

2 Einstiege: Ungleichungssysteme und diskrete Strukturen

In diesem und dem folgenden Kapitel werden einige mögliche Einstiege in das Thema der Optimierung aufgezeigt, ausgehend von der linearen Algebra, der Kombinatorik und der Algorithmik.¹ Als elementare Motivation der Polyedertheorie und des Simplex-Algorithmus ist etwa Sektion 2.1 über die Fourier-Motzkin²-Elimination zur Lösung von linearen Ungleichungssystemen gedacht, mit der auch eine erste Methode zur Lösung von LP-Aufgaben zur Verfügung gestellt wird. Sektionen 2.2 bietet eine Einführung in die Graphentheorie, während sich Sektion 2.3 mit Matroiden befasst und einen Einstieg in verschiedene Probleme der Diskreten Optimierung ermöglicht.

Literatur: [3], [16], [30], [39], [57], [59], [60], [67]

2.1 Von Gleichungs- zu Ungleichungssystemen: Fourier-Motzkin-Elimination

Lineare Gleichungssysteme sind sowohl theoretisch als auch praktisch gut verstanden. Es ist daher naheliegend zu versuchen, die Ergebnisse auch auf Systeme der Form $Ax \leq b$ bzw. $(Ax = b \wedge x \geq 0)$ zu übertragen, um zu entscheiden, wann sie lösbar, d.h. wann die zugrunde liegenden Polyeder nicht leer sind. Insbesondere steht die Frage im Vordergrund, ob es ein der Gauß³-Elimination analoges Verfahren zur Lösung von linearen Ungleichungssystemen gibt. Die Lösbarkeit von Gleichungssystemen kann bekanntlich mit Hilfe der um die Spalte b erweiterten Koeffizientenmatrix (A, b) charakterisiert werden.

2.1.1 Wiederholung. *Das Gleichungssystem $Ax = b$ ist genau dann lösbar, wenn*

$$\text{rang}(A) = \text{rang}(A, b)$$

gilt.

Liegt ein Ungleichungssystem der Form $Ax \leq b$ vor und gilt $\text{rang}(A) = \text{rang}(A, b)$, so ist demnach $Ax \leq b$ (sogar als Gleichungssystem) lösbar. Die Umkehrung gilt allerdings nicht, wie folgende Beispiele zeigen:

2.1.2 Beispiel. (a) *Seien*

$$A := \begin{pmatrix} 1 \\ -1 \end{pmatrix} \quad \wedge \quad b := \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

¹ Abhängig von Vorkenntnissen, Schwerpunktsetzungen und Zielrichtungen können diese Kapitel (zumindest in Teilen) eine notwendige und nützliche Einführung sein oder aber auch weitgehend übersprungen werden. Der gewählte modulare Aufbau soll Einstiege in verschiedene Teile der Optimierung ermöglichen, die in den späteren Kapiteln vertieft werden.

² Joseph Fourier, 1768 – 1830; Theodore Motzkin, 1908 – 1970.

³ Johann Carl Friedrich Gauß, 1777 – 1855.

Dann gilt

$$\text{rang}(A) = 1 \neq 2 = \text{rang} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} = \text{rang}(A, b).$$

Trotzdem ist das Ungleichungssystem $Ax \leq b$ lösbar; es gilt

$$\{x \in \mathbb{R}^1 : Ax \leq b\} = [-1, 1].$$

(b) Für

$$A := \begin{pmatrix} 1 \\ -1 \end{pmatrix} \quad \wedge \quad b := \begin{pmatrix} -1 \\ -1 \end{pmatrix}$$

gilt ebenfalls

$$\text{rang}(A) = 1 \neq 2 = \text{rang} \begin{pmatrix} 1 & -1 \\ -1 & -1 \end{pmatrix} = \text{rang}(A, b),$$

aber diesmal ist $Ax \leq b$ unlösbar.

Bereits dieses einfache Beispiel zeigt, dass eine Charakterisierung der Lösbarkeit von Ungleichungssystemen aus der linearen Algebra herausführt. Anders als für lineare Gleichungssysteme ist auch die allgemeine Lösung eines inhomogenen linearen Ungleichungssystems im Allgemeinen nicht die Summe einer partikulären Lösung und der allgemeinen Lösung des zugehörigen homogenen Ungleichungssystems.

Das nächste Beispiel zeigt Ähnliches auch für Systeme der Form $(Ax = b \wedge x \geq 0)$. Natürlich kann ein solches System höchstens dann lösbar sein, wenn auch $Ax = b$ lösbar ist. Die Bedingung $\text{rang}(A) = \text{rang}(A, b)$ ist somit notwendig; hinreichend ist sie im Allgemeinen aber nicht.

2.1.3 Beispiel. (a) Seien

$$A := (1, 1) \quad \wedge \quad b := 1.$$

Dann gilt

$$\{x \in \mathbb{R}^2 : Ax = b \wedge x \geq 0\} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} + [0, 1] \begin{pmatrix} -1 \\ 1 \end{pmatrix}.$$

(b) Für

$$A := (1, 1) \quad \wedge \quad b := -1$$

ist das Ungleichungssystem $Ax = b \wedge x \geq 0$ hingegen unlösbar; vgl. Abbildung 2.1.

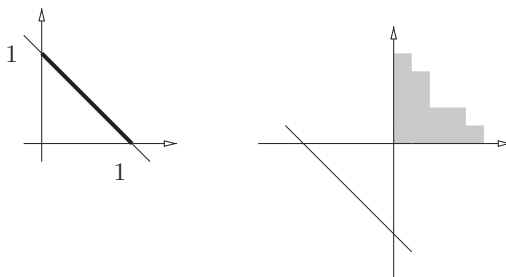
Das folgende Beispiel zeigt, dass – anders als bei linearen Gleichungssystemen – die Addition von Ungleichungen, eine wesentliche Operation für die Gauß-Elimination, die Lösungsgesamtheit eines Systems von linearen Ungleichungen verändern kann.

2.1.4 Beispiel. Gegeben sei das Ungleichungssystem

$$\xi_1 + \xi_2 \leq \alpha \quad \wedge \quad \xi_1 \geq 0 \quad \wedge \quad \xi_2 \geq 0,$$

wobei der Parameter α später festgesetzt wird. Es ist offenbar äquivalent zu

$$\begin{pmatrix} 1 & 1 \\ -1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} \leq \begin{pmatrix} \alpha \\ 0 \\ 0 \end{pmatrix}.$$

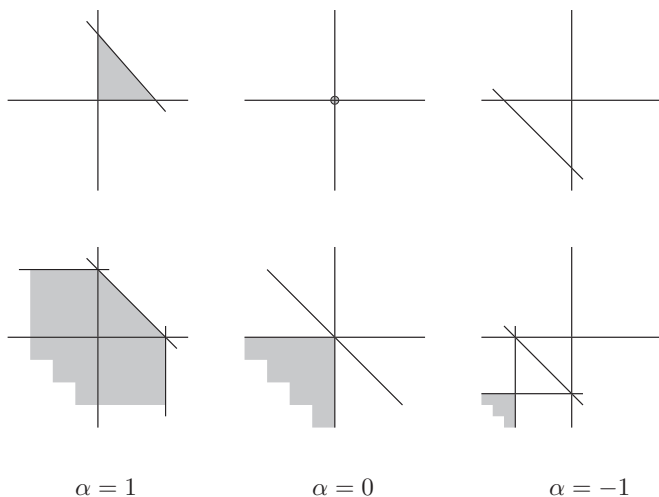


2.1 Abbildung. Lösungsmengen der Ungleichungssysteme in Beispiel 2.1.3 (a) (links) und (b) (rechts).

Addition der ersten Ungleichung zur zweiten und dritten liefert das neue System

$$\begin{pmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} \leq \begin{pmatrix} \alpha \\ \alpha \\ \alpha \end{pmatrix}.$$

Natürlich ist jede Lösung des ursprünglichen Problems auch Lösung des durch ‘elementare Zeilenoperationen’ entstandenen neuen Systems. Die Umkehrung gilt aber im Allgemeinen nicht. Die Skizzen in Abbildung 2.2 veranschaulichen jeweils den zulässigen Bereich des ursprünglichen sowie des neuen Systems für die Parameterwerte $\alpha = 1, 0, -1$.



2.2 Abbildung. Lösungsmengen des ursprünglichen und (darunter) des neuen Systems für $\alpha = 1, 0, -1$ aus Beispiel 2.1.4 (jeweils von links nach rechts).

Offenbar liegt die Vergrößerung des zulässigen Bereichs in Beispiel 2.1.4 daran, dass zwar die Addition von zwei Ungleichungen $a_1^T x \leq \beta_1$ und $a_2^T x \leq \beta_2$ wieder eine gültige Ungleichung liefert, nämlich $(a_1 + a_2)^T x \leq \beta_1 + \beta_2$, dass man von dieser und einer der ursprünglichen – etwa $a_1^T x \leq \beta_1$ – aber nicht mehr auf die andere, $a_2^T x \leq \beta_2$, zurück-schließen kann: Ungleichungen lassen sich nicht einfach subtrahieren wie Gleichungen.

Um wenigstens die Grundidee der Gauß-Elimination zu übertragen, kann man versuchen, sie in zwei ‘Phasen’ aufzuteilen: eine, die Multiplikationen mit negativen Zahlen sowie die Addition von Ungleichungen vermeidet, und eine zweite, in der diese Operationen zugelassen sind. Der erste Teil wird auf Ungleichungssysteme übertragbar sein, der zweite nicht.⁴

Wir betrachten also zunächst einen Schritt der Gauß-Elimination für das lineare Gleichungssystem

$$Ax = b$$

mit

$$m, n \in \mathbb{N} \setminus \{1\} \quad \wedge \quad A := (\alpha_{i,j})_{\substack{i \in [m] \\ j \in [n]}} \quad \wedge \quad b := (\beta_1, \dots, \beta_m)^T,$$

und wir nehmen o.B.d.A. an, dass die Gleichungen so nummeriert sind, dass

$$\alpha_{i,n} \begin{cases} < 0 & \text{für } i = 1, \dots, k \\ = 0 & \text{für } i = k+1, \dots, l \\ > 0 & \text{für } i = l+1, \dots, m \end{cases}$$

mit $k \in \mathbb{N}_0$ und $l - k \in \mathbb{N}_0$ gilt. Man beachte, dass $(k = 0 \wedge l = m)$ bedeutet, dass die letzte Spalte von A der Nullvektor ist; die letzte Variable ist dann überflüssig. Natürlich orientieren wir uns im Folgenden an den nichttrivialen Fälle, in denen tatsächlich eine Elimination erforderlich ist.

Durch Division der i -ten Zeile durch $|\alpha_{i,n}|$ für $i = 1, \dots, k$ bzw. durch $\alpha_{i,n}$ für $i = l+1, \dots, m$ erhalten wir das äquivalente System

$$\begin{aligned} \sum_{j=1}^{n-1} \hat{\alpha}_{i,j} \xi_j - \xi_n &= \hat{\beta}_i & (i = 1, \dots, k) \\ \sum_{j=1}^{n-1} \hat{\alpha}_{i,j} \xi_j &= \hat{\beta}_i & (i = k+1, \dots, l) \\ \sum_{j=1}^{n-1} \hat{\alpha}_{i,j} \xi_j + \xi_n &= \hat{\beta}_i & (i = l+1, \dots, m), \end{aligned}$$

wobei

$$\hat{\alpha}_{i,j} := \begin{cases} \alpha_{i,j}/|\alpha_{i,n}| & \text{für } i = 1, \dots, k \\ \alpha_{i,j} & \text{für } i = k+1, \dots, l \\ \alpha_{i,j}/\alpha_{i,n} & \text{für } i = l+1, \dots, m \end{cases}$$

und $j = 1, \dots, n-1$ sowie

$$\hat{\beta}_i := \begin{cases} \beta_i/|\alpha_{i,n}| & \text{für } i = 1, \dots, k \\ \beta_i & \text{für } i = k+1, \dots, l \\ \beta_i/\alpha_{i,n} & \text{für } i = l+1, \dots, m \end{cases}$$

gilt. Auflösen nach ξ_n liefert somit

$$\begin{aligned} -\hat{\beta}_i + \sum_{j=1}^{n-1} \hat{\alpha}_{i,j} \xi_j &= \xi_n & (i = 1, \dots, k) \\ \xi_n &= \hat{\beta}_i - \sum_{j=1}^{n-1} \hat{\alpha}_{i,j} \xi_j & (i = l+1, \dots, m) \\ \sum_{j=1}^{n-1} \hat{\alpha}_{i,j} \xi_j &= \hat{\beta}_i & (i = k+1, \dots, l). \end{aligned}$$

⁴ Ob das ausreicht, einen Lösungsalgorithmus für lineare Ungleichungssysteme zu entwickeln, ist a priori nicht klar! Wir werden sehen ...

In diesem ersten Teil sind nur Operationen durchgeführt worden, die auch auf Ungleichungssysteme $Ax \leq b$ übertragbar sind. Das gilt auch für die ersten k Indizes, für die ξ_n jeweils einen negativen Koeffizienten hat; wir haben ja lediglich durch Beträge dividiert und die Variablen umgestellt.⁵

Im Gauß-Algorithmus eliminiert man nun die Variable ξ_n durch Gleichsetzung der auftretenden Werte. Sind $k \geq 1$ und $l \leq m - 1$, so können wir die Bedingungen für ξ_n durch die folgenden Gleichungen ersetzen:⁶

$$\sum_{j=1}^{n-1} \hat{\alpha}_{i,j} \xi_j - \hat{\beta}_i = \hat{\beta}_t - \sum_{j=1}^{n-1} \hat{\alpha}_{t,j} \xi_j \quad (i = 1, \dots, k; t = l + 1, \dots, m)$$

oder äquivalent

$$\sum_{j=1}^{n-1} (\hat{\alpha}_{i,j} + \hat{\alpha}_{t,j}) \xi_j = \hat{\beta}_t + \hat{\beta}_i \quad (i = 1, \dots, k; t = l + 1, \dots, m).$$

In dieser Form werden die $k + (m - l)$ Bedingungen für ξ_n durch $k \cdot (m - l)$ Gleichungen ersetzt, von denen natürlich höchstens $k + (m - l) - 1$ für das reduzierte System relevant sind. Zur Reduktion des neuen Systems benutzt man insbesondere die Symmetrie der Äquivalenzrelation ‘=’. Die Gauß-Elimination organisiert den Gesamtschritt eleganter als Subtraktion geeigneter Vielfacher der ‘Pivotzeile’ des Systems $Ax = b$ von den übrigen Gleichungen und gelangt so zu folgendem linearen Gleichungssystem mit $m - 1$ Ungleichungen in $n - 1$ Variablen:

$$\begin{aligned} \sum_{j=1}^{n-1} (\hat{\alpha}_{i,j} - \hat{\alpha}_{1,j}) \xi_j &= \hat{\beta}_i - \hat{\beta}_1 & (i = 2, \dots, k) \\ \sum_{j=1}^{n-1} \hat{\alpha}_{i,j} \xi_j &= \hat{\beta}_i & (i = k + 1, \dots, l) \\ \sum_{j=1}^{n-1} (\hat{\alpha}_{i,j} + \hat{\alpha}_{1,j}) \xi_j &= \hat{\beta}_i + \hat{\beta}_1 & (i = l + 1, \dots, m). \end{aligned}$$

Das ursprüngliche System $Ax = b$ ist genau dann lösbar, wenn das neue System lösbar ist, und der Eliminationsprozess kann sukzessive fortgesetzt werden. Geometrisch wurde die Lösungsmenge des ursprünglichen Systems orthogonal auf den \mathbb{R}^{n-1} der ersten Koordinaten projiziert.

Liegt nun statt des Gleichungssystems $Ax = b$ das Ungleichungssystem

$$Ax \leq b$$

vor, so ist dieses – analog zum Fall linearer Gleichungen – äquivalent zu dem System

$$\begin{aligned} -\hat{\beta}_i + \sum_{j=1}^{n-1} \hat{\alpha}_{i,j} \xi_j &\leq \xi_n & (i = 1, \dots, k) \\ \xi_n &\leq \hat{\beta}_i - \sum_{j=1}^{n-1} \hat{\alpha}_{i,j} \xi_j & (i = l + 1, \dots, m) \\ \sum_{j=1}^{n-1} \hat{\alpha}_{i,j} \xi_j &\leq \hat{\beta}_i & (i = k + 1, \dots, l). \end{aligned}$$

⁵ Das ‘merkwürdige’ Layout soll bereits ‘vorbereiten’, dass bei späteren Ungleichungen die ξ_n betreffenden Zeilen die Form ‘ $\dots \leq \xi_n$ ’ oder ‘ $\xi_n \leq \dots$ ’ haben werden.

⁶ Aus der Perspektive des bekannten Gauß-Algorithmus mag dieser vollständige Vergleich der Bedingungen für $i = 1, \dots, k$ mit denen für $i = l + 1, \dots, m$ skurril anmuten. Diese Interpretation erweist sich aber für Ungleichungen als zielführend.

Ein ‘paarweiser Vergleich’ der ersten k mit den letzten $m - l$ Ungleichungen führt auch hier zu einer Elimination von ξ_n .

2.1.5 Lemma. *Das System $Ax \leq b$ ist genau dann lösbar, wenn*

$$\begin{aligned} \sum_{j=1}^{n-1} (\hat{\alpha}_{i,j} + \hat{\alpha}_{t,j}) \xi_j &\leq \hat{\beta}_t + \hat{\beta}_i & (i = 1, \dots, k; t = l + 1, \dots, m) \\ \sum_{j=1}^{n-1} \hat{\alpha}_{i,j} \xi_j &\leq \hat{\beta}_i & (i = k + 1, \dots, l). \end{aligned}$$

lösbar ist. Ferner ist $\xi_1^, \dots, \xi_{n-1}^*$ genau dann eine Lösung des reduzierten Systems, wenn es ein ξ_n^* gibt, so dass $\xi_1^*, \dots, \xi_{n-1}^*, \xi_n^*$ eine Lösung des Ausgangssystems ist. Für $n \geq 2$ ist die Lösungsmenge des reduzierten Systems somit die orthogonale Projektion des ursprünglichen auf den \mathbb{R}^{n-1} der ersten $n - 1$ Koordinaten.*

Beweis: Ist ξ_1^*, \dots, ξ_n^* eine Lösung des Ausgangssystems, so gilt insbesondere

$$\begin{aligned} \sum_{j=1}^{n-1} \hat{\alpha}_{i,j} \xi_j^* - \hat{\beta}_i &\leq \hat{\beta}_t - \sum_{j=1}^{n-1} \hat{\alpha}_{t,j} \xi_j^* & (i = 1, \dots, k; t = l + 1, \dots, m) \\ \sum_{j=1}^{n-1} \hat{\alpha}_{i,j} \xi_j^* &\leq \hat{\beta}_i & (i = k + 1, \dots, l), \end{aligned}$$

d.h. $\xi_1^*, \dots, \xi_{n-1}^*$ ist eine Lösung des reduzierten Systems. Ist umgekehrt $\xi_1^*, \dots, \xi_{n-1}^*$ eine Lösung des reduzierten Systems, so kann ξ_n^* so gewählt werden, dass gilt

$$\sum_{j=1}^{n-1} \hat{\alpha}_{i,j} \xi_j^* - \hat{\beta}_i \leq \xi_n^* \leq \hat{\beta}_t - \sum_{j=1}^{n-1} \hat{\alpha}_{t,j} \xi_j^* \quad (i = 1, \dots, k; t = l + 1, \dots, m);$$

ξ_1^*, \dots, ξ_n^* erfüllt dann alle Bedingungen des Ausgangssystems, und es folgt die Behauptung. \square

Es ist möglich, dass im Laufe der Elimination eine Variable in keiner Restriktion oder in Ungleichungen von nur einem der beiden Typen ‘ \leq ’ oder ‘ \geq ’ auftritt. Falls etwa ξ_n nur in ‘ \geq ’-Bedingungen vorkommt, so ist $l = m$, d.h. es treten keine Bedingungen des ersten Typs aus Lemma 2.1.5 auf. Tatsächlich stellt ja das ursprüngliche System nur eine einseitige Einschränkung für ξ_n dar; es reicht, ξ_n hinreichend groß zu wählen. Folgerichtig reduziert sich das System auf

$$\sum_{j=1}^{n-1} \hat{\alpha}_{i,j} \xi_j \leq \hat{\beta}_i \quad (i = k + 1, \dots, m).$$

Tritt ξ_n hingegen in beiden Ungleichungstypen auf, so führt jede zulässige Setzung für ξ_1, \dots, ξ_{n-1} zu einer unteren und einer oberen Intervallgrenze für ξ_n , insgesamt also auf ein kompaktes Intervall der dann noch zulässigen Werte für ξ_n .

Es kann im Laufe der Reduktion passieren, dass Ungleichungen auftreten, die gar keine Variable mehr enthalten. Enthält etwa das ursprüngliche Ungleichungssystem die Bedingung $\mu \leq \xi_n \leq \eta$, so wird diese bei der Elimination von ξ_n durch die variablenfreie Ungleichung $\mu \leq \eta$ ersetzt. Ist diese ungültig, so ist bereits bewiesen, dass das System $Ax \leq b$ keine Lösung besitzt.

Man beachte, dass in Lemma 2.1.5 alle gemäß dem vorliegenden System auftretenden Ungleichungskombinationen im reduzierten System noch explizit vorkommen, während in der Gauß-Elimination für lineare Gleichungssysteme eine (triviale) Reduktion redundanter Gleichungen erfolgt.

Durch den Übergang gemäß Lemma 2.1.5 wird die Variable ξ_n durch orthogonale Projektion eliminiert. Das neue System hat

$$k(m-l) + (l-k)$$

Ungleichungen in den Variablen ξ_1, \dots, ξ_{n-1} . Führt man nun sukzessive fort, so reduziert sich das ursprüngliche System $Ax \leq b$ schrittweise. Nach n Schritten liegt ein System ohne Variablen vor. Ist es leer, d.h. ohne jegliche Ungleichung, so ist die Lösungsmenge des System unbeschränkt, und man kann analog zur Gauß-Elimination sukzessive Werte für $\xi_1, \xi_2, \dots, \xi_n$ finden, so dass der zugehörige Vektor das Ungleichungssystem löst. Andernfalls entspricht es einer Ungleichung zwischen zwei reellen Zahlen. Ist diese inkorrekt, so ist $Ax \leq b$ unlösbar. Ist sie korrekt, so kann man wieder sukzessive zulässige Werte für $\xi_1, \xi_2, \dots, \xi_n$ finden.

2.1.6 Bezeichnung. Das beschriebene Verfahren heißt **Fourier-Motzkin-Elimination** zur Lösung von linearen Ungleichungssystemen.

2.1.7 Beispiel. Wir wenden das Fourier-Motzkin-Eliminationsverfahren auf Beispiel 2.1.4 an:

$$\left\{ \begin{array}{rcl} \xi_1 & + & \xi_2 \leq \alpha \\ \xi_1 & & \geq 0 \\ & & \xi_2 \geq 0 \end{array} \right\} \rightarrow \left\{ \begin{array}{rcl} & \xi_2 & \leq \alpha - \xi_1 \\ \xi_1 & & \geq 0 \\ & \xi_2 & \geq 0 \end{array} \right\} \rightarrow$$

$$\rightarrow \left\{ \begin{array}{rcl} 0 & \leq & \alpha - \xi_1 \\ \xi_1 & \geq & 0 \end{array} \right\} \rightarrow \left\{ \begin{array}{rcl} \xi_1 & \leq & \alpha \\ \xi_1 & \geq & 0 \end{array} \right\} \rightarrow \{0 \leq \alpha\}.$$

Der Übergang von der letzten, eine Variable enthaltenden Ungleichung $0 \leq \xi_1 \leq \alpha$ zur variablenfreien Ungleichung $0 \leq \alpha$ entspricht dabei im wesentlichen dem Test, ob das Intervall leer ist.

Für $\alpha = -1$ hat das Problem somit keine Lösung, für $\alpha = 0$ und $\alpha = 1$ ist es lösbar. Für $\alpha = 0$ ergibt sich zwingend $\xi_1 = 0$ und damit auch $\xi_2 = 0$. Für $\alpha = 1$ können wir für ξ_1 jeden Wert aus dem Intervall $[0, 1]$ wählen. Für $\xi_1 = \frac{1}{2}$ etwa liegt dann ξ_2 in dem Intervall $[0, \frac{1}{2}]$. Der Punkt $(\frac{1}{2}, \frac{1}{2})^T$ ist eine Lösung (von vielen).

Man beachte, dass in der Fourier-Motzkin-Elimination die Reihenfolge, in der die Variablen eliminiert werden, beliebig gewählt werden kann. Tatsächlich kann die Eliminationsreihenfolge erheblichen Einfluss auf die entstehenden Zwischensysteme haben; vgl. Übungsaufgabe 2.4.4.

Die folgende Prozedur fasst die Struktur der Fourier-Motzkin-Elimination noch einmal grob zusammen, ohne jedoch die technischen Details der Reduktion erneut auszuführen.⁷

⁷ Hier und im Folgenden werden wir algorithmische Methoden immer wieder strukturiert als Prozeduren zusammenfassen. Diese sind im Allgemeinen lediglich Beschreibungen auf einer 'Metaebene', verweisen auf vorangehende Konstruktionen, ignorieren Fragen nach Datenstrukturen etc., sind also nicht als Programmvorlagen zu verstehen.

2.1.8 Prozedur: *Fourier-Motzkin-Elimination*

INPUT: Matrix $A \in \mathbb{R}^{m \times n}$, Vektor $b \in \mathbb{R}^m$
 OUTPUT: Vektor $x^* \in \mathbb{R}^n$ mit $Ax^* \leq b$, falls ein solcher existiert;
 oder Meldung 'Ungleichungssystem unlösbar'
 BEGIN
 Eliminiere sukzessive alle Variablen gemäß Lemma 2.1.5
 IF das entstehende variablenfreie System ist unlösbar
 THEN Meldung 'Ungleichungssystem unlösbar'
 ELSE Bestimme sukzessive Komponenten ξ_1^*, \dots, ξ_n^*
 eines Lösungsvektors
 END

Nach Konstruktion ist die Fourier-Motzkin-Elimination endlich; genauer gilt der folgende Satz.

2.1.9 Satz. *Das Verfahren der Fourier-Motzkin-Elimination löst ein lineares Ungleichungssystem $Ax \leq b$ in n Variablen und mit m Ungleichungsrestriktionen in n Eliminationsschritten.*

Beweis: In jedem Schritt des Verfahrens wird eine Variable eliminiert. Somit erhält man nach n Eliminationsschritten eine variablenfreie Ungleichung. \square

Auf analoge Weise kann man auch strenge Ungleichungen behandeln. Es ist sogar möglich, mit Hilfe der Fourier-Motzkin-Elimination lineare Funktionale über den Lösungen von linearen Gleichungssystemen zu optimieren; d.h. man kann mit der Fourier-Motzkin-Elimination LP-Probleme lösen.

2.1.10 Bemerkung. *Seien $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$ und $P := \{x \in \mathbb{R}^n : Ax \leq b\} \neq \emptyset$. Ferner sei $\nu \in \mathbb{R} \cup \{\infty\}$. Durch Fourier-Motzkin-Elimination der Variablen ξ_1, \dots, ξ_n gehe das Ungleichungssystem*

$$\begin{array}{rcl} \xi_0 & - & c^T x \leq 0 \\ & & Ax \leq b \end{array}$$

über in

$$\xi_0 \leq \nu,$$

wobei $\nu = \infty$ bedeutet, dass ξ_0 keiner reellen oberen Schranke genügen muss. Dann gilt

$$\max\{c^T x : Ax \leq b\} = \nu.$$

Beweis: Für einen gegebenen Punkt $x^* \in P$ ist die Ungleichung $c^T x^* \geq \xi_0$ äquivalent zu $\xi_0 \in]-\infty, c^T x^*]$. Sei

$$\nu_0 := \max\{c^T x : Ax \leq b\}.$$

Nach Lemma 2.1.5 gibt es genau zu jedem $\xi_0 \in]-\infty, \nu_0]$ Werte ξ_1, \dots, ξ_n , so dass das System

$$\begin{array}{rcl} \xi_0 & - & c^T x \leq 0 \\ & & Ax \leq b \end{array}$$

lösbar ist. Hieraus folgt die Behauptung. \square

Natürlich erlaubt es das Eliminationsverfahren auch hier wieder, durch sukzessives 'Rückwärtslösen' Optimalpunkte zu bestimmen, falls solche existieren.

2.1.11 Beispiel. Gegeben sei die lineare Optimierungsaufgabe

$$\begin{array}{rcll} \max & \xi_1 + \xi_2 & & \\ 3\xi_1 & + \xi_3 & \leq & 3 \\ -3\xi_1 & + \xi_3 & \leq & 0 \\ & \xi_2 + \xi_3 & \leq & 1 \\ & 2\xi_2 + \xi_3 & \geq & 1 \\ & \xi_3 & \geq & 0. \end{array}$$

Der zulässige Bereich ist in Abbildung 2.3 (links) skizziert. Gemäß Bemerkung 2.1.10 fügen wir dem System die 'Zielfunktionsungleichung'

$$\xi_0 - \xi_1 - \xi_2 \leq 0$$

hinzu und führen anschließend die Elimination nach ξ_3 durch. Die ξ_3 enthaltenden Ungleichungen haben die Form

$$\begin{array}{rcll} & \xi_3 & \leq & 3 - 3\xi_1 \\ & \xi_3 & \leq & 3\xi_1 \\ & \xi_3 & \leq & 1 - \xi_2 \\ 1 - 2\xi_2 & \leq & \xi_3 & \\ 0 & \leq & \xi_3; & \end{array}$$

wir erhalten somit folgendes System in ξ_0, ξ_1 und ξ_2 .

$$\begin{array}{rcll} \xi_0 - \xi_1 - \xi_2 & \leq & 0 \\ 3\xi_1 - 2\xi_2 & \leq & 2 \\ 3\xi_1 + 2\xi_2 & \geq & 1 \\ & \xi_2 & \geq & 0 \\ & \xi_1 & \leq & 1 \\ & \xi_1 & \geq & 0 \\ & \xi_2 & \leq & 1. \end{array}$$

Im nächsten Schritt eliminieren wir nun ξ_2 . Zunächst schreiben wir den ξ_2 betreffenden Teil des Systems wieder übersichtlicher in der Form

$$\begin{array}{rcll} \xi_0 - \xi_1 & \leq & \xi_2 \\ -1 & + \frac{3}{2}\xi_1 & \leq & \xi_2 \\ \frac{1}{2} & - \frac{3}{2}\xi_1 & \leq & \xi_2 \\ 0 & & \leq & \xi_2 \\ & & \xi_2 & \leq 1; \end{array}$$

Nach Weglassen der trivialen Bedingung $0 \leq 1$ gelangen wir zu dem (schon in der für die Elimination nach ξ_1 geeigneten Form aufgeschriebenen) neuen System

$$\begin{array}{rcll} -1 + \xi_0 & \leq & \xi_1 \\ & \xi_1 & \leq & \frac{4}{3} \\ -\frac{1}{3} & \leq & \xi_1 \\ 0 & \leq & \xi_1 \\ & \xi_1 & \leq & 1. \end{array}$$

Durch Elimination nach ξ_1 (und Weglassen der redundanten Bedingungen ohne Variablen) erhalten wir

$$\xi_0 \leq 2.$$

Das Maximum der Zielfunktion ist somit 2. Durch Rückeinsetzen erhält man sukzessive die Koordinaten des (in diesem Beispiel eindeutig bestimmten) Optimalpunktes $\xi_1^* = 1$, $\xi_2^* = 1$ sowie $\xi_3^* = 0$.

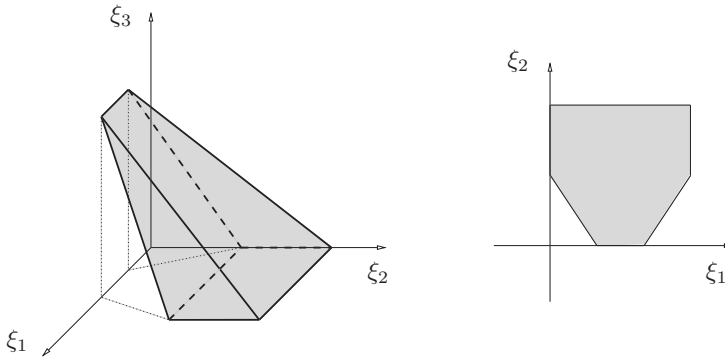
Nach Bemerkung 2.1.10 können wir lineare Optimierungsaufgaben mit Hilfe der Fourier-Motzkin Elimination lösen. Haben wir damit die lineare Optimierung bereits ‘im Griff’? Oder anders ausgedrückt: Ist der gefundene Algorithmus beweisbar effizient, oder wenigstens praktisch schnell genug, um für die ‘im täglichen Leben’ auftretenden linearen Programme in annehmbarer Zeit Optimalpunkte bestimmen zu können?

Wie der Gauß-Algorithmus zur Lösung von linearen Gleichungssystemen, benötigt auch die Fourier-Motzkin Elimination nur n Eliminationsschritte. Anders als bei linearen Gleichungssystemen kann sich im Verlauf der Lösung von linearen Ungleichungssystemen allerdings die Anzahl der Ungleichungen erhöhen.

2.1.12 Beispiel. Betrachten wir noch einmal den zulässigen Bereich aus Beispiel 2.1.11. Durch Elimination von ξ_3 erhält man (wie wir bereits wissen)

$$\begin{array}{rcll} 3\xi_1 & - & 2\xi_2 & \leq 2 \\ 3\xi_1 & + & 2\xi_2 & \geq 1 \\ & & \xi_2 & \geq 0 \\ \xi_1 & & & \leq 1 \\ \xi_1 & & & \geq 0 \\ & & \xi_2 & \leq 1. \end{array}$$

Das Ausgangssystem bestand aus fünf Bedingungen; das durch Elimination von ξ_3 entstandene enthält sechs Ungleichungen, von denen keine redundant ist. Das lässt sich einerseits leicht nachrechnen, andererseits ist es aus der folgenden graphischen Darstellung evident.



2.3 Abbildung. Die Lösungsmengen des Ausgangssystems (links) und des durch Elimination von ξ_3 entstandenen Systems (rechts) aus Beispiel 2.1.12.

Die Anzahl der irredundanten Ungleichungen, die in den Zwischenschritten auftreten, kann sich somit tatsächlich erhöhen.

Speziell für gerades m und $k = l = \frac{m}{2}$ beträgt die ‘unbereinigte’ Zahl $k(m-l) + l - k$ der Ungleichungen nach einem Schritt somit immerhin $\frac{m^2}{4}$. Würde sich dieses Wachstum über s Schritte fortsetzen, so lägen dann

$$\frac{m^{2^s}}{4^{2^s-1}} = 4 \left(\frac{m}{4} \right)^{2^s}$$

Ungleichungen vor.

Für $s \sim \log n$ (und hinreichend großes m und n) hätten wir somit bereits etwa $4\left(\frac{m}{4}\right)^n$ Ungleichungen. Wären solche Zahlen von Ungleichungen tatsächlich unvermeidlich – oder, schlimmer noch, typisch – so wäre das Verfahren zwar prinzipiell in n Schritten fertig, die Zwischensysteme würden aber so groß, dass man sie schon für moderate Größenordnungen von m und n nicht mehr explizit aufstellen könnte. Man vergegenwärtige sich: Für $n = 100$ und $m = 400$ ist $\left(\frac{m}{4}\right)^n = 10^{200}$.

