatz für die Beschreibung einer Nachbarschaft zwischen irreduziblen Plänen. Bevor wir dieses Kapitel jedoch beenden, wollen wir noch einen kurzen Blick auf die Nachbarschaft der irreduziblen Pläne zu der vollständigen Operationenmenge mit 3 Aufträgen und 3 Maschinen werfen.

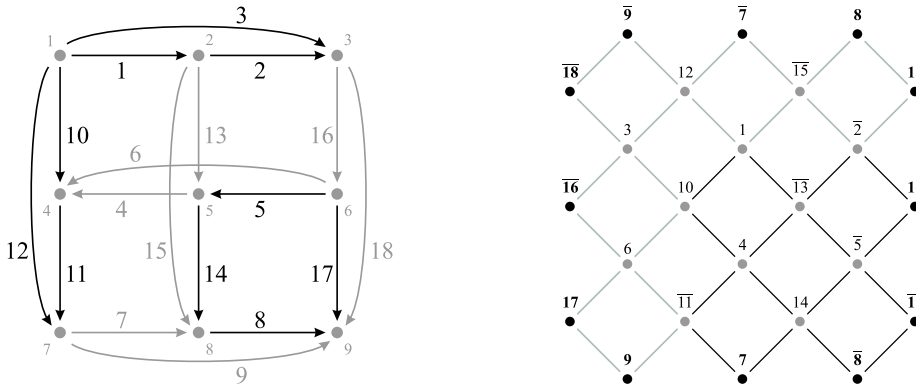


Abbildung 6.10: Die Planimplikationsklassen zu dem Plan $S = B_{35}$ (links), und der zugehörige Reduktionsgraph $G_{\mathcal{R}_H}^*$ bezüglich $H = \emptyset$ (rechts).

Beispiel 6.63 ($K_3 \times K_3$) Im Kapitel 5 haben wir im Zusammenhang mit der Bestimmung von Mengen entferntbarer Schrägkanten aus einem H -Comparabilitygraph (Abschnitt 5.5.1) den Plan B_{34} betrachtet. Er ist ein irreduzibler Plan über der vollständigen Operationenmenge $K_3 \times K_3$.

$$B_{34} = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 2 & 3 & 1 \end{bmatrix} \quad B_{35} = \begin{bmatrix} 1 & 2 & 3 \\ 6 & 5 & 4 \\ 7 & 8 & 9 \end{bmatrix} \quad B_{36} = \begin{bmatrix} 1 & 3 & 2 \\ 2 & 4 & 1 \\ 3 & 1 & 4 \end{bmatrix}$$

Der H -Comparabilitygraph $[B_{34}^{tr}]$ enthält 3 Schrägkanten. Wir wollen nun die Bestimmung eines anderen irreduziblen Plans aus B_{34} betrachten. D.h. wir wollen einen Nachbarn von B_{34} bestimmen. Als Ausgangsplan wählen wir den Plan $S = B_{35}$. Er beschreibt eine lineare Ordnung aller Operationen, und $[B_{35}^{tr}]$ enthält demzufolge sämtliche denkbaren Schrägkanten. Der H -Comparabilitygraph $[S^{tr}]$ ist isomorph zum K_9 , und sowohl jede Planimplikationsklasse, als auch jede Implikationsklasse besteht nur aus einer einzigen Kante. In Abbildung 6.10 (links) ist eine Bezeichnung der Planimplikationsklassen von S angegeben. (Es gilt $B_{34} = S_{589.11.13.16.17.18}$.) Im rechten Teil der Abbildung 6.10 ist der Reduktionsgraph $G_{\mathcal{R}_H}^*$ bezüglich der Schrägkantenmenge $H = \emptyset$ dargestellt. In Tabelle 6.5 ist der Konsequenzgraph $G_{\mathcal{K}}$ von S angegeben.

Der Reduktionsgraph bezüglich der Schrägkantenmenge $H = \{\widehat{1.8}, \widehat{3.5}, \widehat{4.9}\}$ (Schrägkantenmenge von B_{34}) ist konfliktfrei. Nehmen wir an, wir entscheiden uns dafür, aus $[B_{34}^{tr}]$ die Kante $\widehat{1.8}$ zu entfernen. Der resultierende Reduktionsgraph $G_{\mathcal{R}_H}^*$ zu der verbleibenden Kantenmenge $H = \{\widehat{3.5}, \widehat{4.9}\}$ ist in Abbildung 6.11 (links) dargestellt. In $G_{\mathcal{R}_H}^*$ suchen wir nun nach einer Kantenmenge, die alle Wege zwischen \bar{P}_7 und P_7 , P_8 und \bar{P}_8 , \bar{P}_9 und P_9 (alle Wege "von oben nach unten"), sowie zwischen \bar{P}_{16} und P_{16} , P_{17} und \bar{P}_{17} , und \bar{P}_{18} und P_{18} (alle Wege "von links nach rechts") unterbricht. Nehmen wir an, wir entscheiden uns dafür, die Verbindung zwischen P_9 und P_{17} zu zerstören. Dann erweitern wir dadurch die Schrägkantenmenge um die Kante $\widehat{6.7}$. Diese Erweiterung hat zur Folge, dass auch die Verbindung zwischen P_6 und \bar{P}_{11} unterbrochen wird. Entscheiden wir uns anschließend beispielsweise nacheinander für die Unterbrechung der Verbindungen $P_8 + \bar{P}_{15}$ ($\widehat{2.9}$, $\bar{P}_2 + P_{18}$), $\bar{P}_5 + \bar{P}_{13}$ ($\widehat{2.6}$, $\bar{P}_2 + P_{16}$) und $P_4 + P_{10}$ ($\widehat{1.5}$, $P_1 + \bar{P}_{13}$), dann gelangen wir zu dem auf der rechten Seite von Abbildung 6.11 dargestellten Reduktionsgraphen. Die Darstellung dieses Reduktionsgraphen zerfällt in sechs Komponenten, von denen keine einen Konflikt enthält. $G_{\mathcal{R}_H}^*$ ist damit jedoch nicht konfliktfrei. Man gelangt zum Beispiel von P_8 über P_{18} und P_7 zu \bar{P}_8 ,

$$P_8 + P_{18} + \bar{P}_3 + \bar{P}_{12} + P_7 + P_{14} + \bar{P}_8.$$

1.5	$P_1 + \bar{P}_{13}$	$P_4 + P_{10}$
1.6	$P_3 + \bar{P}_{16}$	$P_6 + P_{10}$		
1.8	$P_1 + \bar{P}_{15}$	$P_7 + \bar{P}_{12}$		
1.9	$P_3 + \bar{P}_{18}$	$P_9 + \bar{P}_{12}$		
2.4	$P_4 + \bar{P}_{13}$	$P_1 + P_{10}$		
2.6	$P_2 + \bar{P}_{16}$	$P_5 + P_{13}$		
2.7	$P_1 + P_{12}$	$P_7 + P_{15}$		
2.9	$P_8 + \bar{P}_{15}$	$P_2 + \bar{P}_{18}$		
3.4	$P_3 + P_{10}$	$P_6 + \bar{P}_{16}$		
3.5	$P_2 + P_{13}$	$P_5 + \bar{P}_{16}$		
3.7	$P_3 + P_{12}$	$P_9 + P_{18}$		
3.8	$P_2 + P_{15}$	$P_8 + P_{18}$		
4.8	$P_4 + P_{14}$	$P_7 + \bar{P}_{11}$		
4.9	$P_6 + P_{17}$	$P_9 + \bar{P}_{11}$		
5.7	$P_7 + P_{14}$	$P_4 + \bar{P}_{11}$		
5.9	$P_5 + P_{17}$	$P_8 + \bar{P}_{14}$		
6.7	$P_9 + P_{17}$	$P_6 + \bar{P}_{11}$		
6.9	$P_8 + P_{17}$	$P_5 + \bar{P}_{14}$		

Tabelle 6.5: Der Konsequenzgraph $G_{\mathcal{K}}$ zu dem Plan $S = B_{35} \in \mathcal{S}_{SIJ}$.

Wählen wir nun aus diesem Konflikt beispielsweise die Verbindung $P_{18} + \bar{P}_3$ ($\widehat{1.9}$, $\bar{P}_9 + P_{12}$), dann gilt in dem resultierendem Reduktionsgraphen

$$\begin{array}{l} P_1 + \bar{P}_2 + P_3 + P_6 + \bar{P}_7 + P_{10} + P_{12} + \bar{P}_{15} + \bar{P}_{16}, \\ \hline P_4 + \bar{P}_5 + P_7 + \bar{P}_8 + \bar{P}_{11} + \bar{P}_{13} + P_{14} + \bar{P}_{17}, \\ P_8 + P_{18}, \quad \bar{P}_{18} + \bar{P}_9. \end{array}$$

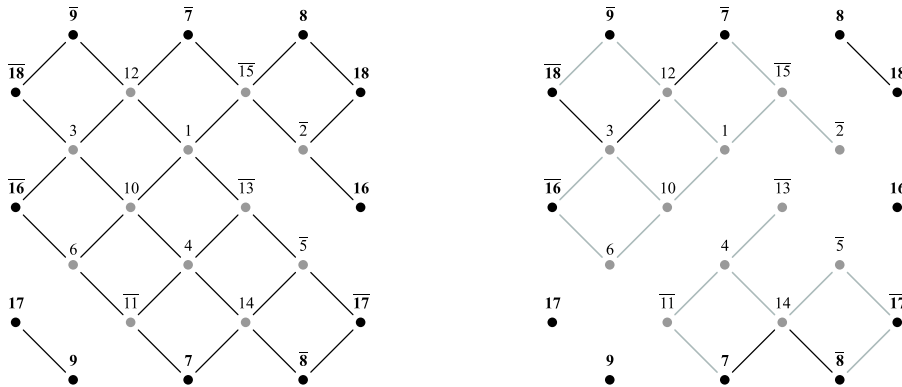


Abbildung 6.11: Der Reduktionsgraph $G_{\mathcal{R}_H}^*$ von $S = B_{35}$ bezüglich $H = \{\widehat{3.5}, \widehat{4.9}\}$ (Plan $B_{34} - \widehat{1.8}$) (links) bzw. $H = \{\widehat{3.5}, \widehat{4.9}, \widehat{6.7}, \widehat{2.9}, \widehat{2.6}, \widehat{1.5}\}$ (rechts).

Er ist damit konfliktfrei, und beschreibt nur eine Planimplikationsklasse. Der resultierende Plan $S_{247.14.15.16} = B_{36}$ ist somit irreduzibel. Von den Schrägkanten $H = \{\widehat{3.5}, \widehat{4.9}, \widehat{6.7}, \widehat{2.9}, \widehat{2.6}, \widehat{1.5}, \widehat{1.9}\}$ entfällt jedoch die Kante $\widehat{2.9}$ wieder, da sie in B_{36}^{tr} nicht enthalten ist. Die Hinzunahme der Kante $\widehat{1.9}$ hatte neben der Unterbrechung von $P_{18} + \bar{P}_3$ auch die Unterbrechung von $P_9 + \bar{P}_{12}$ zur Folge. Damit wurde die durch die Kante $\widehat{2.9}$ erreichte Unterbrechung von $P_8 + \bar{P}_{15}$ überflüssig. Die bestimmte Schrägkantenmenge H beschreibt also keine minimale trennende Kantenmenge in $G_{\mathcal{R}_H}^*$. ■

Im folgenden Kapitel wollen wir die erzielten Resultate zu der Komplexität des Problems IRRED zusammenfassend darstellen.

Kapitel 7

Komplexität von IRREDUCIBILITY

In den Kapiteln 4, 5 und 6 haben wir das Problem IRRED (Irreducibility) umfassend und unter zahlreichen Gesichtspunkten beschrieben. Insbesondere haben wir dabei eine Reihe verschiedener äquivalenter Problemformulierungen für IRRED kennen gelernt, und für jede Beschreibung versucht, Aussagen über die Zeitkomplexität von IRRED abzuleiten. In diesem Kapitel wollen wir diese Aussagen zur Komplexität gesammelt darstellen und kommentieren.

7.1 REDUCIBILITY und Spezialfälle von IRRED

Für das Problem IRRED lautet die Frage “ist $S \in \mathcal{S}_{SIJ}$ irreduzibel?” Für das komplementäre Problem $\text{co-IRRED} = \text{REDUCIBILITY}$ lautet die Frage dagegen “ist $S \in \mathcal{S}_{SIJ}$ reduzierbar?” Bereits in Abschnitt 4.1.3 haben wir festgehalten, dass co-IRRED in der Komplexitätsklasse NP liegt, und IRRED somit in co-NP liegt. Insbesondere ist der Algorithmus `Reducing1` ein co-NP -Test für IRRED.

Bemerkung 7.1 (IRRED in co-NP) *Das Problem IRRED liegt in co-NP , da das Problem REDUCIBILITY in NP liegt (Satz 4.3).*

Offen ist hingegen die Frage, ob IRRED auch in NP liegt. Für spezielle Operationenmengen liegt IRRED andererseits nicht nur in NP, sondern sogar in $P \subseteq NP \cap \text{co-NP}$. Das gilt zum Einen im Falle von nur zwei Maschinen (Satz 4.65), und zum Anderen für Operationenmengen, die *tree-like* sind (Satz 4.66).

7.2 Umkehren von Planimplikationsklassen

In Abschnitt 5.2 haben wir den polynomiell beschränkten Algorithmus `Naives Umkehren` betrachtet, der einen Ausgangsplan $S \in \mathcal{S}_{SIJ}$ durch Umkehren von genau einer Planimplikationsklasse zu reduzieren versucht. Diese Strategie führt genau dann in jedem Fall zum Erfolg, wenn zu jedem Plan S ein Plan S_i existiert, der durch Umkehren von genau einer Planimplikationsklasse $P_i \in \mathcal{P}_S$ entsteht und der S streng reduziert. Auch wenn es für diese Annahme, im Gegensatz zu anderen Vermutungen in diesem Kapitel, außer der Abwesenheit von Gegenbeispielen keine plausiblen Gründe gibt, wollen wir diese Möglichkeit aus methodischen Gründen dennoch als Vermutung formulieren.