Wenn tatsächlich solche großen Zwischensysteme während der Fourier-Motzkin Elimination auftreten könnten, wäre diese Methode wohl kaum praxistauglich. Aber vielleicht sind zwar nicht alle, jedoch wenigstens die meisten auftretenden Ungleichungen redundant.⁸ In Kapitel 4 untersuchen wir die geometrische Struktur der Lösungen linearer Ungleichungssysteme genauer. Hierdurch wird es möglich, die Frage nach der Effizienz der Fourier-Motzkin-Elimination abschließend zu beantworten; vgl. Korollar 4.3.35.

Unabhängig von der Frage nach seiner Effizienz liefert das Verfahren der Fourier-Motzkin Elimination in jedem Fall verschiedene strukturelle Ergebnisse; vgl. etwa Übungsaufgabe 2.4.6. Als Beispiel hierfür beenden wir diese Sektion mit einer geometrischen Folgerung. In jedem Fall bleibt die Anzahl der Ungleichungen bei der Fourier-Motzkin Elimination ja wenigstens endlich. Hieraus folgt eine Aussage für die Projektion von Polyedern auf beliebige lineare Unterräume.

2.1.13 Korollar. *Seien P ein Polyeder des \mathbb{R}^n , L ein linearer Teilraum des \mathbb{R}^n und Q entstehe durch orthogonale Projektion von P auf L . Dann ist Q ein Polyeder.*

Beweis: Sei $k := \dim(L)$. Die Aussage ist trivial für $k = 0$; sei also im Folgenden $k \in \mathbb{N}$. Wir nutzen aus, dass nach Lemma 2.1.5 das nach einem Schritt der Fourier-Motzkin Elimination gefundene Ungleichungssystem die orthogonale Projektion der Lösungsmenge auf den entsprechenden Koordinatenraum beschreibt, und dabei jeweils nur endlich viele Ungleichungen auftreten. Der allgemeine Fall ergibt sich durch Anwendung einer geeigneten orthogonalen Transformation. Wir führen sie im Folgenden explizit durch.

Seien $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ und $P := \{x \in \mathbb{R}^n : Ax \leq b\}$. Ferner seien v_1, \dots, v_n eine orthonormale Basis des \mathbb{R}^n , so dass v_1, \dots, v_k Basis von L ist, und $B := (v_1, \dots, v_n)^T$. Dann ist B eine orthogonale $(n \times n)$ -Matrix, und für $i \in [n]$ gilt $Bv_i = u_i$, wobei u_i gemäß Bezeichnung 1.3.7 wieder der i -te Standardeinheitsvektor ist. Als lineare Abbildung überführt B den Untervektorraum L in den Koordinatenunterraum $\mathbb{R}^k \times \{0\}^{n-k}$. Seien D die Diagonalmatrix mit Einträgen 1 in den ersten k und 0 in den restlichen Diagonalelementen. Dann gilt

$$Q = B^T D B P.$$

Wir führen nun auf $B P = \{y \in \mathbb{R}^n : A B^T y \leq b\}$ die ersten $n - k$ Schritte der Fourier-Motzkin Elimination durch, um aus $y =: (\eta_1, \dots, \eta_n)^T$ die Variablen $\eta_{k+1}, \dots, \eta_n$ zu eliminieren.

⁸ Aus diesem Blickwinkel kann man das Gauß-Eliminationsverfahren als Fourier-Motzkin-Elimination mit sukzessiver ‘Redundanzelimination’ interpretieren.

Mit geeigneten $r \in \mathbb{N}$, $T \in \mathbb{R}^{r \times k}$ und $t \in \mathbb{R}^r$ erhalten wir eine Darstellung

$$\hat{Q} = \{\hat{y} = (\eta_1, \dots, \eta_k)^T \in \mathbb{R}^k : T\hat{y} \leq t\}$$

der orthogonalen Projektion von BP auf den \mathbb{R}^k der ersten k Koordinaten.⁹ Für die Einbettung von \hat{Q} in $\mathbb{R}^k \times \{0\}^{n-k}$ gilt mit $D_1 := (u_1, \dots, u_k)^T$ und $D_2 := (u_{k+1}, \dots, u_n)^T$

$$\hat{Q} \times \{0\}^{n-k} = \{y \in \mathbb{R}^n : TD_1y \leq t \wedge D_2y \leq 0 \wedge -D_2y \leq 0\}.$$

Somit ist

$$Q = B^T(\hat{Q} \times \{0\}^{n-k}) = \{z \in \mathbb{R}^n : TD_1Bz \leq t \wedge D_2Bz \leq 0 \wedge -D_2Bz \leq 0\}.$$

Mit

$$A_Q := \begin{pmatrix} TD_1B \\ D_2B \\ -D_2B \end{pmatrix} \in \mathbb{R}^{(r+2(n-k)) \times n} \quad \wedge \quad b_Q := \begin{pmatrix} t \\ 0 \\ 0 \end{pmatrix} \in \mathbb{R}^{r+2(n-k)}$$

gilt demnach

$$Q = \{x \in \mathbb{R}^n : A_Qx \leq b_Q\};$$

Q ist also ein Polyeder. □

Wie wir in Beispiel 2.1.12 gesehen haben, kann die Beschreibung der Projektion eines Polyeders mehr lineare Ungleichungen benötigen als für die Darstellung des ursprünglichen Polyeders erforderlich sind; vgl. Abbildung 2.3. Die Frage der Effizienz der Fourier-Motzkin-Elimination, genauer nach der Anzahl der nicht redundanten Ungleichungen in den Zwischenschritten, ist somit äquivalent zu der Frage, wie ‘komplex’ Projektionen von Polyedern werden können.

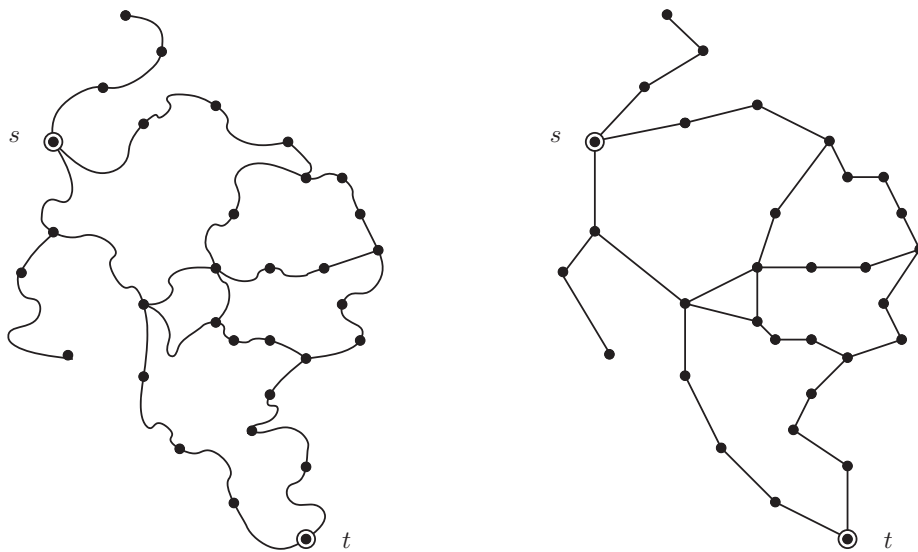
2.2 Graphen

Viele Probleme der kombinatorischen Optimierung und des Operations Research¹⁰ lassen sich am besten mit Hilfe einer einfachen kombinatorischen Struktur formulieren, den *Graphen*. In Sektion 1.2, Abbildung 1.5 zum Problem des Handlungsreisenden etwa lag eine (fiktive) Landkarte zugrunde, bei der die Städte durch Punkte und die möglichen Flugverbindungen zwischen ihnen durch Verbindungskanten dargestellt waren. Ähnliche Darstellungen treten auf, wenn nach kürzesten Reiserouten gefragt wird. Will man etwa auf einer möglichst kurzen Autobahnverbindung von einer Stadt A in eine Stadt B fahren, so kann man jede Autobahnabfahrt, jedes Autobahndreieck und jede Autobahnkreuzung (des relevanten Teils des Autobahnnetzes und geeigneter Zubringerstraßen) als Punkt darstellen und die dazwischen liegenden Autobahnabschnitte einzeichnen. Abbildung 2.4 zeigt ein (fiktives) Straßennetz in zwei unterschiedlichen symbolischen Darstellungen. Links sind die Verbindungen – wie das in ‘realen’ Straßenkarten geschieht¹¹ – grob den wirklichen Straßenverläufen nachempfunden, rechts einfach durch Verbindungsstrecken zwischen Punkten dargestellt.

⁹ Die Setzung $r \in \mathbb{N}$ ist keine Einschränkung, denn durch Verwendung der 0-Matrix und $t = 0$ kann auch der ganze \mathbb{R}^k so dargestellt werden.

¹⁰ Dieser Begriff umfasst allgemeiner die Entwicklung, Analyse und den Einsatz quantitativer Modelle und Algorithmen zur Unterstützung von Entscheidungen.

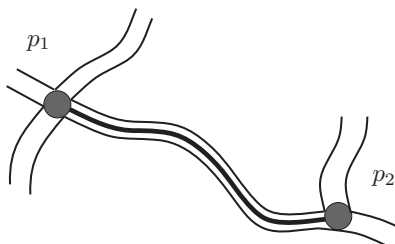
¹¹ Man sollte aber nicht vergessen, dass auch diese ‘realen’ Straßenkarten lediglich symbolische Darstellungen der Wirklichkeit sind.



2.4 Abbildung. Darstellungen eines (fiktiven) Straßennetzes. Die Verbindungskanten symbolisieren die Autobahnteilstücke zwischen Autobahndreiecken und -kreuzungen sowie Auf- bzw. Abfahrten; diese selbst werden als Knoten repräsentiert. Die mit s und t bezeichneten Knoten entsprechen den Städten A und B.

Da auf den Verbindungskanten keine Möglichkeiten zum Abbiegen bestehen, kommt es tatsächlich auf ihren genauen Verlauf gar nicht an. Man kann sie daher mathematisch mit Paaren von Punkten identifizieren (die allerdings häufig in suggestiven Darstellungen wieder symbolisch als Stecken zwischen den Punkten dargestellt werden). Bei Einbahnstraßen muss natürlich auch noch ihre Richtung erfasst werden. Daher wird ihnen ein geordnetes Paar von Punkten zugeordnet. Auf gleiche Weise kann man in Planungsproblemen auch zeitliche oder logistische Abhängigkeiten modellieren; siehe Beispiel 2.2.29.

2.2.1 Beispiel. Sind p_1 und p_2 zwei Straßenkreuzungen eines realen Straßennetzes, zwischen denen kein Abbiegen (oder eine andere, für die zugrunde liegende Aufgabenstellung relevante Aktion) möglich ist, so ordnen wir diesen formale Elemente v_1 und v_2 zu.



2.5 Abbildung. Straßenverlauf zwischen Kreuzungspunkten.

Die Tatsache, dass p_1 und p_2 durch eine Straße verbunden sind, kann dann mittels eines dritten formalen Elements $\{v_1, v_2\}$ modelliert werden, das lediglich symbolisiert,

dass v_1 und v_2 ‘kombinatorisch inzident’ sind. Vom konkreten Straßenverlauf wurde also vollständig abstrahiert. Der Abschnitt des Straßennetzes zwischen p_1 und p_2 wird somit durch die Komponenten v_1, v_2 und $\{v_1, v_2\}$ modelliert, bzw., wenn wir die Komponenten gleichen Typs in Mengen zusammenfassen durch

$$\left(\{v_1, v_2\}, \{\{v_1, v_2\}\} \right).$$

Kann die Straße nur in einer Richtung durchfahren werden, so erhalten wir (je nach der erlaubten Richtung) eine der folgenden beiden Strukturen:

$$\left(\{v_1, v_2\}, \{(v_1, v_2)\} \right) \quad \vee \quad \left(\{v_1, v_2\}, \{(v_2, v_1)\} \right).$$

Allgemeine Graphen: Wir führen nun Modelle ein, die es gestatten, solche Fragestellungen mathematisch zu erfassen. Dabei erlauben wir zunächst sehr allgemeine Strukturen.¹²

2.2.2 Bezeichnung. Sei X eine Menge. Dann bezeichnet 2^X die **Potenzmenge**, d.h. die Menge aller Teilmengen von X . Ferner wird für $k \in \mathbb{N}_0$ durch

$$\binom{X}{k} := \{S \in 2^X : |S| = k\}$$

die Menge der k -elementigen Teilmengen von X angegeben.

Man beachte, dass sich die gewählte Notation nicht nur an den Binomialkoeffizienten orientiert, sondern für endliche Mengen X auch die analoge Identität

$$\bigcup_{k=0}^{|X|} \binom{X}{k} = 2^X$$

zulässt.

Die folgende Definition 2.2.3 enthält gegenüber Beispiel 2.2.1 noch Verallgemeinerungen, die es erlauben, auch Ringstraßen und parallel verlaufende Straßen zwischen denselben Kreuzungspunkten zu modellieren. Um dieses fassen zu können verwenden wir eine Abbildung ν , die – anschaulich gesprochen – den Straßenabschnitten ihre Endpunkte zuordnet.

2.2.3 Definition. Seien V und E endliche Mengen mit $V \cap E = \emptyset$ und

$$\nu : E \rightarrow \binom{V}{1} \cup \binom{V}{2} \cup V^2$$

eine Abbildung.

- (a) Sei $G := (V, E, \nu)$. Dann heißt G **allgemeiner Graph**. Die Elemente von V heißen **Knoten** [engl.: nodes oder vertices], die Elemente von E **Kanten** [engl.: edges oder arcs] von G . Gilt $V = E = \emptyset$, so spricht man von dem **leeren Graphen**.¹³

¹² Für viele der späteren Untersuchungen werden wir diese dann ‘o.B.d.A.’ auf speziellere Klassen einschränken. Allerdings geht das nicht immer.

¹³ So richtig interessant ist der leere Graph nicht, aber mathematisch wegen der Abgeschlossenheit des Begriffs gegenüber Teilmengenbildung nützlich. Wir werden im Folgenden meistens darauf verzichten, in Aussagen den Fall des leeren Graphen zu thematisieren, falls dieser trivial ist, sondern ein ‘O.B.d.A. sei G nicht der leere Graph’ unausgesprochen hinzufügen.

- (b) Sei $e \in E$. Gilt $\nu(e) \in \binom{V}{1} \cup \binom{V}{2}$, so heißt die Kante e **ungerichtet**, ist $\nu(e) \in V^2$, so heißt e **gerichtet**. Gilt $\nu(e) = (v, w)$, so heißen v **Anfangs-** und w **Endknoten** [engl.: head bzw. tail] von e . Sind alle Kanten von G ungerichtet bzw. alle gerichtet, so heißt G **ungerichtet** bzw. **gerichtet**.
- (c) Sind $v \in V$, $e \in E$ und gilt $\nu(e) \in \{\{v\}, (v, v)\}$ oder existiert ein $w \in V$ mit $\nu(e) \in \{\{v, w\}, (v, w), (w, v)\}$, so heißen v und e **inzident**.
- (d) Sei $e \in E$. Gibt es ein $v \in V$ mit $\nu(e) \in \{\{v\}, (v, v)\}$, so heißt e **Schlinge** [engl.: loop]. G heißt **schlingenfrei**, wenn E keine Schlinge enthält.¹⁴ (Im Folgenden wird, wenn nichts anderes gesagt ist, eine Schlinge stets als ungerichtet aufgefasst und in der Form $\{v\}$ geschrieben.)
- (e) Allgemeine schlingenfreie ungerichtete Graphen werden auch **Multigraphen**, allgemeine schlingenfreie gerichtete Graphen **gerichtete Multigraphen** genannt.
- (f) Ist ν injektiv und gibt es keine $e_1, e_2 \in E$ und $v \in V$ mit $\nu(e_1) = \{v\}$ und $\nu(e_2) = (v, v)$, so heißt G **schlicht**.¹⁵ Ist G schlingenfrei, schlicht und gibt es keine Knoten $v, w \in V$ und Kanten e_1, e_2 mit $\nu(e_1) = \{v, w\}$ und $\nu(e_2) \in \{(v, w), (w, v)\}$, so heißt G **einfach**.

Die eingeführte Grundstruktur des allgemeinen Graphen ist allgemein genug, um die in den folgenden Kapiteln auftretenden Routenplanungsprobleme modellieren zu können. Tatsächlich kann man sich in den meisten Fällen sogar auf die später in Bezeichnung 2.2.28 eingeschränkten Klassen von ungerichteten bzw. gerichteten Graphen beschränken. Allerdings gibt es auch relevante Beispiele, in denen nicht nur allgemeine Graphen in natürlicher Weise auftreten, sondern diese auch nicht durch einfache Tricks auf die ‘reinen Fälle’ zurückgeführt werden können.¹⁶

Geometrisch stellt man die Knoten oft als Punkte eines Raumes, meistens der Ebene, dar. Die Kanten können dann als Verbindungen zwischen Knoten gezeichnet werden. Gerichtete Kanten können durch Pfeile dargestellt werden; die Pfeilspitze zeigt auf den Endknoten der Kante, d.h. auf das zweite Element des entsprechenden Knotenpaares.¹⁷

2.2.4 Beispiel. Seien

$$V := \{v_1, v_2, v_3, v_4\} \quad \wedge \quad E := \{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8\}$$

mit $V \cap E = \emptyset$ und $\nu : E \rightarrow \binom{V}{1} \cup \binom{V}{2} \cup V^2$ definiert durch

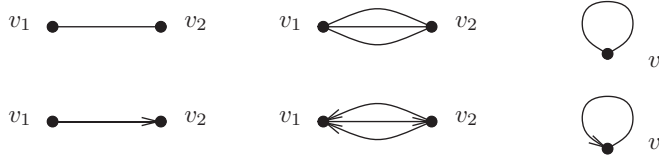
$$\begin{array}{llll} \nu(e_1) & := & \{v_1\} & \wedge \quad \nu(e_2) & := & (v_1, v_2) & \wedge \quad \nu(e_3) & := & (v_2, v_1) \\ \nu(e_4) & := & \{v_1, v_4\} & \wedge \quad \nu(e_5) & := & (v_1, v_3) & \wedge \quad \nu(e_6) & := & \{v_3, v_4\} \\ \nu(e_7) & := & \{v_3, v_4\} & \wedge \quad \nu(e_8) & := & (v_3, v_4) \end{array}$$

¹⁴ Man beachte, dass hier formal die Bilder $\{v\}$ und (v, v) unterschieden werden, um später eine einheitliche Notation für ungerichtete bzw. gerichtete Graphen zu haben. Die in (v, v) suggerierte Orientierung ist jedoch ohne Bedeutung, da für $v_1 := v_2 := v$ die geordneten Paare (v_1, v_2) und (v_2, v_1) nicht unterscheidbar sind.

¹⁵ Ein Graph ist also schlicht, wenn von jedem Kantentyp zwischen bis zu zwei Knoten höchstens ein Exemplar vorhanden ist. Dabei werden die ‘Schlingentypen’ $\{v\}$ und (v, v) identifiziert.

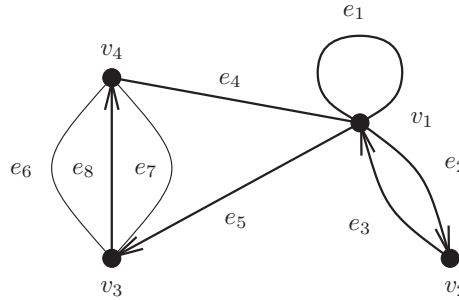
¹⁶ Hierzu gehört das später besprochene Chinese Postman Problem, das sowohl im ungerichteten als auch im gerichteten Fall wesentlich einfacher zu lösen ist als im ‘gemischten’ Fall.

¹⁷ Diese Darstellung ist sehr suggestiv, kann aber auch in die Irre führen. Wir hatten schon darauf hingewiesen, dass bei einer Schlinge die Pfeildarstellungen rechts- oder links herum im Graphen nicht unterscheidbar sind. Ferner brauchen geometrische Eigenschaften der Darstellung, wie etwa Schnittpunkte von Verbindungsstrecken, nichts mit der Struktur des Graphen zu tun zu haben.



2.6 Abbildung. Gebräuchliche ‘graphische Darstellungen’ von Kanten eines allgemeinen Graphen: ungerichtete Kanten (oben), gerichtete (unten); Mehrfachkanten zwischen Knoten (Mitte); Schlingen (rechts).

sowie $G := (V, E, \nu)$. In einer suggestiven geometrischen Darstellung könnte man den allgemeinen Graphen G etwa gemäß Abbildung 2.7 darstellen.



2.7 Abbildung. Geometrische Darstellungen des allgemeinen Graphen aus Beispiel 2.2.4.

Dabei lässt sich mit Hilfe gerichteter Kanten unter anderem modellieren, dass gewisse Strecken nur in einer Richtung durchfahren werden können oder, dass sich die Fahrzeit in der einen Richtung (z.B. wegen einer Baustelle) von der in der entgegengesetzten erheblich unterscheiden kann. Auch kann es durchaus bei der Modellierung von realen Straßennetzen auftreten, dass zwischen zwei Knoten mehrere Kanten gleichen Typs verlaufen. Dann ist der zugrunde liegende allgemeine Graph nicht schlicht. Fragt man aber nach einer kürzesten Route, so kann von allen Straßen gleicher Richtung zwischen einem Paar von Kreuzungen nur eine minimalen Gewichts zu einer solchen gehören; alle bis auf eine ‘kürzeste’ dieser Kanten können also gestrichen werden. Bei Fragen nach kürzesten Wegen ist es naturgemäß nicht erforderlich, ‘Rundstraßen’, die ohne Abzweigungsmöglichkeit wieder zum Ausgangspunkt zurückkehren, zu berücksichtigen – jedenfalls wenn die Längen der Kanten nichtnegativ sind. Für Straßenreinigungs- oder Zustelldienste müssen sie hingegen durchaus in die Modellierung einbezogen werden.

Fixiert man jeweils die Reihenfolge der Elemente von V und E in einem allgemeinen Graphen $G := (V, E, \nu)$, so kann ν mit Hilfe einer $(|V| \times |E|)$ -Matrix angegeben werden.

2.2.5 Bezeichnung. Seien X eine nichtleere endliche Menge und $k := |X|$. Jede Bijektion $\tau : [k] \rightarrow X$ heißt **Ordnung** oder **Reihenfolge** auf X . Ist eine Reihenfolge τ auf X gegeben, so setzt man oftmals

$$x_i := \tau(i) \quad (i \in [k])$$

und schreibt $X = \{x_1, \dots, x_k\}$. Umgekehrt wird durch diese Schreibweise für $i \in [k]$ gemäß $\tau(i) := x_i$ stets eine Reihenfolge τ auf X festgelegt.

2.2.6 Definition. Seien $G := (V, E, \nu)$ ein allgemeiner Graph (ohne Schlingen der Form (v, v)), $n := |V|$, $m := |E|$ und $n, m \geq 1$.

- (a) Seien τ_V eine Reihenfolge auf V und τ_E eine Reihenfolge auf E , d.h. $V = \{v_1, \dots, v_n\}$ und $E = \{e_1, \dots, e_m\}$. Ferner sei die $(n \times m)$ -Matrix $S_G := S_G(\tau_V, \tau_E) := (\sigma_{i,j})_{\substack{i \in [n] \\ j \in [m]}}$ definiert durch¹⁸

$$\sigma_{i,j} := \begin{cases} 1, & \text{falls } v_i \in \nu(e_j) \in \binom{V}{2} \text{ gilt;} \\ 2, & \text{falls } \nu(e_j) = \{v_i\} \text{ ist;} \\ -1, & \text{falls } e_j \text{ gerichtet und } v_i \text{ Anfangsknoten von } e_j \text{ ist;} \\ 1, & \text{falls } e_j \text{ gerichtet und } v_i \text{ Endknoten von } e_j \text{ ist;} \\ 0, & \text{sonst.} \end{cases}$$

Dann heit S_G **(Knoten-Kanten) Inzidenzmatrix** [engl.: (node-arc) incidence matrix] von G . Fr $j \in [m]$ heit der j -te Spaltenvektor von S_G **Inzidenzvektor** von e_j .

- (b) Seien $G_1 := (V_1, E_1, \nu_1)$ und $G_2 := (V_2, E_2, \nu_2)$ allgemeine Graphen mit $n := |V_1| = |V_2|$, $m := |E_1| = |E_2|$ und $n, m \geq 1$. Dann heien G_1 und G_2 **isomorph**, falls es fr $k = 1, 2$ Ordnungen τ_{V_k} auf V_k und τ_{E_k} auf E_k gibt, so dass

$$S_{G_1}(\tau_{V_1}, \tau_{E_1}) = S_{G_2}(\tau_{V_2}, \tau_{E_2})$$

glt.

Die Eintrge von S_G sind weitgehend kanonisch, bis auf vielleicht die Setzung 2 fr Schlingen e_j mit $\nu(e_j) = \{v_i\}$. Wie soll man eigentlich den Eintrag fr eine Schlinge sinnvoll definieren? Da wir ja auf die Unterscheidung zwischen $\{v\}$ und (v, v) verzichten, fhrt ein beliebiger von 0 verschiedener Eintrag zur Rekonstruierbarkeit von G aus S_G . Die Setzung 2 entspricht der Interpretation, dass bei ganz ‘lokaler Betrachtung’ von einem Knoten v bei einer Schlinge zwei Kanten sichtbar sind. Der formal klarere algebraische Grund wird sichtbar, wenn man den Rang von S_G bestimmt¹⁹; vgl. bungsaufgaben 2.4.16, 2.4.17.

2.2.7 Beispiel. Sei $G := (V, E, \nu)$ der Graph aus Beispiel 2.2.4 (mit den dort angegebenen Reihenfolgen auf V und E). Die zugehrige Inzidenzmatrix S_G ist gegeben durch

$$S_G = \begin{pmatrix} 2 & -1 & 1 & 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & -1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}.$$

Es gilt $S_G \in \{-1, 0, 1\}^{4 \times 8}$; die Zeilen von S_G entsprechen den Knoten v_1, v_2, v_3, v_4 , die Spalten den Kanten $e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8$ von G .

¹⁸ Die Vorzeichensetzung ist in der Literatur nicht einheitlich. Bisweilen werden den gerichteten Kanten auch die negativen Spaltenvektoren zugeordnet. Wir verwenden hier die ‘optimistische Kantenrichtung’ von Minus zu Plus.

¹⁹ Die gleiche algebraische Argumentation wrde bei Schlingen (v, v) einen Eintrag 0 rechtfertigen; allerdings wre dann das Vorhandensein der Schlinge in S_G nicht erkennbar. Es besteht also ein Konflikt zwischen den Aspekten ‘Algebra’ und ‘Datenstruktur’. Aus diesen Grund haben wir Schlingen (v, v) hier ausgeschlossen.

Inzidenzmatrizen sind eine für verschiedene Probleme adäquate ‘Datenstruktur’ für Graphen; vgl. Sektion 3.1.

Obwohl sich isomorphe allgemeine Graphen nur durch die Reihenfolgen auf ihren Knoten- und Kantenmengen unterscheiden, ist bislang nicht bekannt, ob Isomorphie effizient getestet werden kann; vgl. Forschungsproblem 3.4.27. Natürlich können höchstens solche Graphen isomorph sein, deren Knoten- und Kantenzahlen übereinstimmen. Mit Hilfe der folgenden Definition kann man weitere Invarianten herleiten.

2.2.8 Bezeichnung. Sei $G := (V, E, \nu)$ ein allgemeiner Graph.

(a) Für $U \subset V$ seien

$$\begin{aligned}\delta(G, U) &:= \left\{ e \in E : \exists (u \in U \wedge v \in V \setminus U) : \nu(e) = \{u, v\} \right\} \\ \delta_{\text{in}}(G, U) &:= \left\{ e \in E : \exists (u \in U \wedge v \in V \setminus U) : \nu(e) = (v, u) \right\} \\ \delta_{\text{aus}}(G, U) &:= \left\{ e \in E : \exists (u \in U \wedge v \in V \setminus U) : \nu(e) = (u, v) \right\} \\ N(G, U) &:= \left\{ v \in V \setminus U : \exists (u \in U \wedge e \in E) : \nu(e) = \{u, v\} \right\} \\ N_{\text{in}}(G, U) &:= \left\{ v \in V \setminus U : \exists (u \in U \wedge e \in E) : \nu(e) = (v, u) \right\} \\ N_{\text{aus}}(G, U) &:= \left\{ v \in V \setminus U : \exists (u \in U \wedge e \in E) : \nu(e) = (u, v) \right\}.\end{aligned}$$

$N(G, U) \cup N_{\text{in}}(G, U) \cup N_{\text{aus}}(G, U)$ wird bisweilen auch als **Nachbarschaft** von U in G bezeichnet. Ferner werden die Abkürzungen $\delta(G, v)$ bzw. $N(G, v)$ für $\delta(G, \{v\})$ bzw. $N(G, \{v\})$ (und analoge Abkürzungen für ihre gerichteten Varianten) verwendet.²⁰

(b) Die Abbildungen

$$\deg_G(v), \deg_{\text{in}G}, \deg_{\text{aus}G} : V \rightarrow \mathbb{N}_0$$

seien für $v \in V$ definiert durch

$$\begin{aligned}\deg_G(v) &:= \left| \delta(G, v) \right| + 2 \left| \{ e \in E : \nu(e) = \{v\} \} \right| \\ \deg_{\text{in}G}(v) &:= \left| \delta_{\text{in}}(G, v) \right| + \left| \{ e \in E : \nu(e) = \{v\} \} \right| \\ \deg_{\text{aus}G}(v) &:= \left| \delta_{\text{aus}}(G, v) \right| + \left| \{ e \in E : \nu(e) = \{v\} \} \right|.\end{aligned}$$

$\deg_G(v)$, $\deg_{\text{in}G}(v)$ bzw. $\deg_{\text{aus}G}(v)$ heißen **Grad** [engl.: degree], **Ingrad** bzw. **Ausgrad** [engl.: indegree bzw. outdegree] von v in G . Wenn der Bezug klar ist, wird meistens nur $\deg(v)$, $\deg_{\text{in}}(v)$ und $\deg_{\text{aus}}(v)$ geschrieben. Bisweilen spricht man statt von (In- bzw. Aus-) Grad auch von (In- bzw. Aus-) **Valenz**.

(c) Sei v ein Knoten. Gilt $\deg_G(v) = \deg_{\text{in}G}(v) = 0$, so wird v als **Quelle** bezeichnet; gilt $\deg_G(v) = \deg_{\text{aus}G}(v) = 0$, so heißt v **Senke**.

Ist v Quelle und Senke, so heißt v **isoliert**.

²⁰ In der Literatur findet man für $\delta(G, U)$ bzw. $N(G, U)$ oft die Bezeichnungen $\delta_G(U)$ bzw. $N_G(U)$ oder verkürzt $\delta(U)$ bzw. $N(U)$. Auch werden δ_{in} bzw. δ_{aus} oft mit δ^- bzw. δ^+ bezeichnet; manchmal jedoch auch umgekehrt.

Offenbar sind $\deg_{\text{in}G}(v)$ bzw. $\deg_{\text{aus}G}(v)$ die Anzahl der ‘in v hinein-’ bzw. ‘aus v herauslaufenden’ gerichteten Kanten von G . In analoger Interpretation gibt $\deg_G(v)$ an, wieviele ungerichtete Kanten ‘in v hinein-’ (bzw. ‘aus v heraus-’) laufen.

Erreichbarkeitsprobleme in allgemeinen Graphen basieren auf der folgenden Definition.

2.2.9 Definition. Seien $G := (V, E, \nu)$ ein allgemeiner Graph und $p \in \mathbb{N}$.

- (a) Ein **Kantenzug** [engl.: walk] der **kombinatorischen Länge** p in G ist eine Sequenz

$$(v_0, e_1, v_1, e_2, \dots, e_p, v_p)$$

von (alternierend) Knoten v_0, v_1, \dots, v_p und Kanten e_1, e_2, \dots, e_p von G , so dass für $k \in [p]$ gilt

$$\nu(e_k) \in \begin{cases} \{\{v\}, (v, v)\}, & \text{falls } v_{k-1} = v_k = v; \\ \{\{v_{k-1}, v_k\}, (v_{k-1}, v_k)\}, & \text{falls } v_{k-1} \neq v_k. \end{cases}$$

Zur Betonung von v_0, v_p spricht man auch von einem v_0 - v_p -**Kantenzug**. Häufig wird ein solcher Kantenzug nur mit (e_1, e_2, \dots, e_p) bezeichnet oder, wenn das eindeutig ist, auch nur mit (v_0, v_1, \dots, v_p) .

- (b) Seien $v, w \in V$. Dann heißt w **von v aus erreichbar**, wenn $v = w$ gilt, oder wenn es einen Kantenzug $(v_0, e_1, v_1, e_2, \dots, e_p, v_p)$ in G gibt mit $v = v_0$ und $w = v_p$.
- (c) Sei $W := (v_0, e_1, v_1, e_2, \dots, e_p, v_p)$ ein Kantenzug in G . W heißt **einfach**²¹ [engl.: simple], wenn e_1, e_2, \dots, e_p paarweise verschieden sind, und (v_0, v_p) -**Weg** [engl.: path], wenn v_0, v_1, \dots, v_p paarweise verschieden sind. v_0 und v_p heißen dann **Endknoten** des Weges, v_1, \dots, v_{p-1} **innere Knoten**.

Der Kantenzug W heißt **geschlossen**, wenn $v_0 = v_p$ gilt, und **Zyklus** [engl.: cycle], wenn W geschlossen und einfach ist. Ist W ein Zyklus, so heißt W **Kreis** [engl.: circuit], wenn $(v_0, e_1, v_1, e_2, \dots, e_{p-1}, v_{p-1})$ ein Weg ist; p heißt dann die **kombinatorische Länge** des Kreises.

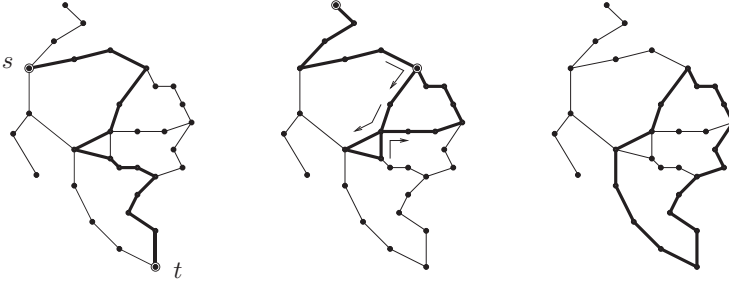
Sind C ein Kreis und v ein Knoten von C , so wird C bisweilen auch als (geschlossener) v - v -Weg bezeichnet. Ferner wird ein einzelner Knoten manchmal als **uneigentlicher** Kantenzug, Weg oder Kreis der Länge 0 aufgefasst; alle anderen werden dann (zur Abgrenzung) **eigentlich** genannt.

Besitzt G keinen eigentlichen Kreis, so heißt G **kreisfrei**.

2.2.10 Bemerkung. Seien $G := (V, E, \nu)$ ein allgemeiner Graph und $s, t \in V$.

- (a) Ist t von s aus erreichbar, so gibt es einen s - t -Weg.
- (b) Jeder (eigentliche) Zyklus enthält einen (eigentlichen) Kreis.
- (c) Ist G gerichtet, so enthält jeder (eigentliche) geschlossene Kantenzug einen (eigentlichen) Kreis.

²¹ Oft werden einfache Kantenzüge in der Literatur auch *Pfad* genannt. Wir verwenden diesen Begriff hier nicht, um eine mögliche Konfusion mit dem englischen Begriff path für Weg zu vermeiden.