Vermutung 7.2 (Umkehren einer Planimplikationsklasse) *Sei $S \in \mathcal{S}_{SIJ}$ ein normaler Plan, und sei $\mathcal{P}_S = \{P_1, \dots, P_k\}$ die Menge der erweiterten Planimplikationsklassen von S . Dann existiert ein $P_i \in \mathcal{P}_S$ mit $S_i = S - P_i + P_i^{-1} \prec S$.*

Satz 7.3 (Umkehren einer Planimplikationsklasse) *Ist Vermutung 7.2 richtig, dann liegt IRRED in P.*

Beweis. Bei Richtigkeit von Vermutung 7.2 liefert der Algorithmus **Naives Umkehren** genau dann einen streng reduzierenden Plan $S^* \prec S$ von S , wenn S reduzierbar ist (Lemma 5.5). ■

Gilt Vermutung 7.2 nicht, dann existiert ein reduzierbarer Plan $S \in \mathcal{S}_{SIJ}$, der nur durch das simultane Umkehren von mindestens zwei verschiedenen erweiterten Planimplikationsklassen streng reduziert werden kann. Ein solcher Plan hat insbesondere mindestens 4 verschiedene erweiterte Planimplikationsklassen. Auswirkungen auf die Komplexität von IRRED hat die Existenz eines solchen Plans nicht.

7.3 Nichttrivial-stabile Schrägkanten

In Abschnitt 4.3.3 haben wir nichttrivial-stabile Schrägkanten kennen gelernt. Dabei haben wir angedeutet, dass zwar einerseits grundsätzlich nichts gegen Konstellationen mit drei oder mehr verschiedenen Planimplikationsklassen spricht, die zu nichttrivial-stabilen Schrägkanten führen. Aber solche Konstellationen andererseits zu Plänen von so großer Komplexität führen, dass die Infragestellung ihrer Existenz keineswegs abwegig erscheint.

Genau wie im Falle der Reduzierbarkeit durch das Umkehren von genau einer Planimplikationsklasse wollen wir die Auswirkung, die die Existenz oder Nichtexistenz solcher Konstellationen auf die Komplexität von IRRED hat, nicht nur durch einen Satz mit einer Implikation formulieren, sondern aus methodischen Gründen vielmehr auch hier eine zitierbare Vermutung formulieren.

Vermutung 7.4 (nichttrivial-stabile Schrägkanten) *Sei $S \in \mathcal{S}_{SIJ}$ ein normaler Plan, und sei $e \in E_{diag}(S)$ eine stabile Schrägkante. Dann ist e entweder trivial-stabil, oder ihre Stabilität wird durch genau zwei verschiedene erweiterte Planimplikationsklassen induziert.*

Können wir sicher sein, dass die Stabilität einer Schrägkante nur durch höchstens zwei Planimplikationsklassen hervorgerufen werden kann, dann können wir mit einem polynomiell beschränktem Algorithmus diese Planimplikationsklassen zu jeder stabilen Schrägkante identifizieren.

Satz 7.5 (nichttrivial-stabile Schrägkanten) *Ist Vermutung 7.4 richtig, dann liegt IRRED in P.*

Beweis. Die Bedingung aus Vermutung 7.4 lässt sich für jede Schrägkante $e = ab \in E_{diag}(S)$ mit polynomiell Aufwand überprüfen. Hierzu muss lediglich jeder von a ausgehende Weg der Kantenlänge zwei, der aus Kanten von erweiterten Planimplikationsklassen besteht, betrachtet werden. Die Anzahl solcher Wege ist durch $|SIJ|^2 = \mathcal{O}(n^2m^2)$ beschränkt. Angenommen, e ist nichttrivial-stabil. Durch die Überprüfung von “ $e \in (P' + P'')^{tr}$?” ($\mathcal{O}(|SIJ|^3)$ für jede Schrägkante) für die erweiterten Planimplikationsklassen P' und P'' jedes Weges der Länge zwei, kann die Menge der Paare von erweiterten Planimplikationsklassen bestimmt werden, die e transitivieren. Gilt für ein Paar P' und P'' auch $e \in (P' + (P'')^{-1})^{tr}$ bzw. $e \in ((P')^{-1} + P'')^{tr}$, dann ist e stabil. Alle diese Überprüfungen können für jede Schrägkante $e \in E_{diag}(S)$ mit polynomiell beschränktem Aufwand realisiert werden ($\mathcal{O}(|E_{diag}(S)| \cdot |SIJ|^2 \cdot (|SIJ|^3 + |SIJ|^3)) = \mathcal{O}(n^7m^7)$). Der Plan S ist genau dann irreduzibel, wenn jede Schrägkante $e \in E_{diag}(S)$ stabil ist. ■

Die Umkehrung von Satz 7.5 gilt nicht. Ist Vermutung 7.4 nicht richtig, dann hat das auf die Komplexität von IRRED keine Auswirkung.

7.4 Getarnte stabile Schrägkanten

Den Begriff der getarnten stabilen Schrägkante haben wir bei der Beschreibung der Reduktionsalgorithmen in Kapitel 6 eingeführt, um nichttrivial-stabile Schrägkanten, die der Algorithmus **Reducing1** unter Umständen nicht als stabil erkennt von solchen nichttrivial-stabilen Schrägkanten abzugrenzen, deren Stabilität von **Reducing1** auf jedem Rechenweg erkannt wird.

Existenz getarnter stabiler Schrägkanten

Die Eigenschaften von Konstellationen, die zu getarnten stabilen Schrägkanten führen, sind für den Aufwand der Reduktionsalgorithmen `Reducing1` bzw. `Reducing2` von entscheidender Bedeutung. Die Konsequenzen der Existenz oder Nichtexistenz bestimmter Konstellationen für die Komplexität von `IRRED` wollen wir nun in einer Folge von drei abgestuften Vermutungen darstellen. Die stärkste Vermutung in dieser Kette, ist die Vermutung, dass getarnte stabile Schrägkanten überhaupt nicht existieren. Aus ihr folgt die Identität von `Reducing1` und `Reducing2` (Folgerung 6.46), und damit die Zugehörigkeit von `IRRED` zu `P`.

Vermutung 7.6 (getarnte stabile Schrägkanten (1)) *Sei $S \in \mathcal{S}_{SIJ}$ ein normaler Plan. Dann enthält S keine getarnte stabile Schrägkante.*

Getarnte stabile Schrägkanten sind nichttrivial-stabile Schrägkanten, deren Eigenschaften die Beteiligung von drei oder mehr verschiedenen erweiterten Planimplikationsklassen an der die Stabilität induzierenden Konstellation nahe legt. Daher ist Vermutung 7.6 möglicherweise eine Abschwächung von Vermutung 7.4. Sicher ist das jedoch nicht.

Bemerkung 7.7 (getarnte stabile Schrägkanten (1)) *Es ist nicht klar, ob aus der Richtigkeit von Vermutung 7.4 auch die Richtigkeit von Vermutung 7.6 folgt.*

Satz 7.8 (getarnte stabile Schrägkanten (1)) *Ist Vermutung 7.6 richtig, dann liegt `IRRED` in `P`.*

Beweis. Nach Folgerung 6.38 versagt der Algorithmus `Reducing1` höchstens, wenn S getarnte stabile Schrägkanten enthält. Mit der Korrektheit und der polynomiellen Beschränktheit von `Reducing1` (Lemma 6.36), sowie mit der polynomiellen Überführung eines beliebigen Plans in einen normalen Plan (Lemma 5.2) folgt die Zugehörigkeit von `IRRED` zu `P`. ■

Die Umkehrung von Satz 7.8 gilt nicht. Das Problem `IRRED` kann auch bei Existenz von Instanzen mit getarnten stabilen Schrägkanten noch polynomiell lösbar sein.

Nicht-Versagen auf jedem Rechenweg

Bei der Beschreibung von `Reducing2` sind wir auf die Bedeutung der Reihenfolge der Schrägkanten in der Liste $Y = V^{**}$ der Startkanten eingegangen (Bemerkung 6.43). Diese Reihenfolge bestimmt den gewählten Rechenweg für beide randomisierten `Reducing`-Algorithmen in erheblichem Maße.

Existieren Instanzen mit getarnten stabilen Schrägkanten, dann ist eine polynomielle Entscheidung der Irreduzibilität für jede dieser Instanzen trotz der Existenz getarnter stabiler Schrägkanten möglich, sofern die beschriebene (oder eine andere deterministische) Anordnung dazu führt, dass die getarnten stabilen Schrägkanten sehr weit hinten in der Liste Y der Startkanten stehen. Bei der Untersuchung dieser Schrägkanten als Startkanten ist dann die Stabilität aller vorher betrachteten Schrägkanten aus Y bekannt, und die getarnten stabilen Schrägkanten sind unter diesen Umständen möglicherweise nicht länger "getarnt".

Vermutung 7.9 (getarnte stabile Schrägkanten (2)) *Sei $S \in \mathcal{S}_{SIJ}$ ein normaler Plan, und sei $e \in E_{diag}(S)$ eine getarnte stabile Schrägkante. Dann versagt der Algorithmus `Reducing1` bei Anordnung der Schrägkanten in Y gemäß Bemerkung 6.43 (Bedeutung der Reihenfolge in Y) nicht.*

Vermutung 7.9 ist offenkundig eine Abschwächung von Vermutung 7.6.

Lemma 7.10 (getarnte stabile Schrägkanten (2)) *Ist Vermutung 7.6 richtig, dann ist auch Vermutung 7.9 richtig.*

Beweis. Existiert kein Plan $S \in \mathcal{S}_{SIJ}$, der eine getarnte stabile Schrägkante enthält, dann ist die Voraussetzung in der Implikation von Vermutung 7.9 nie erfüllt. Die Implikation ist damit immer richtig. ■

Satz 7.11 (getarnte stabile Schrägkanten (2)) *Ist Vermutung 7.9 richtig, dann liegt `IRRED` in `P`.*

Beweis. Versagt der Algorithmus `Reducing1` nicht, dann entscheidet er in polynomiell beschränkter Zeit, ob ein normaler Plan $S \in \mathcal{S}_{SIJ}$ irreduzibel ist, oder nicht (Lemma 6.36). Mit Lemma 5.2 (normaler Plan) folgt damit die Zugehörigkeit von IRRED zu P. ■

Die Umkehrung von Satz 7.11 gilt nicht zwangsläufig. Es ist denkbar, dass Vermutung 7.9 nicht richtig ist, aber IRRED dennoch in P liegt. In diesem Fall wären die beiden vorgeschlagenen Reduktionsalgorithmen, oder die vorgeschlagene Reihenfolge der Startkanten, nicht geeignet, die Polynomialität von IRRED zu beweisen. Vor dem Hintergrund der strukturellen Tiefe der von den Algorithmen `Reducing1` und `Reducing2` genutzten Eigenschaften von Comparabilitygraphen und Planimplikationsklassen muss dieser Fall jedoch als wenig wahrscheinlich betrachtet werden.

Nicht-Versagen auf mindestens einem Rechenweg

Ist Vermutung 7.9 nicht richtig, dann existieren Instanzen mit getarnten stabilen Schrägkanten, für die der Algorithmus `Reducing1` mit der vorgeschlagenen Anordnung der Startkanten nicht entscheiden kann, ob sie irreduzibel sind, oder nicht. In diesem Fall versagt `Reducing1` also auf einem Rechenweg. Das Problem IRRED liegt dann zwar möglicherweise nicht in P, kann aber immerhin in NP liegen. Ein hinreichendes Kriterium hierfür wäre das *Nicht-Versagen* von `Reducing1` auf mindestens einem Rechenweg für jede irreduzible Instanz. Reduzierbare Instanzen müssen in diesem Zusammenhang nicht weiter berücksichtigt werden, da `Reducing1` ohnehin ein NP-Test für co-IRRED ist.

Vermutung 7.12 (getarnte stabile Schrägkanten (3)) Sei $S \in \mathcal{S}_{SIJ}$ ein normaler Plan, und sei $e \in E_{diag}(S)$ eine getarnte stabile Schrägkante. Dann existiert eine Anordnung der Schrägkanten, so dass der Algorithmus `Reducing1` auf mindestens einem Rechenweg nicht versagt.

Vermutung 7.12 ist damit die schwächste aller bisher in diesem Kapitel formulierten Vermutungen. Ihre Gültigkeit folgt unmittelbar aus der Gültigkeit jeder anderen.

Lemma 7.13 (getarnte stabile Schrägkanten (3)) Ist Vermutung 7.9 richtig, dann ist auch Vermutung 7.12 richtig.

Beweis. Versagt `Reducing1` bei der Anordnung der Schrägkanten gemäß Bemerkung 6.43 (Bedeutung der Reihenfolge in Y) nicht, dann existiert mindestens ein Rechenweg, auf dem `Reducing1` nicht versagt. ■

Im Gegensatz zu einigen anderen genannten Vermutungen besteht die Grundlage für die Vermutung 7.12 nicht nur aus der Abwesenheit von Gegenbeispielen. Vor allem die Überlegungen zu der großen Zahl von Schrägkanten, und dem damit verbundenen hohen Anteil trivial-stabiler Schrägkanten in einem irreduziblen Plan mit getarnten stabilen Schrägkanten (Bemerkung 6.47) lassen die Möglichkeit der völligen Abwesenheit eines Rechenweges, der zur alternativlosen "Enttarnung" aller getarnten stabilen Schrägkanten führt, absolut abwegig erscheinen.

Satz 7.14 (getarnte stabile Schrägkanten (3)) Ist Vermutung 7.12 richtig, dann liegt IRRED in NP.