2.8 Abbildung. Ein s - t -Weg (links), ein Kantenzug mit markierten Endknoten (Mitte) und ein Kreis (rechts).

Beweis: (a) Offenbar ist jeder Kantenzug $(v_0, e_1, v_1, e_2, \dots, e_p, v_p)$ mit $s = v_0$ und $t = v_p$ minimaler (kombinatorischer) Länge ein s - t -Weg.

(b) Jeder eigentliche Zyklus minimaler Länge ist ein Kreis.

(c) Jeder eigentliche geschlossene Kantenzug minimaler Länge in einem gerichteten Graphen ist eigentlicher ein Kreis. \square

Man beachte, dass die Aussage (c) in Bemerkung 2.2.10 für ungerichtete Graphen nicht gilt. Sind $v_0, v_1 \in V$, $v_0 \neq v_1$ und $e \in E$ mit $\nu(e) = \{v, w\}$, so ist (v_0, e, v_1, e, v_0) ein geschlossener Kantenzug, enthält aber keinen eigentlichen Kreis.

2.2.11 Bemerkung. Seien $G := (V, E, \nu)$ ein allgemeiner Graph, und die Relation \sim auf $V \times V$ sei für $v, w \in V$ definiert durch

$$v \sim w \quad :\Leftrightarrow \quad w \text{ ist von } v \text{ aus erreichbar.}$$

Dann gelten die folgenden Aussagen:

(a) \sim ist reflexiv und transitiv.

(b) Ist G ungerichtet, so ist \sim symmetrisch.

Beweis: (a) Die Reflexivität folgt aus der Definition; zum Nachweis der Transitivität werden entsprechende Kantenzüge hintereinander geschaltet.

(b) Die Symmetrie im ungerichteten Fall folgt daraus, dass mit $(v_0, e_1, v_1, e_2, \dots, e_p, v_p)$ stets auch $(v_p, e_p, v_{p-1}, e_{p-1}, \dots, e_1, v_0)$ ein Kantenzug ist. \square

Bei der Bestimmung kürzester Wege von einem Startknoten s zu einem Zielknoten t kann man natürlich solche Knoten und Kanten weglassen, die offensichtlich auf keiner solchen Route liegen können. (Hierzu gehört in dem in Abbildung 2.4 dargestellten Graphen etwa der ‘Wurmfortsatz’ von s nach oben.) Man geht somit zu einem Teilgraphen über.

2.2.12 Definition. Seien $G_1 := (V_1, E_1, \nu_1)$ und $G_2 := (V_2, E_2, \nu_2)$ allgemeine Graphen, und es gelte

$$V_1 \subset V_2 \quad \wedge \quad E_1 \subset E_2 \quad \wedge \quad \nu_1 = \nu_2|_{E_1};$$

($\nu_2|_{E_1}$ bezeichnet gemäß Konvention 1.3.12 die Einschränkung von ν_2 auf E_1). Dann heißt G_1 **Teil-, Sub- oder auch Untergraph** [engl.: subgraph] von G_2 . Bisweilen schreibt man dann $G_1 \subset G_2$.

Ist G_1 ein Teilgraph von G_2 und gilt

$$E_1 = \left\{ e \in E_2 : \nu_2(e) \in \binom{V_1}{1} \cup \binom{V_1}{2} \cup V_1^2 \right\},$$

so heit G_1 der durch V_1 in G_2 **induzierte Teilgraph** [engl.: induced subgraph] von G_2 .

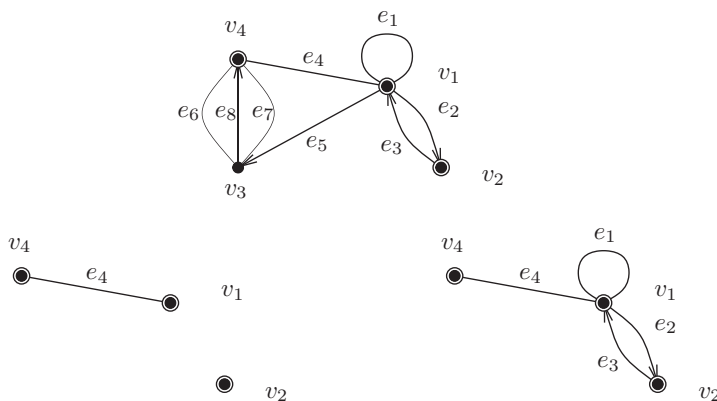
2.2.13 Beispiel. Gegeben sei der allgemeine Graph $G := (V, E, \nu)$ aus Beispiel 2.2.4. Ferner seien

$$U := \{v_1, v_2, v_4\} \quad \wedge \quad E_1 := \{e_4\} \quad \wedge \quad E_2 := \{e_1, e_2, e_3, e_4\}$$

Dann sind

$$S_1 := (U, E_1, \nu|_{E_1}) \quad \wedge \quad S_2 := (U, E_2, \nu|_{E_2})$$

Teilgraphen von G ; S_2 ist der durch die Knotenmenge U induzierte Teilgraph von G ; vgl. Abbildung 2.9.



2.9 Abbildung. Allgemeiner Graph mit markierter Teilknotenmenge (oben), Teilgraph (unten links), induzierter Teilgraph (unten rechts).

2.2.14 Bemerkung. Sei $G := (V, E, \nu)$ ein allgemeiner Graph. Seien

$$W := (v_0, e_1, v_1, e_2, \dots, e_p, v_p)$$

ein Kantenzug in G sowie

$$V_W := \{v : \exists (k \in \{0, \dots, p\}) : v = v_k\} \quad \wedge \quad E_W := \{e : \exists (k \in \{1, \dots, p\}) : e = e_k\}.$$

Dann ist $(V_W, E_W, \nu|_{E_W})$ ein Teilgraph von G .

Oft ist es sinnvoll (und nach Bemerkung 2.2.14 zulssig), Wege oder Kreise als Teilgraphen aufzufassen.

Mit Hilfe der in Definition 2.2.9 eingefhrten Erreichbarkeit kann man den Zusammenhang in Graphen definieren. Dabei gibt es je nach Anwendung durchaus verschiedene Mglichkeiten, mit gerichteten Kanten umzugehen.

2.2.15 Definition. Sei $G := (V, E, \nu)$ ein allgemeiner Graph.

- (a) Seien $s \in V$ und $U(s)$ die Menge aller Knoten von V , die von s aus erreichbar sind. Gilt $V = U(s)$, so heißt G **von s aus zusammenhängend**. G heißt **zusammenhängend**, wenn G von jedem seiner Knoten aus zusammenhängend ist, d.h. wenn je zwei Knoten durch einen Kantenzug verbunden sind.
- (b) Ein Teilgraph S von G heißt genau dann **Zusammenhangskomponente**, wenn S zusammenhängend ist und kein zusammenhängender Teilgraph von G existiert, der S echt enthält.

Der Beispielgraph von Abbildung 2.4 bzw. 2.8 ist zusammenhängend (wie man es von einem solchen Verkehrsnetz auch erwarten sollte).

Da nach Bemerkung 2.2.11 die Erreichbarkeitsrelation in ungerichteten Graphen eine Äquivalenzrelation ist, deren Äquivalenzklassen gerade die Zusammenhangskomponenten sind, zerfallen ungerichtete Graphen in ihre Zusammenhangskomponenten.

Wie die folgende Bemerkung zeigt, lässt sich der Zusammenhang in allgemeinen Graphen auch mit Hilfe der Nachbarschaften charakterisieren.

2.2.16 Bemerkung. Sei $G := (V, E, \nu)$ ein allgemeiner Graph. G ist genau dann zusammenhängend, wenn für jede nichtleere, echte Teilmenge U von V

$$N(G, U) \cup N_{\text{aus}}(G, U) \neq \emptyset$$

gilt.

Beweis: ‘ \Rightarrow ’ Seien $U \subset V$ mit $\emptyset \neq U \neq V$ und $s \in U$, $t \notin U$. Dann gibt es einen s - t -Weg in G . Somit existieren Knoten $v \in U$, $w \in V \setminus U$ sowie eine Kante e mit $\nu(e) \in \{\{v, w\}, (v, w)\}$, und es folgt $N(G, U) \cup N_{\text{aus}}(G, U) \neq \emptyset$.

‘ \Leftarrow ’ Seien $s \in V$ und U die Menge der von s aus erreichbaren Knoten von V . Dann gilt $s \in U$. Angenommen, $U \neq V$. Dann folgt aus $N(G, U) \cup N_{\text{aus}}(G, U) \neq \emptyset$ die Existenz von Knoten $v \in U$, $w \in V \setminus U$ und einer Kante e mit $\nu(e) \in \{\{v, w\}, (v, w)\}$. Da v von s aus erreichbar ist, gibt es einen s - w -Weg in G , d.h. $w \in U$ im Widerspruch zur Annahme.

Insgesamt folgt damit die Behauptung. \square

Für manche Zwecke ist es sinnvoll, einen schwächeren Begriff des Zusammenhangs wie folgt zu definieren.

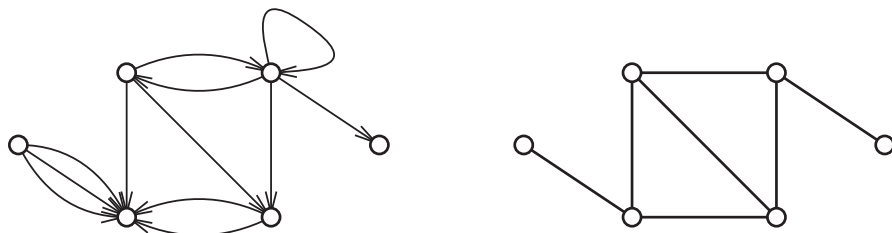
2.2.17 Definition. Sei $G := (V, E, \nu)$ ein allgemeiner Graph. G ist genau dann **schwach zusammenhängend**, wenn für jede echte Teilmenge U von V

$$N(G, U) \cup N_{\text{in}}(G, U) \cup N_{\text{aus}}(G, U) \neq \emptyset$$

gilt. Zur besonderen Betonung des in Definition 2.2.15 eingeführten stärkeren Zusammenhangsbegriffs wird ein zusammenhängender allgemeiner Graph bisweilen auch **stark zusammenhängend** genannt.

Der schwache Zusammenhang entspricht dem starken Zusammenhang in dem Graphen, der durch ‘Ignorierung’ der Kantenrichtungen entsteht. Natürlich kann man auch noch Mehrfachkanten und Schleifen weglassen, ohne den Zusammenhang zu beeinflussen; vgl. Abbildung 2.10.

Hamiltonkreise und Eulertouren: Die folgenden beiden Begriffe spielen für viele Probleme der Logistik eine zentrale Rolle.



2.10 Abbildung. Links: schwach, aber nicht stark zusammenhängender gerichteter allgemeiner Graph G . Rechts: ungerichteter allgemeiner Graph \hat{G} , der durch Ignorierung der Kantenrichtungen und Reduktion von Mehrfachkanten und Schleifen aus G entsteht. \hat{G} ist (stark) zusammenhängend.

2.2.18 Definition. Sei $G := (V, E, \nu)$ ein allgemeiner Graph.

- (a) Jeder Weg bzw. Kreis, der alle Knoten von G enthält, heißt **Hamiltonweg**²² bzw. **Hamiltonkreis**. Ist G leer oder besitzt G einen Hamiltonkreis, so heißt G **hamiltonsch**.
- (b) Jeder einfache Kantenzug bzw. Zyklus, der alle Kanten von G enthält, heißt **Euler-Kantenzug**²³ bzw. **Eulertour**. Ist G leer oder besitzt G eine Eulertour, so heißt G **eulersch**.

Hamiltonkreise sind etwa für die Minimierung von Fahrstrecken bei der Auslieferung von Gütern an verschiedene Kunden relevant; Eulertouren treten unter anderem bei der Plottersteuerung oder der Straßenreinigung auf.²⁴

Im Folgenden werden zwei theoretische Ergebnisse angegeben, die jeweils die Existenz eines Hamiltonkreises oder einer Eulertour aus der ‘Reichhaltigkeit’ bzw. ‘Struktur’ von Nachbarschaften folgern.

Der nachfolgende erste dieser Sätze besagt grob, dass ein allgemeiner Graph G , in dem jeder Knoten nur genügend viele Nachbarn besitzt, bereits hamiltonsch sein muss. Das ist offensichtlich, wenn in G sogar jeder Knoten von jedem anderen aus bereits über eine einzige Kante erreichbar ist, da es dann keinerlei Hindernisse gibt, von einem beliebigen Startknoten aus jeweils so lange zu einem noch nicht besuchten neuen aktuellen Knoten zu gehen, bis alle erreicht sind, und mit der letzten Kante dann den Hamiltonkreis zu schließen. Das nachfolgende Ergebnis verschärft diese Beobachtung bis an ihre mathematische Grenze.

2.2.19 Satz. Sei $G := (V, E, \nu)$ ein allgemeiner Graph mit $|V| \geq 3$, und es gelte für jeden Knoten $v \in V$

$$|N(G, v) \cup N_{\text{aus}}(G, v)| \geq \frac{1}{2}|V| \quad \wedge \quad |N(G, v) \cup N_{\text{in}}(G, v)| \geq \frac{1}{2}|V|.$$

Dann ist G hamiltonsch.

²² Sir William Rowan Hamilton, 1805 – 1865.

²³ Leonhard Euler, 1707 – 1783.

²⁴ Ein ‘klassisches’ Beispiel für Hamiltonkreise ist Eulers Lösung des ‘Rösselsprungproblems’ aus dem Jahr 1757 (Brief an Goldbach), bei dem eine geschlossene Folge von Zügen des Springers auf einem Schachbrett gesucht ist, die jedes Feld genau einmal erreicht. Für die Frage nach Euler-Pfaden sind das ‘Haus des Nikolaus’ oder das Königsberger Brückenproblem bekannte Beispiele.

Beweis: Seien $p \in \mathbb{N}$ und $V_K := \{v_1, \dots, v_p\} \subset V$, so dass

$$K := (v_1, v_2, \dots, v_p, v_1)$$

ein Kreis maximaler Länge in G ist. Gilt $V = V_K$, so ist K ein Hamiltonkreis, und es bleibt nichts zu zeigen. Wir nehmen daher an, dass $V \setminus V_K \neq \emptyset$ gilt. Sei

$$d := \left\lceil \frac{1}{2}|V| \right\rceil.$$

Wir zeigen zunächst, dass

$$p \geq d + 1$$

gilt. Seien dazu $k \in \mathbb{N}$ und $U := (u_0, u_1, \dots, u_k)$ ein Weg maximaler Länge in G ; vgl. Abbildung 2.11 (links). Aus $|N(G, u_i) \cup N_{\text{aus}}(G, u_i)| \geq d$ für alle $i \in \{0\} \cup [k]$ folgt $k \geq d$. Ist $u_0 = u_k$, so ist U ein Kreis von mindestens der Länge $d + 1$. Sei also $u_0 \neq u_k$. Aus der Maximalität von U folgt

$$N(G, u_0) \cup N_{\text{in}}(G, u_0) \subset \{u_1, \dots, u_k\}.$$

Wegen $|N(G, u_0) \cup N_{\text{in}}(G, u_0)| \geq d$ gibt es daher einen Index $i_0 \in [k]$ und eine Kante $e \in E$ mit

$$i_0 \geq d \quad \wedge \quad \nu(e) \in \{\{u_{i_0}, u_0\}, (u_{i_0}, u_0)\}.$$

Somit ist $(u_{i_0}, u_0, u_1, \dots, u_{i_0})$ ein Kreis in G der Länge mindestens $d + 1$. Es gilt also $p \geq d + 1$.

Seien nun $q \in \mathbb{N}_0$ und $V_W := \{w_0, \dots, w_q\} \subset V \setminus V_K$, so dass

$$W := (w_0, \dots, w_q)$$

ein Weg maximaler Länge in dem von $V \setminus V_K$ induzierten Teilgraphen G' von G ist; vgl. Abbildung 2.11 (rechts). Dann gilt insbesondere $p + q + 1 \leq |V|$, und aus $p \geq d + 1$ folgt

$$q \leq |V| - p - 1 \leq |V| - (d + 1) - 1 = |V| - \left\lceil \frac{1}{2}|V| \right\rceil - 2 \leq d - 2.$$

Seien nun (mit der Setzung $v_0 := v_p$)

$$\begin{aligned} X &:= \left\{ i \in [p] : \exists (e \in E) : \nu(e) \in \{\{v_{i-1}, w_0\}, (v_{i-1}, w_0)\} \right\} \\ Y &:= \left\{ i \in [p] : \exists (e \in E) : \nu(e) \in \{\{w_q, v_i\}, (w_q, v_i)\} \right\}. \end{aligned}$$

Angenommen, es wäre $X \cap Y \neq \emptyset$; sei etwa $i \in X \cap Y$. Dann wäre für $i \neq 1$

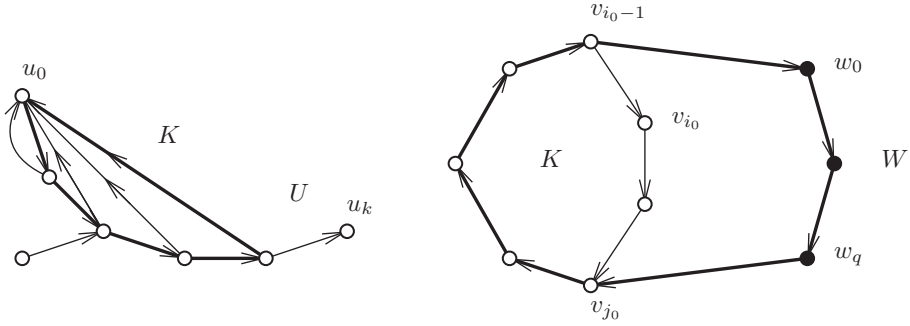
$$(v_{i-1}, w_0, \dots, w_q, v_i, \dots, v_p, v_1, \dots, v_{i-1})$$

bzw. für $i = 1$

$$(v_p, w_0, \dots, w_q, v_1, \dots, v_p)$$

ein Kreis der Länge $(p - 1) + 2 + q = p + q + 1 \geq p + 1$, im Widerspruch zur Maximalität des Kreises K . Somit gilt $X \cap Y = \emptyset$.

Da W ein Weg maximaler Länge in G' ist, gilt



2.11 Abbildung. Konstruktionen im Beweis von Satz 2.2.19. Links: Kreis (hervorgehoben) der Länge $d+1$ in G für $|V| = 7$, $k = 5$ und $d = 4$. Rechts: Längerer Hamiltonkreis (hervorgehoben) für $|V| = 10$, $p = 7$, $q = 2$ und $r = 2$. Nicht alle Kanten von G sind eingezeichnet; die Pfeile zeigen die ‘Durchlaufrichtung’ der konstruierten Kreise auch für solche Kanten an, die ungerichtet sind. Die Knoten von K sind weiß, die von W schwarz ausgefüllt.

$$N(G, w_0) \cup N_{\text{in}}(G, w_0) \subset V_K \cup V_W \quad \wedge \quad N(G, w_q) \cup N_{\text{aus}}(G, w_q) \subset V_K \cup V_W.$$

Nun enthält X gerade die Indizes i aller Knoten $v_{i-1} \in V_K$, von denen w_0 über eine Kante erreichbar ist. Entsprechend ist Y die Menge der Indizes i von Knoten $v_i \in V_K$, die von w_q aus durch eine Kante erreicht werden. Es folgt

$$\begin{aligned} |X| &= |N(G, w_0) \cup N_{\text{in}}(G, w_0)| - |N(G', w_0)| \geq d - q \\ |Y| &= |N(G, w_q) \cup N_{\text{aus}}(G, w_q)| - |N(G', w_q)| \geq d - q \end{aligned}$$

sowie

$$|X \cup Y| = |X| + |Y| \geq 2d - 2q \geq (|V| - q) - q \geq p - q + 1.$$

Da $q \leq d - 2$ ist, sind weder X noch Y leer. Seien nun $i_0 \in X$ und $j_0 \in Y$, so dass der Abstand mod p , d.h.

$$\begin{cases} j_0 - i_0, & \text{falls } i_0 \leq j_0 \\ j_0 + p - i_0, & \text{falls } i_0 > j_0 \end{cases}$$

unter allen solchen Paaren minimal ist, und r sei dieses Minimum. Dann gilt

$$r - 1 \leq p - |X \cup Y| \leq p - (p - q + 1) = q - 1,$$

also $r \leq q$, und der Kreis

$$(v_{i_0-1}, w_0, \dots, w_q, v_{j_0}, \dots, v_{i_0-1})$$

hat somit die Länge $p - r - 1 + q + 2 \geq p + 1$, im Widerspruch zu Maximalität von K . Daher ist $V = V_K$, und es folgt die Behauptung. \square

Satz 2.2.19 impliziert, dass einfache Graphen hoher Valenz immer hamiltonsch sind.

2.2.20 Korollar. (Satz von Dirac²⁵)

Sei $G := (V, E, \nu)$ ein einfacher Graph mit $|V| \geq 3$, und es gelte für jeden Knoten $v \in V$

²⁵ Gabriel Andrew Dirac, 1925 – 1984.

$$\deg_G(v) + \deg_{\text{aus}_G}(v) \geq \frac{1}{2}|V| \quad \wedge \quad \deg_G(v) + \deg_{\text{in}_G}(v) \geq \frac{1}{2}|V|.$$

Dann ist G hamiltonsch.

Beweis: Die Behauptung folgt direkt aus Satz 2.2.19, da für einfache Graphen

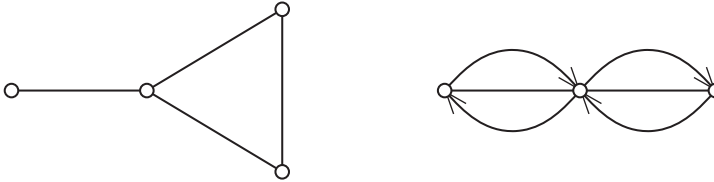
$$\deg_G(v) = |N(G,v)| \quad \wedge \quad \deg_{\text{aus}_G}(v) = |N_{\text{aus}}(G,v)| \quad \wedge \quad \deg_{\text{in}_G}(v) = |N_{\text{in}}(G,v)|$$

sowie

$$N(G,v) \cap N_{\text{aus}}(G,v) = N(G,v) \cap N_{\text{in}}(G,v) = \emptyset$$

gilt. □

Abbildung 2.12 (links) zeigt, dass die Bedingungen in Satz 2.2.19 und Korollar 2.2.20 bereits für ungerichtete einfache allgemeine Graphen scharf sind. Abbildung 2.12 (rechts) gibt einen schlichten allgemeinen Graphen an, für den Korollar 2.2.20 nicht gilt.



2.12 Abbildung. Links: Einfacher, ungerichteter nicht-hamiltonscher Graph. Die Bedingungen von Satz 2.2.19 und Korollar 2.2.20 sind lediglich für einen Knoten verletzt. Rechts: schlichter nicht-hamiltonscher allgemeiner Graph.

Wir kommen nun zu dem angekündigten Satz über die Existenz von Eulertouren. Für die Eigenschaft allgemeiner Graphen G , eulersch zu sein, können natürlich isolierte Knoten ignoriert werden. Außerdem muss G zusammenhängend sein.²⁶ Satz 2.2.25 enthält eine Charakterisierung für allgemeine Graphen. Zur Vorbereitung beweisen wir eine entsprechende Aussage für gerichtete allgemeine Graphen. Anschaulich beruhen das nachfolgende Lemma sowie der anschließende Satz auf der Beobachtung, dass eine Eulertour in jeden Knoten genauso oft ‘hinein-’ wie ‘herauslaufen’ muss.

2.2.21 Lemma. Seien $G := (V, E, \nu)$ ein gerichteter allgemeiner Graph und $E \neq \emptyset$. Dann sind die folgenden beiden Aussagen äquivalent:

- (a) E ist die Vereinigung der Kantenmengen eigentlicher kantendisjunkter geschlossener einfacher Kantenzüge in G .
- (b) Für jeden Knoten v von G gilt $\deg_{\text{in}}(v) = \deg_{\text{aus}}(v)$.

Beweis: ‘(a) \Rightarrow (b)’ Seien $r \in \mathbb{N}$ und K_1, \dots, K_r eigentliche kantendisjunkte geschlossene einfache Kantenzüge in G , deren Kantenmengen insgesamt E ergeben. Sei $K := (v_0, e_1, v_1, e_2, \dots, e_p, v_p) \in \{K_1, \dots, K_r\}$. Dann gilt

$$v_0 = v_p \quad \wedge \quad (i \in [p] \Rightarrow \nu(e_i) = (v_{i-1}, v_i)).$$

²⁶ In der Formulierung der nachfolgenden Ergebnisse kommen wir mit dem schwachen Zusammenhang als genereller Voraussetzung aus, da dieser mit den weiteren kombinatorischen Bedingungen den starken Zusammenhang impliziert.

Für jedes $i \in [p]$ ist v_i Endknoten von e_i und Anfangsknoten von e_{i+1} (bzw. e_1 für $i = p$). Die Gradbedingung gilt also in K . Da jede Kante von G in genau einem der Kantenzüge K_1, \dots, K_r auftritt, stimmt daher an jedem Knoten der Ingrad mit dem Ausgrad überein.

‘(b) \Rightarrow (a)’ Wir führen den Beweis als vollständige Induktion über $m := |E|$. Für $m = 1$ ist die einzige Kante eine Schlinge, und es gilt die Behauptung. Im folgenden sei also $m \geq 2$, und wir setzen voraus, dass die Behauptung für alle kleineren Kantenzahlen gilt. Besitzt G eine Schlinge, so kann diese entfernt werden, ohne die Gradbedingung zu verletzen, und die Behauptung folgt aus der Induktionsannahme. Wir können daher im Folgenden annehmen, dass G schlingenfrei ist.

Seien nun $e_1 \in E$, $E_1 := \{e_1\}$, $v_0, v_1 \in V$ mit $\nu(e_1) = (v_0, v_1)$ und $T_1 := (v_0, e_1, v_1)$. Wegen $\deg_{\text{in}}(v_1) = \deg_{\text{aus}}(v_1) \geq 1$ gibt es eine Kante $e_2 \in E \setminus E_1$ und einen Knoten $v_2 \in V$ mit $\nu(e_2) = (v_1, v_2)$. Seien $E_2 := E_1 \cup \{e_2\}$ und $T_2 := (v_0, e_1, v_1, e_2, v_2)$.

Wir fahren mit dieser Konstruktion solange fort, bis wir an einen Knoten v_p kommen, der nicht über eine Kante aus $E \setminus E_p$ verlassen werden kann. Wegen $\deg_{\text{in}}(v_p) = \deg_{\text{aus}}(v_p) \geq 1$ gilt $v_p \in \{v_0, \dots, v_{p-1}\}$, d.h. T_p enthält einen geschlossenen einfachen Kantenzug C . Jeder Knoten von C ist natürlich gleich oft Anfangs- wie Endknoten von Kanten in C . Entfernt man also alle Kanten von C aus G und ebenso alle dann isolierten Knoten, so entsteht ein Teilgraph, dessen Knoten nach wie vor die Gradvoraussetzung erfüllen. Nach Induktionsannahme folgt daher (b). \square

Wir erhalten nun die folgende Charakterisierung der eulerschen gerichteten allgemeinen Graphen.

2.2.22 Satz. (Satz von Euler²⁷ (gerichtete Version))

Sei $G := (V, E, \nu)$ ein schwach zusammenhängender gerichteter allgemeiner Graph. G ist genau dann eulersch, wenn $\deg_{\text{in}}(v) = \deg_{\text{aus}}(v)$ für jeden Knoten v von G gilt.

Beweis: O.B.d.A. sei $E \neq \emptyset$.

‘ \Rightarrow ’ Ist G eulersch, so ist E die Kantenmenge jeder Eulertour, und die Aussage folgt aus Lemma 2.2.21.

‘ \Leftarrow ’ Gemäß Lemma 2.2.21 seien $r \in \mathbb{N}$ und K_1, \dots, K_r eigentliche kantendisjunkte geschlossene einfache Kantenzüge in G , die insgesamt alle Kanten von G enthalten. Wir zeigen die Behauptung mittels vollständiger Induktion nach r .

Ist $r = 1$, so ist K_1 eine Eulertour. Sei also $r \geq 2$, und die Behauptung gelte für alle kleineren Anzahlen. Da G schwach zusammenhängend ist, gibt es ein $i_0 \in \{2, \dots, r\}$, so dass K_1 und K_{i_0} (mindestens) einen Knoten gemeinsam haben. O.B.d.A. sei $i_0 = 2$, und v_0 sei ein Knoten in K_1 und K_2 . Benutzen wir v_0 als Start- und Endknoten der geschlossenen einfachen Kantenzüge K_1 und K_2 , so sind diese darstellbar als

$$K_1 = (v_0, e_1, v_1, e_2, \dots, e_p, v_0) \quad \wedge \quad K_2 = (v_0, e_{p+1}, v_{p+1}, e_{p+2}, \dots, e_q, v_0)$$

mit $p \in \mathbb{N}$ und $q - p \in \mathbb{N}$. Dann ist die Hintereinanderausführung

$$K_{1,2} := (v_0, e_1, v_1, e_2, \dots, e_p, v_0, e_{p+1}, v_{p+1}, e_{p+2}, \dots, e_q, v_0)$$

ein geschlossener einfacher Kantenzug, und $K_{1,2}, K_3, \dots, K_r$ sind kantendisjunkte geschlossene einfache Kantenzüge in G , deren Kantenmengen E überdecken. Nach Induktionsannahme ist G somit eulersch. \square

²⁷ Leonhard Euler, 1707 – 1783.

Wir wollen nun charakterisieren, wann ein allgemeiner Graph $G = (V, E, \nu)$ eulersch ist. Besitzt G eine Eulertour, so wird durch Festlegung einer ihrer Durchlaufrichtungen jede in G ungerichtete Kante gerichtet. Die Bedingung von Satz 2.2.22 für den gerichteten Fall besagt dann, dass an jedem Knoten der Ingrad mit dem Ausgrad übereinstimmt. Somit ist eine Bedingung dafür, dass G eulersch ist, dass in G genügend ungerichtete Kanten vorhanden sein müssen, um an jedem Knoten v eine möglicherweise vorhandene Differenz zwischen $\deg_{\text{in}}(v)$ und $\deg_{\text{aus}}(v)$ ausgleichen zu können. Eine solche Bedingung kann jedoch nicht rein lokal auf jeden einzelnen Knoten beschränkt sein, da die Wahl einer Durchlaufrichtung einer ungerichteten Kante die Graddifferenz an beiden betroffenen Knoten verändert.

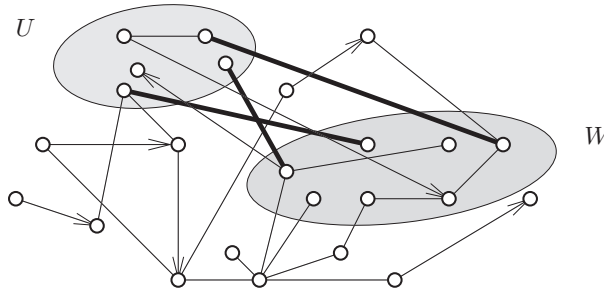
Bevor wir den entsprechenden Satz formulieren, führen wir noch weitere Bezeichnungen zur Ergänzung von Bezeichnung 2.2.8 ein.

2.2.23 Bezeichnung. Seien $G := (V, E, \nu)$ ein allgemeiner Graph und $U, W \subset V$. Dann seien

$$E(U, W) := \{e \in E : \exists(u \in U \setminus W \wedge w \in W \setminus U) : \nu(e) = \{u, w\}\}$$

$$E_{\text{in}}(U, W) := \{e \in E : \exists(u \in U \setminus W \wedge w \in W \setminus U) : \nu(e) = (w, u)\}$$

$$E_{\text{aus}}(U, W) := \{e \in E : \exists(u \in U \setminus W \wedge w \in W \setminus U) : \nu(e) = (u, w)\}$$



2.13 Abbildung. Ein allgemeiner Graph $G := (V, E, \nu)$, Teilmengen $U, W \subset V$ (grau hervorgehoben) sowie die Kanten der Menge $E(U, W)$ (fett).

Natürlich stehen die eingeführten Kantenmengen in enger Beziehung zu den Mengen aus Bezeichnung 2.2.8.

2.2.24 Bemerkung. Seien $G := (V, E, \nu)$ ein allgemeiner Graph und $U, W \subset V$. Dann gelten die folgenden Aussagen:

$$(a) \quad E(U, W) = E(W, U);$$

$$(b) \quad E_{\text{in}}(U, W) = E_{\text{aus}}(W, U);$$

$$(c) \quad \delta(G, U) = E(U, G \setminus U);$$

$$(d) \quad \delta_{\text{in}}(G, U) = E_{\text{in}}(U, G \setminus U);$$

$$(e) \quad \delta_{\text{aus}}(G, U) = E_{\text{aus}}(U, G \setminus U);$$

$$(f) \quad |\delta(G, U)| + |\delta(G, W)| = |\delta(G, U \cup W)| + |\delta(G, U \cap W)| + 2|E(U, W)|;$$

$$\begin{aligned}
(g) \quad & |\delta_{\text{in}}(G, U)| + |\delta_{\text{in}}(G, W)| = |\delta_{\text{in}}(G, U \cup W)| + |\delta_{\text{in}}(G, U \cap W)| + |E_{\text{in}}(U, W)| + |E_{\text{in}}(W, U)|; \\
(h) \quad & |\delta_{\text{aus}}(G, U)| + |\delta_{\text{aus}}(G, W)| = \\
& \quad |\delta_{\text{aus}}(G, U \cup W)| + |\delta_{\text{aus}}(G, U \cap W)| + |E_{\text{aus}}(U, W)| + |E_{\text{aus}}(W, U)|.
\end{aligned}$$

Beweis: Die ersten Aussagen folgen direkt aus den Definitionen, die anderen ergeben sich durch geeignete Aufteilung der auftretenden Kantenmengen; vgl. Übungsaufgabe 2.4.14. \square

Für allgemeine Graphen erhalten wir nun die folgende Charakterisierung.

2.2.25 Satz. (Satz²⁸ von Ford und Fulkerson²⁹)

Sei $G := (V, E, \nu)$ ein schwach zusammenhängender allgemeiner Graph. G ist genau dann eulersch, wenn für alle $u \in V$ und $U \subset V$ gilt

$$|\delta_{\text{aus}}(G, u)| + |\delta_{\text{in}}(G, u)| + |\delta(G, u)| \equiv 0 \pmod{2} \quad \wedge \quad |\delta_{\text{aus}}(G, U)| - |\delta_{\text{in}}(G, U)| \leq |\delta(G, U)|.$$

Beweis: ‘ \Rightarrow ’ Sei $W := (v_0, e_1, v_1, e_2, \dots, e_p, v_0)$ eine Eulertour in G . Seien e eine Kante und $v, w \in V$, so dass (v, e, w) ein Abschnitt von W ist. Ist e gerichtet, so gilt $\nu(e) = (v, w)$, und wir setzen $\hat{\nu}(e) := \nu(e)$. Ist e hingegen ungerichtet, so ist $\nu(e) = \{v, w\}$, und wir setzen $\hat{\nu}(e) := (v, w)$. Diese Orientierung aller ungerichteten Kanten ‘gemäß einem Durchlaufen von W ’ liefert insgesamt einen eulerschen Digraphen $\hat{G} := (V, E, \hat{\nu})$. Nach Satz 2.2.22 gilt für jeden Knoten $u \in V$

$$\deg_{\text{in}\hat{G}}(u) = \deg_{\text{aus}\hat{G}}(u).$$

Seien nun für $U \subset V$

$$\begin{aligned}
\delta^+(G, U) &:= \{e \in E : \exists(u \in U \wedge v \in V \setminus U) : \nu(e) = \{u, v\} \wedge \hat{\nu}(e) = (u, v)\} \\
\delta^-(G, U) &:= \{e \in E : \exists(u \in U \wedge v \in V \setminus U) : \nu(e) = \{u, v\} \wedge \hat{\nu}(e) = (v, u)\}.
\end{aligned}$$

Dann folgt für $u \in V$

$$|\delta_{\text{aus}}(G, u)| + |\delta^+(G, u)| = |\delta_{\text{in}}(G, u)| + |\delta^-(G, u)|,$$

also

$$\begin{aligned}
& |\delta_{\text{aus}}(G, u)| + |\delta_{\text{in}}(G, u)| + |\delta(G, u)| \\
& \quad = |\delta_{\text{aus}}(G, u)| + |\delta^+(G, u)| + |\delta_{\text{in}}(G, u)| + |\delta^-(G, u)| \\
& \quad \equiv 0 \pmod{2},
\end{aligned}$$

und es folgt die erste Aussage.

Zum Beweis der behaupteten Ungleichung sei wieder $U \subset V$. Da jede Kante, deren beide Endknoten in U liegen, den Gesamtbeitrag 0 in nachfolgender Summation liefert, gilt

$$\begin{aligned}
0 &= \sum_{u \in U} |\delta_{\text{aus}}(G, u)| - |\delta_{\text{in}}(G, u)| + |\delta^+(G, u)| - |\delta^-(G, u)| \\
&= |\delta_{\text{aus}}(G, U)| - |\delta_{\text{in}}(G, U)| + |\delta^+(G, U)| - |\delta^-(G, U)|.
\end{aligned}$$

Somit folgt

²⁸ Man beachte, dass dieser Satz weder der einzige noch der bekannteste Satz von Ford und Fulkerson ist.

²⁹ Lester Randolph Ford jun., geb. 1927; Delbert Ray Fulkerson, 1924 – 1976.

$$|\delta_{\text{aus}}(G, U)| - |\delta_{\text{in}}(G, U)| = |\delta^-(G, U)| - |\delta^+(G, U)| \leq |\delta(G, U)|.$$

‘ \Leftarrow ’ Wir führen den Beweis mittels vollständiger Induktion über die Anzahl k der ungerichteten Kanten von G . Im Fall $k = 0$ folgt aus den Voraussetzungen insbesondere für alle $u \in V$

$$|\delta_{\text{aus}}(G, u)| - |\delta_{\text{in}}(G, u)| \leq 0.$$

Da jede gerichtete Kante einen Anfangs- und einen Endknoten hat, summieren sich die In- und Ausgrade über alle Knoten von V jeweils zu derselben Zahl auf. Es folgt daher

$$0 = \left(\sum_{u \in V} |\delta_{\text{aus}}(G, u)| \right) - \left(\sum_{u \in V} |\delta_{\text{in}}(G, u)| \right) = \sum_{u \in V} (|\delta_{\text{aus}}(G, u)| - |\delta_{\text{in}}(G, u)|) \leq 0.$$

Für alle $u \in V$ gilt somit $|\delta_{\text{aus}}(G, u)| - |\delta_{\text{in}}(G, u)| = 0$, also $\deg_{\text{in}}(u) = \deg_{\text{aus}}(u)$, und die Aussage folgt aus Satz 2.2.22.

Gilt für alle $U \subset V$ mit $|\delta(G, U)| \neq 0$

$$|\delta_{\text{aus}}(G, U)| - |\delta_{\text{in}}(G, U)| < |\delta(G, U)|,$$

so orientieren wir eine ungerichtete Kante beliebig, und die Behauptung folgt aus der Induktionsannahme. Seien daher $U^* \subset V$ mit

$$|\delta_{\text{aus}}(G, U^*)| - |\delta_{\text{in}}(G, U^*)| = |\delta(G, U^*)| > 0$$

sowie

$$e^* \in \delta(G, U^*) \quad \wedge \quad u^* \in U^* \quad \wedge \quad v^* \in V \setminus U^* \quad \wedge \quad \nu(e^*) = \{u^*, v^*\}.$$

Dann sei $\hat{\nu} : E \rightarrow \binom{V}{1} \cup \binom{V}{2} \cup V^2$ definiert durch

$$\hat{\nu}(e) := \begin{cases} \nu(e) & \text{für } e \in E \setminus \{e^*\} \\ (v^*, u^*) & \text{für } e = e^*, \end{cases}$$

und wir setzen $\hat{G} := (V, E, \hat{\nu})$. Der allgemeine Graph \hat{G} entsteht also aus G durch Orientierung von e^* , so dass die Kante in U^* hineinläuft.

Wir zeigen nun, dass die Voraussetzungen auch für \hat{G} erfüllt sind; die Behauptung folgt dann aus der Induktionsannahme.

Die Paritätsbedingung ist weiterhin erfüllt, denn es gilt

$$\begin{aligned} |\delta_{\text{in}}(\hat{G}, u^*)| &= |\delta_{\text{in}}(G, u^*)| + 1 & \wedge & & |\delta_{\text{in}}(\hat{G}, v^*)| &= |\delta_{\text{in}}(G, v^*)| \\ |\delta_{\text{aus}}(\hat{G}, u^*)| &= |\delta_{\text{aus}}(G, u^*)| & \wedge & & |\delta_{\text{aus}}(\hat{G}, v^*)| &= |\delta_{\text{aus}}(G, v^*)| + 1 \end{aligned}$$

sowie für $w \in \{u^*, v^*\}$

$$|\delta(\hat{G}, w)| = |\delta(G, w)| - 1.$$

Wir nehmen nun an, dass die Ungleichungsbedingung nicht für alle Teilmengen von V erfüllt ist. Sei also $U \subset V$ mit

$$|\delta_{\text{aus}}(\hat{G}, U)| - |\delta_{\text{in}}(\hat{G}, U)| > |\delta(\hat{G}, U)|.$$

Da für $u_1, u_2 \in U$ die gerichtete Kante (u_1, u_2) durch den Term $|\delta_{\text{aus}}(\hat{G}, u_1)| + |\delta_{\text{in}}(\hat{G}, u_2)|$ den Beitrag 2 zur nachfolgenden Summe leistet, und $\{u_1, u_2\}$ durch $|\delta(\hat{G}, u_1)| + |\delta(\hat{G}, u_2)|$ ebenfalls 2 beiträgt, folgt

$$\begin{aligned}
& |\delta_{\text{aus}}(\hat{G}, U)| - |\delta_{\text{in}}(\hat{G}, U)| - |\delta(\hat{G}, U)| \\
&= \left(|\delta_{\text{aus}}(\hat{G}, U)| + |\delta_{\text{in}}(\hat{G}, U)| + |\delta(\hat{G}, U)| \right) - 2|\delta_{\text{in}}(\hat{G}, U)| - 2|\delta(\hat{G}, U)| \\
&\equiv \sum_{u \in U} \left(|\delta_{\text{aus}}(\hat{G}, u)| + |\delta_{\text{in}}(\hat{G}, u)| + |\delta(\hat{G}, u)| \right) \equiv 0 \pmod{2}
\end{aligned}$$

und damit sogar

$$|\delta_{\text{aus}}(\hat{G}, U)| - |\delta_{\text{in}}(\hat{G}, U)| \geq |\delta(\hat{G}, U)| + 2.$$

Es gilt daher

$$|\delta_{\text{aus}}(G, U)| - |\delta_{\text{in}}(G, U)| = |\delta(G, U)| \quad \wedge \quad e^* \in \delta_{\text{aus}}(\hat{G}, U),$$

also $v^* \in U \cap (V \setminus U^*)$. Es folgt somit (unter Verwendung von Bezeichnung 2.2.23)

$$e^* \in E(U^*, U) \neq \emptyset.$$

Die Kante e^* wird also – anschaulich gesprochen – benötigt, um die Graddefizite bez. U^* und bez. U zu korrigieren, müsste dafür aber unterschiedlich orientiert werden. Formal erhält man einen Widerspruch zur vorausgesetzten Ungleichung für $U^* \cup U$ (unter Verwendung der entsprechenden Ungleichung für $U \cap U^*$). Mit Hilfe von Bemerkung 2.2.24 folgt nämlich

$$\begin{aligned}
0 &\geq |\delta_{\text{aus}}(G, U^* \cup U)| - |\delta_{\text{in}}(G, U^* \cup U)| - |\delta(G, U^* \cup U)| \\
&= \left(|\delta_{\text{aus}}(G, U)| + |\delta_{\text{aus}}(G, U^*)| - |\delta_{\text{aus}}(G, U \cap U^*)| - |E_{\text{aus}}(U, U^*)| - |E_{\text{aus}}(U^*, U)| \right) \\
&\quad - \left(|\delta_{\text{in}}(G, U)| + |\delta_{\text{in}}(G, U^*)| - |\delta_{\text{in}}(G, U \cap U^*)| - |E_{\text{in}}(U, U^*)| - |E_{\text{in}}(U^*, U)| \right) \\
&\quad - \left(|\delta(G, U)| + |\delta(G, U^*)| - |\delta(G, U \cap U^*)| - 2|E(U^*, U)| \right) \\
&= \left(|\delta_{\text{aus}}(G, U)| - |\delta_{\text{in}}(G, U)| - |\delta(G, U)| \right) + \left(|\delta_{\text{aus}}(G, U^*)| - |\delta_{\text{in}}(G, U^*)| - |\delta(G, U^*)| \right) \\
&\quad - \left(|\delta_{\text{aus}}(G, U \cap U^*)| - |\delta_{\text{in}}(G, U \cap U^*)| - |\delta(G, U \cap U^*)| - 2|E(U^*, U)| \right) \\
&= \left(-|\delta_{\text{aus}}(G, U \cap U^*)| + |\delta_{\text{in}}(G, U \cap U^*)| + |\delta(G, U \cap U^*)| \right) + 2|E(U^*, U)| \\
&\geq 2|E(U^*, U)| \geq 2.
\end{aligned}$$

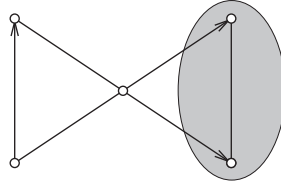
Das ist der angekündigte Widerspruch zur Gültigkeit der vorausgesetzten Ungleichung für $U^* \cup U$, und insgesamt folgt damit die Behauptung. \square

Als Korollar ergibt sich direkt die folgende ungerichtete Version des Satzes von Euler.

2.2.26 Korollar. (*Satz von Euler (ungerichtete Version)*)

Sei $G := (V, E, \nu)$ ein zusammenhängender ungerichteter allgemeiner Graph. G ist genau dann eulersch, wenn jeder Knoten v von G geraden Grad hat.

Beweis: Die Aussage folgt unmittelbar aus Satz 2.2.25; die Kongruenz ist die angegebene Gradbedingung, die Ungleichung ist trivial, und natürlich fallen in ungerichteten allgemeinen Graphen die Begriffe ‘zusammenhängend’ und ‘schwach zusammenhängend’ zusammen. \square



2.14 Abbildung. Jeder Knoten bzw. jede einelementige Knotenmenge erfüllt das Kriterium von Satz 2.2.25; die Bedingung für die gekennzeichnete 2-elementige Knotenmenge ist jedoch verletzt. Der Graph ist somit nicht eulersch.

Man beachte, dass die Bedingungen aus Satz 2.2.22 und Korollar 2.2.26 leicht überprüft werden können, da sie nur die Nachbarschaften einzelner Knoten betreffen. In Satz 2.2.25 kommt man bei den Ungleichungsbedingungen jedoch nicht mit einelementigen Teilmengen U aus; Abbildung 2.14.

Tatsächlich enthält das Kriterium von Satz 2.2.25 eine Ungleichungsbedingung für *jede* der $2^{|V|}$ Teilmengen von V . Natürlich kann man auf die Ungleichungen für $U = \emptyset$ und $U = V$ verzichten, aber es bleiben immer noch exponentiell viele übrig, so dass man sie für große Graphen nicht einfach der Reihe nach durchgehen kann. In der angegebenen Form führt das Kriterium von Satz 2.2.25 für große Knotenzahl daher nicht direkt zu einem praktisch tauglichen Verfahren, um zu überprüfen, ob ein gegebener allgemeiner Graph eulersch ist. Dennoch kann (wie wir später noch sehen werden) diese Eigenschaft effizient überprüft werden.³⁰

In vielen Problemen sind neben den kombinatorischen auch ‘numerische’ Daten gegeben, etwa Fahrzeiten, Bearbeitungszeiten o.ä.. Diese werden häufig mit Hilfe von Kantengewichten modelliert. Kantengewichte sind bisweilen auch relevant, spielen aber in der Theorie eine eher untergeordnete Rolle. Häufig kann man sie mittels Kantengewichten modellieren; vgl. Beispiel 2.2.29.

2.2.27 Definition. Seien $G := (V, E, \nu)$ ein allgemeiner Graph und $\phi : E \rightarrow \mathbb{R}$. Dann heißt $(G; \phi)$ **gewichteter allgemeiner Graph** bzw. **allgemeiner Graph mit Kantengewichten**. Für jede Kante $e \in E$ heißt $\phi(e)$ **Gewicht** (oder, je nach Anwendung, **Kosten**, **Länge**, **Kapazität** etc.) von e . Die Funktion ϕ heißt **Kantengewichtung**.

Ist $W := (v_0, e_1, v_1, e_2, \dots, e_p, v_p)$ ein Kantenzug in G , so heißt

$$\varphi(W) := \sum_{i=1}^p \phi(e_i)$$

Länge (bez. ϕ) von W . Ist W ein Weg oder ein Kreis, so wird $\varphi(W)$ auch **Länge** des Weges bzw. Kreises genannt.

Eine **Kantengewichtung** von G ist eine Funktion $\theta : V \rightarrow \mathbb{R}$ auf V .

In Definition 2.2.27 sind auch negative Gewichte zugelassen. In vielen Anwendungen werden die Längen von Kanten aber nichtnegativ sein.

³⁰ Allerdings zeigt Sektion 3.3 für ein eng verwandtes Problem, dass durchaus ein massiver Komplexitätssprung beim Übergang von ‘reinen’ (gerichteten oder ungerichteten) zu allgemeinen Graphen auftreten kann.

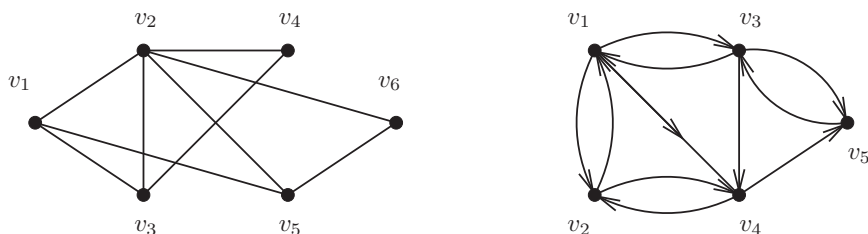
Graphen und Digraphen: Im Folgenden werden wir uns auf die Grundtypen der einfachen ungerichteten oder gerichteten Graphen konzentrieren und die Kanten e mit ihren Bildern $\nu(e)$ unter der definierenden Abbildung ν identifizieren. Wir vereinbaren daher nun für diese Klassen eine äquivalente, aber vereinfachte (und durch die Identifizierung vielleicht intuitivere) Konvention und Kurzbezeichnung.

2.2.28 Bezeichnung. (a) Seien $G := (V, E, \nu)$ ein schlingenfreier, schlichter allgemeiner Graph, $\hat{E} := \nu(E)$ und $\hat{G} := (V, \hat{E})$. Ist G ungerichtet, so heißt \hat{G} **Graph**; ist G gerichtet, so wird \hat{G} **gerichteter Graph** bzw. **Digraph** genannt.

(b) Sind $G := (V, E)$ ein Graph bzw. ein gerichteter Graph und $\phi : E \rightarrow \mathbb{R}$, so heißt $G := (V, E; \phi)$ **gewichteter Graph** bzw. **gewichteter Digraph**.

(c) Seien $G := (V, E)$ ein Graph bzw. Digraph und $U \subset V$. Der durch U induzierte Teilgraph von G wird mit G_U bzw. (U, E_U) bezeichnet. Bisweilen wird für E_U auch $E(U)$ geschrieben.

Die Kanten eines Graphen sind also zweielementige Teilmengen, die eines Digraphen geordnete Paare von Knoten. Offenbar ist (mit der Setzung $\nu(\{v, w\}) := \{v, w\}$) jeder (gewichtete) Graph gemäß Bezeichnung 2.2.28 ein (gewichteter) einfacher, ungerichteter allgemeiner Graph gemäß Definition 2.2.3. Analog ist die Beziehung im gerichteten Fall. Abbildung 2.15 zeigt ‘graphische Darstellungen’ eines ungerichteten Graphen (links) und eines gerichteten Graphen (rechts).



2.15 Abbildung. Graphische Darstellungen eines zusammenhängenden ungerichteten Graphen (links) und eines zusammenhängenden gerichteten Graphen (rechts).

Digraphen treten in einer Vielzahl von unterschiedlichen Problemen auf. Allgemeine Planungsprobleme mit zeitlichen oder logistischen Abhängigkeiten zwischen einzelnen Tätigkeiten können etwa mit Methoden der *Netzplantechnik* behandelt werden. Beispiel 2.2.29 behandelt exemplarisch die Frage nach der Mindestausführungszeit eines Vorhabens.

2.2.29 Beispiel. Bei der Errichtung eines Gebäudes sind von der Planung bis zur Fertigstellung verschiedene Teilgewerke A_1, \dots, A_9 durchzuführen, die zeitlich voneinander abhängen. Die folgende Tabelle enthält zu jedem Teilprojekt seine unmittelbaren ‘Vorgängergewerke’.³¹

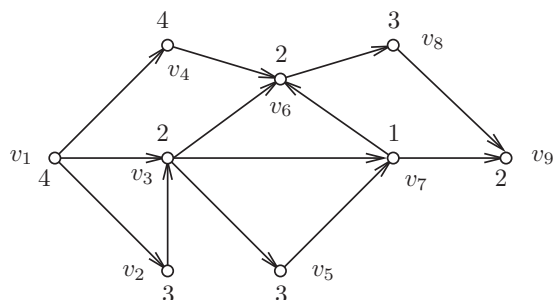
³¹ Natürlich ist die Vorgängerrelation transitiv; der Effizienz und Übersichtlichkeit halber führen wir aber nicht alle Vorgänger auf. Man beachte, dass in einer minimalen graphischen Darstellung gemäß Abbildung 2.16 auch die Kanten (v_1, v_3) , (v_3, v_7) und (v_7, v_9) überflüssig sind, da sie in der transitiven Hülle aller Übrigen liegen.

Gewerk	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9
Vorgänger	–	A_1	A_1, A_2	A_1	A_3	A_3, A_4, A_7	A_3, A_5	A_6	A_7, A_8

Gefragt ist nach der Mindestbauzeit (etwa in Stunden) in Abhängigkeit von den jeweiligen Einzelausführungszeiten γ_i von A_i gemäß

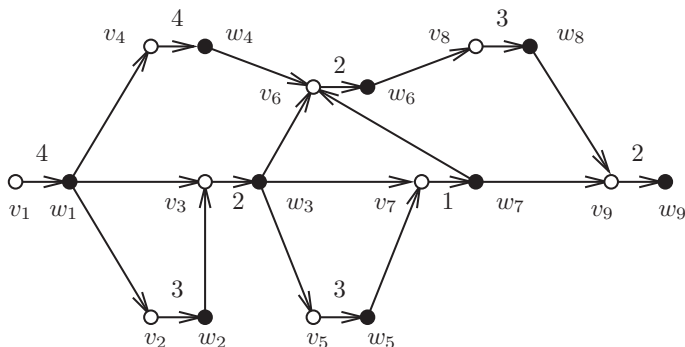
$$\gamma_1 := \gamma_4 := 4 \quad \wedge \quad \gamma_2 := \gamma_5 := \gamma_8 := 3 \quad \wedge \quad \gamma_3 := \gamma_6 := \gamma_9 := 2 \quad \wedge \quad \gamma_7 := 1.$$

Man kann diese Situation mittels eines Digraphen modellieren: seine Knoten entsprechen den einzelnen Teilgewerken und tragen als Gewichte die einzelnen Ausführungszeiten, während seine (ungewichteten) Kanten ihre Abhängigkeiten beschreiben; vgl. Abbildung 2.16.



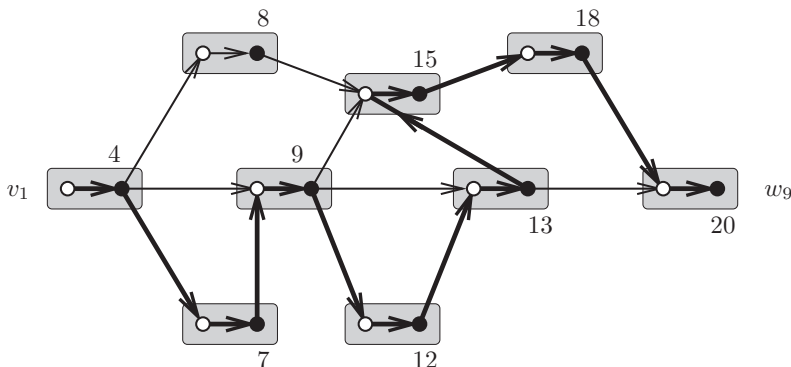
2.16 Abbildung. Graphische Darstellung des Planungsprozesses von Beispiel 2.2.29. Die Knoten entsprechen den Tätigkeiten, die Kanten ihren Abhängigkeiten. Die Ausführungszeiten sind an den Knoten vermerkt.

Man kann die Aufgabe aber auch leicht mittels Kantengewichten modellieren. Jedem einzelnen Gewerk A_i werden dabei zwei Knoten v_i und w_i zugeordnet. Der entstehende gewichtete Digraphen G enthält dann Kanten zweier verschiedener Typen. Zum einen wird für jedes $i = 1, \dots, 9$ die Ausführung von A_i durch die mit $\phi(e) := \gamma_i$ bewertete Kante $e := (v_i, w_i)$ repräsentiert. Zum anderen wird durch mit $\phi(e) := 0$ bewertete Kanten $e := (w_i, v_j)$ modelliert, dass die Fertigstellung von A_i Voraussetzung für die Fertigstellung von A_j ist; vgl. Abbildung 2.17.



2.17 Abbildung. Graphische Darstellung des zugehörigen gewichteten Digraphen. Die von 0 verschiedenen Ausführungszeiten sind an den Kanten vermerkt.

Zur Bestimmung der minimalen Gesamtausführungszeit, kommt es darauf an, wann eine Tätigkeit A_i unter Berücksichtigung ihrer eigenen Ausführungszeit, aber auch der Ausführungstermine aller der Tätigkeiten, die zur Ausführung von A_i bereits zwingend abgeschlossen sein müssen, frühestens ausgeführt sein kann. Diese Zeiten sind in Abbildung 2.18 für das Planungsproblem aus Abbildung 2.17 an den Knoten w_i vermerkt.



2.18 Abbildung. Akkumulierte Gesamtausführungsdauern der einzelnen Tätigkeiten und kritischer v_1 - w_9 -Weg (hervorgehoben) in Beispiel 2.2.29.

Die Mindestausführungszeit entspricht somit der Länge eines längsten v_1 - w_9 -Weges in G bez. der Gewichtsfunktion ϕ . Jeder solche längste v_1 - w_9 -Weg ist insofern **kritisch** (und wird auch so genannt), als die Verzögerung einer einzelnen ‘seiner’ Tätigkeiten die Gesamtausführungszeit verlängert.

Zur Darstellung von Graphen oder Digraphen kann man statt $(|V| \times |E|)$ -Inzidenzmatrizen auch einfachere $(|V| \times |V|)$ -Matrizen verwenden.

2.2.30 Definition. Seien $G := (V, E)$ ein Graph (bzw. ein Digraph) mit $n := |V| \geq 1$ und τ_V eine gegebene Ordnung auf V , d.h. $V =: \{v_1, \dots, v_n\}$. Ferner seien für $i, j \in [n]$

$$\alpha_{i,j} := \begin{cases} 1 & \text{falls } \{v_i, v_j\} \in E \text{ (bzw. } (v_i, v_j) \in E) \text{ gilt;} \\ 0 & \text{sonst.} \end{cases}$$

und

$$A_G := A_G(\tau_V) := (\alpha_{i,j})_{i,j \in [n]}.$$

Dann heißt A_G **Adjazenzmatrix** von G .

2.2.31 Beispiel. Die Adjazenzmatrizen der Graphen aus Abbildung 2.15 sind (mit der durch ihren Index angegebenen Reihenfolge ihrer Knoten):

$$A_{G_1} = \begin{pmatrix} 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix} \quad \wedge \quad A_{G_2} = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}.$$

Offenbar ist A_{G_1} eine symmetrische Matrix, A_{G_2} aber nicht.

2.2.32 Bemerkung. Seien $G := (V, E)$ ein Graph und τ_V eine Reihenfolge auf V . Dann ist A_G symmetrisch.

Beweis: Es gilt

$$\alpha_{i,j} = \alpha_{j,i} = \begin{cases} 1, & \text{falls } \{v_i, v_j\} \in E; \\ 0, & \text{falls } \{v_i, v_j\} \notin E. \end{cases}$$

A_G ist somit symmetrisch. □

Tatsächlich stimmt die Adjazenzmatrix eines Graphen $G := (V, E)$ mit der des Digraphen $G' := (V, E')$ überein, der durch Ersetzen jeder Kante $\{v, w\}$ von G durch die beiden Kanten (v, w) und (w, v) entsteht. In vielen Anwendungen ist es nützlich und in den meisten sogar erlaubt, Graphen als Digraphen aufzufassen.

2.2.33 Bemerkung. Durch Ersetzen jeder Kante $\{v, w\}$ durch die beiden gerichteten Kanten (v, w) und (w, v) kann man einen (ungerichteten) Graphen auch als gerichtet auffassen; siehe Abbildung 2.19.

Ist der Graph gewichtet, so überträgt sich das Gewicht von $\{v, w\}$ jeweils auf beide gerichteten Kanten (v, w) und (w, v) .



2.19 Abbildung. Transformation eines ungerichteten in einen gerichteten Graphen.

Im Rest dieser Sektion beschränken wir uns der Einfachheit halber auf den ungerichteten Fall. Die meisten (wenn auch nicht alle) der folgenden Definitionen übertragen sich in kanonischer Weise auf Digraphen oder allgemeine Graphen.

Wir betrachten zunächst noch einmal das Routenplanungsproblem aus Abbildung 2.4 und fragen danach, ob wir unser Modell vielleicht noch weiter reduzieren können.³² Offenbar sind die meisten Autobahnabfahrten für s - t -Wege ja irrelevant, da man hier ohnehin nicht auf andere Autobahnen wechseln kann. Wir können daher alle entsprechenden ‘Zwischenknoten’ wegzulassen und nur noch den kleineren Graphen in Abbildung 2.20 (rechts) zu verwenden.

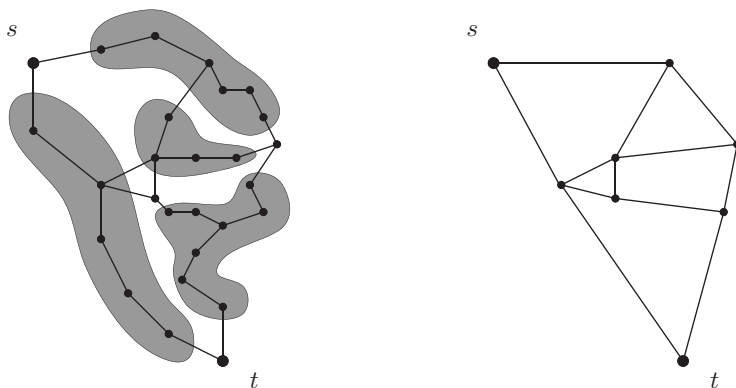
Er besteht aus s, t , allen Knoten, die Autobahndreiecke und -kreuze symbolisieren sowie Kanten, die die entsprechenden dazwischen liegenden Autobahnabschnitte repräsentieren. Graphentheoretisch spricht man von einer *Kontraktion*, da die in Abbildung 2.20 (links) grau unterlegten Teilgraphen jeweils zu einem einzigen Knoten zusammengezogen werden.

2.2.34 Definition. Seien $G := (V, E)$ ein Graph, $S := (U, F)$ ein Teilgraph von G mit $|U| \geq 1$ und

$$S_i := (U_i, F_i) \quad (i \in [k])$$

die Zusammenhangskomponenten von S . Ferner seien $Y := \{y_1, \dots, y_k\}$ eine zu $(V \setminus U) \cup E$ disjunkte Menge der Kardinalität k und

³² Wenn man schnell und effizient die ‘Größe des Modells’ verringern kann (‘preprocessing’), so reduziert sich in der Regel die Laufzeit nachfolgender Algorithmen. Für algorithmisch schwierige Probleme kann es durchaus hiervon abhängen, ob eine konkrete Aufgabe noch bewältigt werden kann oder praktisch nicht mehr lösbar ist.



2.20 Abbildung. Links: Der (um die ‘Sackgassen’ reduzierte) Graph G aus Abbildung 2.4. Grau unterlegt sind zusammenhängende Teilgraphen S_1, S_2, S_3, S_4 , die jeweils zu einer Ecke ‘zusammen gezogen’ werden sollen. Rechts: Der kontrahierte ‘Autobahngraph’.

$$\begin{aligned}
 K := & \left(\{ \{v, w\} : v, w \in V \setminus U \} \cap E \right) \\
 & \cup \bigcup_{\substack{i, j=1 \\ i \neq j}}^k \left\{ \{y_i, y_j\} : \exists (u_i \in U_i \wedge u_j \in U_j) : \{u_i, u_j\} \in E \right\} \\
 & \cup \bigcup_{i=1}^k \left\{ \{y_i, v\} : v \in V \setminus U \wedge \exists (u_i \in U_i) : \{u_i, v\} \in E \right\}.
 \end{aligned}$$

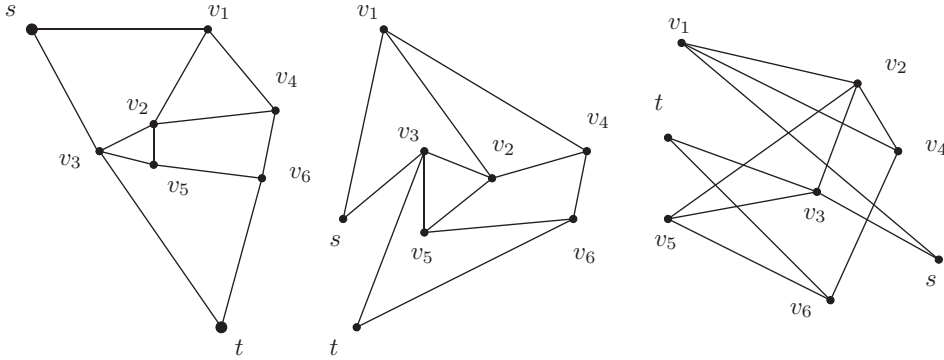
Dann heißt der Graph $((V \setminus U) \cup Y, K)$ **Kontraktion** von S in G und wird mit G/S bezeichnet.

Offenbar ist der in Abbildung 2.20 rechts dargestellte Graph die Kontraktion des grau unterlegten Teilgraphen des links angegebenen Graphen G .

Wir kommen nun noch einmal auf die Frage der Darstellung von Graphen zurück. Zwar sind Graphen rein kombinatorische Objekte, aber geometrische Darstellungen, in denen die Knoten bzw. Kanten durch Punkte bzw. sie verbindende Strecken in der Ebene repräsentiert werden, sind oft wesentlich intuitiver als kombinatorische Angaben. Allerdings können solche Darstellungen desselben Graphen sehr verschieden aussehen. Die drei Skizzen von Abbildung 2.21 geben tatsächlich isomorphe Graphen an; lediglich die *Einbettungen* in die Ebene sind verschieden. Die rechte Darstellung ist nicht einmal überschneidungsfrei, d.h. es gibt Schnittpunkte von (zu) Kanten (gehörenden Strecken), die keinem Knoten entsprechen. Man wird sie daher als ‘weniger intuitiv’ ansehen als die beiden anderen.³³

Aufgrund der passenden Beschriftungen der Knoten in Abbildung 2.21 kann man leicht überprüfen, dass es sich bei dem dargestellten kombinatorischen Objekt um denselben Graphen handelt. Ohne diese Identifikation entsprechender Knoten ist das jedoch schwieriger. Aufgrund ihrer einfacheren Struktur, kann die Isomorphie von Graphen statt

³³ Tatsächlich ist die Frage nach einem ‘guten Layout’ eines Graphen von großer Bedeutung für eine Vielzahl von Anwendungen, in denen etwa Planungen oder Abhängigkeitsstrukturen sichtbar gemacht werden sollen.



2.21 Abbildung. Drei Darstellungen des Autobahngraphen aus Abbildung 2.20 (rechts). Die Nummerierung der Knoten definiert Isomorphismen gemäß Bezeichnung 2.2.36.

mit Hilfe von Definition 2.2.6 allerdings auch mit Hilfe des folgenden Kriteriums festgestellt werden.

2.2.35 Lemma. Seien $G_1 := (V_1, E_1)$ und $G_2 := (V_2, E_2)$ Graphen. G_1 und G_2 sind (mit der Identifikation gemäß Bezeichnung 2.2.28) genau dann isomorph, wenn es eine bijektive Abbildung $\tau : V_1 \rightarrow V_2$ gibt, so dass für alle v, w aus V_1 gilt

$$\{v, w\} \in E_1 \iff \{\tau(v), \tau(w)\} \in E_2.$$

Beweis: ‘ \Rightarrow ’ Für $k = 1, 2$ sei $\nu_k : E_k \rightarrow \binom{V_k}{2}$ definiert durch

$$e \in E_k \wedge v, w \in V_k \wedge e = \{v, w\} \Rightarrow \nu_k(e) := \{v, w\}.$$

Ferner seien τ_{V_k} und τ_{E_k} Ordnungen auf V_k bzw. E_k mit $S_{G_1} = S_{G_2}$. Nach Definition der Inzidenzmatrizen folgt daher für $i = 1, \dots, |V_k|$ und $j = 1, \dots, |E_k|$

$$\tau_{V_1}(i) \in \nu_1(\tau_{E_1}(j)) \iff \tau_{V_2}(i) \in \nu_2(\tau_{E_2}(j)).$$

Mit

$$\tau := \tau_{V_2} \circ \tau_{V_1}^{-1}$$

folgt daher

$$\{v, w\} \in E_1 \iff \{\tau(v), \tau(w)\} \in E_2.$$

‘ \Leftarrow ’ Sei $\hat{\tau} : E_1 \rightarrow E_2$ für $e := \{v, w\}$ definiert durch

$$\hat{\tau}(e) := \{\tau(v), \tau(w)\}.$$

Seien nun τ_{V_1} bzw. τ_{E_1} beliebige Ordnungen auf V_1 bzw. E_1 und setze

$$\tau_{V_2} := \tau \circ \tau_{V_1} \quad \wedge \quad \tau_{E_2} := \hat{\tau} \circ \tau_{E_1}.$$

Dann sind τ_{V_2} bzw. τ_{E_2} Ordnungen auf V_2 bzw. E_2 , und es gilt

$$S_{G_1}(\tau_{V_1}, \tau_{E_1}) = S_{G_2}(\tau_{V_2}, \tau_{E_2}).$$

Insgesamt folgt damit die Behauptung. □

2.2.36 Bezeichnung. Seien $G_1 := (V_1, E_1)$, $G_2 := (V_2, E_2)$ Graphen und $\tau : V_1 \rightarrow V_2$ eine bijektive Abbildung mit

$$\{v, w\} \in E_1 \iff \{\tau(v), \tau(w)\} \in E_2$$

für alle $v, w \in V_1$. Dann heißt τ **Isomorphismus** und für $G_1 = G_2$ **Automorphismus**.