Beweis. Der Algorithmus `Reducing1` lehnt jede abzulehnende (d.h. reduzierbare) Instanz von IRRED auf jedem erlaubten Rechenweg ab (es wird keine zulässige Entfernungsmenge gefunden) (Lemma 6.36). Versagt `Reducing1` nicht, dann entscheidet er in polynomiell beschränkter Zeit, ob ein normaler Plan $S \in \mathcal{S}_{SIJ}$ irreduzibel ist, oder nicht (Lemma 6.36). Insbesondere nimmt er in diesem Fall eine anzunehmende (d.h. irreduzible) Instanz an (Ausgabe "S ist irreduzibel"). Existiert nun zu jeder anzunehmenden Instanz ein erlaubter Rechenweg, auf dem `Reducing1` die Instanz annimmt, d.h. kann `Reducing1` die Irreduzibilität jedes irreduziblen normalen Plans auf mindestens einem Rechenweg nachweisen, dann ist `Reducing1` ein NP-Algorithmus für das Problem IRRED (Definition 2.2). Mit Lemma 5.2 (normaler Plan) folgt die Zugehörigkeit von IRRED zu NP. ■

Auch von Satz 7.14 gilt die Umkehrung nicht zwangsläufig. Selbst bei Ungültigkeit von Vermutung 7.12 könnte IRRED immernoch in NP liegen. Beispielsweise könnte der mit einer polynomiellen Zeitschranke versehene Algorithmus `Reducing2` irreduzible Instanzen entscheiden, die zwar auf jedem Rechenweg an einzelnen Stellen zu Verzweigungen führen, deren Größe aber auf einigen Rechenwegen polynomiell beschränkt bleibt, so dass der Algorithmus `Reducing2` trotz der Verzweigungen innerhalb der begrenzten Zeit terminiert.

ZPP* und ZPP

Mit der Zugehörigkeit von IRRED zu co-NP folgt aus der Richtigkeit von Vermutung 7.12 unmittelbar die Zugehörigkeit von IRRED zu der Komplexitätsklasse ZPP* (zero-error probabilistic polynomial time) aller Entscheidungsprobleme, für die ein randomisierter Algorithmus mit polynomieller maximaler Laufzeit existiert, der für jede Eingabe der Länge n eine durch $\varepsilon(n) < 1$ beschränkte Versagenswahrscheinlichkeit besitzt. Der Algorithmus **Reducing1** liefert in diesem Fall für *jede* Instanz $S \in \mathcal{S}_{SIJ}$ mit positiver Wahrscheinlichkeit $1 - \varepsilon(|E_{[S^{tr}]}|) > 0$ eine korrekte Antwort für IRRED.

Folgerung 7.15 (getarnte stabile Schrägkanten (3)) *Ist Vermutung 7.12 richtig, dann liegt IRRED in $ZPP^* = NP \cap co-NP$. Insbesondere liegt IRRED dann weder in NPC, noch in co-NPC, sofern $NP \neq co-NP$ gilt, sondern entweder in P, oder in $NPI \cap co-NPI$ (Abbildung 2.2).*

Bemerkung 7.16 (beschränktes Versagen) *Die Überlegungen zu der Anordnung der Schrägkanten in der Liste Y der Startkanten legen nahe, dass bei Richtigkeit von Vermutung 7.12 der Algorithmus **Reducing1** nicht nur auf einigen wenigen, sondern auf sehr vielen Rechenwegen nicht versagt. Ist die Versagenswahrscheinlichkeit $\varepsilon(|E_{[S^{tr}]}|)$ nach oben durch $\frac{1}{2}$ beschränkt, dann liegt IRRED sogar in $ZPP = ZPP(\frac{1}{2})$.*

7.5 IRRED nicht in NP?

Bisher haben wir versucht, die Zugehörigkeit von IRRED zu NP, oder sogar zu P zu begründen. Andersherum können wir uns auch fragen, was gelten muss, wenn IRRED nicht in NP liegt. Welche Eigenschaften müssen irreduzible Instanzen von IRRED haben, die nicht durch einen nichtdeterministischen Algorithmus mit polynomiell beschränkter maximaler Laufzeit als irreduzibel erkannt werden können?

Vermutung 7.12 und NP

Die Nicht-Zugehörigkeit von IRRED zu NP folgt beispielsweise noch nicht aus der Ungültigkeit der schwächsten aller bisher formulierten Vermutungen (Vermutung 7.12). Die Existenz von irreduziblen Instanzen, für die **Reducing1** auf keinem erlaubten Rechenweg die Irreduzibilität nachweisen kann, ist nicht hinreichend für die Existenz von irreduziblen Instanzen $S \in \mathcal{S}_{SIJ}$, für die *kein* nichtdeterministischer Algorithmus mit positiver Wahrscheinlichkeit $1 - \varepsilon(|E_{[S^{tr}]}|) > 0$ die Irreduzibilität nachweisen kann.

Bemerkung 7.17 (Vermutung 7.12 und NP) *Ist Vermutung 7.12 nicht richtig, dann folgt nicht $IRRED \notin NP$.*

co-NPI und co-NPC

Unabhängig von der Frage, ob IRRED in NP liegt, oder nicht, ist klar, dass IRRED in co-NP liegt (Satz 4.3). Aus den Betrachtungen der Komplexitätswelt in $NP \cup co-NP$ (Abbildung 2.2 auf Seite 42) geht hervor, dass aus der Nicht-Zugehörigkeit von IRRED zu NP noch nicht die co-NP-Vollständigkeit von IRRED folgt.

Bemerkung 7.18 (co-NPI und co-NPC) *Liegt IRRED nicht in NP, dann liegt IRRED entweder in co-NPC (co-NP-complete) oder in co-NPI (co-NP-incomplete). Genauer: IRRED liegt dann entweder in co-NPC oder in $co-NPI - (NPI \cap co-NPI)$ (Abbildung 2.2).*

Während nichtdeterministische polynomielle Verfahren zur Lösung von NP-vollständigen Problemen üblicherweise eine sehr große Versagenswahrscheinlichkeit $\varepsilon(n) < 1$ haben, legen die Untersuchungen zu den Reduktionsalgorithmen **Reducing1** bzw. **Reducing2** und zu dem Algorithmus **Naives Umkehren** nahe, dass für reduzierbare Instanzen von co-IRRED die Versagenswahrscheinlichkeit $\varepsilon(n)$ nicht nur kleiner als 1 ist, sondern tatsächlich nahe Null ist. Die Wahrscheinlichkeit, dass ein reduzierbarer Ausgangsplan $S \in \mathcal{S}_{SIJ}$ nicht polynomiell streng reduziert werden kann, ist für jeden der drei Algorithmen sehr klein.

Liegt also IRRED nicht in NP, dann gibt es gute Gründe für die Vermutung, dass co-IRRED ein NP-unvollständiges Problem ist.

Vermutung 7.19 (co-IRRED nicht in NPC) *Das Problem co-IRRED liegt nicht in NPC.*

Satz 7.20 (NP und co-NP) *Gilt Vermutung 7.12, aber Vermutung 7.19 nicht, dann folgt $NP = co-NP$.*

Beweis. Die Aussage folgt direkt aus Satz 2.11. ■

Satz 7.21 (co-IRRED nicht in NPC) *Gilt $NP \neq co-NP$, dann folgt Vermutung 7.19 aus Vermutung 7.12.*

Beweis. Gilt $NP \neq co-NP$, dann folgt $NP \cap co-NPC = \emptyset$ (Satz 2.11). Mit Satz 7.14 folgt dann IRRED \notin co-NPC. ■

Nichtdeterministisch nicht erkennbare irreduzible Pläne

Zum Abschluss dieses Kapitels wollen wir die Eigenschaften einer irreduziblen Instanz sammeln, die auf keinem Rechenweg eines nichtdeterministischen Verfahrens mit polynomiell beschränkter Laufzeit als irreduzibel erkannt werden kann. Offenkundig ist bei der Nicht-Zugehörigkeit von IRRED zu NP keine der in den Abschnitten 7.2 bis 7.4 dieses Kapitels formulierten Vermutungen gültig.

Die in Bemerkung 7.22 aufgelisteten Eigenschaften leiten sich überwiegend aus der Ungültigkeit dieser Vermutungen ab.

Bemerkung 7.22 (Instanzen ohne NP-Test) *Liegt IRRED nicht in NP, dann existiert zu jeder natürlichen Zahl d eine irreduzible Instanz $S \in \mathcal{S}_{SIJ}$, die auf keinem erlaubten Rechenweg eines durch $\mathcal{O}(|SIJ|^d) = \mathcal{O}(n^d m^d)$ beschränkten Verfahrens als irreduzibel erkannt wird. Sei $\mathcal{P}_S = \{P_1, \dots, P_k\}$ die Menge der erweiterten Planimplikationsklassen von S . Dann hat S die folgenden Eigenschaften:*

- (i) *Für alle $L \subseteq \{1, \dots, k\}$ mit $|L| \leq l = l(d)$ ist $S_L = S - (\bigcup_{i \in L} P_i) + (\bigcup_{i \in L} P_i^{-1})$ entweder zyklisch, oder enthält mindestens eine neue Schrägkante, $E_{diag}(S_L) \not\subseteq E_{diag}(S)$.*
- (ii) *Insbesondere ist $S_i = S - P_i + P_i^{-1}$ für alle $i \in \{1, \dots, k\}$ entweder nicht kreisfrei, oder enthält mindestens eine neue Schrägkante.*
- (iii) *Insbesondere gilt $|\mathcal{P}_S| = k \geq 4$.*
- (iv) *S enthält mindestens zwei getarnte stabile Schrägkanten, die in **Reducing2**, unabhängig von der Kenntnis der Stabilität anderer Schrägkanten, jeweils zu einer Verzweigung des Suchbaums führen, die nur durch $\mathcal{O}(n^{d^*} m^{d^*})$ mit $d^* > d$ beschränkt werden kann.*
- (v) *Jede trennende Kantenmenge in $G_{\mathcal{R}_H}^*$ bezüglich $H = \emptyset$, die jeden Knoten $v \in V_{G_{\mathcal{R}_H}^*}$ von $v^{-1} \in V_{G_{\mathcal{R}_H}^*}$ trennt, führt zu einem Plan S^* mit mindestens einer neuen Schrägkante, $E_{diag}(S^*) \not\subseteq E_{diag}(S)$.*

Jede dieser Eigenschaften führt für sich genommen, zu einem Plan mit vielen Schrägkanten. Nicht nur die Existenz von getarnten stabilen Schrägkanten induziert eine große Zahl weiterer Schrägkanten, sondern auch die große Zahl der Planimplikationsklassen. Während zwar irreduzible Pläne mit beliebig großer Anzahl von Planimplikationsklassen existieren (Bemerkung 4.61), verträgt sich die sehr große Anzahl von Schrägkanten in einem solchen nichtdeterministisch nicht erkennbaren irreduziblen Plan $S \in \mathcal{S}_{SIJ}$ jedoch nur äußerst schlecht mit der Annahme, dass ein signifikanter Anteil dieser Schrägkanten nicht sofort als trivial-stabil erkennbar sein soll (vgl. Bemerkung 4.57). Noch viel weniger verträglich damit ist die Annahme, dass sich unter diesen nicht sofort als trivial-stabil erkennbaren Schrägkanten mehrere Schrägkanten befinden sollen, die, jede für sich genommen, selbst bei Kenntnis der Stabilität aller bereits betrachteten Schrägkanten, erst nach der Bearbeitung eines umfangreichen und in keinem Fall polynomiell beschränkbaren Suchbaum, als stabil erkennbar sein soll.

Damit schließen wir unsere Untersuchungen zur Zeitkomplexität von IRRED ab, und beenden den zweiten Teil dieser Arbeit.

Schlussbemerkungen

Ziel dieser Arbeit war die Einordnung der Zeitkomplexität der Erkennung der Reduzierbarkeit von Lösungen des Open-Shop Schedulingproblems in die Komplexitätsklassen innerhalb von $\text{NP} \cup \text{co-NP}$. Zur Beschreibung des Problems IRRED (Irreducibility) stehen mehrere äquivalente Formulierungen zur Verfügung. Neben der ursprünglichen Fragestellung nach der Irreduzibilität eines Plans, waren die alternativen Fragestellungen nach der Nichtexistenz eines echt enthaltenen H-Comparabilitygraphen, bzw. nach der Stabilität aller Schrägkanten bereits bekannt. Neu sind dagegen die äquivalenten Fragestellungen nach der Unzulässigkeit aller Rekombinationen der Planimplikationsklassen, sowie der Unzulässigkeit aller Entfernungsmengen.

Die Modellierung von Plänen zum Open-Shop Problem erfolgt dabei mittels H-Comparabilitygraphen, d.h. mittels induzierter Teilgraphen des Hamming-Graphen $K_n \times K_m$, die jeweils um eine geeignete Menge von Schrägkanten erweitert worden sind. Jede der alternativen Problemformulierungen von IRRED liefert unterschiedliche Einsichten in die Eigenschaften von H-Comparabilitygraphen.

Von zentraler Bedeutung für die Reduzierbarkeit von Plänen ist dabei der Begriff der Planimplikationsklasse. Neben der bereits bekannten Definition als Äquivalenzklassen der transitiven Hülle der Γ_S -Relation, wurde eine alternative Auffassung als $\dot{\Gamma}_F$ -Komponenten bezüglich der Kantenmenge $F = E_{diag}(S)$ vorgestellt. Mit Hilfe dieser Darstellung der Planimplikationsklassen wurde die Bedeutung des Begriffs der trennenden Kantenmenge $F_{i,j}$ zwischen zwei Planimplikationsklassen P_i und P_j herausgearbeitet.

Ausgangspunkt für die Suche nach einer Charakterisierung irreduzibler Pläne war die Vermutung, dass die hinreichende Bedingung der trivialen Stabilität aller Schrägkanten auch notwendig ist. Die Betrachtung der Analogien zwischen immer-transitiven Kanten in verschiedenen Kombinationen der Implikationsklassen eines Comparabilitygraphen und stabilen Schrägkanten in verschiedenen Kombinationen der Planimplikationsklassen eines H-Comparabilitygraphen führte zur Falsifizierung dieser Vermutung. Aus diesen Überlegungen wurde der zentrale Begriff der nichttrivial-stabilen Schrägkanten abgeleitet.

Aus dem Ansatz der Bestimmung einer zulässig entfernbaren Schrägkantenmenge aus dem H-Comparabilitygraphen $[S^{tr}]$ eines Ausgangsplans $S \in \mathcal{S}_{SIJ}$, wurde ein Verfahren entwickelt, das zu einer nicht zulässigen Entfernungsmenge $M \subseteq E_{diag}(S)$ geeignete Erweiterungskanten identifiziert.

Mit Hilfe dieses Reduktionsschritts wurden zwei Reduzierungsalgorithmen formuliert. Während der Algorithmus **Reducing1** in polynomiell beschränkter Laufzeit mit hoher Wahrscheinlichkeit einen streng reduzierenden Plan findet, wenn der Ausgangsplan $S \in \mathcal{S}_{SIJ}$ reduzierbar ist, findet der Algorithmus **Reducing2** einen solchen mit Sicherheit, und mit hoher Wahrscheinlichkeit mit demselben Aufwand wie **Reducing1**. Ist der Ausgangsplan irreduzibel, dann versagt **Reducing1** unter Umständen, während **Reducing2** die Irreduzibilität mit Sicherheit erkennt, jedoch unter Umständen nur mit exponentieller Laufzeit.

Sowohl die Laufzeit von **Reducing2**, als auch das mögliche Versagen von **Reducing1**, hängen dabei von der Erkennbarkeit der Stabilität bestimmter nichttrivial-stabiler Schrägkanten ab. Diese Schrägkanten wurden als getarnte stabile Schrägkanten bezeichnet. Für einen Ausgangsplan, der keine dieser Schrägkanten enthält, sind die beiden Algorithmen **Reducing1** und **Reducing2** in Ablauf und Laufzeit identisch. Ob überhaupt Pläne existieren, die getarnte stabile Schrägkanten enthalten, ist offen. Die mutmaßliche Mindestgröße solcher Pläne, verbunden mit der völligen Abwesenheit von globalen Eigenschaften, die lokal zur Einschränkung des Suchraums genutzt werden können, lassen die Suche nach solchen Plänen wenig aussichtsreich erscheinen.

Aus einer alternativen Interpretation des Reduktionsgraphen wurde sowohl eine Möglichkeit zur Konstruktion von irreduziblen Plänen, als auch eine Nachbarschaft zwischen irreduziblen Plänen abgeleitet. Beide Ideen haben jedoch keine Auswirkung auf die Komplexität von IRRED, da die so erzeugten irreduziblen Pläne unter Umständen nicht den gegebenen Ausgangsplan reduzieren.

Ausgangspunkt für die Untersuchung der Komplexität von IRRED war die Kenntnis der Zugehörigkeit von IRRED zu co-NP. Vor dem Hintergrund der offenen Frage “NP \neq co-NP?”, und der damit verbundenen Frage “P \neq NP?”, lag der Fokus der Untersuchung auf der Bestimmung von Bedingungen, unter denen IRRED in NP liegt. Existieren keine Instanzen von IRRED, die getarnte stabile Schrägkanten enthalten, dann versagt der polynomiell beschränkte Algorithmus `Reducing1` auf keinem Rechenweg. Das Problem IRRED liegt dann nicht nur in NP, sondern sogar in P.

Existieren zwar Instanzen von IRRED, die getarnte stabile Schrägkanten enthalten, aber bewirkt die Anordnung der Schrägkanten in der Liste der Startkanten nach einem beschriebenen Muster, dass die getarnten stabilen Schrägkanten erst im hinteren Teil des Verfahrens betrachtet werden, dann versagt `Reducing1` ebenfalls auf keinem Rechenweg. Das Problem IRRED liegt auch unter diesen Bedingungen in P.

Existieren hingegen irreduzible Instanzen von IRRED, die getarnte stabile Schrägkanten enthalten, für die die beschriebene Anordnung der Startkanten zu einem Rechenweg führen kann, auf dem `Reducing1` versagt, dann ist die Zugehörigkeit von IRRED zu NP davon abhängig, ob in solchen Fällen ein anderer Rechenweg existiert, auf dem die Stabilität aller Schrägkanten durch `Reducing1` nachgewiesen werden kann. Aus den Eigenschaften von getarnten stabilen Schrägkanten konnte abgeleitet werden, dass eine solche irreduzible Instanz eine große Anzahl von Schrägkanten enthält, deren Stabilität sich sehr leicht nachweisen lässt. Der Algorithmus `Reducing1` ist damit genau dann ein NP-Test für IRRED, wenn jede getarnte stabile Schrägkante bei vorherigem Nachweis der Stabilität aller nicht getarnten stabilen Schrägkanten, auf mindestens einem erlaubten Rechenweg als stabile Schrägkante erkannt werden kann. Es wurde begründet, dass die Existenz von Instanzen, für die `Reducing1` kein NP-Test ist, kaum denkbar ist.

Die Ergebnisse dieser Arbeit sind in die graphentheoretische Grundlagenforschung zur Schedulingtheorie einzuordnen. Der Aspekt der möglichen NP- bzw. co-NP-Unvollständigkeit von IRRED liefert einen Beitrag zur Komplexitätstheorie. Aus den Untersuchungen ergeben sich unter anderem die folgenden weiterführenden Fragestellungen:

- Kann quantitativ ermittelt werden, ob `Reducing1` versagen kann? Können quantitativ Instanzen ermittelt werden, für die `Reducing2` keine durch den Aufwand von `Reducing1` begrenzte Laufzeit hat?
- Führt die deutliche Verkleinerung des Suchraums bei der vorgeschlagenen Nachbarschaft zwischen irreduziblen Plänen trotz der aufwendigen Bestimmung eines Nachbarn zu besseren iterativen Verbesserungsverfahren?
- Können die Ergebnisse und deren Bewertungen zu den Bedingungen für die Existenz eines NP-Tests auf Irreduzibilität auf andere reguläre Zielfunktionen übertragen werden?
- Ist das Problem IRRED ein Kandidat für ein NP-unvollständiges Problem?

Liegt IRRED in NP, dann folgt die Zugehörigkeit von IRRED zu der Klasse $ZPP^* = NP \cap co-NP$. Könnte in diesem Fall die Zugehörigkeit von IRRED zu P ausgeschlossen werden, dann wäre IRRED ein Kandidat für ein NP-unvollständiges Problem. Könnte dann die NP- oder die co-NP-Vollständigkeit ausgeschlossen werden, dann würde NP \neq co-NP, und damit P \neq NP folgen. Die Vermutung, dass IRRED nicht co-NP-vollständig ist, ist vor dem Hintergrund der Vermutung NP \neq co-NP eine noch schwächere Vermutung als die Zugehörigkeit von IRRED zu NP.

Anhang A

REDUCING1 und REDUCING2

In diesem Anhang wollen wir die beiden im Kapitel 6 entwickelten Varianten eines Reduktionsalgorithmus und die enthaltenen Prozeduren vollständig und ein wenig detaillierter darstellen.

Eingabe für beide Verfahren ist ein normaler Plan $S \in \mathcal{S}_{SIJ}$ (Abschnitt 5.1) über einer Operationenmenge $SIJ \subseteq I \times J$ mit n Aufträgen und m Maschinen. Für die Anzahl der Knoten und Kanten im H-Comparabilitygraphen $[S^{tr}] = (SIJ, E_{reg}(S) + E_{diag}(S))$ gelten $|SIJ| = nm$, bzw. $|E_{reg}(S) + E_{diag}(S)| = |E_{reg}(S)| + |E_{diag}(S)| = \mathcal{O}(n^2m^2) + \mathcal{O}(n^2m^2) = \mathcal{O}(n^2m^2)$.

A.1 Teilprozeduren

Aktualisierung des Reduktionsgraphen (updateRedGraph)

Die Prozedur `updateRedGraph` (Abschnitt 6.3.1, Seite 130) aktualisiert den Reduktionsgraphen $G_{\mathcal{R}_M}$, nachdem die Entfernungsmenge M um die Schrägkante e^* erweitert worden ist. Sie wird von der rekursiven Schleife im Algorithmus `Reducing2` (Prozedur `findPossibleEdges`), aber auch direkt von den Algorithmen `Reducing1` und `Reducing2` aufgerufen. Der Aufwand von `updateRedGraph` ist beschränkt durch $\mathcal{O}(n^5m^5)$ (Lemma 6.26).

```
updateRedGraph( $M, G_{\mathcal{R}_M}, G_{\mathcal{K}}, \hat{e}^*$ ) {
   $G_{\mathcal{R}_{M^*}} := (G_{\mathcal{R}_M} - e^* - \bar{e}^*) + G_{\mathcal{K}_{e^*}}$ ;
  while ( $\exists \widehat{vv'} \in E_{\mathcal{R}_{M^*}}$  mit  $v, v' \in V_{M^*}^*$ ) do {
    Kontrahiere ggf. Knoten aus  $V_{M^*}^*$ ;
    Bestimme die transitiven Hüllen der Planimplikationsklassen  $v \in V_{M^*}^*$ ;
    Kontrahiere ggf. trivial-stabile Schrägkanten  $w \in V_{M^*}^{**}$  mit
      zugehörigen Planimplikationsklassen  $v \in V_{M^*}^{**}$ ;
  };
  return  $G_{\mathcal{R}_{M^*}}$ ;
}
```

Bestimmung eines Konflikts (findNextEdge)

Die Prozedur `findNextEdge` (Abschnitt 6.3.2, Seite 133) findet einen Konflikt in einem nicht zulässigen Reduktionsgraphen $G_{\mathcal{R}_M} = (V_M^* + V_M^{**}, E_{\mathcal{R}_M})$. Dabei wird zunächst direkt nach Konflikten der Ordnung 0 bzw. 1 gesucht. Nur wenn $G_{\mathcal{R}_M}$ weder direkte Konflikte, noch Konflikte erster Ordnung enthält, wird nach einem beliebigen Konflikt höherer Ordnung gesucht. Ist ein Konflikt identifiziert, wird in der Menge $X \subseteq V_M^{**}$ der beteiligten Schrägkanten nach einer Kante \hat{e}^* gesucht, deren Entfernung aus $[S^{tr}] - M$ nicht sofort zu einem direkten Konflikt führt. Zurückgegeben wird entweder die Kante \hat{e}^* , oder, als Variante, die Menge X der möglichen Erweiterungskanten des identifizierten Konflikts, ggf. bereinigt um einige sofort als stabil bezüglich M erkennbaren Schrägkanten.

Die Prozedur `findNextEdge` wird von den Prozeduren `findX` und `findPossibleEdges`, sowie direkt vom Algorithmus `Reducing2` aufgerufen. Der Aufwand von `findNextEdge` ist beschränkt durch $\mathcal{O}(n^4m^4)$ (Lemma 6.30).

```

findNextEdge( $M, G_{\mathcal{R}_M}, G_{\mathcal{K}}$ ) {
  // (1) Teste auf Existenz direkter Konflikte
  forall  $v \in V_M^*$  do {
    if ( $\widehat{vv^{-1}} \in E_{\mathcal{R}_M}$ ) then return “ $M$  nicht zulässig erweiterbar”;
  };
  // (2) Teste auf Existenz Konflikter 1. Ordnung
  forall  $v \in V_M^*$  do {
    forall  $\widehat{vw} \in E_{\mathcal{R}_M}$  do{
      if ( $\widehat{wv^{-1}} \in E_{\mathcal{R}_M}$ ) then {
         $e^* := w \in V_M^{**} \subseteq E_{diag}(S)$ ;
        if (Konsequenzen von  $e^*$  führen zu direktem Konflikt in  $G_{\mathcal{R}_M}$  or
           $\hat{e}^*$  “stabil”) then {
          return “ $M$  nicht zulässig erweiterbar”;
        };
        else return  $\hat{e}^*$ ;
      };
    };
  };
  // (3) Finde irgendeinen Konflikt höherer Ordnung
  forall  $v \in V_M^*$  do {
    if ( $v$  und  $v^{-1}$  liegt in gemeinsamer Komponente) then {
      Finde kürzesten Weg  $W \subseteq V_{\mathcal{R}_M}$  zwischen  $v$  und  $v^{-1}$ ;
       $X := W \cap V_M^{**}$ ;
      while ( $X \neq \emptyset$ ) do {
        Wähle eine Schrägkante  $e^* \in X$ ;
        if (Konsequenzen von  $e^*$  führen zu direktem Konflikt in  $G_{\mathcal{R}_M}$  or
           $\hat{e}^*$  “stabil”) then {
           $X := X - \hat{e}^*$ ;
        };
        else return  $\hat{e}^*$ ;
      };
      return “ $M$  nicht zulässig erweiterbar”;
    };
  };
}

```

Bestimmung einer Menge von möglichen Erweiterungskanten in Reducing1 (findX)