Im Allgemeinen ist es nicht klar, wie man ohne ‘Durchprobieren’ (wenigstens eines signifikanten Teils) aller möglicher Bijektionen zwischen den Knotenmengen von zwei gegebenen Graphen entscheiden kann, ob diese isomorph sind. Tatsächlich versteckt sich hinter dieser Frage eines der bekanntesten und wichtigsten offenen algorithmischen Probleme der Diskreten Mathematik; vgl. Forschungsproblem 3.4.27.

Wir führen nun einige Klassen von besonders wichtigen Graphen und Teilgraphen ein. Von zentraler Bedeutung für viele Probleme der kombinatorischen Optimierung sind die vollständigen Graphen, die untereinander genau dann isomorph sind, wenn ihre Knotenzahl übereinstimmt.

2.2.37 Definition. Sei $G := (V, E)$ ein Graph, und es gelte

$$v \in V \wedge w \in V \setminus \{v\} \implies \{v, w\} \in E.$$

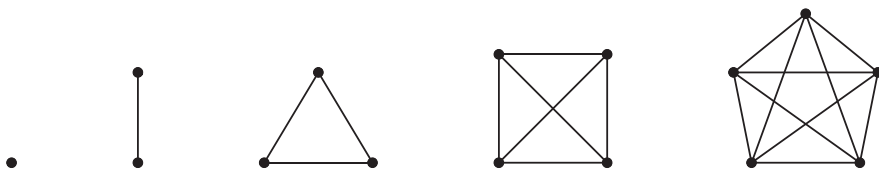
Dann heißt G **vollständig**. Der vollständige Graph auf der Knotenmenge V wird auch mit K_V bezeichnet. Interessiert nur die Isomorphieklasse und ist $n := |V|$, so schreibt man auch K_n und spricht von dem vollständigen Graphen auf n Knoten.

Ersetzt man die Bedingung durch $E := \{(v, w) : v \in V \wedge w \in V \setminus \{v\}\}$ so erhält man den **vollständigen gerichteten Graphen** auf V .

2.2.38 Bemerkung. Sei $G := (V, E)$ ein vollständiger Graph. Dann gilt $|E| = \binom{|V|}{2}$.

Beweis: Jede 2-elementige Teilmenge von V ist eine Kante; hiervon gibt es genau $\binom{|V|}{2}$. \square

2.2.39 Beispiel. Abbildung 2.22 skizziert die vollständigen Graphen auf 1 bis 5 Knoten.

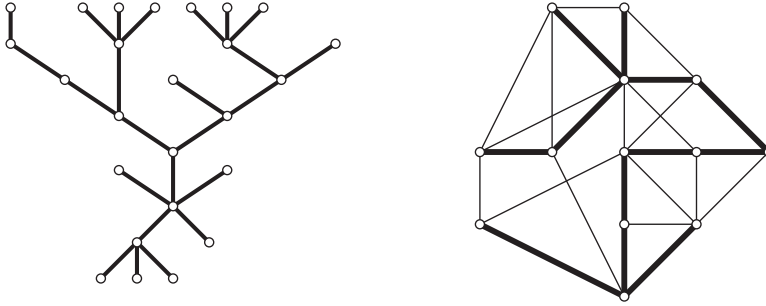


2.22 Abbildung. Darstellungen von K_1 bis K_5 .

Bäume und Wälder: Wir betrachten nun eine andere Klasse von speziellen Graphen, Bäume und Wälder. Sie treten direkt in der Praxis auf, etwa wenn Versorgungsleitungen eines gegebenen Netzes möglichst kostengünstig angemietet werden sollen. Bäume sind aber auch Grundlage für eine Vielzahl von effizienten (approximativen) Algorithmen für andere Probleme der Optimierung.

2.2.40 Definition. Sei $G := (V, E)$ ein Graph.

- (a) G heißt **Baum** [engl.: tree], wenn G zusammenhängend und kreisfrei ist.
- (b) Ist G kreisfrei, so heißt G **Wald** [engl.: forest].
- (c) Ist G ein Wald, so heißt jeder Knoten v von G mit $\deg(v) = 1$ **Blatt** [engl.: leaf].
- (d) Ein Teilgraph $T := (V_T, E_T)$ von G heißt **aufspannender Baum** oder **Spannbaum** [engl.: spanning tree], falls T ein Baum ist und $V_T = V$ gilt.



2.23 Abbildung. Links: ein Baum. Rechts: ein Graph mit (fett eingezeichnetem) Spannbaum.

Außer in trivialen Fällen besitzen Bäume Blätter.

2.2.41 Bemerkung. Sei $G := (V, E)$ ein Baum mit $|E| \geq 1$. Dann besitzt G mindestens zwei Blätter.

Beweis: Sei W ein Weg in G maximaler (kombinatorischer) Länge. Dann sind die Endknoten von W Blätter von G . □

Die folgenden Aussagen enthalten nützliche Charakterisierungen von Bäumen.

2.2.42 Lemma. Sei $G := (V, E)$ ein Graph. Dann sind die folgenden Aussagen äquivalent:

- (a) G ist ein Baum.
- (b) Für je zwei Knoten $v, w \in V$ existiert genau ein v - w -Weg in G .
- (c) G ist zusammenhängend, aber für kein $e \in E$ ist $(V, E \setminus \{e\})$ zusammenhängend.
- (d) G ist kreisfrei, aber für alle $v, w \in V$ mit $v \neq w$ und $e := \{v, w\} \notin E$ enthält $(V, E \cup \{e\})$ einen Kreis.

Beweis: ‘(a) \Rightarrow (b)’ Da G ein Baum ist, existiert für je zwei Knoten $v, w \in V$ mindestens ein v - w -Weg in G . Gäbe es zwei verschiedene v - w -Wege, so enthielte G einen Zyklus und nach Bemerkung 2.2.10 (b) einen Kreis.

‘(b) \Rightarrow (c)’ Wäre $(V, E \setminus \{e\})$ für ein $e \in E$ zusammenhängend, so wären die beiden zu e gehörenden Knoten durch zwei verschiedene Wege in G miteinander verbunden.

‘(c) \Rightarrow (d)’ Enthielte G einen Kreis, so könnte man eine beliebige seiner Kanten entfernen, ohne den Zusammenhang zu verlieren. Wäre für ein Paar $v, w \in V$ mit $v \neq w$ und $e := \{v, w\} \notin E$ der Graph $(V, E \cup \{e\})$ kreisfrei, so wären v und w in G nicht verbunden.

‘(d) \Rightarrow (a)’ Enthielte G zwei verschiedene Zusammenhangskomponenten S_1 und S_2 , so könnte zwischen einem beliebigen Knoten aus S_1 und einem beliebigen Knoten aus S_2 eine Kante eingefügt werden, ohne, dass ein Kreis entsteht. \square

2.2.43 Korollar. Sei $G := (V, E)$ ein Baum mit $|V| \geq 1$. Dann gilt $|E| = |V| - 1$.

Beweis: Der Beweis wird mittels vollständiger Induktion über die Knotenzahl $|V|$ geführt. Für $|V| = 1$ ist nichts zu zeigen. Seien also $G := (V, E)$ ein Baum mit $|V| \geq 2$, v ein Blatt von G und e eine v enthaltende Kante von G . Dann ist nach Lemma 2.2.42 (b) $(V \setminus \{v\}, E \setminus \{e\})$ ein Baum, und es folgt nach Induktionsvoraussetzung

$$|E| = |E \setminus \{e\}| + 1 = |V \setminus \{v\}| = |V| - 1,$$

und damit die Behauptung. \square

2.2.44 Korollar. Sei $G := (V, E)$ ein Graph mit $|V| \geq 1$. Dann sind die folgenden Aussagen äquivalent:

- (a) G ist ein Baum.
- (b) G ist kreisfrei, und es gilt $|E| = |V| - 1$.
- (c) G ist zusammenhängend, und es gilt $|E| = |V| - 1$.

Beweis: ‘(a) \Rightarrow (b) \wedge (c)’ folgt direkt aus Korollar 2.2.43.

‘(b) \Rightarrow (a)’ Sei $T := (V, E_T)$ ein kantenmaximaler kreisfreier Graph, der G als Teilgraph enthält. Nach Lemma 2.2.42 (d) ist T ein Baum, und aus Korollar 2.2.43 folgt $|E_T| = |V| - 1$, also $E = E_T$ und damit $G = T$.

‘(c) \Rightarrow (a)’ Sei $T := (V, E_T)$ ein kantenminimaler zusammenhängender Teilgraph von G . Nach Lemma 2.2.42 (c) ist T ein Baum, und aus Korollar 2.2.43 folgt $|E_T| = |V| - 1 = |E|$, also $G = T$. \square

2.2.45 Korollar. Sei $G := (V, E)$ ein zusammenhängender Graph. Dann besitzt G einen Spannbaum.

Beweis: Sei $T := (V, E_T)$ ein zusammenhängender Teilgraph minimaler Kantenzahl. Dann ist T nach Lemma 2.2.42 (c) ein Baum, somit ein Spannbaum. \square

Matchings und der Satz von Hall: Wir kommen nun zu einer anderen wichtigen Struktur, den Matchings. Zu ihren praktischen Anwendungsfeldern gehören Probleme wie die Einteilung von Arbeitskräften (Manpower-Planning), die Chromosomen-Klassifikation, die Flugverkehrskontrolle, das Job Scheduling auf parallelen Maschinen und viele andere.

Daneben treten Matchings auch als ‘Substruktur’ bei der Behandlung zahlreicher anderer Probleme der kombinatorischen Optimierung auf.

2.2.46 Definition. Seien $G := (V, E)$ ein Graph und $M \subset E$.

(a) Sei $v \in V$. Der Knoten v wird von M **überdeckt**, wenn

$$v \in \bigcup_{e \in M} e$$

gilt. M heißt **Knotenüberdeckung** von G , wenn jeder Knoten aus V von M überdeckt wird.

(b) M heißt **Paarung** oder **Matching**, wenn gilt

$$\{v_1, w_1\} \in M \wedge \{v_2, w_2\} \in M \wedge \{v_1, w_1\} \cap \{v_2, w_2\} \neq \emptyset \implies \{v_1, w_1\} = \{v_2, w_2\}.$$

(c) M heißt **perfektes Matching**, wenn M ein Matching und eine Knotenüberdeckung von G ist.

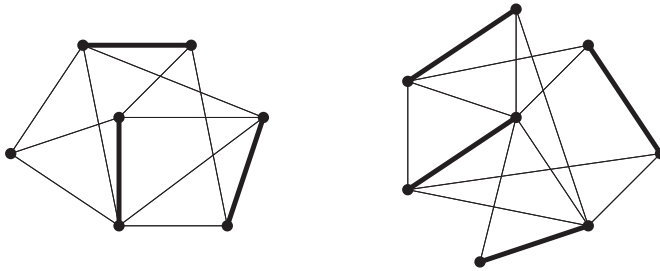
(d) Ist $\phi : E \rightarrow \mathbb{R}$ und ist M^* ein Matching bzw. perfektes Matching in G , so heißt M^* genau dann ein (bez. ϕ) **maximales Matching** bzw. **maximales perfektes Matching**, wenn M^* den Wert

$$\sum_{e \in M} \phi(e)$$

unter allen Matchings bzw. perfekten Matchings in G maximiert.

Sind speziell $\phi \equiv 1$ und M^* ein bez. ϕ maximales Matching, so heißt M^* ein **kardinalitätsmaximales Matching**.

2.2.47 Beispiel. Die beiden Graphen G_1 und G_2 seien wie in Abbildung 2.24 gegeben. Die hervorgehobenen Kanten geben jeweils Matchings an. Beide sind kardinalitätsmaximal, das rechts abgebildete ist perfekt.



2.24 Abbildung. Kardinalitätsmaximale Matchings in Graphen G_1 (links) und G_2 (rechts). Das Matching in G_2 ist perfekt.

Von besonderer Bedeutung für Anwendungen, aber auch wegen ihrer (später noch genauer analysierten) besonders ‘gutartigen Struktur’ sind Matchings in der folgenden eingeschränkten Klasse von Graphen.

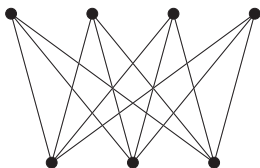
2.2.48 Definition. Ein Graph $G := (V, E)$ heißt **bipartit**, wenn es eine **zugehörige Partition** $\{V_1, V_2\}$ von V gibt mit

$$\{v, w\} \in E \implies \{v, w\} \cap V_1 \neq \emptyset \wedge \{v, w\} \cap V_2 \neq \emptyset.$$

Sind G bipartit, $\{V_1, V_2\}$ eine zugehörige Partition und gilt

$$v_1 \in V_1 \wedge v_2 \in V_2 \implies \{v_1, v_2\} \in E,$$

so heißt G **vollständig bipartiter Graph**. Mit $n_1 := |V_1|$ und $n_2 := |V_2|$ wird er (bzw. seine Isomorphieklasse) mit K_{n_1, n_2} bezeichnet.



2.25 Abbildung. Darstellung des vollständigen bipartiten Graphen $K_{4,3}$.

2.2.49 Beispiel. In dem schon in Sektion 1.2 beschriebenen Beispiel der Zuordnung von Personen P_1, P_2, P_3, P_4 vom Geschlecht Γ_1 zu Personen Q_1, Q_2, Q_3 vom Geschlecht Γ_2 mit $\Gamma_1 \neq \Gamma_2$ zur Maximierung der Summe der Kompatibilitätskoeffizienten $\gamma_{i,j}$ kann man die Personen vom Geschlecht Γ_1 bzw. Γ_2 als Knotenmenge $V_1 := \{v_1, v_2, v_3, v_4\}$ bzw. $V_2 := \{v_5, v_6, v_7\}$ auffassen. Wird für jedes (getrennt geschlechtliche) Paar eine Kante gebildet, so erhält man den vollständigen bipartiten Graphen $K_{4,3}$; vgl. Abbildung 2.25. Durch $\phi(\{v_i, v_j\}) := \gamma_{i,j}$ für $i = 1, 2, 3, 4$ und $j = 5, 6, 7$ wird eine Funktion ϕ auf den Kanten von $K_{4,3}$ definiert. Gesucht ist also ein bez. ϕ maximales Matching in $K_{4,3}$.

Das folgende Lemma gibt eine Charakterisierung bipartiter Graphen mittels Kreisen.

2.2.50 Lemma. Sei $G := (V, E)$ ein Graph. G ist genau dann bipartit, wenn G keinen Kreis ungerader Länge enthält.

Beweis: ‘ \Rightarrow ’ Sei (V_1, V_2) eine Knotenpartition, so dass die induzierten Teilgraphen G_{V_1} und G_{V_2} keine Kanten enthalten. Jeder Kreis muss gleich viele Knoten in V_1 wie in V_2 enthalten, also gerade Länge besitzen.

‘ \Leftarrow ’ Offenbar reicht es, die Aussage für Zusammenhangskomponenten zu beweisen. Sei also G zusammenhängend. Nach Korollar 2.2.45 besitzt G einen Spannbaum; sei $T := (V, E_T)$ ein solcher. Ferner sei $s \in V$. Nach Lemma 2.2.42 (b) gibt es zu jedem Knoten $v \in V$ genau einen s - v -Weg $W(v)$ in T . Sei $p(v)$ seine Länge, und für $k = 0, 1$ sei

$$V_{k+1} := \{v \in V : p(v) \equiv k \pmod{2}\}.$$

Dann ist $\{V_1, V_2\}$ eine Partition von V . Natürlich liegen die beiden Knoten jeder Kante von T in verschiedenen Mengen der Partition, und es bleibt zu zeigen, dass das auch für die restlichen Kanten von G gilt. Seien also $v, w \in V$ mit $e := \{v, w\} \in E \setminus E_T$. Nach Lemma 2.2.42 (d) enthält $(V, E_T \cup \{e\})$ einen Kreis, und nach Voraussetzung hat dieser gerade Länge. Somit haben der s - v -Weg in T und der s - w -Weg in T unterschiedliche Länge modulo 2, d.h. v und w liegen in verschiedenen Mengen der Partition.

Insgesamt folgt damit die Behauptung. \square

In Beispiel 2.2.49 existiert kein perfektes Matching. Das liegt natürlich daran, dass es mehr Personen vom Geschlecht Γ_1 als vom Geschlecht Γ_2 gibt. Allgemeiner ist es für die Existenz eines perfekten Matchings in einem bipartiten Graph G mit zugehöriger Knotenpartition (V_1, V_2) offenbar notwendig, dass es kein $k \in \{1, 2\}$ und keine Teilmenge U_k von V_k gibt, zu der weniger als $|U_k|$ Knoten benachbart sind. Tatsächlich ist diese Bedingung der ‘Reichhaltigkeit der Nachbarschaft’ sogar hinreichend.

2.2.51 Satz. (Satz von Hall³⁴)

Seien $G := (V, E)$ ein bipartiter Graph und $\{Y, Z\}$ eine zugehörige Partition von V . Dann sind die folgenden beiden Aussagen äquivalent:

- (a) Es gibt ein Matching M in G mit $|M| = |Y|$.
- (b) Für jede Teilmenge U von Y gilt $|N(G, U)| \geq |U|$.

Beweis: Da die Richtung ‘(a) \Rightarrow (b)’ trivial ist, bleibt nur die Umkehrung ‘(b) \Rightarrow (a)’ zu zeigen. Wir führen einen Widerspruchsbeweis.

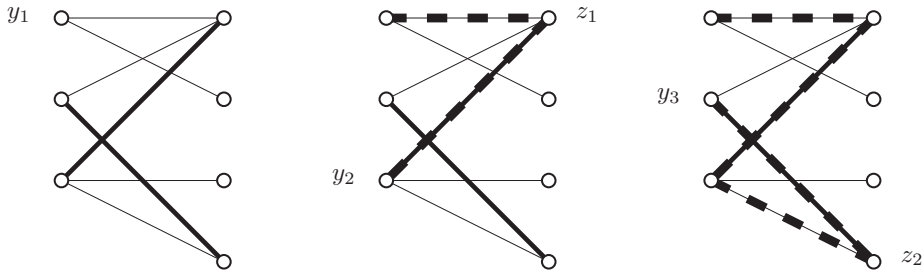
Sei M ein kardinalitätsmaximales Matching in G , aber es gelte $|M| < |Y|$. Sei y_1 ein Knoten von Y , der in keiner Kante aus M enthalten ist. Mit $Y_1 := \{y_1\}$ gilt nach Voraussetzung $|N(G, Y_1)| \geq |Y_1| = 1$. Seien

$$z_1 \in N(G, Y_1) \quad \wedge \quad Z_1 := \{z_1\} \quad \wedge \quad e_1 := \{y_1, z_1\} \quad \wedge \quad E_1 := \{e_1\}.$$

Wäre z_1 in keiner Kante von M enthalten, so wäre $M \cup \{e_1\}$ ein um eine Kante größeres Matching, im Widerspruch zur Maximalität von M . Seien also

$$y_2 \in Y \setminus Y_1 \quad \wedge \quad m_1 := \{y_2, z_1\} \in M \quad \wedge \quad Y_2 := Y_1 \cup \{y_2\} \quad \wedge \quad M_1 := \{m_1\}.$$

Nach Voraussetzung gilt $|N(G, Y_2)| \geq |Y_2| = 2 > 1 = |Z_1|$. Seien daher $z_2 \in Z \setminus Z_1$, $Z_2 := Z_1 \cup \{z_2\}$ und $e_2 \in \delta(G, Y_2)$ mit $z_2 \in e_2$ und $E_2 := E_1 \cup \{e_2\}$. Da y_1 nicht von M überdeckt wird und $y_2 \in m_1$ gilt, folgt $e_2 \notin M$.



2.26 Abbildung. Links: Bipartiter Graph G mit Matching M , nicht überdeckter Knoten y_1 . Mitte: 1. Schritt: Konstruktion von z_1 und y_2 . Rechts: 2. Schritt.

Induktiv setzen wir nun die Konstruktion solange wie möglich fort. Nach Voraussetzung gilt stets

$$|N(G, Y_i)| \geq |Y_i| = i > i - 1 = |Z_{i-1}|.$$

Somit muss der letzte hinzukommende Knoten z_k in Z liegen. Die Konstruktion bricht also mit den Knotenmengen

³⁴ Philip Hall, 1904 – 1982.

$$Y_k \subset Y \quad \wedge \quad Z_k \subset Z$$

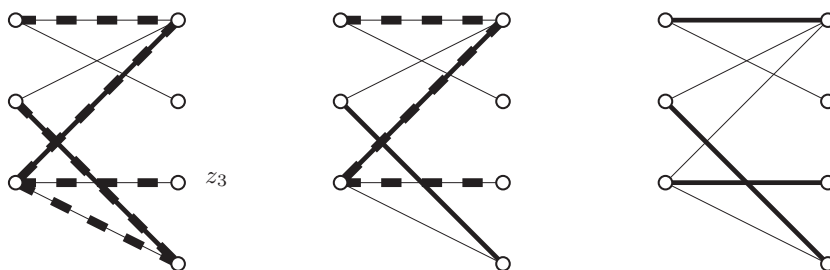
und den Kantenmengen

$$E_k \subset E \setminus M \quad \wedge \quad M_{k-1} \subset M$$

ab. Sei

$$T := (Y_k \cup Z_k, E_k \cup M_{k-1}).$$

Nach Konstruktion ist T zusammenhängend und kreisfrei, also ein Baum.



2.27 Abbildung. Links: Mit z_3 bricht die Konstruktion ab. Die gestrichelten Kanten bilden den Baum T . Mitte: y_1 - z_3 -Weg W . Rechts: Vergrößertes, maximales Matching M^* .

Seien W der nach Lemma 2.2.42 (b) eindeutig bestimmte y_1 - z_k -Weg in T , E_W die Menge seiner Kanten sowie $M_W := E_W \cap M$ und $N_W := E_W \setminus M$. Da W nicht mehr verlängert werden kann, liegt z_k in keiner Kante von M . Setzt man daher

$$M^* := (M \setminus M_W) \cup N_W,$$

so ist M^* ein Matching in G mit

$$|M^*| = |M| + 1,$$

im Widerspruch zur Maximalität von M . Somit folgt $|M| = |Y|$ und damit die Behauptung. \square

Als Folgerung aus Satz 2.2.51 erhält man ein Kriterium für die Existenz perfekter Matchings.

2.2.52 Korollar. (Heiratssatz)

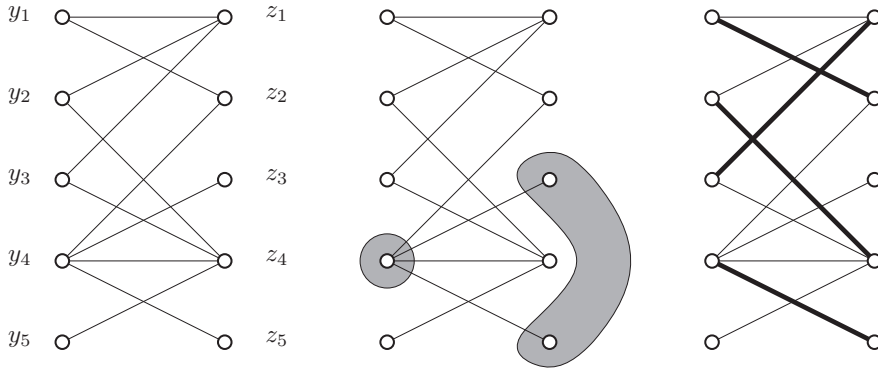
Seien $G := (V, E)$ ein bipartiter Graph und $\{V_1, V_2\}$ eine zugehörige Partition von V . Es gibt genau dann ein perfektes Matching in G , wenn für $i = 1, 2$ gilt

$$U_i \subset V_i \implies |N(G, U_i)| \geq |U_i|.$$

Beweis: ‘ \implies ’ folgt direkt aus Satz 2.2.51.

‘ \Leftarrow ’ Nach Satz 2.2.51 gibt es für $i = 1, 2$ ein Matching M_i mit $|M_i| = |V_i|$. Es folgt $|V_1| \leq |V_2|$ sowie $|V_1| \geq |V_2|$, also $|V_1| = |V_2|$, und die Kanten von M_i überdecken ganz V . \square

2.2.53 Beispiel. Sei G der in Abbildung 2.28 (links) gegebene bipartite Graph. Besitzt G ein perfektes Matching? Sei $Z := \{z_3, z_5\}$. Dann gilt $N(G, Z) = \{y_4\}$. Somit ist



2.28 Abbildung. Links: Bipartiter Graph G . Mitte: zweielementige Knotenmenge in Z und zugehörige einelementige Menge $N(G, Z)$. Rechts: kardinalitätsmaximales Matching.

$$2 = |Z| > |N(G, Z)| = 1,$$

und nach dem Heiratssatz 2.2.52 muss mindestens ein Knoten von Z ‘ungepaart’ bleiben.

Es gibt aber ein Matching M der Kardinalität 4; es ist in Abbildung 2.28 (rechts) angegeben.

Im Beweis von Satz 2.2.51 war der konstruierte Weg W von zentraler Bedeutung, da man mit seiner Hilfe ein gegebenes Matching vergrößern konnte. Solche Wege führen auch in allgemeinen Graphen zur Charakterisierung kardinalitätsmaximaler Matchings.³⁵

2.2.54 Definition. Seien $G := (V, E)$ ein Graph, M ein Matching, W ein eigentlicher Weg in G mit Kantenmenge E_W und $A := E_W \setminus M$. W heißt **alternierend** bez. M , wenn A ein Matching in G ist. W heißt **Augmentationsweg**³⁶ bez. M in G , wenn W alternierend ist und seine Endknoten nicht von M überdeckt werden.

2.2.55 Bemerkung. Seien $G := (V, E)$ ein Graph, M ein Matching, W ein Augmentationsweg in G mit Kantenmenge E_W und $A := E_W \setminus M$. Ferner sei

$$M^* := (M \setminus E_W) \cup A.$$

Dann ist M^* ein Matching mit $|M^*| = |M| + 1$.

Beweis: Da die beiden Endknoten von W nicht von M überdeckt werden und jeder innere Knoten in genau einer Kante von M liegt, ist kein Knoten von G in mehr als einer Kante von M^* enthalten; M^* ist also ein Matching. Aus $|A| - |M \cap E_W| = 1$ folgt die Behauptung. \square

2.2.56 Bezeichnung. Das Matching M^* gemäß Bemerkung 2.2.55 wird als (*mittels W*) **augmentiertes Matching** bezeichnet.

Nach Bemerkung 2.2.55 besitzt ein kantenmaximales Matching keinen Augmentationsweg. Tatsächlich gilt auch die Umkehrung, und wir erhalten die folgende Charakterisierung.

³⁵ Entsprechend gewichtete Verallgemeinerungen werden später ebenfalls von Bedeutung sein.

³⁶ augmen (lat.) Vermehrung, Zuwachs.

2.2.57 Lemma. Seien $G := (V, E)$ ein Graph und M ein Matching in G . Das Matching M ist genau dann kardinalitätsmaximal, wenn es keinen Augmentationsweg bez. M in G gibt.

Beweis: Die Aussage ‘ \Rightarrow ’ folgt aus Bemerkung 2.2.55.

‘ \Leftarrow ’ Sei M ein nicht kardinalitätsmaximales Matching in G . Dann gibt es ein Matching größerer Kardinalität in G . Sei M^* ein solches. Seien

$$F := (M \setminus M^*) \cup (M^* \setminus M) \quad \wedge \quad U := \bigcup_{e \in F} e.$$

Dann ist $S := (U, F)$ ein nichtleerer Teilgraph von G , und es gilt

$$u \in U \quad \Rightarrow \quad \deg_S(u) \leq 2.$$

Die Zusammenhangskomponenten von S sind also Wege und Kreise, wobei die Kanten von M und M^* jeweils alternieren. Die Kreise enthalten also gleichviele Kanten von M und M^* ; bei den Wegen ist die Differenz höchstens 1. Wegen $|M^*| \geq |M| + 1$ gibt es einen Weg in S , der eine Kante mehr aus M^* enthält als aus M ; dieser ist ein gesuchter Augmentationsweg. \square

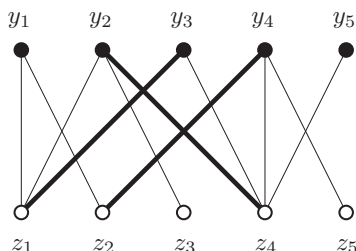
Prinzipiell kann man bereits mit Lemma 2.2.57 kardinalitätsmaximale Matchings konstruieren. Man starte mit einem beliebigen (etwa dem leeren) Matching und vergrößere dieses sukzessive mittels eines Augmentationsweges. Ist also M das aktuelle Matching, so ist G systematisch nach potentiellen Augmentationswegen abzusuchen, um M zu verbessern oder nachzuweisen, dass M bereits optimal ist. Nach Übungsaufgabe 2.4.11 können bipartite Graphen allerdings exponentiell viele Wege enthalten, die man im Allgemeinen nicht alle ‘durchsuchen’ kann. Man kann jedoch die Suche effizient strukturieren. Betrachten wir zunächst ein Beispiel.

2.2.58 Beispiel. Sei $G := (V, E)$ der bipartite Graph aus Abbildung 2.29. Die Knoten sind in die Mengen

$$Y := \{y_1, y_2, y_3, y_4, y_5\} \quad \wedge \quad Z := \{z_1, z_2, z_3, z_4, z_5\}$$

partitioniert. Gegeben sei ferner das Matching

$$M := \left\{ \{y_2, z_4\}, \{y_3, z_1\}, \{y_4, z_2\} \right\}.$$



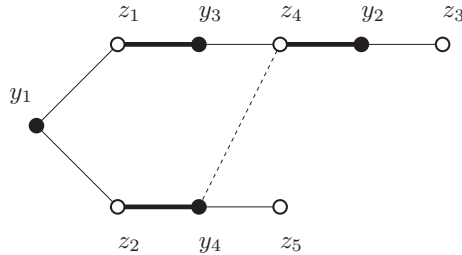
2.29 Abbildung. Bipartiter Graph G mit Matching M .

Offenbar ist M nicht kardinalitätsmaximal, und wir suchen (ganz analog zum Beweis von Satz 2.2.51) systematisch einen Augmentationsweg. Da y_1 von M nicht überdeckt wird, fangen wir mit y_1 an.

Jeder der in y_1 beginnenden potentiellen Augmentationswege startet alternativ mit der Kante $\{y_1, z_1\}$ oder der Kante $\{y_1, z_2\}$. Da sowohl z_1 als auch z_2 in Kanten aus M enthalten sind, werden als nächstes die Matchingkanten $\{y_3, z_1\}$ bzw. $\{y_4, z_2\}$ durchlaufen. Von y_3 kann man über $\{y_3, z_4\}$ weiter laufen; danach schließt sich die Matchingkante $\{y_2, z_4\}$ an. Von y_2 aus kommt man zu z_3 und erhält den Augmentationsweg

$$W_1 := (y_1, z_1, y_3, z_4, y_2, z_3).$$

(Von y_2 könnte man prinzipiell auch in den Knoten z_1 ‘zurücklaufen’. Diese Möglichkeit braucht aber nach dem Beweis von Satz 2.2.51 nicht berücksichtigt zu werden und führt auch zu keinem Augmentationsweg.)



2.30 Abbildung. Die nicht gestrichelten Kanten bilden einen zu y_1 gehörenden ‘alternierenden Baum’ bez. M . Die gestrichelte Kante entfällt, weil in der verwendeten ‘Suchreihenfolge’ ein Endknoten bereits erfasst war, bevor sie an die Reihe kam.

Geht man über z_2 , so spaltet sich der Teilgraph in y_4 erneut auf: wir können im Prinzip über jede der beiden Kanten $\{y_4, z_4\}$ oder $\{y_4, z_5\}$ weiterlaufen. Der vorher noch nicht erreichte Knoten z_5 wird nicht von M überdeckt; also ist

$$W_2 := (y_1, z_2, y_4, z_5)$$

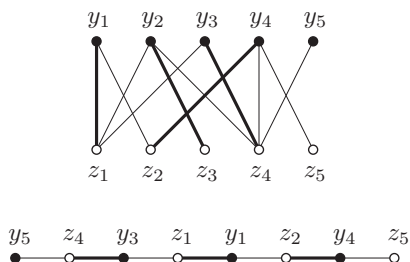
ein Augmentationsweg; vgl. Abbildung 2.30. Die (gestrichelt gezeichnete) Kante $\{y_4, z_4\}$ hingegen erreicht keinen neuen Knoten, da wir zu z_4 ja bereits über z_1 gelangt waren. Aber es gibt durchaus einen Augmentationsweg, der diese Kante enthält, nämlich

$$W_3 := (y_1, z_2, y_4, z_4, y_2, z_3).$$

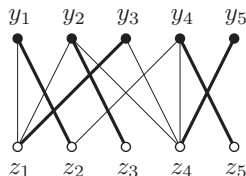
Da der Kreis $(y_1, z_2, y_4, z_4, y_3, z_1, y_1)$ sechs Kanten besitzt, also gerade Länge hat (wie sich das für Kreise in bipartiten Graphen gehört), besitzen beide y_1 - z_4 -Wege gerade oder beide ungerade Länge. Da sie alternierend sind und y_1 nicht überdeckt ist, muss ihre Länge ungerade sein; hier ist sie jeweils 3. Die Fortsetzung über z_4 hinaus hängt also nicht davon ab, welcher der beiden alternierenden Wege zu z_4 durchlaufen wurde. Die Nichtberücksichtigung der Kante $\{y_4, z_4\}$ bei der Suche nach Augmentationswegen verringert zwar die Anzahl der ‘findbaren’ Augmentationswege, hat aber keinen Einfluss darauf, ob unsere Suche überhaupt erfolgreich ist. Man kann sich also auf einen Baum beschränken. Da y_1 nach einer Augmentierung bereits überdeckt ist, kommt es auch gar nicht auf die genaue Anzahl der von y_1 ausgehenden Augmentationswege an.

Wählen wir zur Augmentation W_1 , so erhalten wir das in Abbildung 2.31 (oben) dargestellte augmentierte Matching M' . Es ist noch nicht optimal, da der unten dargestellte ‘Augmentationsbaum’ noch einen Augmentationsweg enthält (ja tatsächlich selbst ein solcher ist).

Das entsprechend augmentierte Matching M^* ist in Abbildung 2.32 skizziert. Es ist perfekt, also natürlich auch kardinalitätsmaximal.



2.31 Abbildung. Matching M' und Augmentationsweg.



2.32 Abbildung. Optimales Matching M^* .

Tatsächlich kann man mit der exemplifizierten Methode kardinalitätsmaximale Matchings in bipartiten Graphen G effizient finden. Zentral ist dabei die in Beispiel 2.2.58 festgestellte Eigenschaft, dass zur Verbesserung nur sukzessiv konstruierte Teilbäume von G betrachtet zu werden brauchen, da es wegen der Bipartitheit nur darauf ankommt, *ob* man einen Knoten erreicht, nicht aber *wie*, also auf welchen Wegen, man ihn erreichen kann. Genauer gilt die folgende Aussage.

2.2.59 Lemma. Seien $G := (V, E)$ ein bipartiter Graph, $v_1 := w_1 := v \in V$, $p, q \in \mathbb{N}$, $v_p := w_q := w \in V$ und $W_1 := (v_1, \dots, v_p)$, $W_2 := (w_1, \dots, w_q)$ zwei v - w -Wege in G . Sind $u \in V$, $i \in [p]$ und $j \in [q]$ mit $u = v_i = w_j$, so gilt $i \equiv j \pmod{2}$.