Die Prozedur `findX` wird nur vom Algorithmus `Reducing1` benutzt. Erkennt `Reducing1` in einem Ast des Suchbaums die aktuelle Entfernungsmenge M als unzulässig, d.h. nicht mehr zulässig erweiterbar, dann geht er zum letzten Entscheidungsknoten zurück. Die Prozedur `findX` stellt sicher, dass bei einem solchen Rückschritt (die Variable `StepBack` ist dann auf `TRUE` gesetzt), keine neue Menge X möglicher Erweiterungskanten bestimmt wird, sondern die bereits bestimmte erneut betrachtet wird. Der Aufwand von `findX` ist identisch zu dem von `findNextEdge` ($\mathcal{O}(n^4m^4)$).

```

findX( $M, G_{\mathcal{R}_M}, G_{\mathcal{K}}, StepBack$ ) {
  if ( $StepBack = FALSE$ ) then {
    Bestimme alternativloses  $\hat{e}^*$  (nicht stabil) nach Schritt (1), (2) oder (3)
    in findNextEdge( $M, G_{\mathcal{R}_M}, G_{\mathcal{K}}$ ), und setze  $X := \{\hat{e}^*\}$ ,
    oder bestimme  $X := W \cap V_M^{**}$  nach Schritt (3);
  };
  else  $X := X'$ ;
   $StepBack := FALSE$ ;
}

```


Wahl einer Verzweigung in Reducing1 (branch)

Die Prozedur `branch` wird nur vom Algorithmus `Reducing1` aufgerufen. Sie speichert lediglich die Situation an einem Entscheidungsknoten im Suchbaum. Sie wird aufgerufen, wenn die Menge X der (noch) möglichen Erweiterungskanten aus dem identifizierten Konflikt mindestens zwei verschiedene Schrägkanten enthält. Bei einem Rückschritt wird auf diese Kopien von M , X und \hat{e} zurückgegriffen. Der Aufwand von `branch` ist beschränkt durch $\mathcal{O}(|M|) = \mathcal{O}(|X|) = \mathcal{O}(|E_{diag}(S)|) = \mathcal{O}(n^2m^2)$.

```
branch( $M, G_{\mathcal{R}_M}, G_{\mathcal{K}}, X$ ) {
  Wähle ein  $\hat{e}^* \in X$  (nicht stabil);
   $M' := M$ ;
   $X' := X$ ;
   $\hat{e}' := \hat{e}^*$ ;
}
```

Verfolgung eines alternativlosen Astes im Suchbaum bei Reducing1 (dontBranch)

Die Prozedur `dontBranch` wird nur vom Algorithmus `Reducing1` aufgerufen. Ist die im aktuellen Schritt gewählte Erweiterungskante \hat{e}^* alternativlos, und führt dennoch zu einem direkten Konflikt (d.h. M ist nicht mehr zulässig erweiterbar), dann wird ein Rückschritt im Suchbaum zum letzten Entscheidungsknoten initiiert (Variable `StepBack` wird auf `TRUE` gesetzt), sofern dort mindestens eine noch nicht betrachtete Alternative zur Auswahl steht ($|X'| \geq 2$). Andernfalls wird der gesamte Ast des Suchbaums verworfen (Variable `NewEdge` wird auf `TRUE` gesetzt). Unter Umständen kann die Startkante \hat{e} als stabile Kante identifiziert werden.

Der Aufwand von `dontBranch` ist durch die Prüfung der zulässigen Erweiterbarkeit von M bestimmt. Dazu müssen alle Kanten in $G_{\mathcal{K}_{\hat{e}^*}}$ betrachtet werden, d.h. $\mathcal{O}(nm)$ viele (Lemma 6.5). Alle übrigen Schritte sind ebenfalls in $\mathcal{O}(nm)$ realisierbar.

```
dontBranch( $M, M', G_{\mathcal{R}_M}, G_{\mathcal{K}}, X, X', Y, \hat{e}, \hat{e}', NewEdge, StepBack$ ) {
  if (“ $M$  nicht zulässig erweiterbar”) then {
    if ( $|X'| = 1$ ) then {
       $NewEdge := TRUE$ ;
       $Y := Y - \hat{e}$ ;
      if (alle Kanten in  $M - \hat{e}$  sind alternativlos) then Markiere  $\hat{e}$  als stabil;
    };
    else {
       $StepBack := TRUE$ ;
       $M := M'$ ;
       $X' := X - \hat{e}'$ ;
    };
  };
}
```

Rekursive Schleife in Reducing2 (findPossibleEdges)

Die Prozedur `findPossibleEdges` wird nur vom Algorithmus `Reducing2` und von sich selbst aufgerufen. Sie regelt die Verzweigung an jedem echten Entscheidungsknoten im Suchbaum.

Der Aufwand eines Durchlaufens der Schleife (bis zum Selbstaufzuruf) ist bestimmt durch die Erkennung einer Menge X von Schrägkanten eines Konflikts in $G_{\mathcal{R}_M}$, $\mathcal{O}(n^4m^4)$, und durch die Aktualisierung von $G_{\mathcal{R}_M}$, $\mathcal{O}(n^5m^5)$. Durch die Verzweigung in der inneren Schleife ist die Gesamtlaufzeit eines Aufrufs von `findPossibleEdges` unter Umständen nur durch $\mathcal{O}(n^5m^5 \cdot |E_{diag}(S)|^{|E_{diag}(S)|})$ beschränkt. Bei Vermeidung der mehrfachen Betrachtung einer Entfernungsmenge M kann der Aufwand auf $\mathcal{O}(n^5m^5 \cdot 2^{|E_{diag}(S)|}) = \mathcal{O}(n^5m^5 \cdot 2^{n^2m^2})$ reduziert werden (Lemma 6.44).

```
findPossibleEdges( $M, G_{\mathcal{R}_M}, G_{\mathcal{K}}$ ) {
  while ( $V_M^{**} - M \neq \emptyset$ ) do {
```

```

Bestimme alternativloses  $\hat{e}^*$  (nicht stabil) nach Schritt (1), (2) oder (3)
  in findNextEdge( $M, G_{\mathcal{R}_M}, G_{\mathcal{K}}$ ), und setze  $X := \{\hat{e}^*\}$ ,
  oder bestimme  $X := W \cap V_M^{**}$  nach Schritt (3);
if not (“ $M$  nicht zulässig erweiterbar”) then {
  while ( $X \neq \emptyset$ ) do {
    Wähle ein  $\hat{e}^* \in X$  (nicht stabil);
     $M := M + \hat{e}^*$ ;
    updateRedGraph( $M, G_{\mathcal{R}_M}, G_{\mathcal{K}}, \hat{e}^*$ );
    if ( $G_{\mathcal{R}_M}$  ist konfliktfrei) then return  $M$ ;
    else findPossibleEdges( $M, G_{\mathcal{R}_M}, G_{\mathcal{K}}$ );
  };
};
}

```

A.2 REDUCING1

Der Algorithmus `Reducing1` (Abschnitt 6.4.1, Seite 138) durchsucht, ausgehend von einer Startkante $\hat{e} \in Y$, im Wesentlichen einen Ast des Suchbaums aller denkbaren Entfernungsmengen, die \hat{e} enthalten. Dadurch bleibt der Aufwand polynomiell beschränkt ($\mathcal{O}(n^9 m^9)$, Lemma 6.36). Vor dem Abbruch der Betrachtung dieses Astes im Suchbaum zu \hat{e} erfolgt lediglich ein Schritt zurück zum letzten echten Entscheidungsknoten. Dadurch wird die Wahrscheinlichkeit, eine zulässige Entfernungsmenge M zu finden, erhöht, aber gleichzeitig bleibt der Gesamtaufwand polynomiell beschränkt.

Dieser Rückschritt zum jeweils letzten echten Entscheidungsknoten wird durch die boolesche Variable `StepBack` realisiert (in der Prozedur `dontBranch`). Der betreffende Entscheidungsknoten wird durch die kopierten Mengen M' , X' und \hat{e}' beschrieben. Wird im untersuchten Ast des Suchbaums zu der Startkante $\hat{e} \in Y$ keine zulässige Entfernungsmenge M entdeckt, dann wird die Startkante \hat{e} verworfen, und eine noch unbetrachtete Schrägkante aus Y als neue Startkante betrachtet. Der Abbruch der Betrachtung der Startkante \hat{e} wird durch die boolesche Variable `NewEdge` realisiert.

Könnte bei der Betrachtung von $\hat{e} \in Y$ die Stabilität der Startkante \hat{e} nachgewiesen werden, dann wird \hat{e} als stabil markiert, und entsprechend nicht mehr berücksichtigt. Könnte die Stabilität von \hat{e} nicht zweifelsfrei nachgewiesen werden, dann wird \hat{e} nicht als stabil markiert.

Während die Rückgaben einer zulässigen Entfernungsmenge M und die Feststellung der Irreduzibilität von S immer fehlerfrei sind, ist bei der Rückgabe “?” nicht zweifelsfrei klar, ob der Ausgangsplan S reduzierbar ist, und nur keine zulässige Entfernungsmenge entdeckt werden konnte, oder ob er irreduzibel ist, aber die Stabilität der Schrägkanten nicht in jedem Fall zweifelsfrei nachgewiesen werden konnte. Vor dem Hintergrund der Frage der Existenz oder Nichtexistenz von Instanzen mit getarnten stabilen Schrägkanten, bzw. deren Häufigkeit, muss die Wahrscheinlichkeit, dass ein reduzierbarer Ausgangsplan durch `Reducing1` nicht reduziert werden kann, als sehr gering eingeschätzt werden. Enthält der Ausgangsplan keine getarnten stabilen Schrägkanten, dann löst `Reducing1` das Problem `IRRED` in polynomieller Laufzeit ($\mathcal{O}(n^7 m^7)$, Satz 6.39).

Algorithmus REDUCING1

Input: $[S^{tr}]$ zu normalem Plan $S \in \mathcal{S}_{SIJ}$.

Output: zulässige Entfernungsmenge M , “ S ist irreduzibel”, oder “?” (Versagen).

BEGIN {

Bestimme erweiterte Planimplikationsklassen $\mathcal{P}_S = \{P_1, \dots, P_k\}$;

Bestimme Faktorgraph $G_{\mathcal{F}}$ und Konsequenzgraph $G_{\mathcal{K}}$;

Setze $M := \emptyset$;

Setze $G_{\mathcal{R}_M} = (V_M^* + V_M^{**}, E_{\mathcal{R}_M}) := G_{\mathcal{F}}$;

Setze $Y := V_M^{**}$;

while ($Y \neq \emptyset$) **do** {

Setze `NewEdge`: = `FALSE`;

Setze `StepBack`: = `FALSE`;

Wähle ein $\hat{e} \in Y$ (nicht stabil);

```

    M: = {ê};
    updateRedGraph(M, GRM, GK, ê);
    while (VM** - M ≠ ∅) and (NewEdge = FALSE) do {
        findX(M, GRM, GK);
        if (X = {ê*}) then {
            dontBranch(M, M', GRM, GK, X, X', Y, ê, ê', NewEdge, StepBack);
        };
        else {
            branch(M, GRM, GK, X);
        };
        M: = M + ê*;
        updateRedGraph(M, GRM, GK, ê*);
        if (GRM ist konfliktfrei) then return M;
    };
};
if (alle e ∈ Y sind stabil) then return "S ist irreduzibel";
else return "?";
};
END.

```

A.3 REDUCING2

Der Algorithmus **Reducing2** (Abschnitt 6.4.2, Seite 143) durchsucht, ausgehend von einer Startkante $\hat{e} \in Y$, in einer Tiefensuche den gesamten Suchbaum zu dieser Startkante, bis eine zulässige Entfernungsmenge M mit $\hat{e} \in M$ gefunden ist. Wird keine gefunden, dann ist \hat{e} stabil. **Reducing2** ist ein randomisierter Algorithmus, der das Problem **IRRED** ohne Fehler- oder Versagenswahrscheinlichkeit mit exponentieller Laufzeit löst (Lemma 6.44). Um die Laufzeit so gering wie möglich zu halten, sollten die Kanten in der Menge Y geeignet (Qualität der jeweiligen Konsequenzen) sortiert werden (Bemerkung 6.43).

Für reduzierbare Pläne $S \in \mathcal{S}_{SIJ}$ unterscheidet sich **Reducing2** kaum von **Reducing1**. Während **Reducing1** solche Pläne in polynomieller Laufzeit mit sehr hoher Wahrscheinlichkeit reduzieren kann, findet **Reducing2** mit Sicherheit einen streng reduzierenden Plan, und diesen mit hoher Wahrscheinlichkeit mit derselben Laufzeit wie **Reducing1**.

Im Falle von irreduziblen Plänen $S \in \mathcal{S}_{SIJ}$ steht die Irreduzibilität erst nach Betrachtung des gesamten Suchbaums fest, und der Aufwand kann daher nur exponentiell abgeschätzt werden. Da im Laufe des Verfahrens der Suchbaum durch die nachgewiesene Stabilität aller bereits betrachteten Startkanten immer kleiner wird, und der Nachweis der Stabilität einer einzelnen Kante in vielen Fällen sehr schnell erreicht werden kann, gibt es keinen Grund anzunehmen, dass die durchschnittliche Laufzeit deutlich über der Laufzeit der Bestimmung eines reduzierenden Plans eines reduzierbaren Ausgangsplans liegt. Insbesondere ist der Aufwand von **Reducing2** auch für irreduzible Ausgangspläne $S \in \mathcal{S}_{SIJ}$ polynomiell ($\mathcal{O}(n^7 m^7)$), wenn S keine getarnten stabilen Schrägkanten enthält (Lemma 6.45).

Algorithmus REDUCING2

Input: $[S^{tr}]$ zu normalem Plan $S \in \mathcal{S}_{SIJ}$.

Output: zulässige Entfernungsmenge M oder "S ist irreduzibel".