Beweis: Wir führen einen Widerspruchsbeweis. Sei (G, v, w, W_1, W_2) ein in folgendem Sinn minimales Gegenbeispiel: W_1 und W_2 sind Wege, für die die Behauptung nicht gilt, mit minimaler symmetrischer Differenz der Kantenmengen.

Sei S der aus allen Knoten und Kanten von W_1 und W_2 gebildete Teilgraph von G . Dann enthält S einen Kreis K . Die Kanten von W_1 in K bzw. W_2 in K bilden jeweils einen Weg $W_1(K)$ und $W_2(K)$. Da G bipartit ist, hat K gerade Länge, d.h. die Paritäten der Kantenzahlen von $W_1(K)$ und $W_2(K)$ stimmen überein. S kann nicht selbst der Kreis K sein, da sonst kein Gegenbeispiel vorläge. Sei W'_1 der aus W_1 durch Ersetzung von $W_1(K)$ durch $W_2(K)$ entstehende Weg. Dann ist W'_1, W_2 noch immer ein Gegenbeispiel, allerdings mit kleinerer symmetrischer Differenz der Kantenmengen. Das ist ein Widerspruch zur Voraussetzung, und des folgt die Behauptung. \square

Mit Lemma 2.2.59 ist es möglich, die Anzahl der zu betrachtenden Wege dramatisch zu reduzieren und sich tatsächlich auf Bäume zu beschränken. (Man vergleiche auch noch einmal die Konstruktion im Beweis von Satz 2.2.51.)

2.2.60 Definition. Seien $G := (V, E)$ ein bipartiter Graph mit zugehöriger Partition $\{Y, Z\}$ von V , $y \in Y$ und M ein Matching in G , das y nicht überdeckt. Sei $T := (V_T, E_T)$ ein (inklusions-) maximaler Baum in G mit $y \in V_T$, so dass jeder Weg in T mit Endknoten y alternierend ist. Dann heißt T **Augmentationsbaum** zu y in G .

Das folgende Korollar zeigt, dass man sich in bipartiten Graphen bei der Suche nach Augmentationswegen durch einen nicht überdeckten Knoten y auf einen beliebigen Augmentationsbaum zu y in G beschränken kann.

2.2.61 Korollar. *Seien $G := (V, E)$ ein bipartiter Graph mit zugehöriger Partition $\{Y, Z\}$ von V , $y \in Y$, M ein Matching in G , das y nicht überdeckt, und $T := (V_T, E_T)$ ein Augmentationsbaum zu y in G . Es gibt genau dann einen Augmentationsweg bez. M in G mit Endknoten y , wenn es einen solchen in T gibt.*

Beweis: Die Richtung ‘ \Leftarrow ’ ist klar. Zum Beweis von ‘ \Rightarrow ’ sei W ein Augmentationsweg in G mit Endknoten y und z . Wir zeigen zunächst mittels Widerspruchsbeweises, dass alle Knoten von W in T liegen. Seien w der von y aus erste Knoten aus W , der nicht in T liegt, v sein Vorgänger und $e := \{v, w\}$. Nach Lemma 2.2.59 hat w in W gleichen Abstand modulo 2 von y wie in T . Der alternierende y - v -Weg in T kann somit mittels e verlängert werden, im Widerspruch zur Maximalität von T . Somit liegen tatsächlich alle Knoten von W in T , insbesondere also z .

Sei W_T der eindeutig bestimmte y - z -Weg in T . Da W ein Augmentationsweg ist, ist nach Lemma 2.2.59 auch W_T ein solcher, und es folgt die Behauptung. \square

Insgesamt erhält man nun folgenden Algorithmus zur Konstruktion kardinalitätsmaximaler Matchings in bipartiten Graphen.

2.2.62 Prozedur: *Kardinalitätsmaximales Matching in bipartiten Graphen (Grundstruktur).*

```

INPUT:   Bipartiter Graph  $G := (V, E)$ , zugehörige Partition  $\{Y, Z\}$  von  $V$ 
OUTPUT:  Kardinalitätsmaximales Matching  $M$  in  $G$ 
BEGIN     $M \leftarrow \emptyset$ ;  $U \leftarrow Y$ 
          BEGIN
            WHILE  $U \neq \emptyset$  DO
              BEGIN
                Wähle  $y \in U$ 
                Konstruiere einen Augmentationsbaum  $T := (V_T, E_T)$  zu  $y$  in  $G$ 
                IF  $T$  enthält einen Augmentationsweg  $W$ 
                  THEN  $M \leftarrow$  mittels  $W$  augmentiertes Matching;  $U \leftarrow U \setminus \bigcup_{e \in M} e$ 
                  ELSE  $U \leftarrow U \setminus \{y\}$ 
                END
              END
            END
          END
        END
      END
    END
  END
END

```

2.2.63 Satz. *Sei $G := (V, E)$ ein bipartiter Graph. Dann findet Prozedur 2.2.62 ein kardinalitätsmaximales Matching.*

Beweis: Da jeder Augmentationsweg einen Endknoten in Y und einen in Z haben muss, ist die im Algorithmus vorgenommene Einschränkung auf Y irrelevant.

Sei y nicht überdeckt. Eine Augmentation führt dazu, dass y überdeckt wird. Alle vorher überdeckten Knoten bleiben ferner überdeckt.

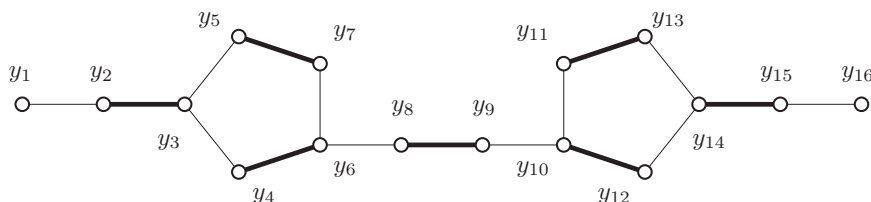
Nach Korollar 2.2.61 findet der Algorithmus solange sukzessive Augmentationswege, bis kein solcher mehr in G existiert. Nach Lemma 2.2.57 ist das gefundene Matching dann aber maximal. \square

Prozedur 2.2.62 benötigt höchstens $|Y|$ Schritte. Der Rechenaufwand pro Schritt besteht im Wesentlichen in der Konstruktion eines Augmentationsbaums. Das kann man einfach als eine spezielle ‘Suche’ in G organisieren, nämlich als ‘Breitensuche’. Sektion 3.1 geht näher auf Suchstrategien in Graphen ein.

Der Algorithmus kann noch in mehrfacher Hinsicht ‘verkürzt’ werden. Falls es etwa zu einem Knoten y keinen Augmentationsweg gibt, so muss jeder Augmentationsweg in G knotendisjunkt zu jedem Augmentationsbaum zu y sein; vgl. Übungsaufgabe 2.4.22. Das Verfahren kann ferner eleganter (wenn auch nicht wirklich übersichtlicher) gefasst werden, wenn man ausnutzt, dass die Matchingkanten in Augmentationsbäumen ja keine Alternativen erlauben, also eigentlich kontrahiert werden können.

Zum Abschluss dieser Sektion soll noch gezeigt werden, dass das ‘Fehlen’ von Lemma 2.2.59 im Fall nicht bipartiter Graphen die Suche nach Augmentationswegen wesentlich komplizierter macht.

2.2.64 Beispiel. Seien $G := (V, E)$ der in Abbildung 2.33 dargestellte Graph mit dem aus den fett gezeichneten Kanten bestehenden Matching M .

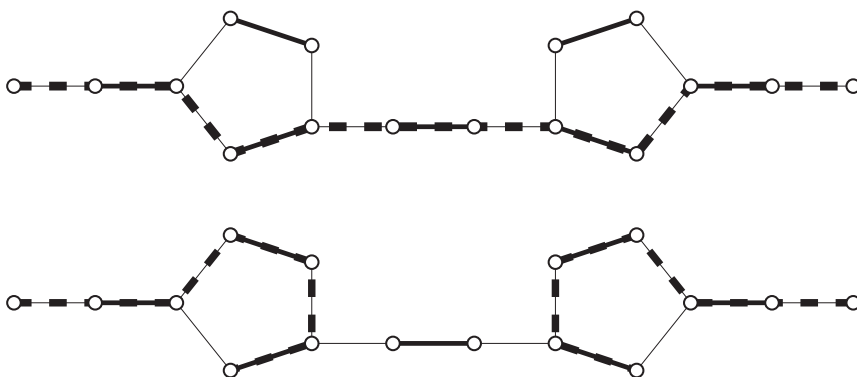


2.33 Abbildung. Nicht-bipartiter Graph G mit Matching M .

Das Matching M ist nicht maximal;

$$(y_1, y_2, y_3, y_4, y_6, y_8, y_9, y_{10}, y_{12}, y_{14}, y_{15}, y_{16})$$

ist ein Augmentationsweg; vgl. Abbildung 2.34 (oben). Unten in derselben Abbildung sind zwei Augmentationsbäume markiert, zu y_1 bzw. zu y_{16} . Keiner von ihnen enthält einen Augmentationsweg, d.h. Korollar 2.2.61 ist für nicht bipartite Graphen im Allgemeinen falsch. Das liegt daran, dass bereits Lemma 2.2.59 nicht mehr gilt. Je nachdem, ob man von y_1 aus ‘oben durch das Fünfeck’ oder ‘unten herum’ zu y_4 geht, ist y_4 der siebte oder vierte Knoten; die Parität hängt also sehr wohl davon ab, wie man zu y_4 gelangt.



2.34 Abbildung. Augmentationsweg (oben), Augmentationsbäume (unten), jeweils schraffiert.

Beschränkt man also die Suche auf jeweils einen beliebigen Augmentationsbaum, so

hat man keine Garantie mehr, einen Augmentationsweg zu finden. Der Suchprozess kann somit nicht in gleicher Weise reduziert werden, wie im bipartiten Fall.

Wir werden später sehen, dass auch das generelle Matchingproblem effizient gelöst werden kann; es ist allerdings wesentlich komplizierter.

Es gibt vielfältige praktische Anwendungen für dieses und andere Probleme dieser Sektion, unter anderem in der Telekommunikation, der Arbeitsplanung für städtische Dienste (Müllabfuhr, Schneeräumdienste) oder bei der Maschinenbelegungsplanung. Auf verschiedene dieser Probleme, ihre mathematische Modellierung und algorithmische Behandlung wird in späteren Kapiteln eingegangen.

2.3 Von Basen zu diskreten Strukturen: Matroide

Diese Sektion verallgemeinert sowohl den Begriff der Basen endlich-dimensionaler Vektorräume aus der linearen Algebra als auch den Begriff der Kreise in Graphen. Insbesondere wird eine Struktur eingeführt, die dadurch charakterisiert werden kann, dass sich bestimmte Optimierungsprobleme leicht durch eine einfache ‘lokale Methode des größten Gewinns’ lösen lassen.

Unabhängigkeitssysteme und Matroide: Wir beginnen mit zentralen Begriffen und Konzepten, geben einige typische Beispiele und leiten erste strukturelle Ergebnisse her.

2.3.1 Definition. Seien E eine endliche Menge und \mathcal{E} eine nichtleere Menge von Teilmengen von E . Das Paar (E, \mathcal{E}) heißt **Unabhängigkeitssystem** [engl.: *independence system*], wenn \mathcal{E} abgeschlossen unter Inklusion ist, d.h. wenn gilt:

$$I \in \mathcal{E} \wedge J \subset I \implies J \in \mathcal{E}.$$

Sei (E, \mathcal{E}) ein Unabhängigkeitssystem. Dann heißt E **Grundmenge** von (E, \mathcal{E}) , die Elemente von \mathcal{E} heißen **unabhängig** oder **unabhängige Mengen**, die Elemente von $2^E \setminus \mathcal{E}$ **abhängig**. Gilt ferner

$$I, J \in \mathcal{E} \wedge |I| + 1 = |J| \implies \exists (e \in J \setminus I) : I \cup \{e\} \in \mathcal{E},$$

so heißt (E, \mathcal{E}) **Matroid**. Die letzte Bedingung wird **Austausch-** oder **Ergänzungsbedingung** genannt.

2.3.2 Bemerkung. Sei (E, \mathcal{E}) ein Unabhängigkeitssystem. Dann gilt $\emptyset \in \mathcal{E}$. Ferner ist (E, \mathcal{E}) genau dann ein Matroid, wenn

$$I, J \in \mathcal{E} \wedge |I| < |J| \implies \exists (e \in J \setminus I) : I \cup \{e\} \in \mathcal{E}$$

gilt.

Beweis: Da nach Voraussetzung $\mathcal{E} \neq \emptyset$ gilt und \mathcal{E} unter Inklusion abgeschlossen ist, folgt $\emptyset \in \mathcal{E}$.

Die Richtung ‘ \Leftarrow ’ der zweiten Aussage ist klar, ‘ \Rightarrow ’ folgt durch Anwendung der Austauschbedingung auf eine beliebige Teilmenge von J der Kardinalität $|I| + 1$. \square

Wir geben nun einige Standardbeispiele für Matroide.

2.3.3 Lemma. Für $i = 1, 2, 3, 4$ seien E_i endliche Mengen und $\mathcal{E}_i \subset 2^{E_i}$ wie nachfolgend spezifiziert:

- (a) $\mathcal{E}_1 := 2^{E_1}$;
- (b) Seien $k \in \mathbb{N}_0$ und $\mathcal{E}_2 := \{I \subset E_2 : |I| \leq k\}$;
- (c) Sei X ein endlich-dimensionaler Vektorraum über einem Körper \mathbb{K} , $E_3 \subset X$ und \mathcal{E}_3 die Menge aller linear unabhängigen Teilmengen von E_3 ;
- (d) Seien $s \in \mathbb{N}$, $\{A_1, \dots, A_s\}$ eine Partition von E_4 , $\beta_1, \dots, \beta_s \in \mathbb{N}_0$ und

$$\mathcal{E}_4 = \{I \subset E_4 : |I \cap A_1| \leq \beta_1 \wedge \dots \wedge |I \cap A_s| \leq \beta_s\}.$$

Dann sind (E_1, \mathcal{E}_1) , (E_2, \mathcal{E}_2) , (E_3, \mathcal{E}_3) und (E_4, \mathcal{E}_4) Matroide.

Beweis: In allen vier Fällen ist die Abgeschlossenheit gegenüber Inklusionen klar³⁷. Ferner genügen (E_1, \mathcal{E}_1) und (E_2, \mathcal{E}_2) trivialerweise der Austauschbedingung. Für (E_3, \mathcal{E}_3) folgt diese aus dem Basisergänzungssatz der linearen Algebra.

Wir zeigen nun abschließend, dass auch (E_4, \mathcal{E}_4) die Austauschbedingung erfüllt. Seien daher $I, J \in \mathcal{E}_4$ mit $|I| + 1 = |J|$. Seien $k \in \mathbb{N}_0$ und die Mengen A_1, \dots, A_s so nummeriert, dass

$$|I \cap A_1| = \beta_1 \wedge \dots \wedge |I \cap A_k| = \beta_k \quad \wedge \quad |I \cap A_{k+1}| < \beta_{k+1} \wedge \dots \wedge |I \cap A_s| < \beta_s$$

gilt. Zur Abkürzung seien

$$X := A_1 \cup \dots \cup A_k \quad \wedge \quad Y := A_{k+1} \cup \dots \cup A_s.$$

Wir nehmen zunächst an, dass $J \setminus I \subset X$ gilt. Dann folgt $k \geq 1$ und $J \cap Y \subset I$, also insbesondere $|J \cap Y| \leq |I \cap Y|$. Die Voraussetzung $|I| < |J|$ impliziert somit

$$|J \cap X| > |I \cap X| = \beta_1 + \dots + \beta_k.$$

Die Menge J verletzt also mindestens eine der ersten k Kardinalitätsbedingungen, im Widerspruch zu $J \in \mathcal{E}_4$.

Somit gibt es einen Index $i_0 \in \{k+1, \dots, s\}$ mit

$$(J \setminus I) \cap A_{i_0} \neq \emptyset.$$

Man kann nun I ein beliebiges Element $e \in (J \setminus I) \cap A_{i_0}$ hinzufügen, ohne die Kardinalitätsbedingungen zu verletzen, d.h. $I \cup \{e\} \in \mathcal{E}_4$, und es folgt die Behauptung. \square

2.3.4 Bezeichnung. Seien

$$(E_1, \mathcal{E}_1), \quad (E_2, \mathcal{E}_2), \quad (E_3, \mathcal{E}_3), \quad (E_4, \mathcal{E}_4)$$

die Matroide aus Lemma 2.3.3. (E_1, \mathcal{E}_1) heißt das **freie Matroid** über E_1 , (E_2, \mathcal{E}_2) heißt **uniformes Matroid**, (E_3, \mathcal{E}_3) heißt **vекtoriell** oder **Matrix-Matroid**³⁸, und (E_4, \mathcal{E}_4) heißt **Partitionsmatroid**.

³⁷ Im Falle $i = 3$ wird, wie üblich, die leere Menge von Vektoren als linear unabhängig aufgefasst.

³⁸ Dem Namen Matrix-Matroid liegt die Vorstellung zugrunde, dass die Vektoren aus E_3 Spalten einer Matrix sind.

Die Matroid-Eigenschaft bleibt unter Teilmengenbildung erhalten. Genauer gilt die folgende Aussage.

2.3.5 Bemerkung. Seien (E, \mathcal{E}) ein Unabhängigkeitssystem, $A \subset E$ und $\mathcal{E}_A := \{I \in \mathcal{E} : I \subset A\}$.

- (a) Dann ist (A, \mathcal{E}_A) ein Unabhängigkeitssystem.
- (b) Ist (E, \mathcal{E}) ein Matroid, so ist auch (A, \mathcal{E}_A) ein Matroid.

Beweis: Die Aussagen folgen direkt aus den Definitionen. □

2.3.6 Bezeichnung. Seien $U := (E, \mathcal{E})$ ein Unabhängigkeitssystem bzw. ein Matroid, $A \subset E$ und $\mathcal{E}_A := \{I \in \mathcal{E} : I \subset A\}$. Dann heißt (A, \mathcal{E}_A) das durch A in (E, \mathcal{E}) **induzierte Unabhängigkeitssystem** bzw. **Matroid** und wird mit U_A bezeichnet.

Offenbar erhalten nicht nur Inklusionen sondern auch Bijektionen die Matroid-Eigenschaften. Um diese Aussage adäquat formulieren zu können, wird zunächst eine Bijektion auf natürliche Weise auf ein Mengensystem übertragen.

2.3.7 Definition. Seien X_1 und X_2 Mengen und $\theta : X_1 \rightarrow X_2$ eine Abbildung. Dann heißt die durch $\Theta(I) := \{\theta(x) : x \in I\}$ für alle $I \in 2^{X_1}$ definierte Funktion $\Theta : 2^{X_1} \rightarrow 2^{X_2}$ die **natürliche Erweiterung** von θ auf die Potenzmengen.³⁹

2.3.8 Bemerkung. Seien (E_1, \mathcal{E}_1) ein Unabhängigkeitssystem bzw. ein Matroid, E_2 eine endliche Menge, $\theta : E_1 \rightarrow E_2$ eine Bijektion, Θ ihre natürliche Erweiterung und

$$\mathcal{E}_2 := \left\{ \Theta(I) : I \in \mathcal{E}_1 \right\}.$$

Dann ist (E_2, \mathcal{E}_2) ein Unabhängigkeitssystem bzw. Matroid.

Beweis: Die Aussage folgt direkt aus der Tatsache, dass sich die Bijektivität der Funktion θ auf ihre natürliche Erweiterung überträgt. □

Als Spezialfall von Lemma 2.3.3 (c) erhalten wir ein einem Graphen zugeordnetes Matroid.

2.3.9 Korollar. Seien $G := (V, E)$ ein Graph, $V := \{v_1, \dots, v_n\}$, $E := \{e_1, \dots, e_m\}$ und $m \geq 1$. Ferner sei $Z \subset \mathbb{Z}_2^m$ die Menge der Inzidenzvektoren der Kanten von G (d.h. der Spaltenvektoren von S_G), $\theta : Z \rightarrow E$ ordne jedem Inzidenzvektor z seine zugehörige Kante $e \in E$ zu, und Θ sei die natürliche Erweiterung von θ . Bezeichnet (Z, \mathcal{E}) das vektorielle Matroid über Z und ist $\mathcal{F} := \Theta(\mathcal{E})$, so ist (E, \mathcal{F}) ein Matroid.

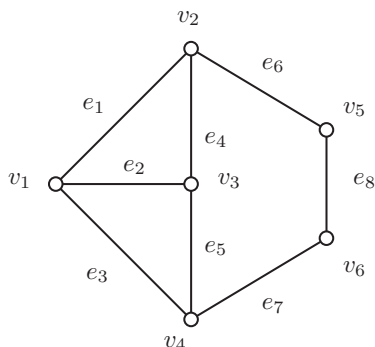
Beweis: Die Aussage folgt direkt aus Bemerkung 2.3.8. □

Natürlich werden wir im Folgenden oftmals Kanten und zugehörige Inzidenzvektoren identifizieren und von (E, \mathcal{F}) als vektorielltem Matroid über \mathbb{Z}_2 sprechen.

2.3.10 Beispiel. Gegeben sei der in Abbildung 2.35 dargestellte Graph $G := (V, E)$.

Es gilt

³⁹ Meistens wird bezüglich der Notation nicht zwischen θ und Θ unterschieden und auch die natürliche Erweiterung wieder θ genannt.



2.35 Abbildung. Ein Graph $G := (V, E)$.

$$S_G = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \in \mathbb{Z}_2^{6 \times 8}.$$

Die Matrix S_G hat den Rang 5; die Summe der Zeilen ist 0 (mod 2), aber die zu den Kanten e_1, e_2, e_3, e_6, e_7 gehörenden Spalten sind linear unabhängig über \mathbb{Z}_2 . Insbesondere ist also jede Teilmenge von mindestens 6 Kanten von G im Matroid (E, \mathcal{F}) abhängig. Modulo 2 addieren sich aber auch die Inzidenzvektoren der Kantenmengen

$$\{e_1, e_2, e_4\}, \{e_2, e_3, e_5\}, \{e_1, e_3, e_4, e_5\}, \{e_1, e_3, e_6, e_7, e_8\}$$

zu 0 und weitere, die sich modulo 2 aus diesen kombinieren lassen;⁴⁰ vgl. Abbildung 2.36. Unabhängig sind hingegen unter anderem

$$\{e_1, e_3, e_5, e_7, e_8\}, \{e_2, e_3, e_4, e_7, e_8\}, \{e_2, e_4, e_5, e_6\}, \{e_2, e_3, e_6, e_8\}.$$

In Beispiel 2.3.10 enthalten die abhängigen Mengen Kreise, die unabhängigen Mengen hingegen nicht. Das folgende Lemma zeigt, dass das kein Zufall ist: die unabhängigen Mengen von (E, \mathcal{F}) sind genau die (Kantenmengen der) Wälder in G .

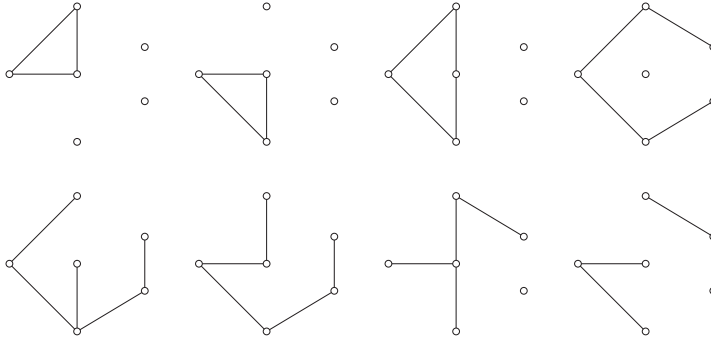
2.3.11 Lemma. Seien $G := (V, E)$ ein Graph, (E, \mathcal{F}) das zugehörige vektorielle Matroid über \mathbb{Z}_2 und $T \subset E$. Die Teilmenge T ist genau dann unabhängig, wenn (V, T) kreisfrei ist.

Beweis: ‘ \Rightarrow ’ Sei K ein Kreis in (V, T) . Da jeder Knoten von K genau zwei Kanten enthält⁴¹, addieren sich die zugehörigen Inzidenzvektoren zu 0 (mod 2), und nach Korollar 2.3.9 gilt $T \notin \mathcal{F}$.

‘ \Leftarrow ’ Seien nun andererseits (V, T) kreisfrei, $T = \{e_1, \dots, e_s\}$, z_1, \dots, z_s die zugehörigen Inzidenzvektoren und

⁴⁰ Man beachte, dass sich (im Sinne der Addition ihrer Inzidenzvektoren mod 2) die ersten beiden abhängigen Mengen $\{e_1, e_2, e_4\}$ und $\{e_2, e_3, e_5\}$ zur dritten $\{e_1, e_3, e_4, e_5\}$ addieren.

⁴¹ Hier geht ein, dass G schlingenfrei ist!



2.36 Abbildung. Vier verschiedene abhängige (oben) und unabhängige Mengen (unten) des zu G gehörigen Matroids (E, \mathcal{F}) .

$$\lambda_1, \dots, \lambda_s \in \mathbb{Z}_2 \quad \wedge \quad \sum_{i=1}^s \lambda_i z_i = 0.$$

Wir nehmen an, dass nicht alle λ_i gleich 0 sind, etwa mit $k \in \mathbb{N}$ o.B.d.A.

$$\lambda_1 = \dots = \lambda_k = 1 \quad \wedge \quad \lambda_{k+1} = \dots = \lambda_s = 0.$$

Sei $F := \{e_1, \dots, e_k\}$. Es gilt also $|F| \geq 1$, und nach Bemerkung 2.2.41 enthält $S := (V, F)$ Knoten vom Grad 1. Die entsprechenden Komponenten von

$$\sum_{i=1}^s \lambda_i z_i$$

sind somit kongruent 1 (mod 2), im Widerspruch zur Annahme. Somit ist 0 nur auf triviale Weise als \mathbb{Z}_2 -Linearkombination von z_1, \dots, z_s darstellbar; die Vektoren sind also linear unabhängig. \square

2.3.12 Bezeichnung. Seien $G := (V, E)$ ein Graph und \mathcal{F} die Menge (der Kanten) der Wälder in G . Dann heißt (E, \mathcal{F}) das zu G gehörige **graphische Matroid**.

2.3.13 Definition. Sei $U := (E, \mathcal{E})$ ein Unabhängigkeitssystem. Jede (inklusions-) maximale unabhängige Menge von \mathcal{E} heißt **Basis**. (Inklusions-) Minimale abhängige Mengen heißen **Kreise** [engl.: circuit]. Seien

$$r_+(U) := \max\{|B| : B \text{ ist Basis von } U\} \quad \wedge \quad r_-(U) := \min\{|B| : B \text{ ist Basis von } U\}.$$

$r_+(U)$ heißt **oberer Rang**, $r_-(U)$ **unterer Rang** von U . Gilt $r_+(U) = r_-(U)$, so heißt $r(U) := r_+(U) = r_-(U)$ der **Rang** von U .

Mit Hilfe der Austauschbedingung erhält man das folgende Ergebnis.

2.3.14 Lemma. Sei $U := (E, \mathcal{E})$ ein Unabhängigkeitssystem. U ist genau dann ein Matroid, wenn für jedes $A \subset E$ und $U_A := (A, \mathcal{E}_A)$

$$r_+(U_A) = r_-(U_A)$$

gilt, d.h. wenn alle Basen von U_A stets dieselbe Kardinalität haben.

Beweis: ‘ \Rightarrow ’ folgt mit Bemerkung 2.3.5 direkt aus der Austauschbedingung.

‘ \Leftarrow ’ Seien $I, J \in \mathcal{E}$ mit $|I| + 1 = |J|$. Wir setzen $A := I \cup J$. Es gilt $I, J \in \mathcal{E}_A$. Wegen $|I| < |J|$ und $r_+(U_A) = r_-(U_A)$ ist I nicht maximal unabhängig, d.h. es gibt ein Element $e \in A \setminus I = J \setminus I$ mit $I \cup \{e\} \in \mathcal{E}_A \subset \mathcal{E}$, und es folgt die Behauptung. \square

Das folgende Lemma beschreibt insbesondere einen Zusammenhang von Basen und Kreisen in Matroiden, der die entsprechende Aussage von Lemma 2.2.42 verallgemeinert.

2.3.15 Lemma. *Seien $M := (E, \mathcal{E})$ ein Matroid, $I \in \mathcal{E}$, $e \in E$ sowie $I \cup \{e\} \notin \mathcal{E}$. Dann enthält $I \cup \{e\}$ genau einen Kreis.*

Beweis: Da $I \cup \{e\}$ abhängig ist, enthält $I \cup \{e\}$ einen Kreis. Seien K_1, K_2 zwei verschiedene Kreise in $I \cup \{e\}$. Da I unabhängig ist und $(K_1 \cup K_2) \setminus \{e\} \subset I$ gilt, folgt

$$e \in K_1 \cap K_2 \quad \wedge \quad (K_1 \cup K_2) \setminus \{e\} \in \mathcal{E}.$$

Sei nun $a \in K_1 \setminus K_2$. Dann ist $K_1 \setminus \{a\} \in \mathcal{I}$, da Kreise minimal abhängig sind. Durch (ggf. sukzessive) Anwendung der Austauschbedingung auf $K_1 \setminus \{a\}$ und $(K_1 \cup K_2) \setminus \{e\}$ erhält man eine Menge $J \in \mathcal{E}$ mit

$$K_1 \setminus \{a\} \subset J \quad \wedge \quad |J| = |(K_1 \cup K_2) \setminus \{e\}| \quad \wedge \quad J \subset K_1 \cup K_2.$$

Sei $\{b\} := (K_1 \cup K_2) \setminus J$. Falls $b = a$ ($\in K_1 \setminus K_2$) gilt, so folgt $K_2 \subset J$, d.h. $K_2 \in \mathcal{E}$, im Widerspruch zur Voraussetzung. Also gilt $b \neq a$. Wegen $a \in K_1$ folgt somit $a \in J$. Da nach Konstruktion ferner $K_1 \setminus \{a\} \subset J$ ist, folgt $K_1 \subset J$ im Widerspruch zu $K_1 \in 2^E \setminus \mathcal{E}$. Hieraus folgt die Behauptung. \square

Maximierung über Unabhängigkeitssystemen und der Greedy-Algorithmus: Wir untersuchen nun Optimierungsprobleme über Unabhängigkeitssystemen. Das Hauptergebnis wird eine *algorithmische Charakterisierung* von Matroiden sein. Genauer liegt folgendes Problem zugrunde.

2.3.16 Definition. *Eine Maximierungsaufgabe über einem Unabhängigkeitssystem bzw. Matroid ist spezifiziert durch folgende Daten*

$$\begin{aligned} &\text{Unabhängigkeitssystem bzw. Matroid } (E, \mathcal{E}) \\ &\phi : E \rightarrow [0, \infty[. \end{aligned}$$

Sei $\varphi : \mathcal{E} \rightarrow [0, \infty[$ definiert durch $\varphi(\emptyset) := 0$ und für $I \in \mathcal{E} \setminus \{\emptyset\}$ durch

$$\varphi(I) := \sum_{e \in I} \phi(e).$$

φ heißt **zugehörige Zielfunktion**. Gesucht ist eine Menge

$$I^* \in \operatorname{argmax}_{I \in \mathcal{E}} \varphi(I).$$

Die Menge aller solchen Aufgaben heißt **Maximierungsproblem über Unabhängigkeitssystemen bzw. Matroiden**.

Liegen spezielle Klassen von Unabhängigkeitsproblemen oder Matroiden zugrunde, so werden diese auch in der Bezeichnung des entsprechenden Maximierungsproblems verwendet (ohne noch einmal formal eingeführt zu werden). Wird etwa das Maximierungsproblem auf graphische Matroide eingeschränkt, so spricht man von dem **Problem maximaler Wälder**.

Man beachte, dass die Voraussetzung $\phi(e) \geq 0$ für jedes $e \in I$ ohne Einschränkung der Allgemeinheit erfolgt, da aufgrund der Abgeschlossenheit von Unabhängigkeitssystemen unter Inklusion negativ bewertete Elemente entfernt werden können.

Wir betrachten nun einen einfachen Algorithmus zur Behandlung des Maximierungsproblems über Unabhängigkeitssystemen, den *Greedy-Algorithmus*.⁴²

2.3.17 Prozedur: *Greedy-Algorithmus (für das Maximierungsproblem über Unabhängigkeitssystemen).*⁴³

```

INPUT:   Unabhängigkeitssystem  $(E, \mathcal{E})$ ,  $\phi : E \rightarrow [0, \infty[$ 
OUTPUT:   $I^g \in \mathcal{E}$ 
BEGIN     $A \leftarrow E$ ;  $I^g \leftarrow \emptyset$ 
          WHILE  $A \neq \emptyset$  DO
            BEGIN
              Sei  $a \in \operatorname{argmax}_{a \in A} \phi(a)$ ; setze  $A \leftarrow A \setminus \{a\}$ 
              IF  $I^g \cup \{a\} \in \mathcal{E}$  THEN  $I^g \leftarrow I^g \cup \{a\}$ 
            END
          END
END

```

Man beachte, dass man den Maximierungsschritt im Greedy-Algorithmus am einfachsten durchführen kann, wenn die Elemente E bereits nach absteigendem Funktionswert sortiert sind; vgl. Sektion 3.1.

Die Laufzeit von Prozedur 2.3.17 wird naturgemäß wesentlich von den erforderlichen Unabhängigkeitstests bestimmt, die von den konkreten zugrunde liegenden Problemklassen abhängen. Einige Beispiele hierfür werden später in dieser Sektion gegeben.

2.3.18 Bezeichnung. *Durch $(E, \mathcal{E}; \phi)$ sei eine Maximierungsaufgabe über einem Unabhängigkeitssystem spezifiziert. Jede durch den Greedy-Algorithmus gefundene unabhängige Menge I^g heißt **Greedy-Lösung** der Aufgabe.*

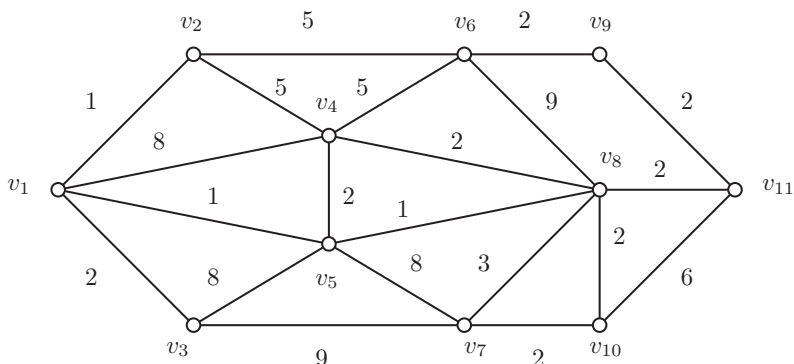
Da der Greedy-Algorithmus jeweils ‘lokal maximal’ operiert, besteht natürlich die Hoffnung, dass Greedy-Lösungen I^g ‘ziemlich groß’ sind. Wir betrachten zunächst als Beispiel die Maximierung über graphischen Matroiden; gesucht sind also maximale Wälder in gewichteten Graphen.

2.3.19 Beispiel. *Seien $G := (V, E; \phi)$ ein gewichteter Graph mit $\phi \geq 0$ und (E, \mathcal{F}) das zugehörige graphische Matroid. Dann wird durch $(E, \mathcal{F}; \phi)$ die Aufgabe spezifiziert, einen maximalen Wald in G zu finden.*

Im Greedy-Algorithmus werden die Kanten der Größe ihres Gewichts nach – beginnend mit der größten – der Reihe nach hinzugenommen, falls hierdurch noch kein Kreis erzeugt wird. Im Beispielgraphen von Abbildung 2.37 haben die Kanten (der Größe nach sortiert) die Gewichte 9, 8, 6, 5, 3, 2, 1, wobei einige mehrfach auftreten.