BEGIN {

Bestimme erweiterte Planimplikationsklassen $\mathcal{P}_S = \{P_1, \dots, P_k\}$;

Bestimme Faktorgraph $G_{\mathcal{F}}$ und Konsequenzgraph $G_{\mathcal{K}}$;

Setze $M: = \emptyset$;

Setze $G_{\mathcal{R}_M} = (V_M^* + V_M^{**}, E_{\mathcal{R}_M}) : = G_{\mathcal{F}}$;

Setze $Y: = V_M^{**}$;

while ($Y \neq \emptyset$) **do** {

Wähle ein $\hat{e} \in Y$ (nicht stabil);

$M: = \{\hat{e}\}$;

```
    updateRedGraph( $M, G_{\mathcal{R}_M}, G_{\mathcal{K}}, \hat{e}$ );  
    findPossibleEdges( $M, Y, \hat{e}$ );  
    Markiere  $\hat{e}$  als stabil;  
     $Y := Y - \hat{e}$ ;  
  };  
  return "S ist irreduzibel";  
};  
END.
```

Symbolverzeichnis

$B\alpha A$	Problem B ist auf Problem A reduzierbar	39
$ab\Gamma cd$	ab und cd direkt Γ -verbunden	9
$e\Gamma P_i$	e direkt Γ -verbunden zu einer Kante aus P_i	79
$ab\Gamma^* cd$	ab und cd durch Γ -Kette verbunden	10
$ab\dot{\Gamma}_F cd$	$ab\Gamma cd$ und $ab, cd \notin F \subseteq E$	76
$ab\dot{\Gamma}_F^* cd$	ab und cd durch $\dot{\Gamma}$ -Kette verbunden	76
$ab\Gamma_S cd$	ab und cd direkt $\dot{\Gamma}_{E_{diag}(S)}$ -verbunden	75
$ab\Gamma_S^* cd$	ab und cd liegen in einer Planimplikationsklasse	75
$\chi(G), \chi_w(G)$	chromatische Zahl und Intervall-Färbungszahl von G	33, 34
$\omega(G), \omega_w(G)$	Cliquenzahl (gewichtet) von G	33, 34
$A + B = A \cup B$	Summe (Vereinigung) zweier Kantenmengen	7
$A - B = A \setminus B$	Differenz zweier Kantenmengen	7
$A^{-1} = \bar{A} = \{e^{-1} : e \in A\}$	Menge der Umkehrkanten von A	6
$\hat{A} = A + A^{-1}$	symmetrischer Abschluss der Kantenmenge A	6
$[A] = (V_A, A + A^{-1})$	ungerichteter Graph der Kanten aus A	6
$[A^{tr}] = (V_A, A^{tr} + (A^{tr})^{-1})$	ungerichteter Graph der Kanten aus A^{tr} , H-Comparabilitygraph für $A \in \mathcal{S}$	13, 55
$A^r \subseteq A$	transitive Reduktion von A	13
$A^{tr} \supseteq A$	transitiver Abschluss von A	13
$A \preceq B, A \prec B$	Plan A reduziert Plan B (streng)	66, 66
$A \sim B$	Pläne A und B sind ähnlich, $A \preceq B$ und $B \preceq A$	58, 66
$C = [c_{ij}]$	Matrix der Bearbeitungszeitpunkte von $ij \in SIJ$	50
C_n	Kreis der Länge n	7
$E \subseteq V \times V$	Kantenmenge	5
E_{V^*}	durch $V^* \subseteq V$ induzierte Kantenmenge	6
E^c	Menge der Komplementkanten zu E	7
$E_{diag}(S)$	Menge der Schrägkanten in $[S^{tr}]$	56
$E_{reg}(S) = E_{SIJ}$	Menge der regulären Kanten in $[S^{tr}]$	55
$e = ab \in E$	gerichtete Kante	5
$\hat{e} = \widehat{ab}$	ungerichtete Kante	5
$e^{-1} = (ab)^{-1} = ba$	Umkehrkante von e	5 5
$F(P_i)$	trennende Kantenmenge (isoliert P_i)	79
$F(P_1, \dots, P_r)$	trennende Kantenmenge zwischen P_1, \dots, P_r	80
$F^*(P_i) \subseteq F(P_i)$	Einschränkung von $F(P_i)$ auf angrenzende Kanten	79
$\mathcal{F}_G = \{\hat{I}_1, \dots, \hat{I}_k\}$	Menge der Farbklassen von G	11

$G = (V, E)$	Graph mit Knotenmenge V und Kantenmenge E	5
$G - \hat{e} = (V, E - \hat{e})$	Teilgraph von G , Entfernen von $\hat{e} \in E$	7
$G - \hat{A} = (V, E - \hat{A})$	Teilgraph von G , Entfernen von $\hat{A} \subseteq E$	7
$G^c = (V, E^c)$	Komplementgraph zu $G = (V, E)$	7
$G(S) = (SIJ, S)$	Plangraph zu $S \in \mathcal{S}_{SIJ}$	53
$G_{\mathcal{F}} = (V_{\mathcal{F}}, E_{\mathcal{F}})$	Faktorgraph zu $[S^{tr}]$	78
$G_{\Gamma} = (E, E_{\Gamma})$	Γ -Graph zu $G = (V, E)$	12
$G_{\Gamma_S} = (E_{reg}(S), \Gamma_S)$	Γ_S -Graph zu $[S^{tr}]$	105
$G_{JO} = (SIJ, JO)$	gerichteter Graph der organisat. Reihenfolgen	54
$G_{\mathcal{K}} = (V_{\mathcal{K}}, E_{\mathcal{K}})$	Konsequenzgraph zu $[S^{tr}]$	121
$G_{\mathcal{K}_{\hat{e}}} = (V_{\mathcal{K}}, E_{\mathcal{K}_{\hat{e}}})$	Teilgraph aller Kanten der Farbe $\hat{e} \in E_{diag}(S)$	121
$G_{MO} = (SIJ, MO)$	gerichteter Graph der technologischen Reihenfolgen	54
$G_{MOJO} = (SIJ, MO + JO)$	Plangraph für $S = MO + JO$	54
$G_{\mathcal{R}_M} = (V_M^* + V_M^{**}, E_{\mathcal{R}_M})$	Reduktionsgraph bzgl. der Entfernungsmenge M	124
$G_{\mathcal{R}_H}^* = (V_{\emptyset}^*, E_{\mathcal{R}_H}^*)$	Reduktionsgraph ^(*) bzgl. der Schrägkantenmenge H	149
$G_{SIJ} = (SIJ, E_{SIJ})$	ungerichteter H-Graph zu SIJ , $E_{SIJ} = E_{reg}$	53
$G_{E^*} = (V_{E^*}, E^*)$	durch $E^* \subseteq E$ induzierter Teilgraph von $G = (V, E)$	6
$G_{V^*} = (V^*, E_{V^*})$	durch $V^* \subseteq V$ induzierter Teilgraph von $G = (V, E)$	6
$I(e) \in \mathcal{I}_G$	Implikationsklasse, die $e \in E$ enthält	10
$\mathcal{I}_G = \{I_1, \dots, I_k, I_1^{-1}, \dots, I_k^{-1}\}$	Menge der Implikationsklassen	11
$I = \{1, \dots, n\}$	(Index-) Menge der Aufträge	46
$\hat{I} \in \mathcal{F}_G$	Farbklasse	10
\dot{I}	\dot{I} -Komponente	29
$J = \{1, \dots, m\}$	(Index-) Menge der Maschinen	46
$JO = [jo_{ij}]$	Matrix der organisat. Reihenfolgen von $ij \in SIJ$	50
K_n	vollständiger Graph auf n Knoten	7
$K_{n,m}$	vollständig bipartiter Graph	7
$K_n \times K_m$	Hamminggraph (H-Graph)	52
$LR = [lr_{ij}]$	Matrix der Ränge von $ij \in SIJ$	50
$M \subseteq E_{diag}(S)$	Entfernungsmenge	119
$MO = [mo_{ij}]$	Matrix der technolog. Reihenfolgen von $ij \in SIJ$	50
$\mathcal{O}(f(n))$	Landausches Symbol	35
$O_{ij} \in SIJ$	Operation zu Index $ij \subseteq I \times J$	46
$P = [p_{ij}]$	Matrix der Bearbeitungszeiten von $ij \in SIJ$	50
$\mathcal{P}_S = \{P_1, \dots, P_k\}$	Menge der Planimplikationsklassen von $S \in \mathcal{S}_{SIJ}$	76
$\mathcal{P}_{[S^{tr}]} = \mathcal{P}_S + \mathcal{P}_{S^{-1}}$	Menge der Planimplikationsklassen in $[S^{tr}]$	76
\dot{P}_i	\dot{I} -Komponente zu Planimplikationsklasse P_i	79
$r(v)$	Rang von Knoten v	9
$S_M \in \mathcal{S}_{SIJ}$	aus $S \in \mathcal{S}_{SIJ}$ abgeleiteter Plan, $M \subseteq 2^{ \mathcal{P}_S }$	82
\mathcal{S}_{SIJ}	Menge aller Pläne über SIJ	48
$SIJ \subseteq I \times J$	Menge der Operationen	46
$t(G)$	Anzahl der transitiven Orientierungen von G	18
$\mathcal{T}_G = \{T_1, \dots, T_t\}$	Menge der transitiven Orientierungen von G	8
V	Knotenmenge	5
V_{E^*}	durch $E^* \subseteq E$ induzierte Knotenmenge	6
$V^* = V_M^* \subseteq V_{\mathcal{R}_M}$	Knoten, die Planimplikationsklassen repräsentieren	130
$V^{**} = V_M^{**} \subseteq V_{\mathcal{R}_M}$	Knoten, die Schrägkanten repräsentieren	130

Index

Symbols

α -Relation	39
Γ	
\sim -Graph	12
\sim -Graph, konsistent	12
\sim -Kette	10
\sim -Kette, kanonisch	14
\sim -Nachbarschaft	9
\sim -Relation	9
\sim -Weg	<i>siehe</i> Γ -Kette
\sim -verbunden	10
Γ^*	
\sim -Relation	10
$\dot{\Gamma}$	
\sim -Kette	29, 76
\sim -Komponente	29, 76
\sim -Nachbarschaft	29
\sim -Relation	28, 76
\sim -verbunden	29, 76
$\dot{\Gamma}^*$	
\sim -Relation	29
Γ_S	
\sim -Graph	105
\sim -Kette	75
\sim -Nachbarschaft	75
\sim -Relation	75
\sim -verbunden	75
Γ_S^*	
\sim -Relation	75
3-farbiges Dreieck	19

A

Ablaufgraph	<i>siehe</i> Plangraph
Abschluss	
symmetrischer \sim	6
transitiver \sim	13
adjazent	6
ähnlich	
Plan	58, 66
aktiv	49
semi \sim	49
Algorithmus	
branch	169
deterministisch	37

dontBranch	169
effizient	36
findNextEdge	133, 168
findPossibleEdges	169
findX	168
Naives Umkehren	110
nichtdeterministisch	37
randomisiert	37
Reducing1	138, 170
Reducing2	143, 171
updateRedGraph	130, 167
allgemein reduziert	74
alternativlos	134
äquivalent	
komplexitätstheoretisch	39
Auftrag	46
Aufwand	36
azyklisch	8

B

baumartig	<i>siehe</i> tree-like
Bearbeitungszeit	46
berühren	<i>siehe</i> inzidieren
Berge-Graph	33
bipartiter Graph	7
Bogen	<i>siehe</i> Kante
paralleler \sim	<i>siehe</i> Multikante
BPP	43
Branch and Bound	48

C

chromatische Zahl	33
CLIQUE	37
Clique	7, 37
Cliquenzahl	33
gewichtet	34
co-NP	41
COMP-DEL	96
COMP-ED	40
COMP-GS	97
COMPARABILITY	36
Comparability	
\sim graph	8
H- \sim graph	55, 61

completion time..... *siehe* Fertigstellungszeit

D

Dekomposition
 modulare $\tilde{}$ 31
 deterministisch 37
 diagonal edge *siehe* Schrägkante
 Dimension 25
 direkte Konsequenz 121
 direkter Konflikt 125
 disjunktiv
 $\tilde{}$ e Kante 52
 $\tilde{}$ er Graph 52
 $\tilde{}$ es Graphenmodell 51
 Dispatching Rules 48
 dominiert *siehe* reduziert, 74
 Dreieck
 3-farbig 19
 Dreieckslemma 14
 Dreieckssehne 24
 dynamische Optimierung 48

E

echte Implikationsklasse 10
 Ecke 57
 Eckkante 57
 effizient 66
 effizienter Algorithmus 36
 Einbettung
 $H\text{-}\tilde{}$ 115
 reguläre $H\text{-}\tilde{}$ 115
 eindeutig orientierbar 8
 einfacher Graph *siehe* schlichter Graph
 Einmaschinenproblem 45
 einseitig
 $\tilde{}$ er Fehler 37
 $\tilde{}$ er Irrtum 37
 $\tilde{}$ es Versagen 37
 Entfernen einer Kante 7, 29
 Entfernungsmenge 119
 unzulässig 119
 zulässig 119
 zulässig erweiterbar 119
 Entscheidungsproblem 38
 EP 42
 erweiterte Planimplikationsklasse 86
 Erweiterungskante
 alternativlos 134

F

f(n)-leicht 36

f(n)-schwierig 36
 Färbung
 $\tilde{}$ szahl 33
 Intervall- $\tilde{}$ 34
 Intervall- $\tilde{}$ szahl 34
 Knoten- $\tilde{}$ 33
 FACT 41
 Faktorgraph 78
 konsistent 78
 Farbklasse 10
 Fehler
 einseitig 37
 zweiseitig 43
 Fertigstellungszeit 47
 Flow-Shop Problem 46