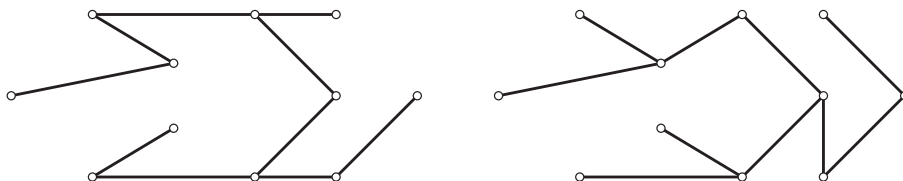
⁴² greedy [gridi] (engl.) gierig

⁴³ Zwar wirkt die strukturierte Beschreibung 2.3.17 des Greedy-Algorithmus bereits detaillierter als etwa Prozedur 2.2.62 zur Bestimmung kantenmaximaler Matchings in bipartiten Graphen, aber auch diese ist nicht als Implementierungsvorschrift zu verstehen, sondern dient lediglich einer einfachen Erfassung der Struktur des Verfahrens. Es fehlen insbesondere Aussagen über Datenstrukturen und Datenformate, aber auch die konkrete Beschreibung der erforderlichen Unabhängigkeitstests; vgl. Sektionen 3.1. Natürlich ist es auch für alle praktischen und komplexitätstheoretischen Belange erforderlich, genau zu spezifizieren, wie das Unabhängigkeitssystem gegeben sein soll. Man wird im Allgemeinen natürlich nicht als Input alle unabhängigen Mengen explizit auflisten, da dieses wegen der kombinatorischen Explosion zu Zeit- und Platzproblemen führt; vgl. Sektion 3.1.



2.37 Abbildung. Ein gewichteter Graph.

Als Reihenfolge, in der die Kanten bei Gleichheit der Gewichte betrachtet werden, legen wir eine (lexikographische) Reihung nach aufsteigendem kleinsten und bei Gleichheit nach aufsteigendem zweiten Knotenindex fest. Zunächst werden somit die Kanten $\{v_3, v_7\}$ und $\{v_6, v_8\}$ aufgenommen. Als nächstes kommen die mit 8 bewerteten Kanten $\{v_1, v_4\}$, $\{v_3, v_5\}$ hinzu. Die Kante $\{v_5, v_7\}$ hat ebenfalls Gewicht 8, kann aber nicht mehr hinzugefügt werden, da sie dann mit den schon aufgenommenen Kanten $\{v_3, v_5\}$, $\{v_3, v_7\}$ einen Kreis bildet. Das nächst kleinere Gewicht 6 trägt nur die Kante $\{v_{10}, v_{11}\}$; sie kann aufgenommen werden, da sie keinen Kreis schließt. Danach kommen $\{v_2, v_4\}$ und $\{v_2, v_6\}$ hinzu. $\{v_4, v_6\}$ kann nicht mehr aufgenommen werden, da wieder ein Kreis geschlossen würde. Es folgt $\{v_7, v_8\}$. Die Kanten mit 2 bewerteten $\{v_1, v_3\}$, $\{v_4, v_5\}$, $\{v_4, v_8\}$ schließen jeweils wieder einen Kreis, können also nicht aufgenommen werden. $\{v_6, v_9\}$, $\{v_7, v_{10}\}$ werden hinzugefügt, $\{v_8, v_{10}\}$, $\{v_8, v_{11}\}$, $\{v_9, v_{11}\}$ hingegen nicht. Zuletzt werden (der Reihe nach) die Kanten $\{v_1, v_2\}$, $\{v_1, v_5\}$, $\{v_5, v_8\}$ mit Gewicht 1 betrachtet. Keine von ihnen kann hinzu genommen werden, ohne dass ein Kreis entsteht. Insgesamt erhalten wir somit die in Abbildung 2.38 links abgebildete unabhängige Menge. Da der zugrunde liegende Graph zusammenhängend ist, bildet sie einen Spannbaum der Länge 57; vgl. Abbildung 2.38 (links).



2.38 Abbildung. Greedy-Lösungen.

Werden die Kanten gleichen Gewichts in einer anderen Reihenfolge abgearbeitet, so entstehen im Allgemeinen andere Lösungen. Abbildung 2.38 (rechts) zeigt einen anderen Spannbaum der gleichen Länge 57, der ebenfalls mit dem Greedy-Algorithmus gewonnen werden kann. Beide Greedy-Lösungen sind maximal.

Im Beispiel des Problems maximaler Wälder, bei dem die zugrunde liegenden Unabhängigkeitssysteme Matroide sind, scheint der Greedy-Algorithmus gut zu funktionieren.

ren. Andererseits kann er im Allgemeinen aber auch sehr schlechte Ergebnisse liefern.

2.3.20 Beispiel. Seien $m \geq 2$, $E := \{e_1, \dots, e_m\}$, und \mathcal{E} bestehe aus $\{e_1\}$ und der Potenzmenge von $\{e_2, \dots, e_m\}$. Dann ist $U := (E, \mathcal{E})$ ein Unabhängigkeitssystem (aber kein Matroid). Sei $\gamma \in]0, \infty[$, und $\phi : E \rightarrow [0, \infty[$ sei definiert durch

$$\phi(e) = \begin{cases} \gamma + 1 & \text{für } e = e_1 \\ \gamma & \text{für } e \in E \setminus \{e_1\}. \end{cases}$$

Ferner seien

$$I^g := \{e_1\} \quad \wedge \quad I^* := \{e_2, \dots, e_m\}.$$

Dann ist I^g die Greedy-Lösung, und es gilt

$$\varphi(I^g) = \gamma + 1 \quad \wedge \quad \varphi(I^*) = \gamma(m-1).$$

Für $m = 2$ oder $\gamma \leq \frac{1}{m-2}$ ist I^g optimal. Für

$$m \geq 3 \quad \wedge \quad \gamma > \frac{1}{m-2}$$

hingegen hat I^g einen kleineren Zielfunktionswert als I^* ; tatsächlich ist I^* optimal.

Der Greedy-Algorithmus kann also gründlich daneben gehen: abhängig von den Gewichten kann die Differenz der Gesamtgewichte der optimalen und der Greedy-Lösung beliebig groß werden. Allerdings ist der Quotient

$$\frac{\varphi(I^g)}{\varphi(I^*)} = \frac{\gamma + 1}{\gamma(m-1)} = \left(1 + \frac{1}{\gamma}\right) \frac{1}{m-1}$$

stets größer als $1/(m-1)$. Das Verhältnis von $\varphi(I^g)$ und $\varphi(I^*)$ liegt somit im Intervall

$$]1/(m-1), 1],$$

dessen Grenzen nur von m abhängen. Man beachte, dass $r_-(U) = 1$ und $r_+(U) = m-1$ gilt, der relative Fehler hier also durch das Verhältnis von $r_-(U)$ und $r_+(U)$ beschränkt ist.

Der nachfolgende Satz 2.3.23 zeigt, dass das relative Approximationsverhalten in Beispiel 2.3.20 kein Zufall war.

2.3.21 Definition. Sei $U := (E, \mathcal{E})$ ein Unabhängigkeitssystem. Für $A \subset E$ setze

$$\kappa(A) := \begin{cases} 1, & \text{falls } r_+(U_A) = 0; \\ \frac{r_-(U_A)}{r_+(U_A)}, & \text{falls } r_+(U_A) > 0. \end{cases}$$

Dann heißt

$$\rho(U) := \min\{\kappa(A) : A \subset E\}$$

Rangquotient von U .

Zum Beweis des Approximationssatzes 2.3.23 benutzen wir ein kombinatorisches Lemma über monotone Mengenfamilien; es ist bereits in der für unseren Kontext adäquaten Weise formuliert.

2.3.22 Lemma. Seien $E := \{e_1, \dots, e_m\}$, $I \in 2^E$, $\phi : E \rightarrow \mathbb{R}$ und für $j \in [m]$ seien $E_j := \{e_1, \dots, e_j\}$ und $I_j := I \cap E_j$. Mit den (formalen) Setzungen $E_0 := I_0 := \emptyset$ und $\phi(e_{m+1}) := 0$ für ein $e_{m+1} \notin E$ folgt dann

$$\sum_{e \in I} \phi(e) = \sum_{j=1}^m (\phi(e_j) - \phi(e_{j+1})) \cdot |I_j|.$$

Beweis: Da für $j \in [m]$

$$|I_j| - |I_{j-1}| = 1 \quad \Leftrightarrow \quad e_j \in I_j \setminus I_{j-1}$$

gilt, folgt

$$\begin{aligned} \sum_{e \in I} \phi(e) &= \sum_{j=1}^m \phi(e_j) \cdot (|I_j| - |I_{j-1}|) \\ &= \sum_{j=1}^m \phi(e_j) |I_j| - \sum_{j=0}^{m-1} \phi(e_{j+1}) \cdot |I_j| = \sum_{j=1}^m (\phi(e_j) - \phi(e_{j+1})) \cdot |I_j|, \end{aligned}$$

und damit die Behauptung. \square

2.3.23 Satz. Durch $(E, \mathcal{E}; \phi)$ sei eine Maximierungsaufgabe über dem Unabhängigkeitssystem $U := (E, \mathcal{E})$ spezifiziert, φ bezeichne die zugehörige Zielfunktion, I^* sei eine optimale, I^g eine Greedy-Lösung. Dann gilt

$$\varphi(I^g) \geq \rho(U) \cdot \varphi(I^*).$$

Es gibt eine Gewichtsfunktion ϕ , für die in obiger Ungleichung Gleichheit angenommen wird.

Beweis: Für $E = \emptyset$ ist nichts zu zeigen. Seien also $m \geq 1$ und $E = \{e_1, \dots, e_m\}$ mit $\phi(e_1) \geq \dots \geq \phi(e_m)$. Für $j \in [m]$ seien $E_j := \{e_1, \dots, e_j\}$ und $U_j := U_{E_j}$ das durch E_j induzierte Unabhängigkeitssystem. Für $I \in \mathcal{E}$ und $j \in [m]$ sei ferner $I_j := I \cap E_j$. Dann gilt für $j \in [m]$

$$r_-(U_j) \leq |I_j^g| \quad \wedge \quad |I_j^*| \leq r_+(U_j).$$

Mit der (formalen) Setzung $\phi(e_{m+1}) := 0$ für ein $e_{m+1} \notin E$ folgt mittels Lemma 2.3.22

$$\begin{aligned} \varphi(I^g) &= \sum_{j=1}^m (\phi(e_j) - \phi(e_{j+1})) \cdot |I_j^g| \geq \sum_{j=1}^m (\phi(e_j) - \phi(e_{j+1})) r_-(U_j) \\ &\geq \rho(U) \sum_{j=1}^m (\phi(e_j) - \phi(e_{j+1})) r_+(U_j) \geq \rho(U) \sum_{j=1}^m (\phi(e_j) - \phi(e_{j+1})) \cdot |I_j^*| \\ &= \rho(U) \cdot \varphi(I^*). \end{aligned}$$

Wir konstruieren nun noch eine Gewichtsfunktion, für die Gleichheit gilt. Seien $A^* \subset E$ mit

$$\kappa(A^*) = \min\{\kappa(A) : A \subset E\} = \rho(U)$$

und

$$\phi(e) := \begin{cases} 1 & \text{für } e \in A^*, \\ 0 & \text{für } e \in E \setminus A^*, \end{cases}$$

Seien nun I, J Basen in U_{A^*} mit

$$|I| = r_-(U_{A^*}) \quad \wedge \quad |J| = r_+(U_{A^*}).$$

Wird die Reihenfolge der Elemente von E so gewählt, dass zunächst die von I , danach die von $J \setminus I$ und abschließend alle anderen durchlaufen werden, so liefert der Greedy-Algorithmus $|I|$, obwohl $|J|$ der maximale Zielfunktionswert ist. Insgesamt folgt die Behauptung. \square

In dem im Beweis von Satz 2.3.23 konstruierten Beispiel dafür, dass die Schranke in Satz 2.3.23 scharf ist, wählt der Greedy-Algorithmus unter den Elementen mit Gewicht 1 zunächst die der kleineren Basis, und das ist offenbar eine schlechte Wahl. Man könnte sich fragen, ob nicht erheblich bessere Abschätzungen gelten, wenn der Greedy-Algorithmus bei ‘Gleichstand’ jeweils die bestmögliche Wahl treffen könnte. Unabhängig von der Frage, wie man eine solche Zusatzregel effizient umsetzen sollte, würde sie aber auch keine bessere Abschätzung erlauben.

2.3.24 Bemerkung. *Die Abschätzung in Satz 2.3.23 ist auch dann nicht substantiell zu verbessern, wenn man den Greedy-Algorithmus konzeptionell mit der Fähigkeit ausstattet, unter allen möglichen, gleich bewerteten nächsten Elementen ‘global das beste’ zu wählen.*

Beweis: Seien $\rho(U) < 1$, $A^* \subset E$ mit

$$\kappa(A^*) = \min\{\kappa(A) : A \subset E\}$$

sowie I, J Basen in U_{A^*} mit

$$|I| = r_-(U_{A^*}) \quad \wedge \quad |J| = r_+(U_{A^*}).$$

Seien $k \in \mathbb{N}$ und $l \in \mathbb{N}$ mit

$$I = \{e_1, \dots, e_k\} \quad \wedge \quad J \setminus I = \{e_{k+1}, \dots, e_l\} \quad \wedge \quad E \setminus (I \cup J) = \{e_{l+1}, \dots, e_m\}$$

sowie $\epsilon \in]0, \frac{1}{2}[$. Wir setzen

$$\phi(e_j) := \begin{cases} 1 + \epsilon^j & \text{für } j = 1, \dots, l, \\ 0 & \text{für } j = l+1, \dots, m. \end{cases}$$

Dann sind I die Greedy-Lösung und J optimal, und es gilt

$$\varphi(I) = r_-(U_{A^*}) + \epsilon \frac{1 - \epsilon^k}{1 - \epsilon} \quad \wedge \quad \varphi(J) \geq r_+(U_{A^*}) + \epsilon^{k+1} \frac{1 - \epsilon^{l-k}}{1 - \epsilon}.$$

Für $\epsilon \rightarrow 0$ folgt die Behauptung. \square

Als Korollar zu Satz 2.3.23 erhalten wir den folgenden zentralen Charakterisierungssatz für Matroide.

2.3.25 Korollar. *(Edmonds-Rado⁴⁴ Theorem)*

Sei (E, \mathcal{E}) ein Unabhängigkeitssystem. (E, \mathcal{E}) ist genau dann ein Matroid, wenn der Greedy-Algorithmus für jedes $\phi : E \rightarrow [0, \infty[$ ein Optimum der durch $(E, \mathcal{E}; \phi)$ spezifizierten Maximierungsaufgabe findet.

⁴⁴ Jack R. Edmonds, geb. 1934; Richard Rado, 1906 – 1989.

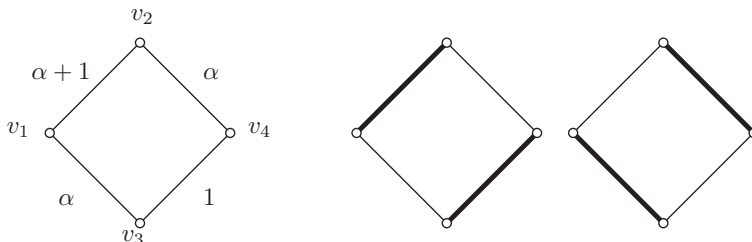
Beweis: Die Behauptung folgt direkt mittels Lemma 2.3.14 aus Satz 2.3.23. \square

Das Edmonds-Rado Theorem 2.3.25 ist etwas ganz Besonderes: Eine *kombinatorische Struktur* wird vollständig dadurch charakterisiert, dass ein *Algorithmus* eine Klasse von Optimierungsaufgaben über dieser Struktur löst.

Das Ergebnis charakterisiert aber natürlich nicht alle Tupel $(E, \mathcal{E}; \phi)$, für die der Greedy-Algorithmus funktioniert. So findet er trivialerweise immer ein Optimum, falls ϕ identisch 0 ist, unabhängig davon, ob (E, \mathcal{E}) ein Matroid ist oder nicht. Es kann also durchaus bei einzelnen praktischen Aufgaben passieren, dass der Greedy-Algorithmus auch dann überraschend gute Ergebnisse liefert, wenn das zugrunde liegende Unabhängigkeitssystem kein Matroid ist.

In dem folgenden Beispiel wird der Greedy-Algorithmus auf das Matching-Problem angewendet.

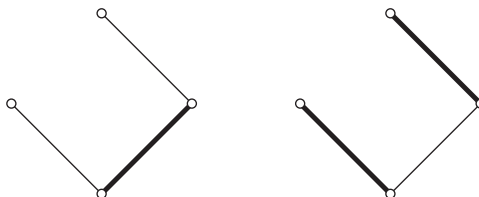
2.3.26 Beispiel. Sei $G := (V, E; \phi)$ der in Abbildung 2.39 skizzierte gewichtete Graph mit $\alpha \in \mathbb{N} \setminus \{1, 2\}$; gesucht ist ein Matching M mit maximalem Gewicht $\varphi(M) := \sum_{e \in M} \phi(e)$. Seien \mathcal{E} die Menge aller Matchings von G und $U := (E, \mathcal{E})$. Dann ist U ein Unabhängigkeitssystem.



2.39 Abbildung. Ein gewichteter Graph (links), das vom Greedy-Algorithmus produzierte Matching (Mitte) sowie das optimale Matching (rechts).

Der Greedy-Algorithmus liefert das aus den Kanten $\{v_1, v_2\}$ und $\{v_3, v_4\}$ bestehende Matching mit Zielfunktionswert $\alpha + 2$, obwohl das aus $\{v_1, v_3\}$ und $\{v_2, v_4\}$ bestehende Matching den Zielfunktionswert 2α besitzt, also größeres Gewicht hat. Nach Korollar 2.3.25 kann U somit kein Matroid sein.⁴⁵

Für die volle Kantenmenge sowie für jede beliebige Teilmenge von einer oder zwei Kanten stimmen der obere und der untere Rang überein. Anders ist es für drei Kanten; vgl. Abbildung 2.40.



2.40 Abbildung. Inklusionsmaximale Matchings unterschiedlicher Kardinalität.

⁴⁵ Man beachte übrigens, dass G sogar bipartit ist.

Da weder das aus beiden ‘Randkanten’ des entstehenden Weges der Länge 3 noch das aus der ‘Mittelkante’ bestehende Matching durch Hinzunahme weiterer Kanten vergrößert werden kann, ist der obere Rang 2, der untere aber 1. Satz 2.3.23 liefert daher die Abschätzung

$$\alpha + 2 = \varphi(I^g) \geq \frac{1}{2}\varphi(I^*) = \alpha,$$

deren relativer Fehler für $\alpha \rightarrow \infty$ gegen 1 geht.

Tatsächlich gilt die in Beispiel 2.3.26 festgestellte Abschätzung auch allgemeiner.

2.3.27 Lemma. Seien $G := (V, E; \phi)$ ein gewichteter Graph mit $\phi \geq 0$, \mathcal{E} die Menge aller Matchings von G , $\varphi(M) := \sum_{e \in M} \phi(e)$ für $M \in \mathcal{E}$ und $U := (E, \mathcal{E})$. Ferner seien I^* ein (bez. ϕ) maximales Matching und I^g eine Greedy-Lösung. Dann gilt

$$2\varphi(I^g) \geq \varphi(I^*).$$

Beweis: Wir zeigen, dass für beliebige nichtleere Teilmengen A von E stets

$$r_+(U_A) \leq 2r_-(U_A)$$

gilt. Angenommen, es gäbe eine Menge $A \subset E$ und maximale unabhängige Mengen I_1, I_2 in U_A mit $|I_1| = r_-(U_A)$, $|I_2| = r_+(U_A)$ und $r_+(U_A) \geq 2r_-(U_A) + 1$. Für die Knotenmengen $V_k := \bigcup_{e \in I_k} e$ für $k = 1, 2$ wäre dann

$$|V_2| = 2|I_2| = 2r_+(U_A) \geq 2(2r_-(U_A) + 1) = 4|I_1| + 2 = 2|V_1| + 2.$$

Es gibt also eine Kante e in I_2 , die zwei Knoten von G verbindet, die nicht zu V_1 gehören. Damit ist $I_1 \cup \{e\} \in \mathcal{E}$ im Widerspruch zur Voraussetzung, dass I_1 eine maximale unabhängige Menge sei. Also gilt $\rho(U) \geq 1/2$. Aus Satz 2.3.23 folgt

$$\varphi(I^g) \geq \rho(U) \cdot \varphi(I^*) \geq \frac{1}{2}\varphi(I^*)$$

und somit die Behauptung. □

Wie Beispiel 2.3.26 zeigt, weist die Frage nach Matchings maximaler Kantengewichte keine Matroid-Struktur auf; man vgl. aber Übungsaufgabe 2.4.25.

2.3.28 Bemerkung. Sei $G := (V, E)$ ein Graph, und \mathcal{E} bezeichne die Menge aller Teilmengen I von V , so dass ein Matching M in G existiert, das alle Knoten von I überdeckt. Dann ist (V, \mathcal{E}) ein Matroid.

Beweis: Übungsaufgabe 2.4.25 □

Minimierung über Basissystemen: Wie wir gesehen haben, ist das Problem der maximalen Wälder für nichtnegative Zielfunktionen ein Maximierungsproblem über dem zugehörigen graphischen Matroid. Ist insbesondere G zusammenhängend, so ist die Greedy-Lösung ein maximaler Spannbaum. Wenn wir hingegen einen *minimalen* Spannbaum konstruieren wollen, so müssen wir uns bei der Optimierung auf Basen des graphischen Matroids beschränken.

2.3.29 Definition. Seien (E, \mathcal{E}) ein Unabhängigkeitssystem und

$$\mathcal{B} := \{I \in \mathcal{E} : I \text{ ist Basis}\}.$$

Dann heißt (E, \mathcal{B}) (das zu (E, \mathcal{E}) gehörige) **Basissystem**.

Eine **Minimierungsaufgabe über einem Basissystem** ist durch folgende Daten spezifiziert⁴⁶

$$\begin{aligned} &\text{Unabhängigkeitssystem } (E, \mathcal{E}) \\ &\phi : E \rightarrow \mathbb{R}. \end{aligned}$$

Sei (E, \mathcal{B}) das Basissystem von (E, \mathcal{E}) , und die ϕ zugehörige Zielfunktion $\varphi : \mathcal{B} \rightarrow \mathbb{R}$ sei definiert durch $\varphi(B) := \sum_{e \in B} \phi(e)$ für $B \in \mathcal{B}$. Gesucht ist eine Menge

$$B^* \in \operatorname{argmin}_{B \in \mathcal{B}} \varphi(B).$$

Die Menge aller solchen Aufgaben heißt **Minimierungsproblem über Basissystemen**.

Wie bereits in Definition 2.3.16 werden auch hier für spezielle Klassen von Unabhängigkeitsproblemen entsprechende deskriptive Bezeichnungen verwendet (ohne noch einmal formal eingeführt zu werden). Wird etwa das Minimierungsproblem auf graphische Matroide eingeschränkt, so spricht man von dem **Problem minimaler Spannbäume** [engl.: *minimal spanning tree problem*] oder kürzer vom MST-Problem.

Tatsächlich lassen sich viele Probleme der diskreten Optimierung als Maximierungsproblem über Unabhängigkeitssystemen oder als Minimierungsproblem über Basissystemen formulieren; vgl. Übungsaufgabe 2.4.26. Auch für Minimierungsprobleme über Basissystemen kann man direkt einen zu Korollar 2.3.25 analogen Satz beweisen. Einfacher ist aber folgende Transformation.

2.3.30 Bemerkung. Seien $U := (E, \mathcal{E})$ ein Unabhängigkeitssystem, (E, \mathcal{B}) das zugehörige Basissystem, $\phi : E \rightarrow \mathbb{R}$, φ die zugehörige Zielfunktion und

$$\beta := \max\{0, \max\{\phi(e) : e \in E\}\}.$$

Ferner seien $\psi : E \rightarrow [0, \infty[$ definiert durch $\psi(e) := \beta - \phi(e)$ für $e \in E$ und $\Psi : \mathcal{E} \rightarrow [0, \infty[$ definiert durch $\Psi(I) := \sum_{e \in I} \psi(e)$ für $I \in \mathcal{E}$. Dann gilt

$$\beta r_-(U) - \max\{\Psi(I) : I \in \mathcal{E}\} \leq \min\{\varphi(B) : B \in \mathcal{B}\} \leq \beta r_+(U) - \max\{\Psi(I) : I \in \mathcal{E}\}.$$

Beweis: Für jedes $I \in \mathcal{E}$ gilt $\varphi(I) = \beta|I| - \Psi(I)$. Wegen $\beta \geq 0$ folgt

$$\min\{\varphi(B) : B \in \mathcal{B}\} = \min\{\beta|B| - \Psi(B) : B \in \mathcal{B}\} \leq \beta r_+(U) - \max\{\Psi(I) : I \in \mathcal{E}\}.$$

Die Abschätzung nach unten ergibt sich analog. □

Aus Bemerkung 2.3.30 folgt, dass minimale Spannbäume mit Hilfe des Greedy-Algorithmus konstruiert werden können. Formuliert man das Verfahren für die ursprüngliche Zielfunktion, so erhält man den folgenden Algorithmus von Kruskal⁴⁷.

⁴⁶ Man beachte, dass hier nicht auf gleiche Weise wie in Definition 2.3.16 für die Maximierung über Unabhängigkeitssystemen gerechtfertigt ist, sich auf den Fall nichtnegativer Gewichte zu beschränken; vgl. jedoch Bemerkung 2.3.30.

⁴⁷ Joseph Bernard Kruskal, 1928 – 2010.

2.3.31 Prozedur: *Algorithmus von Kruskal.*

INPUT: Zusammenhängender gewichteter Graph $G := (V, E; \phi)$ mit $|E| = m$
 OUTPUT: Spannbaum T minimalen Gewichts
 BEGIN Sortiere die Kanten nach aufsteigendem Gewicht, d.h.
 $E = \{e_1, \dots, e_m\}$ mit $\phi(e_1) \leq \dots \leq \phi(e_m)$
 $T \leftarrow \emptyset$
 FOR $i = 1, \dots, m$ DO
 BEGIN
 IF $T \cup \{e_i\}$ ist kreisfrei THEN $T \leftarrow T \cup \{e_i\}$
 END
 END

Der Test auf Kreisfreiheit kann dadurch geführt werden, dass man überprüft, ob mindestens eine der beiden Ecken einer abzuarbeitenden Kante außerhalb der Eckenmenge des aktuellen Teilbaums liegt.

Wir geben im Folgenden eine Variante des Greedy-Algorithmus zur Bestimmung minimaler Spannbäume an, bei der die Bäume von einer beliebigen Wurzel aus zusammenhängend wachsen. Hierbei wird die bereits erreichte aktuelle Knotenmenge W sukzessive durch Hinzunahme eines nächsten Knotens vergrößert.

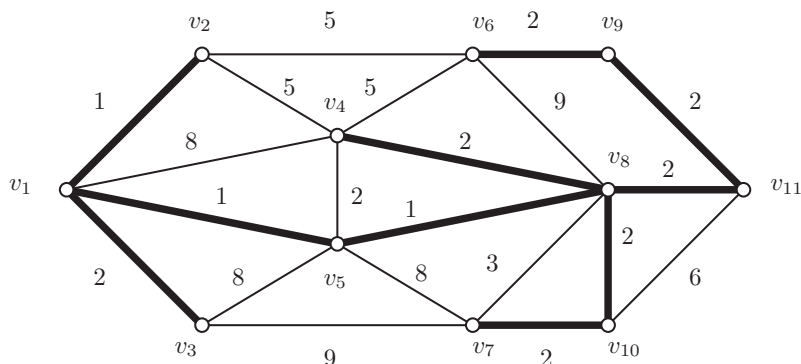
2.3.32 Prozedur: *Algorithmus von Prim*⁴⁸.

INPUT: Zusammenhängender gewichteter Graph $G = (V, E; \phi)$
 OUTPUT: Spannbaum (V, T) minimalen Gewichts
 BEGIN Sei $w \in V$
 $W \leftarrow \{w\}; T \leftarrow \emptyset$
 WHILE $N(G, W) \neq \emptyset$ DO
 BEGIN
 Seien $w \in W$ und $z \in V \setminus W$ mit $e^* := \{w, z\} \in \operatorname{argmin} \{\phi(e) : e \in \delta(G, W)\}$
 $W \leftarrow W \cup \{z\}; T \leftarrow T \cup \{e^*\}$
 END
 END

2.3.33 Beispiel. Wir wenden den Algorithmus von Prim auf den Graph von Beispiel 2.3.19 an und beginnen mit dem Startknoten v_1 ; vgl. Abbildung 2.41. Zur Auswahl für die erste Kante stehen $\{v_1, v_2\}$, $\{v_1, v_3\}$, $\{v_1, v_4\}$ und $\{v_1, v_5\}$; minimales Gewicht 1 haben $\{v_1, v_2\}$ und $\{v_1, v_5\}$, so dass wir als erste etwa die Kante $\{v_1, v_2\}$ aufnehmen können. Danach stehen alle Kanten zur Auswahl, die mit v_1 oder v_2 inzident sind. Da $\{v_1, v_5\}$ unter diesen minimales Gewicht 1 besitzt, wird diese Kante aufgenommen. Unter allen mit v_1 , v_2 oder v_5 inzidenten Kanten ist $\{v_5, v_8\}$ minimal und wird in den Baum aufgenommen. Im nächsten Schritt stehen alle Kanten zur Auswahl, die genau mit einem der Knoten v_1 , v_2 , v_5 und v_8 inzident sind. Das minimale Gewicht aller dieser Kanten ist 2, es wird von $\{v_1, v_3\}$, $\{v_4, v_5\}$, $\{v_4, v_8\}$, $\{v_8, v_{10}\}$ und $\{v_8, v_{11}\}$ realisiert. Sukzessive können etwa $\{v_1, v_3\}$, $\{v_8, v_{10}\}$ und $\{v_8, v_{11}\}$ aufgenommen werden sowie eine der Kanten $\{v_4, v_5\}$ und $\{v_4, v_8\}$. Wir wählen etwa $\{v_4, v_8\}$. Danach werden die Kanten $\{v_7, v_{10}\}$, $\{v_9, v_{11}\}$ und zuletzt $\{v_6, v_9\}$ aufgenommen. Der konstruierte Spannbaum hat die Länge 17.

Es kann leicht direkt überprüft werden, dass dieser Spannbaum tatsächlich minimal ist, da er alle Kanten mit dem minimalen Gewicht 1 und sonst nur Kanten mit dem zweitkleinsten Gewicht 2 enthält.

⁴⁸ Robert Clay Prim, geb. 1921.



2.41 Abbildung. Gewichteter Graph und (ein mit Hilfe des Algorithmus von Prim konstruierter) minimaler Spannbaum.

Während die direkte Anwendung des Greedy-Algorithmus auf das Problem minimaler Spannbäume und damit auch der Algorithmus von Kruskal nach Bemerkung 2.3.30 und Korollar 2.3.25 tatsächlich minimale Spannbäume liefern, muss dieses für den Algorithmus von Prim noch gesondert nachgewiesen werden, da dieser ja nicht eine jeweils bestmögliche Kante hinzufügt, sondern nur eine beste solche, die an einem schon erreichten Knoten ansetzt.

2.3.34 Satz. *Der Algorithmus von Prim löst das Problem minimaler Spannbäume korrekt.*

Beweis: Sei $G := (V, E; \phi)$ zusammenhängend. Offenbar erzeugt der Algorithmus einen aufspannenden Baum. Mittels Induktion nach $|W|$ zeigen wir, dass der in einem Schritt des Algorithmus erzeugte aktuelle Baum $B := (W, T)$ stets zu einem minimalen Spannbaum ergänzt werden kann. Dieses ist trivial für $W_1 := \{v_1\}$, da $B_1 := (W_1, \emptyset)$ keine Kanten besitzt, v_1 aber natürlich in jedem Spannbaum enthalten ist. Die Behauptung gelte für $W_k := \{v_1, \dots, v_k\}$. Seien $T_k := \{e_1, \dots, e_{k-1}\}$ und $B_k := (W_k, T_k)$ der zugehörige aktuelle Baum.

Der Algorithmus wählt nun eine kürzeste Kante mit genau einem Endpunkt in W_k , etwa $e_k := \{v_{i_0}, v_{k+1}\}$. Seien $W_{k+1} := W_k \cup \{v_{k+1}\}$, $T_{k+1} := T_k \cup \{e_k\}$ und $B_{k+1} := (W_{k+1}, T_{k+1})$.

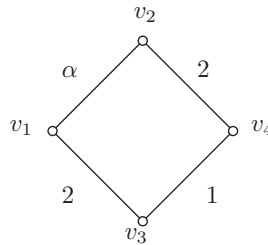
Wir nehmen an, es gäbe einen Spannbaum $S := (V, F)$ mit $B_k \subset S$, der kürzer ist, als alle aufspannenden Bäume, die B_{k+1} enthalten. $(V, F \cup \{e_k\})$ besitzt nach Lemma 2.3.15 genau einen Kreis K . Der Kreis K enthält eine von e_k verschiedene Kante e in $\delta(G, W_k)$. Nach Konstruktion gilt $\phi(e) \geq \phi(e_k)$; sonst hätte der Algorithmus nicht e_k hinzugenommen.

Mit $F' := (F \cup \{e_k\}) \setminus \{e\}$ folgt, dass der Teilgraph $S' := (V, F')$ ein aufspannender Baum in G ist mit $\varphi(S') \leq \varphi(S)$ und $B_{k+1} \subset S'$, im Widerspruch zur Annahme. Nach Induktionsvoraussetzung gibt es also einen minimalen Spannbaum, der B_{k+1} enthält, und es folgt die Behauptung. \square

Für Matroide löst der Greedy-Algorithmus nach Korollar 2.3.25 und Bemerkung 2.3.30 sowohl das Maximierungsproblem als auch das Minimierungsproblem über ihren Basissystemen. Allerdings unterscheidet sich seine Approximationsqualität bei diesen beiden Aufgaben über Unabhängigkeitssystemen beträchtlich. Die Fehlergarantie für das

Maximierungsproblem über Unabhängigkeitssystemen von Satz 2.3.23 hängt nicht von den (numerischen) Gewichten ab, sondern nur von der kombinatorischen Struktur des zugrunde liegenden Unabhängigkeitssystems (nämlich seinem Rangquotienten). Wie das folgende Beispiel zeigt, existiert bei dem Minimierungsproblem über Basissystemen im Allgemeinen keine entsprechende Abschätzung.

2.3.35 Beispiel. Wir betrachten denselben Graphen wie in Abbildung 2.39, allerdings mit anderen Gewichten. Sei $\alpha \in \mathbb{N}$ mit $\alpha \geq 4$. Anders als in Beispiel 2.3.26 suchen wir jetzt kein maximales Matching sondern minimieren über dem zugehörigen Basissystem. Gesucht ist also ein ‘minimales maximales Matching’, d.h. ein minimales Matching, bei dem allerdings keine Kante mehr hinzugenommen werden kann, ohne die Matching-Eigenschaft zu verletzen, hier also ein perfektes Matching.