G

G-Zerlegung 23
 genetische Algorithmen 48
 gerichtet
 $\tilde{}$ e Kante 5
 $\tilde{}$ er Graph 5
 $\tilde{}$ er Weg 6
 Gesamtbearbeitungszeit 47
 getarnte stabile Schrägkante 135
 gewichtete Cliquenzahl 34
 GI 41
 Grad *siehe* Valenz
 Graph 5
 $\Gamma\text{-}\tilde{}$ 12
 $\Gamma_S\text{-}\tilde{}$ 105
 Berge- $\tilde{}$ 33
 bipartit 7
 Comparability- $\tilde{}$ 8
 disjunktiver $\tilde{}$ 52
 eindeutig orientierbar 8
 einfach *siehe* schlicht
 Faktor- $\tilde{}$ 78
 gerichtet 5
 $H\text{-}\tilde{}$ 52
 $H\text{-Comparability}\text{-}\tilde{}$ 55, 61
 Hamming- $\tilde{}$ 52
 induzierter Teil- $\tilde{}$ 6
 Komplement- $\tilde{}$ 7
 Konsequenz- $\tilde{}$ 121
 Multi- $\tilde{}$ 5
 orientiert 8
 perfekt 33
 Plan- $\tilde{}$ 53
 PO- $\tilde{}$ 25
 prim 8
 Reduktions- $\tilde{}$ 124, 149
 schlicht 5
 Semi- $\tilde{}$ 6

superperfekt 34
 Teil~ 6
 ungerichtet 5
 UPO-~ 32
 vollständig 7
 vollständig bipartit 7
 zusammenhängend 5
 GRAPH SANDWICH 97
 Graphenmodell
 disjunktives ~ 51

H

H-
 Comparabilitygraph 55, 61
 Einbettung 115
 Graph 52
 reguläre ~Einbettung 115
 Hülle
 symmetrische ~ 6
 transitive ~ 13
 Halbordnung *siehe* partielle Ordnung
 Hamming-Graph 52
 Heuristik
 iterativ 48
 konstruktiv 48
 horizontale Kante 53

I

immer-transitive Kante 28
 Implikationsklasse 10
 echt 10
 erweiterte Plan~ 86
 Plan~ 75, 77
 Umkehr~ 11
 Umkehren von ~n 17, 70
 verschieden 11
 induziert
 ~e Kantenmenge 6
 ~e Knotenmenge 6
 ~er Teilgraph 6
 Instanz 36
 Intervall-Färbung 34
 ~szahl 34
 inzidieren 6
 IRED 67 f., 82 f., 120
 irreduzibel 66
 Irrtum
 einseitig 37
 isolierende Kantenmenge 79
 isolierter Knoten 6
 iterative Heuristik 48

J

Job 46
 job order... *siehe* organisatorische Reihenfolge
 Job-Shop Problem 46
 Jobcharakteristika 47

K

kanonisch
 ~e Γ -Kette 14
 ~e Partition 31
 Kante 5
 ~nlänge eines Weges 6
 adjazent 6
 alternativlose Erweiterungs~ 134
 disjunktiv 52
 Eck~ 57
 Entfernen einer ~ 7, 29
 gerichtet 5
 getarnte stabile Schräg~ 135
 horizontal 53
 immer-transitiv 28
 induzierte ~menge 6
 isolierende ~menge 79
 konjunktiv 52
 Multi~ 5
 neue Schräg~ 83
 Nicht~ 6, 12
 nichttrivial-stabile Schräg~ 91
 regulär 55
 Schräg~ 56
 stabile Schräg~ 81
 transitiv 8
 trennende ~menge 79
 trivial-stabile Schräg~ 81, 86
 Umkehr~ 5
 ungerichtet 5
 vertikal 53
 Kette
 Γ -~ 10
 Γ_S -~ 75
 $\bar{\Gamma}$ -~ 29, 76
 Klasse *siehe* Komplexitätsklasse
 erweiterte Planimplikations~ 86
 Farb~ 10
 Implikations~ 10
 Planimplikations~ 75, 77
 Knoten 5
 ~färbung 33
 ~länge eines Weges 6
 adjazent 6
 induzierte ~menge 6
 isoliert 6
 stabile ~menge 7

Kombination	
$\tilde{R}e$	17, 82
zulässig	16
Komplement	
\tilde{a} res Problem	41
\tilde{g} raph	7
Komplexität	
\tilde{s} theoretisch äquivalent	39
Zeit \tilde{t}	36
Komplexitätsklasse	
BPP	43
$BPP(\varepsilon(n))$	42
co-NP	41
EP	42
NP	38
NPC, NP-complete	39
NPI, NP-incomplete	40
P	38
PP	43
RP	43
$RP(\varepsilon(n))$	43
RP^*	43
ZPP	43
$ZPP(\varepsilon(n))$	42
ZPP^*	43
Komponente	5
$\tilde{\Gamma}$	29, 76
Konflikt	125
direkter \tilde{t}	125
l-ter Ordnung	125
konfliktfrei	125
Konsequenz	121
\tilde{g} raph	121
direkte \tilde{t}	121
konsistent	
$\tilde{\Gamma}$ -Graph	12
Faktorgraph	78, 121
Reduktionsgraph	125
konstruktive Heuristik	48
Kreis	7
nicht-einfach	24
kreisfrei	<i>siehe</i> azyklisch
kunjunktive Kante	52

L

Löschen einer Kante	<i>siehe</i> Entfernen einer Kante
Landausches Symbol	35
lateinisches Rechteck	50
leicht	
$f(n)$ - \tilde{t}	36
lineare Ordnung	25
Lokale Suche	48
LP	41

M

M-Zerlegung	18
machine order <i>siehe</i> technologische Reihenfolge	
makespan	<i>siehe</i> Gesamtbearbeitungszeit
Maschine	46
Maschinenumgebung	47
Matrix	
C	50
JO	50
LR	50
MO	50
P	50
Rang \tilde{t}	50
maximal	
\tilde{t} es Multiplex	18
\tilde{t} es Simplex	18
minimal	
\tilde{t} er Teilweg	57
\tilde{t} er Weg	57
Mixed-Shop Problem	47
Modul	30
überlappende \tilde{e}	30
\tilde{t} are Dekomposition	31
quasimaximal	30
stark	30
Multikante	5
Multiplex	18
maximal	18
r - \tilde{t}	18

N

Nachbarschaft	48
$\tilde{\Gamma}$	9
$\tilde{\Gamma}_S$	75
$\tilde{\Gamma}$	29
neue Schrägkante	83
nicht-einfacher Kreis	24
nichtdeterministisch	37
Nichtkante	6, 12
nichttrivial-stabile Schrägkante	91
normaler Plan	109
NP	38
\tilde{t} -complete	39
\tilde{t} -hard	<i>siehe</i> NP-schwer
\tilde{t} -incomplete	40
\tilde{t} -schwer	39, 63
\tilde{t} -unvollständig	<i>siehe</i> NP-incomplete
\tilde{t} -vollständig	<i>siehe</i> NP-complete
NPC	39
NPI	40

O

O-Notation	35
------------------	----

Open-Shop Problem 46, 61, 63
 Operation 46
 Operationenmenge 46
 tree-like 98
 unvollständig 46
 vollständig 46
 Optimierungsproblem 38
 Ordnung
 lineare \sim 25
 partielle \sim 25
 organisatorische Reihenfolge 46
 orientierter Graph 8
 Orientierung 7
 azyklisch 8
 Plan \sim 58
 potentiell transitiv 16
 pseudotransitiv 23
 transitiv 8
 Umkehr \sim 8

P

P 38
 paralleler Bogen *siehe* Multikante
 Parallelmaschinenproblem 45
 partielle Ordnung 25
 Partition
 kanonische \sim 31
 perfekt 33
 Pfad *siehe* Weg
 Plan 48, 50
 ähnlich 58, 66
 \sim graph 53
 \sim orientierung 58
 allgemein reduziert 74
 dominiert 74
 effizient 66
 irreduzibel 66
 normal 109
 reduzierend 66
 reduziert 66
 reduziert streng 66
 resultierend 82
 Umkehr \sim 58
 zyklisch 73
 Planimplikationsklasse 75, 77
 erweiterte \sim 86
 Umkehren von \sim n 82
 PO-Graph *siehe* Comparabilitygraph, 25
 poset 25
 potentiell transitive Orientierung 16
 PP 43
 primer Comparabilitygraph 8
 PRIMES 41
 Problem 36

\sim instanz 36
 CLIQUE 37
 COMP-DEL 96
 COMP-ED 40
 COMP-GS 97
 COMPARABILITY 36
 Einmaschinen \sim 45
 Entscheidungs \sim 38
 FACT 41
 Flow-Shop 46
 GI 41
 IRRD 67 f., 82 f., 120
 Job-Shop 46
 komplementäres \sim 41
 LP 41
 Mixed-Shop 47
 Open-Shop 46, 61, 63
 Optimierungs \sim 38
 Parallelmaschinen \sim 45
 PRIMES 41
 REDUCIBILITY 67
 REDUCING 67
 SAT 39
 SEQUENCE-SUBGRAPH (SEQ-SG) 100
 Shop \sim 45
 SUBGRAPH 40
 TRANSITIVE ORIENTATION 36
 Processing Time 46
 pseudotransitiv 23

Q

quasimaximales Modul 30
 quasitransitiv *siehe* pseudotransitiv
 Quelle 9

R

r-Multiplex 18
 r-Simplex 17
 randomisiert 37
 Rang
 \sim eines Knotens 9
 \sim eines Multiplexes 18
 \sim eines Simplexes 17
 \sim matrix 50
 REDUCIBILITY 67
 REDUCING 67
 Reduktion
 polynomielle \sim 39
 transitive \sim 13
 Reduktionsgraph 124, 149
 konfliktfrei 125
 konsistent 125

- reduzierend 66
 reduziert 66
 allgemein 74
 streng 66
 regulär 47
 ~e H-Einbettung 115
 ~e Kante 55
 ~er Weg 55
 Reihenfolge
 organisatorische ~ 46
 technologische ~ 46
 Reihungsregeln 48
 Rekombination 17, 82
 zulässig 17, 83
 Relation
 Γ -~ 9
 Γ^* -~ 10
 Γ_S -~ 75
 Γ_S^* -~ 75
 α -~ 39
 $\hat{\Gamma}$ -~ 28, 76
 $\hat{\Gamma}^*$ -~ 29
 \preceq (Reduktion) 66
 resultierender Plan 82
 RP 43
- S**
- SAT 39
 Schedule 49
 aktiv 49
 semiaktiv 49
 Schedulingproblem
 Einmaschinen-~ 45
 Flow-Shop 46
 Job-Shop 46
 Mixed-Shop 47
 Open-Shop 46
 Parallelmaschinen-~ 45
 Shop ~ 45
 schlichter Graph 5
 Schlinge 5
 Schrägkante 56
 getarnt stabil 135
 neu 83
 nichttrivial-stabil 91
 stabil 81
 trivial-stabil 81, 86
 schwierig
 $f(n)$ -~ 36
 semiaktiv 49
 Semigraph 6
 Senke 9
 sequence *siehe* Plan
 SEQUENCE-SUBGRAPH (SEQ-SG) 100
- Sequenzbedingung 50
 Shop Problem 45
 Simplex 17
 maximal 18
 r -~ 17
 Simulated Annealing 48
 Spalte 53
 stabile Knotenmenge 7
 stabile Schrägkante 81
 getarnt 135
 nichttrivial-~ 91
 trivial-~ 81, 86
 starkes Modul 30
 Struktursatz
 Gallai 31
 Golumbic 18
 SUBGRAPH 40
 superperfekt 34, 62
 symmetrisch
 ~e Hülle 6
 ~er Abschluss 6
- T**
- technologische Reihenfolge 46
 Teilgraph 6
 induzierter ~ 6
 Teilordnung *siehe* partielle Ordnung
 Teilweg 56
 minimal 57
 Threshold Accepting 48
 totale Ordnung *siehe* lineare Ordnung
 transitiv
 ~e Hülle 13
 ~e Kante 8
 ~e Orientierung 8
 ~e Orientierung, verschieden 8
 ~e Reduktion 13
 ~er Abschluss 13
 potentiell ~e Orientierung 16
 pseudo~ 23
 TRANSITIVE ORIENTATION 36
 tree-like 98
 trennende Kantenmenge 79
 trivial-stabile Schrägkante 81, 86
 nicht~ 91
- U**
- überlappende Module 30
 Umkehr
 ~implikationsklasse 11
 ~kante 5
 ~orientierung 8

$\tilde{\text{plan}}$	58
Umkehren	
von Implikationsklassen	17, 70
von Planimplikationsklassen	82
unabhängige Knotenmenge	<i>siehe</i> stabile Knotenmenge
ungerichtet	
$\tilde{\text{e}}$ Kante	5
$\tilde{\text{er}}$ Graph	5
universell-optimale Menge	66, 74
Untergraph	<i>siehe</i> Teilgraph
unvergleichbar	25
unvollständig	
Operationenmenge	46
unzulässig	
Entfernungsmenge	119
UPO-Graph	32

V

Valenz	6
verbunden	
Γ - $\tilde{\text{}}$	10
Γ_S - $\tilde{\text{}}$	75
$\hat{\Gamma}$ - $\tilde{\text{}}$	29, 76
vergleichbar	25
Vergleichbarkeitsgraph	<i>siehe</i> Comparabilitygraph
Versagen	
einseitig	37
verschieden	
Implikationsklassen	11
transitive Orientierungen	8
verschmelzen	
von Planimplikationsklassen	121
vertikale Kante	53
vollständig	
$\tilde{\text{}}$ bipartiter Graph	7
$\tilde{\text{er}}$ Graph	7
Operationenmenge	46

W

Weg	6, 55
Γ - $\tilde{\text{}}$	10
gerichtet	6
Kantenlänge	6
Knotenlänge	6
minimal	57
minimaler Teil $\tilde{\text{}}$	57
regulär	55
Teil $\tilde{\text{}}$	56