2.42 Abbildung. Gewichteter Graph.

Der Greedy-Algorithmus wählt zunächst die Kante $\{v_3, v_4\}$, liefert also insgesamt das aus den Kanten $\{v_1, v_2\}$ und $\{v_3, v_4\}$ bestehende perfekte Matching mit Wert $\alpha + 1$. Andererseits besitzt das aus den Kanten $\{v_1, v_3\}$ und $\{v_2, v_4\}$ bestehende perfekte Matching den (offenbar optimalen) Zielfunktionswert 4. Das Verhältnis dieser Werte wächst mit α gegen unendlich.

Durchschnitt von Matroiden: Im letzten Abschnitt dieser Sektion soll das Verhältnis allgemeiner Unabhängigkeitssysteme und Matroide noch genauer analysiert werden.

Natürlich sind Matroide sehr spezielle Unabhängigkeitssysteme, die dadurch charakterisiert sind, dass der Greedy-Algorithmus stets optimale unabhängige Mengen liefert. Tatsächlich zeigt sich aber, dass jedes Unabhängigkeitssystem bereits Durchschnitt von Matroiden ist.

2.3.36 Definition. Für $j \in [k]$ seien $U_j := (E, \mathcal{E}_j)$ Unabhängigkeitssysteme auf derselben Grundmenge E , $\mathcal{E} := \bigcap_{j=1}^k \mathcal{E}_j$ und $U := (E, \mathcal{E})$. Dann heißt U **Durchschnitt** von U_1, \dots, U_k . Oft wird statt $(E, \bigcap_{j=1}^k \mathcal{E}_j)$ auch die (suggestivere) Schreibweise $U = \bigcap_{j=1}^k U_j$ verwendet.

Der Durchschnitt von Unabhängigkeitssystemen ist selbst wieder ein Unabhängigkeitssystem.

2.3.37 Bemerkung. Für $j \in [k]$ seien $U_j := (E, \mathcal{E}_j)$ Unabhängigkeitssysteme und $U := \bigcap_{j=1}^k U_j$. Dann ist U ein Unabhängigkeitssystem.

Beweis: Es seien $I \in \mathcal{E}$ und $J \subset I$. Für alle $j \in [k]$ gilt daher $I \in \mathcal{E}_j$. Da (E, \mathcal{E}_j) abgeschlossen unter Inklusion ist, folgt $J \in \mathcal{E}_j$ für alle $j \in [k]$, insgesamt also $J \in \mathcal{E}$. \square

Insbesondere ist also der Durchschnitt von Matroiden ein Unabhängigkeitssystem, im Allgemeinen allerdings kein Matroid mehr; vgl. Beispiel 2.3.40. Das folgende Lemma zeigt, dass das Konzept des Durchschnitts von Matroiden jedoch einen ‘fließenden’ Übergang zwischen Matroiden und Unabhängigkeitssystemen erzeugt: Unabhängigkeitssysteme sind Durchschnitte von Matroiden.

2.3.38 Lemma. *Sei $U := (E, \mathcal{E})$ ein Unabhängigkeitssystem. Dann gibt es ein $k \in \mathbb{N}$ und Matroide M_1, \dots, M_k mit $U = \bigcap_{j=1}^k M_j$.*

Beweis: Seien \mathcal{K} die Menge der Kreise in U , $K \in \mathcal{K}$ und

$$\mathcal{E}(K) := \{I \subset E : K \not\subset I\}.$$

Offenbar ist $(E, \mathcal{E}(K))$ ein Unabhängigkeitssystem, tatsächlich sogar ein Matroid. Sind nämlich $I, J \in \mathcal{E}(K)$ mit $|I| + 1 = |J|$. Dann gilt nach Definition $K \not\subset I$ sowie $K \not\subset J$. Angenommen für jedes Element $e \in J \setminus I$ wäre $K \subset I \cup \{e\}$. Dann wäre

$$|K \setminus I| = 1 \quad \wedge \quad |J \setminus I| = 1,$$

also $I \subset J$ und damit $K \subset J$, im Widerspruch zu $J \in \mathcal{E}(K)$.

Natürlich gilt $\mathcal{E} \subset \mathcal{E}(K)$. Seien nun $S \subset E$ mit $S \notin \mathcal{E}$ und $K \in \mathcal{K}$ mit $K \subset S$. Dann gilt $S \notin \mathcal{E}(K)$, es folgt

$$\mathcal{E} = \bigcap_{K \in \mathcal{K}} \mathcal{E}(K)$$

und damit die Behauptung. \square

Die Anzahl der Matroide, die in Lemma 2.3.38 konstruiert werden, ist im Allgemeinen sehr groß. Von besonderem Interesse sind aber diejenigen Strukturen, die sich als Durchschnitt weniger Matroide darstellen lassen.

Im nachfolgenden Beispiel verwenden wir eine natürliche Verallgemeinerung der in Definition 2.2.48 eingeführten bipartiten Graphen.

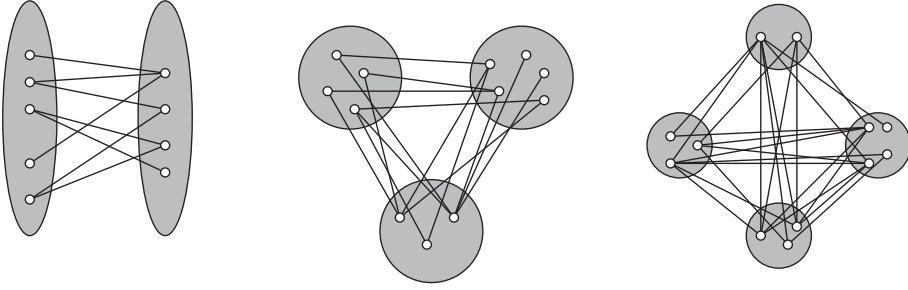
2.3.39 Definition. *Seien $G := (V, E)$ ein Graph, $\{V_1, \dots, V_k\}$ eine Partition von V , und es gelte*

$$j \in [k] \wedge e \in E \implies |e \cap V_j| \leq 1.$$

*Dann heißt G **k -partit**, und $\{V_1, \dots, V_k\}$ wird eine **zugehörige Partition** von V genannt.*

In einem k -partiten Graphen G verbindet also keine Kante zwei Ecken, die zur selben Partitionsmenge gehören. Natürlich ist jeder Graph $G := (V, E)$ stets $|V|$ -partit.

2.3.40 Beispiel. *Seien $k \in \mathbb{N}$, $G := (V, E)$ ein k -partiter Graph, $\{V_1, \dots, V_k\}$ eine zugehörige Partition von V , \mathcal{E} die Menge aller Matchings in G und $U := (E, \mathcal{E})$. Dann ist U ein Unabhängigkeitssystem und nach Lemma 2.3.38 somit Durchschnitt aller Matroide $(E, \mathcal{E}(K))$ mit $\mathcal{E}(K) := \{I \subset E : K \not\subset I\}$ für Kreise $K \subset E$ in U . Obwohl eine Kantensmenge bereits abhängig ist, wenn sie zwei verschiedene, nicht disjunkte Kanten enthält, ist U Durchschnitt von lediglich k Partitionsmatroiden. Für $j \in [k]$ sei nämlich*



2.43 Abbildung. k -partite Graphen für (von links nach rechts) $k = 2$, $k = 3$ und $k = 4$.

$$\mathcal{P}_j := \{\delta(G, v) : v \in V_j\} \cup \{E \setminus \delta(G, V_j)\}.$$

Dann ist \mathcal{P}_j eine Partition von E . Offenbar zerfällt \mathcal{P}_j in eine Partition von $\delta(G, V_j)$ und die Menge der Kanten, die keinen Knoten von V_j enthalten. Seien M_1, \dots, M_k die zugehörigen Partitionsmatroide, wobei die zu den Mengen $\delta(G, v)$ gehörigen Schranken $\beta_{j,i}$ jeweils 1 und die zu $E \setminus \delta(G, V_j)$ gehörige Schranke gleich $|E|$ gesetzt wird. In M_j ist somit die Tatsache kodiert, dass kein Knoten von V_j in mehr als einer Kante eines Matchings liegen darf. Offenbar ist dann $U = M_1 \cap \dots \cap M_k$.

Insbesondere liegt den Matchings in bipartiten Graphen also die kombinatorische Struktur des Durchschnitts von zwei Partitionsmatroiden zugrunde.

2.3.41 Beispiel. In einer Aufgabe des Rekonstruktionsproblems für kristalline Strukturen aus *Sektion 1.2* sind Liniensummandaten in m Gitterrichtungen $z_1, \dots, z_m \in \mathbb{Z}^n \setminus \{0\}$ gegeben, aus denen eine zugrunde liegende Punktmenge rekonstruiert werden soll.⁴⁹

Es zeigt sich, dass diese Aufgabe als Durchschnitt von m Partitionsmatroiden geschrieben werden kann. Wir betrachten dazu alle Geraden parallel zu einer der m gegebenen Richtungen, die – nach den gegebenen Liniensummandaten – einen Gitterpunkt der unbekannten Menge enthalten müssen. Für $j \in [m]$ seien also

$$T_{j,1}, \dots, T_{j,k_j}$$

alle Gittergeraden parallel zu z_j , für die die gegebenen Messungen von 0 verschieden sind. Die Geraden $T_{j,1}, \dots, T_{j,k_j}$ bilden demnach den Träger der Messungen in Richtung z_j . Die zugehörigen Messdaten seien $\beta_{j,1}, \dots, \beta_{j,k_j}$. Sei ferner

$$K := \mathbb{Z}^n \cap \bigcap_{j=1}^m \bigcup_{i=1}^{k_j} T_{j,i}.$$

Offenbar ist K eine endliche Obermenge für alle Lösungen des Rekonstruktionsproblems; die Kandidatenmenge. Wir setzen für $j \in [m]$

$$\mathcal{E}_j := \{I \in 2^K : |I \cap T_{j,1}| \leq \beta_{j,1} \wedge \dots \wedge |I \cap T_{j,k_j}| \leq \beta_{j,k_j}\} \quad \wedge \quad M_j := (K, \mathcal{E}_j).$$

sowie

⁴⁹ Wir formulieren die Aufgabe hier für m allgemeine Richtungen; vorher lagen speziell die drei Standardkoordinatenrichtungen des \mathbb{R}^3 zugrunde.

$$U := M_1 \cap \dots \cap M_m.$$

Dann ist U Durchschnitt der m Partitionsmatroide M_j , und das Rekonstruktionsproblem entspricht dem Maximierungsproblem über U mit den Gewichten 1. Genauer existiert dann, und nur dann, eine mit den Messdaten kompatible Lösung der Rekonstruktionsaufgabe, wenn es eine in U unabhängige Menge I^* gibt mit

$$|I^*| = \sum_{i=1}^{k_1} \beta_{1,i} = \dots = \sum_{i=1}^{k_m} \beta_{m,i}.$$

Nachdem wir gesehen haben, dass die Optimierung über Matroiden einfach ist, liegt es nahe zu hoffen, dass auch wenigstens über dem Durchschnitt von wenigen Matroiden effizient optimiert werden kann.

2.3.42 Definition. Sei $k \in \mathbb{N}$. Eine **Maximierungsaufgabe über dem Durchschnitt von k Matroiden** ist spezifiziert durch folgende Daten

$$\begin{aligned} \text{Matroide } M_1 &:= (E, \mathcal{E}_1), \dots, M_k := (E, \mathcal{E}_k) \\ \phi &: E \rightarrow [0, \infty[. \end{aligned}$$

Seien $U := (E, \mathcal{E}) := M_1 \cap \dots \cap M_k$ und $\varphi : \mathcal{E} \rightarrow [0, \infty[$ definiert durch

$$\varphi(I) := \sum_{e \in I} \phi(e) \quad (I \in \mathcal{E}).$$

Gesucht ist eine Menge

$$I^* \in \operatorname{argmax}_{I \in \mathcal{E}} \varphi(I).$$

Die Menge aller solchen Aufgaben heißt **Maximierungsproblem über dem Durchschnitt von k Matroiden** [engl.: *weighted k -matroid intersection problem*].

Das folgende Beispiel zeigt, dass sich auch eine bereits vorher untersuchte Eigenschaft von Graphen als Maximierungsproblem über dem Durchschnitt von (in diesem Fall drei) Matroiden feststellen lässt, nämlich, ob ein gegebener Graph einen Hamiltonweg besitzt.

2.3.43 Beispiel. Sei $G := (V, E)$ ein zusammenhängender Graph, von dem entschieden werden soll, ob in ihm ein Hamiltonweg existiert. Um diese Eigenschaft mithilfe von Matroiden zu kodieren, müssen wir zum einen ausdrücken, dass in einem Hamiltonweg H kein Knoten in mehr als zwei Kanten liegt, andererseits aber auch alle Knoten enthalten sind. Die zweite Bedingung ließe sich leicht mit Hilfe des zugehörigen graphischen Matroids ausdrücken, denn ein Hamiltonweg ist insbesondere ein Spannbaum. Um die erste Bedingung mit Hilfe von Partitionsmatroiden zu fassen, benutzen wir, dass durch das Durchlaufen von H seinen Kanten eine Richtung zugewiesen wird, und dann in jeden inneren Knoten von H genau eine Kante hinein- und aus ihm eine herausläuft. Wir gehen also zu dem Digraphen $\hat{G} := (V, \hat{E})$ über, der dadurch entsteht, dass jede ungerichtete Kante $\{v, w\} \in E$ durch die beiden gerichteten Kanten (v, w) und (w, v) ersetzt wird. Dann seien

$$\mathcal{P}_1 := \{\delta_{\text{aus}}(\hat{G}, v) : v \in V\} \quad \wedge \quad \mathcal{P}_2 := \{\delta_{\text{in}}(\hat{G}, v) : v \in V\}$$

sowie für jeden Knoten $v \in V$

$$\beta_1(v) = \beta_2(v) = 1$$

und M_1, M_2 die zugehörigen Partitionsmatroide. Bezüglich des graphischen Matroids müssen wir jetzt allerdings ebenfalls mit dem Digraphen \hat{G} arbeiten. Wir tun dieses, indem wir für jeden kreisfreien Untergraphen von G jede beliebige Orientierung seiner Kanten zulassen. Genauer setzen wir

$$\hat{\mathcal{I}} := \{\hat{I} \subset \hat{E} : (v, w) \in \hat{I} \Rightarrow (w, v) \notin \hat{I}\}$$

und für $\hat{I} \in \hat{\mathcal{I}}$

$$I(\hat{I}) := \{\{v, w\} : (v, w) \in \hat{I}\}$$

sowie

$$\mathcal{E} := \{\hat{I} \in \hat{\mathcal{I}} : I(\hat{I}) \text{ ist kreisfrei}\} \quad \wedge \quad M_3 := (\hat{E}, \mathcal{E}).$$

Dann ist M_3 ein Matroid. Ferner besitzt G genau dann einen Hamiltonweg, wenn $M_1 \cap M_2 \cap M_3$ eine unabhängige Menge der Kardinalität $|V| - 1$ besitzt.

Beispiel 2.3.43 zeigt, dass das Maximierungsproblem über dem Durchschnitt von drei Matroiden jedenfalls nicht leichter ist, als zu entscheiden, ob ein gegebener Graph einen Hamiltonweg besitzt.⁵⁰

Zum Abschluss dieser Sektion geben wir eine Gütegarantie für den Greedy-Algorithmus bei der Maximierung über dem Durchschnitt von k Matroiden an. Sie folgt mit Satz 2.3.23 aus einer Abschätzung des Rangverhältnisses. Diese basiert auf dem folgenden Korollar zu Lemma 2.3.15.

2.3.44 Bemerkung. Seien $k \in \mathbb{N}$, M_1, \dots, M_k Matroide, $U := (E, \mathcal{E})$ ihr Durchschnitt, $I \in \mathcal{E}$, $e \in E$ sowie $I \cup \{e\} \notin \mathcal{E}$. Dann enthält $I \cup \{e\}$ höchstens k Kreise.

Beweis: Ist $K \subset I \cup \{e\}$ ein Kreis in U , so gibt es mindestens ein $j_0 \in [k]$, so dass K in M_{j_0} abhängig ist. Nach Lemma 2.3.15 enthält $I \cup \{e\}$ andererseits aber nur einen einzigen Kreis in M_{j_0} . Somit kann $I \cup \{e\}$ insgesamt höchstens k Kreise in U enthalten, nämlich höchstens einen Kreis in M_j für jedes $j \in [k]$. \square

2.3.45 Lemma. Durch $(M_1, \dots, M_k; \phi)$ sei eine Maximierungsaufgabe über dem Durchschnitt $U := (E, \mathcal{E})$ von k Matroiden spezifiziert. Dann gilt für den Rangquotienten

$$k\rho(U) \geq 1.$$

Beweis: Seien $A^* \subset E$ und I, J Basen von U_{A^*} mit

$$r_-(U_{A^*}) = |I| \quad \wedge \quad r_+(U_{A^*}) = |J| \quad \wedge \quad r_-(U_{A^*}) = \rho(U)r_+(U_{A^*}).$$

O.B.d.A. sei $|I| \geq 1$. Ferner seien $s \in \mathbb{N}$ und $I \setminus J =: \{e_1, \dots, e_s\}$. Nach Bemerkung 2.3.44 enthält $J \cup \{e_1\}$ höchstens k Kreise. Da I unabhängig ist, schneidet jeder solche Kreis $J \setminus I$. Es gibt daher eine Teilmenge A_1 von $J \setminus I$ mit

$$|A_1| \leq k \quad \wedge \quad (J \setminus A_1) \cup \{e_1\} \in \mathcal{E}.$$

Sei nun $J_1 := (J \setminus A_1) \cup \{e_1\}$. Dann gilt $I \setminus J_1 = \{e_2, \dots, e_s\}$.

⁵⁰ Vgl. Satz 3.4.12.

Fahren wir sukzessive fort, so haben wir nach höchstens s Schritten eine Teilmenge A von $J \setminus I$ mit

$$|A| \leq ks \quad \wedge \quad (J \setminus A) \cup \{e_1, \dots, e_s\} \in \mathcal{E}.$$

Es folgt

$$I = (I \cap J) \cup (I \setminus J) \subset (J \setminus A) \cup (I \setminus J) \in \mathcal{E}$$

und, da I eine Basis ist, sogar

$$I = (J \setminus A) \cup (I \setminus J).$$

Daher gilt

$$J \setminus I = J \setminus ((J \setminus A) \cup (I \setminus J)) = A \cap J = A,$$

also

$$|J \setminus I| \leq ks = k \cdot |I \setminus J|.$$

Es folgt

$$|J| = |J \cap I| + |J \setminus I| \leq |I \cap J| + k \cdot |I \setminus J| \leq k \cdot |I|,$$

somit

$$|I| = \rho(U)|J| \leq k\rho(U)|I|,$$

und damit die Behauptung. \square

Hiermit erhalten wir folgendes Approximationsergebnis.

2.3.46 Satz. *Durch $(M_1, \dots, M_k; \phi)$ sei eine Maximierungsaufgabe über dem Durchschnitt von k Matroiden spezifiziert; φ sei die zugehörige Zielfunktion. Ferner bezeichne I^* eine optimale, I^g eine Greedy-Lösung. Dann gilt*

$$\varphi(I^*) \leq k \cdot \varphi(I^g).$$

Beweis: Die Aussage folgt mit Lemma 2.3.45 direkt aus Satz 2.3.23. \square

Nach Satz 2.3.46 kann man also etwa in Beispiel 2.3.41 stets (in beliebiger, die oberen Schranken respektierender Weise) mindestens den Anteil $1/m$ der optimalen Punktzahl in K platzieren.

2.4 Übungsaufgaben

2.4.1 Übungsaufgabe. Gegeben sei das lineare Ungleichungssystem

$$\begin{array}{rclcl} \xi_1 & + & \xi_2 & - & 3\xi_3 & \leq & 1 & (1) \\ \xi_1 & - & \xi_2 & + & 3\xi_3 & \leq & 1 & (2) \\ -\xi_1 & + & \xi_2 & - & \xi_3 & \leq & 1 & (3) \\ -\xi_1 & - & \xi_2 & + & 5\xi_3 & \leq & 1 & (4) \\ & & & & -\xi_3 & \leq & 0 & (5). \end{array}$$

Man wende die Fourier-Motzkin-Elimination an, um zu entscheiden, ob das System lösbar ist, und bestimme ggf. einen zulässigen Punkt.

2.4.2 Übungsaufgabe. Das lineare Ungleichungssystem bestehe aus den Ungleichungen (1) – (4) von Übungsaufgabe 2.4.1 sowie der neuen strikten Ungleichung

$$\xi_3 < 0 \quad (5').$$

Man führe eine (entsprechend angepasste) Fourier-Motzkin-Elimination durch, entscheide, ob das System lösbar ist, und bestimme ggf. einen zulässigen Punkt.

2.4.3 Übungsaufgabe. Man benutze die Fourier-Motzkin-Elimination, um einen Optimalpunkt des linearen Programms

$$\begin{array}{rcll} \max & \xi_1 + \xi_2 & & \\ -2\xi_1 & - & \xi_2 & \leq -4 \\ \xi_1 & - & 2\xi_2 & \leq 0 \\ 2\xi_1 & + & \xi_2 & \leq 11 \\ -2\xi_1 & + & 6\xi_2 & \leq 17 \end{array}$$

zu bestimmen.

2.4.4 Übungsaufgabe. Man konstruiere ein Ungleichungssystem in n Variablen, für das zwei Eliminationsreihenfolgen der Variablen existieren, so dass in allen (mit Ausnahme von höchstens konstant vielen) Schritten der Fourier-Motzkin-Elimination die Anzahl der Ungleichungen des entstehenden Systems bei der ersten fällt, bei der zweiten aber wächst.

2.4.5 Übungsaufgabe. Das Polytop P sei durch die \mathcal{H} -Darstellung (m, n, A, b) gegeben, d.h. es gilt $P := \{x \in \mathbb{R}^n : Ax \leq b\}$. Auf das System $Ax \leq b$ werde nun die Fourier-Motzkin-Elimination angewendet, um alle Variablen zu eliminieren. Man beweise oder widerlege die folgenden Aussagen:

- (a) Erhält man für jede Eliminationsreihenfolge der Variablen eine Ungleichung der Form $\gamma \leq \delta$ mit $\gamma < \delta$, so besitzt P innere Punkte.
- (b) Besitzt P innere Punkte, so erhält man für jede Eliminationsreihenfolge der Variablen eine Ungleichung der Form $\gamma \leq \delta$ mit $\gamma < \delta$.

2.4.6 Übungsaufgabe. Seien $A \in \mathbb{R}^{m \times n}$ und $b \in \mathbb{R}^m$. Man beweise die folgenden Aussagen:

- (a) Seien $c \in \mathbb{R}^n$, $\gamma \in \mathbb{R}$, und die Ungleichung $c^T x \leq \gamma$ sei durch Schritte der Fourier-Motzkin-Elimination auf das Systems $Ax \leq b$ hervorgegangen. Dann gibt es einen Vektor $y \in \mathbb{R}^m$ mit

$$y \geq 0 \quad \wedge \quad y^T A = c \quad \wedge \quad y^T b = \gamma.$$

- (b) $Ax \leq b$ ist genau dann unlösbar, wenn es einen Vektor $y \in [0, \infty]^m$ mit $y^T A = 0$ und $y^T b < 0$ gibt.⁵¹

2.4.7 Übungsaufgabe. Sei A ein Algorithmus, der als Input Paare (S, L) akzeptiert, die aus einem linearen Ungleichungssystem S und einer zusätzlichen linearen Ungleichung L bestehen, und der korrekt entscheidet, ob L für S redundant ist. Man zeige, dass mit Hilfe von endlich vielen Aufrufen von A entschieden werden kann, ob S zulässig ist.

2.4.8 Übungsaufgabe. Um Patienten mit einer Injektion gleich gegen mehrere Infektionskrankheiten zu impfen, werden in modernen Kombinationsimpfstoffen verschieden Vakzine gleichzeitig verabreicht. Für die Zusammenstellung eines solchen Impfstoffs stehen n verschiedene Seren s_1, \dots, s_n zur Auswahl. Ihre einzelnen 'Impfwerte' seien unabhängig voneinander durch positive reelle Zahlen $\gamma_1, \dots, \gamma_n$ quantifiziert und addieren sich bei ihrer Kombination. Bei der Zusammenstellung eines Kombinationsimpfstoffs gibt es keine Einschränkung bezüglich der Anzahl der beteiligten Vakzine. Allerdings sind nicht alle miteinander verträglich. Für jedes $i \in [n]$ gibt es daher eine 'schwarze Liste' $S_i \subset \{s_1, \dots, s_n\}$ solcher Impfstoffe, die mit s_i nicht kombiniert werden dürfen. Gesucht ist ein Kombinationsimpfstoff mit größtem Impfwert.

- (a) Man beschreibe die gegebene Aufgabe als Suche nach einer passenden Struktur in einem geeigneten Graphen.
- (b) Man formuliere die Fragestellung als ganzzahlige lineare Optimierungsaufgabe.

Eine naheliegende Variante der Aufgabe fragt nach einer Menge von Kombinationsimpfstoffen kleinster Kardinalität, deren Vereinigung jedoch alle Wirkstoffe $\{s_1, \dots, s_n\}$ enthält.

- (c) Man formuliere auch diese Variante als ganzzahlige lineare Optimierungsaufgabe.

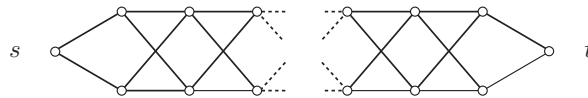
2.4.9 Übungsaufgabe. Seien $G := (V, E)$ ein Graph, τ_V bzw. τ_E jeweils eine Reihenfolge auf V bzw. E . Man leite eine (nicht völlig triviale) Identität zwischen der zugehörigen Inzidenzmatrix S_G und der entsprechenden Adjazenzmatrix A_G her.

2.4.10 Übungsaufgabe. Seien $G := (V, E)$ ein Graph, $n := |V|$ und $k \in \mathbb{N}$. Dann heißt G k -regulär, wenn $\deg_G(v) = k$ für alle $v \in V$ gilt. Man beweise die folgenden Aussagen:

⁵¹ Diese Aussage ist auch als Lemma von Farkas bekannt; vgl. Beispiel 4.2.10 und Korollar 4.2.36.

- (a) G ist genau dann k -regulär, wenn **1** Eigenvektor der Adjazenzmatrix A_G zum Eigenwert k ist.
- (b) Der Graph G sei k -regulär, $y := (\eta_1, \dots, \eta_n)^T$ sei Eigenvektor zum Eigenwert λ von A_G , und es sei $\lambda \neq k$. Dann gilt $\sum_{i=1}^n \eta_i = 0$.
- (c) Der Graph G sei k -regulär. G ist genau dann zusammenhängend, wenn der Eigenwert k die Vielfachheit 1 besitzt.
- (d) Man verallgemeinere (c) auf den Fall, dass G mehr als eine Zusammenhangskomponente besitzt.

2.4.11 Übungsaufgabe. In nachfolgender Skizze ist eine von einem Parameter $r \in \mathbb{N}$ abhängige Familie von Graphen G_r mit $2r + 1$ Knoten dargestellt.



Skizze zu Übungsaufgabe 2.4.11

Man bestimme die Anzahl der s - t -Wege in G_r .

2.4.12 Übungsaufgabe. Seien s, t zwei verschiedene Knoten des vollständigen Graphen K_n , und α_n bezeichne die Anzahl der verschiedenen s - t -Wege in K_n .

- (a) Man beweise

$$\alpha_n = (n-2)! \sum_{k=0}^{n-2} \frac{1}{k!}.$$

- (b) Man zeige

$$\alpha_n \geq \left\lceil 2\sqrt{2\pi}(n-2)^{n-\frac{3}{2}} e^{-n+2} \right\rceil.$$

- (c) Gegeben sei ein Algorithmus \mathcal{A} , der auf der Suche nach einem kürzesten s - t -Weg der Reihe nach alle solchen Wege durchprobiert. Unter der (sehr optimistischen) Annahme, dass \mathcal{A} pro Weg mit nur einer Gleitkommaoperation auskommt, wie lange benötigt der Algorithmus für $n = 22$ bzw. $n = 26$ dann auf einem Teraflop-Rechner?

2.4.13 Übungsaufgabe. Sei $n \in \mathbb{N}$. Man zeige, dass es in K_n genau n^{n-2} verschiedene Spannbäume gibt.

Hinweis: Man zeige zunächst, dass man einen Baum T auf $[n]$ mit einem $(n-2)$ -Tupel $(\eta_1, \dots, \eta_{n-2})$ aus $[n]$ identifizieren kann, dem sogenannten Prüfer-Code⁵². Es wird iterativ durch Entfernen des Blattes mit kleinster Nummer konstruiert, bis nur noch zwei Knoten übrig sind. Hat im Schritt i der eindeutig bestimmte Nachbar des zu entfernenden Knotens in T die Nummer j , so wird $\eta_i := j$ gesetzt.

2.4.14 Übungsaufgabe. Seien $G = (V, E, \nu)$ ein allgemeiner Graph und $U, W \subset V$. Man gebe einen detaillierten Beweis von Bemerkung 2.2.24 und zeige die folgenden Aussagen:

$$\begin{aligned} |\delta(G, U)| + |\delta(G, W)| &= |\delta(G, U \cup W)| + |\delta(G, U \cap W)| + 2|E(U, W)|; \\ |\delta_{\text{in}}(G, U)| + |\delta_{\text{in}}(G, W)| &= |\delta_{\text{in}}(G, U \cup W)| + |\delta_{\text{in}}(G, U \cap W)| + |E_{\text{in}}(U, W)| + |E_{\text{in}}(W, U)|; \\ |\delta_{\text{aus}}(G, U)| + |\delta_{\text{aus}}(G, W)| &= |\delta_{\text{aus}}(G, U \cup W)| + |\delta_{\text{aus}}(G, U \cap W)| + |E_{\text{aus}}(U, W)| + |E_{\text{aus}}(W, U)|. \end{aligned}$$

2.4.15 Übungsaufgabe. Seien $A \in \{0, 1\}^{m \times n}$. Man beweise oder widerlege die folgenden Aussagen für das Verhältnis der Ränge von A bezüglich der Körper \mathbb{Z}^2 bzw. \mathbb{R} :

- (a) $\text{rang}_{\mathbb{Z}^2}(A) \leq \text{rang}_{\mathbb{R}}(A)$; (b) $\text{rang}_{\mathbb{R}}(A) \leq \text{rang}_{\mathbb{Z}^2}(A)$.

2.4.16 Übungsaufgabe. Seien G ein allgemeiner Graph mit n Knoten und γ die Anzahl seiner schwachen Zusammenhangskomponenten.

- (a) Man zeige $\text{rang}_{\mathbb{Z}^2}(S_G) = n - \gamma$.
- (b) Sei G ein Digraph; man bestimme $\text{rang}_{\mathbb{R}}(S_G)$.

⁵² Heinz Prüfer, 1896 – 1933.

2.4.22 Übungsaufgabe. Seien $G := (V, E)$ ein bipartiter Graph, $x, y \in V$, M ein Matching in G , das x und y nicht überdeckt, $T := (V_T, E_T)$ ein Augmentationsbaum zu y in G , der x nicht enthält, und es gebe keinen Augmentationsweg bez. M in G mit Endknoten y . Seien ferner S der durch $V \setminus V_T$ induzierte Teilgraph von G und $F := (V_F, E_F)$ ein Augmentationsbaum zu x in S .

- Man zeige, dass genau dann ein Augmentationsweg bez. M in G mit Endknoten x existiert, wenn es einen solchen in F gibt. Bleibt diese Aussage auch dann richtig, wenn es einen Augmentationsweg bez. M in G mit Endknoten y gibt? (Beweis oder Gegenbeispiel)
- Man benutze (a), um eine 'verkürzte Variante' von Prozedur 2.2.62 anzugeben.

2.4.23 Übungsaufgabe. Seien $M := (E, \mathcal{E})$ ein Matroid, $I \in \mathcal{E}$ und $e \in E$. In Lemma 2.3.15 wurde gezeigt, dass $I \cup \{e\}$ entweder unabhängig ist oder genau einen Kreis enthält.

- Sei $M := (E, \mathcal{E})$ ein Matrix-Matroid. Man formuliere und beweise die Aussage in der Sprache der linearen Algebra.
- Sei $M := (E, \mathcal{E})$ ein graphisches Matroid. Man formuliere und beweise die Aussage in der Sprache der Graphentheorie.

2.4.24 Übungsaufgabe. Seien G ein Graph, $\mathcal{E}(G)$ die Menge der Kantenmengen von Matchings in G . Man charakterisiere alle Graphen $G := (V, E)$, für die $(E, \mathcal{E}(G))$ ein Matroid ist.

2.4.25 Übungsaufgabe. Sei $G := (V, E)$ ein Graph. Ferner bezeichne \mathcal{E} die Menge aller Teilmengen $I \subset V$, so dass ein Matching M in G existiert, das I überdeckt, d.h. das alle Knoten von I enthält. Man zeige, dass (V, \mathcal{E}) ein Matroid ist.

2.4.26 Übungsaufgabe. Man formuliere die folgenden Probleme in Graphen als Maximierungsproblem über einem Unabhängigkeitssystem bzw. Minimierungsproblem über einem Basissystem:

- das Problem SPP kürzester Wege;
- das Traveling Salesman Problem TSP;
- das Problem maximaler Matchings.

2.4.27 Übungsaufgabe. Statt des Greedy-Algorithmus von Prozedur 2.3.17 verwende man für das Maximierungsproblem über Unabhängigkeitssystemen die folgende 'Streichungsvariante':

```

INPUT:   Unabhängigkeitssystem  $(E, \mathcal{E})$ ,  $\phi : E \rightarrow [0, \infty[$ 
OUTPUT:   $S^g \in \mathcal{E}$ 
BEGIN     $A \leftarrow E$ ;  $S^g \leftarrow E$ 
          WHILE  $A \neq \emptyset$  DO
            BEGIN
              Sei  $a \in \operatorname{argmin}_{a \in A} \phi(e)$ ; setze  $A \leftarrow A \setminus \{a\}$ 
              IF  $S^g \setminus \{a\}$  enthält eine Basis THEN  $S^g \leftarrow S^g \setminus \{a\}$ 
            END
          END
END
```

Man zeige, dass diese Variante genau dann das Maximierungsproblem korrekt löst, wenn (E, \mathcal{E}) ein Matroid ist.

2.4.28 Übungsaufgabe. Zu einem gegebenen Graphen $G := (V, E)$ sei $G' := (V', E')$ der durch Unterteilung jeder Kante $e \in E$ mittels einer neuen Ecke entstehende bipartite Graph, d.h. es ist

$$V' := V \cup E \quad \wedge \quad E' := \{\{v, e\} : v \in V \wedge e \in E \wedge v \in e\}.$$

Ferner seien M_1 das durch

$$\{\delta(G', v) : v \in V\} \quad \wedge \quad \beta(v) = 2 \quad (v \in V)$$

definierte Partitionsmatroid und M_2 das graphische Matroid zu G' . Schließlich bezeichne $\zeta(G)$ das Maximum der Kardinalitäten unabhängiger Mengen in $M_1 \cap M_2$.

Man beweise oder widerlege die folgende Aussage: Es gibt es eine Funktion $\psi : \mathbb{N}_0 \times \mathbb{N}_0 \rightarrow \mathbb{N}_0$, so dass ein Graph $G := (V, E)$ genau dann einen Hamiltonweg besitzt, wenn $\zeta(G) \geq \psi(|V|, |E|)$ ist.

Grundlagen der Mathematischen Optimierung

Diskrete Strukturen, Komplexitätstheorie,

Konvexitätstheorie, Lineare Optimierung,

Simplex-Algorithmus, Dualität

Gritzmann, P.

2013, XVII, 525 S. 160 Abb., Softcover

ISBN: 978-3-528-07290-2