Z

Zahl	
chromatische $\tilde{\text{}}$	33
Cliques $\tilde{\text{}}$	33
gewichtete Cliques $\tilde{\text{}}$	34
Intervall-Färbungs $\tilde{\text{}}$	34
Zeile	53
Zeit	
$\tilde{\text{}}$ komplexität	36
Bearbeitungs $\tilde{\text{}}$	46
Fertigstellungs $\tilde{\text{}}$	47
Gesamtbearbeitungs $\tilde{\text{}}$	47
Zerlegung	
$\tilde{\text{}}$ in Farbklassen	11
$\tilde{\text{}}$ in Implikationsklassen	11
$\tilde{\text{}}$ in Planimplikationsklassen	76
G- $\tilde{\text{}}$	23
M- $\tilde{\text{}}$	18
Zielfunktion	47
regulär	47
zulässig	
$\tilde{\text{}}$ erweiterbare Entfernungsmenge	119
$\tilde{\text{e}}$ Kombination	16
$\tilde{\text{e}}$ Rekombination	17, 83
Entfernungsmenge	119
un $\tilde{\text{}}$ Entfernungsmenge	119
zusammenhängender Graph	5
Zusammenhangskomponente	5
zweiseitig	
$\tilde{\text{er}}$ Fehler	43
zyklisch	73

Literaturverzeichnis

- [1] Agrawal, M.; Kayal, N. and Saxena, N. (2004). PRIMES is in P. *Ann. of Math. (2004)* **160**:2 781-793.
- [2] Aigner, Martin and Prins, Geert (1971). Uniquely partially orderable graphs. *J. London Math. Soc. (2) (1971)* **3** 260-266.
- [3] Andresen, Michael (2005). *Über transitive Orientierungen auf $G - \hat{e}$* . Diplomarbeit. Otto-von-Guericke-Universität Magdeburg.
- [4] Andresen, Michael (2006). *On transitive orientations of $G - \hat{e}$* . Preprint. Otto-von-Guericke-Universität Magdeburg. Accepted for publication in *Discussiones Mathematicae Graph Theory*.
- [5] Andresen, Michael; Bräsel, Heidemarie; Mörig, Marc; Tusch, Jan; Werner, Frank and Willenius, Per (2008). Simulated Annealing and Genetic Algorithms for Minimizing Mean Flow Time in an Open Shop. *Math. and Comp. Modelling (2008)* **48** 1279 - 1293.
- [6] Andresen, Michael; Bräsel, Heidemarie; Plauschin, Mathias and Werner, Frank (2008). Using Simulated Annealing for Open Shop Scheduling with Sum Criteria. In Tan, Cher Ming, editor. *Simulated Annealing*. 49-76, I-Tech Education and Publishing, 2008.
- [7] Bang-Jensen, Jørgen and Gutin, Gregory (2001). *Digraphs: Theory, Algorithms and Applications*. Springer, 2001.
- [8] Berge, Claude (1961). *Färbung von Graphen, deren sämtliche bzw. deren ungerade Kreise starr sind*. *Wiss. Z. Martin-Luther-Univ. Halle-Wittenberg Math.-Natur. Reihe (1961)* 114-115.
- [9] Błażewicz, Jacek; Ecker, Klaus H.; Pesch, Erwin; Schmidt, Günther and Weglarz, Jan (2007). *Handbook of Scheduling. From Theory to Applications*. Springer, 2007.
- [10] Bräsel, Heidemarie (1990). *Lateinische Rechtecke und Maschinenbelegung*. Habilitationsschrift. Technische Universität Otto-von-Guericke Magdeburg.
- [11] Bräsel, Heidemarie (2006). Matrices in Shop Scheduling Problems. In Morlock, M.; Schwindt, C.; Trautmann, N. and Zimmermann, J., editors. *Perspectives on Operations Research - Essays in Honor of Klaus Neumann*, 17-43, Gabler Edition Wissenschaft, Deutscher Universitätsverlag, 2006.
- [12] Bräsel, Heidemarie; Dhamala, Tankanath and Matzke, Bettina (2007). *On Superclasses and Subclasses of H-Comparability Graphs with Applications in Scheduling Theory and Discrete Optimization*. Preprint. Otto-von-Guericke-Universität Magdeburg.
- [13] Bräsel, Heidemarie; Harborth, Martin; Tautenhahn, Thomas and Willenius, Per (1999). On the set of solutions of an open shop problem. *Ann. Oper. Res. (1999)* **92** 241-263.
- [14] Bräsel, Heidemarie; Harborth, Martin; Tautenhahn, Thomas and Willenius, Per (1999). On the hardness of the classical job shop problem. *Ann. Oper. Res. (1999)* **92** 265-279.
- [15] Bräsel, Heidemarie; Herms, Andre; Mörig, Marc; Tautenhahn, Thomas; Tusch, Jan and Werner, Frank (2008). Heuristic Constructive Algorithms for Open Shop Scheduling to Minimize Mean Flow Time. *Europ. J. Oper. Res. (2008)* **189** 856 - 870.

- [16] Bräsel, Heidemarie and Kleinau, Martina (1996). New steps in the amazing world of sequences and schedules. *Math. Comput. Modelling (1996)* **43** 195-214.
- [17] Brucker, Peter (1995). *Scheduling Algorithms*. Springer, 1995.
- [18] Brucker, Peter (2007). *Scheduling Algorithms*. 5th ed., Springer, 2007.
- [19] Brucker, Peter and Knust, Siegfried (2002). *Complexity Results of Scheduling Problems*. Universität Osnabrück, Fachbereich Mathematik/Informatik. <http://www.mathematik.uni-osnabrueck.de/research/OR/class>.
- [20] Chudnovsky, Maria; Cornuéjols, Gérard; Liu, Xinming; Seymour, Paul and Vukovic', Kristina (2005). Recognizing Berge graphs. *Comb. (2005)* **25:2** 143-186.
- [21] Chudnovsky, Maria; Robertson, Neil; Seymour, Paul and Thomas, Robin (2006). The strong perfect graph theorem. *Ann. Math. (2006)* **164** 51-229.
- [22] Cook, Stephen (1971). The complexity of theorem proving procedures. *Proceedings of the Third Annual ACM Symposium on Theory of Computing (1971)* 151-158.
- [23] Cournier, A. and Habib, M. (1994). A new linear algorithm for modular decomposition. In Tison, S., editor, *Trees in Algebra and Programming, CAAP '94, 19th International Colloquium*, vol 787 of *Lect. Notes in Comp. Sci.* 68-82, Springer, 1994.
- [24] Dillworth, R. P. (1950). A decomposition theorem for partially ordered sets. *Ann. Math. (1950)* **51** 161-166.
- [25] Dushnik, B. and Miller, E. W. (1941). Partially Ordered Sets. *Amer. J. Math. (1941)* **63** 600-610.
- [26] Fulkerson, D. R. (1971). Blocking and Anti-Blocking Pairs of Polyhedra. *Math. Program. (1971)* **1** 168-194.
- [27] Gallai, Tibor (1967). Transitiv orientierbare Graphen. *Acta Math. Acad. Sci. Hungar. (1967)* **18** 25-66.
- [28] M. R. Garey and D. S. Johnson (1979). *Computers and intractability: A guide to NP-completeness*. Freeman, 1979.
- [29] Ghoulâ-Houri, Alain (1962). Caractérisation des graphes non orientés dont on peut orienter les arrêtes de manière à obtenir le graphe d'une relation d'ordre. *C.R. Acad. Sci. Paris (1962)* **254** 1370-1371.
- [30] Gilmore, Paul C. and Hoffman, Alan J. (1964). A characterization of comparability graphs and of interval graphs. *Canad. J. Math. (1964)* **16** 539-548.
- [31] Golumbic, Martin Cha (1977a). Comparability graphs and a new matroid. *J. Combin. Theory B (1977)* **22** 68-90.
- [32] Golumbic, Martin Cha (1977b). The complexity of comparability graph recognition and coloring. *Computing (1977)* **18** 199-208.
- [33] Golumbic, Martin Cha (1980). *Algorithmic graph theory and perfect graphs*. Academic Press, 1980.
- [34] Golumbic, Martin Cha (2004). *Algorithmic graph theory and perfect graphs*. 2nd ed. *Ann. of Disc. Math.* **57**, Elsevier, 2004.
- [35] Golumbic, M. C.; Kaplan, H. and Shamir, R. (1995). Graph sandwich problems. *J. Alg. (1995)* **19** 449-473.
- [36] Gonzalez, T. and Sahni, S. (1976). Open shop scheduling to minimize finish time. *J. Assoc. Comp. Mach. (1976)* **23** (4) 665-679.

- [37] Graham, R. L.; Lawler, E. L.; Lenstra, J. K. and Rinnoy Kan, A. H. G. (1979). Optimization and approximation in deterministic sequencing and scheduling: A survey. *Ann. of Disc. Math. (2007)* **5** 287-326.
- [38] Hougardy, Stefan (2006). Classes of Perfect Graphs. *Disc. Math. (2006)* **306** (19-20) 2529-2571.
- [39] *Information System on Graph Class Inclusions v2.0*. Universität Rostock, Institut für Informatik, Lehrstuhl Theoretische Informatik. <http://www.teo.informatik.uni-rostock.de/isgci/classes.cgi>.
- [40] Leonid Khachiyan (1979). A polynomial algorithm in linear programming. *Soviet Math. Doklady (1979)* **20** 191-194 (Russian).
- [41] Kleinau, Martina (1992). *Zur Struktur von Shop Scheduling Problemen: Anzahlprobleme, Reduzierbarkeit und Komplexität*. Dissertationsschrift. Otto-von-Guericke-Universität Magdeburg.
- [42] Ladner, R. E. (1975). On the structure of polynomial time reducibility. *J. of the ACM (1975)* **22** 155-171.
- [43] Lageweg, B. J.; Lawler, E. L.; Lenstra, J. K. and Rinnooy Kan, A. H. G. (1981). *Computeraided complexity classification of deterministic scheduling problems*. Report BM 138, Centre for Mathematics and Computer Science, Amsterdam.
- [44] Lawler, E. L.; Lenstra, J. K. and Rinnoy Kan, A. H. G. (1981). Minimizing maximum lateness in a two-machine open shop. *Math. Oper. Res. (1981)* **6** (1) 153-158.
- [45] Levin, Leonid (1973). Universal search problems. *Problems of Information Transmission (1973)* **9** (3) 265-266 (Russian).
- [46] *LiSA - A Library of Scheduling Algorithms*. Otto-von-Guericke-Universität Magdeburg, Fakultät für Mathematik, Institut für Algebra und Geometrie. <http://lisa.math.uni-magdeburg.de/>
- [47] Lovász, L. (1972). Normal Hypergraphs and the Perfect Graph Conjecture. *Disc. Math. (1972)* **2** 253-267.
- [48] McConnell, Ross M. and Montgolfier, Fabien de (2005). Linear-time modular decomposition of directed graphs. *Disc. App. Math. (2005)* **145** 198-209.
- [49] McConnell, Ross M. and Spinrad, Jeremy P. (1994). Linear-time modular decomposition and efficient transitive orientation of comparability graphs. *Proceedings of the 5th Annual ACM-SIAM Symposium on Discrete Algorithms (1994)* **5** 536-545.
- [50] McConnell, Ross M. and Spinrad, Jeremy P. (1999). Modular decomposition and transitive orientation. *Disc. Math. (1999)* **201** 189-241.
- [51] McConnell, Ross M. and Spinrad, Jeremy P. (2000). Ordered Vertex Partitioning. *Disc. Math. and Theor. and Comp. Sci. (2000)* **4** 45-60.
- [52] Miller, G. L. (1976). Riemann's hypothesis and tests for primality. *J. Comp. and System Sci. (1976)* **13** 300-317.
- [53] Möhring, Rolf H. (1984). Almost all comparability graphs are UPO. *Disc. Math. (1984)* **50** 63-70.
- [54] Mörig, Marc A. (2006). *Modulare Dekomposition durch geordnete Partitionierung der Knotenmenge: Grundlagen und Implementierung*. Diplomarbeit. Otto-von-Guericke-Universität Magdeburg.
- [55] Natanzon, Assaf; Shamir, Ron and Sharan, Roded (2001). Complexity classification of some edge modification problems. *Disc. App. Math. (2001)* **113** 109-128.
- [56] Pinedo, Michael (1995). *Scheduling: Theory, Algorithms, and Systems*. Prentice-Hall, 1995.
- [57] Prins, C. (1994). An overview of scheduling problems arising in satellite communications. *J. Oper. Res. Soc. (1994)* **45** 611-623.

- [58] Reischuk, K. Rüdiger (1999). *Komplexitätstheorie, Band I: Grundlagen*. Teubner, 1999.
- [59] Roy, B. and Sussmann, B. (1964). Les problèmes d'ordonnancement avec contraintes disjonctives. SEMA Note D. S. 9, 1964.
- [60] Simon, Klaus (1992). *Effiziente Algorithmen für perfekte Graphen*. Teubner, 1992.
- [61] Solovay, Robert M. and Strassen, Volker (1977). A fast Monte-Carlo test for primality. *SIAM J. Comput.* (1977) **6** (1) 84-85.
- [62] Trotter, William T. (1992). *Combinatorics and partially ordered sets: Dimension theory*. Johns Hopkins University Press, 1992.
- [63] Tautenhahn, Thomas (1996). *Irreducible sequences - an approach to intervall edge coloring trees*. Preprint. OR83, University of Southampton.
- [64] Tautenhahn, Thomas (2000). *Irreducible sequences on tree-like operation sets*. Preprint. Otto-von-Guericke-Universität Magdeburg.
- [65] Wegener, Ingo (2003). *Komplexitätstheorie*. Springer, 2003.
- [66] Willenius, Per (2000). *Irreduzibilitätstheorie bei Shop-Scheduling-Problemen*. Dissertationsschrift. Shaker Verlag, 2000.
- [67] Williamson, D. P.; Hall, L. A.; Hoogeveen, J. A.; Hurkens, C. A. J.; Lenstra, J. K.; Sevastianov, S. V. and Shmoys, D. B. (1997). Short Shop Schedules. *Oper. Res.* (1997) **45** 288-294.
- [68] Yannakakis, M. (1981). Edge deletion problems. *SIAM J. Comput.* (1981) **10** (2) 297-309.