Univ.-Prof. Dr. Klaus Neumann
Univ.-Prof. Dr. Dietrich Ohse
Univ.-Prof. Dr. Wilhelm Rödder
Dr. Friedhelm Kulmann
Dr. Heinz Peter Reidmacher

31801
Problemlösen in graphischen Strukturen

Leseprobe

Fakultät für
**Wirtschafts-
wissenschaft**

FernUniversität in Hagen

Der Inhalt dieses Studienbriefs wird gedruckt auf Recyclingpapier (80 g/m^2, weiß), hergestellt aus 100 % Altpapier.

Modul 31801
»Problemlösen in graphischen Strukturen«

Einheit 1
»Grundlagen der Graphentheorie und Netzwerkoptimierung«

1. Grundbegriffe
- 1.1. Praktische Probleme, die auf Graphen und Netzwerke führen
- 1.2. Grundlegende Definitionen
- 1.3. Kantenfolgen in Graphen und Pfeilfolgen in Digraphen
- 1.4. Erreichbarkeit und Zusammenhang in Graphen und Digraphen
- 1.5. Charakteristische Matrizen zu Graphen und Digraphen
- 1.6. Für Anwendungen wichtige Klassen von Graphen und Digraphen
- 1.7. Bewertete Graphen und Netzwerke

2. Graphen und Computer
- 2.1. Rechenaufwand von Algorithmen
- 2.2. Einige wichtige Graphen-Algorithmen

3. Minimalgerüste und kürzeste Wege
- 3.1. Minimalgerüste
- 3.2. Kürzeste Wege von einem zu allen Knoten (Baumalgorithmen)
- 3.3. Bellmans Verfahren für zyklenfreie Netzwerke
- 3.4. Der Algorithmus von Dijkstra bei nichtnegativer Bewertung
- 3.5. Das Verfahren von Ford für Netzwerke mit beliebiger Bewertung
- 3.6. Der Tripel-Algorithmus von Floyd und Warshall

4. Flüsse in Netzwerken
- 4.1. Flüsse und Schnitte in Netzwerken
- 4.2. Flüsse in Graphen und Zirkulationsflüsse
- 4.3. Der Algorithmus von Ford und Fulkerson
- 4.4. Bestimmung eines zulässigen Anfangsflusses
- 4.5. Kostenminimale Flüsse und Zirkulationsflüsse
- 4.6. Optimalitätsbedingungen für Zirkulationsflüsse
- 4.7. Der Out-of-Kilter-Algorithmus

Lösungen zu den Übungsaufgaben

1
Grundbegriffe

In Kapitel 1 werden Sie grundlegende mit Graphen und Netzwerken zusammenhängende Begriffe kennenlernen, die in diesem Modul benötigt werden. Es ist in der Graphentheorie von Bedeutung, dass auch für die Beschreibung einfacher Modelle und Probleme eine formale Notation und genaue Bezeichnungsweisen verwendet werden. Viele der Begriffe sind unmittelbar einleuchtend und durch ähnliche oder gleich lautende Bezeichnungen aus dem gewöhnlichen Sprachgebrauch leicht ableitbar. Umso wichtiger ist es, hier Klarheit zu schaffen.

1.1 Praktische Probleme, die auf Graphen und Netzwerke führen

Um einen Eindruck von der Verschiedenartigkeit der Probleme zu vermitteln, die mit Hilfe von Graphen beschrieben werden können, stehen in diesem einführenden Abschnitt zunächst einige Beispiele.

Beispiel 1.1

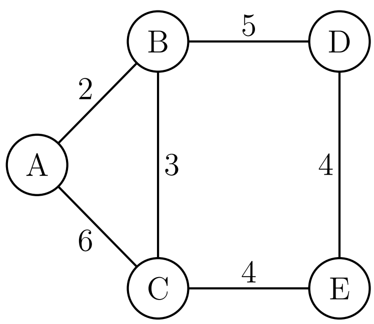

Abbildung 1.1: Straßennetz

5 Orte A, B, C, D, E sind durch ein Straßennetz miteinander verbunden, das in Abbildung 1.1 veranschaulicht ist. Die Orte sind durch sogenannte *Knoten* und die direkten Straßenverbindungen zwischen benachbarten Orten durch Linien, im Folgenden *Kanten* genannt, dargestellt. Die Entfernungen (in [km]) sind an den betreffenden Kanten notiert. Gesucht sind die (kürzesten) Entfernungen zwischen je 2 Orten sowie die entsprechenden Straßenverbindungen.

Knoten

Kante

Beispielsweise beträgt die Entfernung der Orte A und E 9 km, und die zugehörige Straßenverbindung führt über die Orte B und C.

◇[1]

Das in Abbildung 1.1 veranschaulichte System von Knoten und Kanten wird *Graph* genannt. Da die Kanten des Graphen mit Zahlen (in Beispiel 1.1 Entfernungen) „bewertet" sind, spricht man auch von einem *bewerteten Graphen*.

Graph

bewerteter Graph

[. . .]

3.2 Kürzeste Wege von einem zu allen Knoten (Baumalgorithmen)

[. . .]

Wir legen stets ein Netzwerk $\vec{N} = \langle V, E; c\rangle$ mit reellwertiger Gewichtsfunktion c und Knotenmenge $V = \{1, \dots, n\}$ zugrunde. Wie in Unterkapitel 1.7 setzen wir

$$c_{ij} := \begin{cases} 0, \text{ falls } i = j \\ c\langle i, j\rangle, \text{ falls } \langle i, j\rangle \in E \\ \infty, \text{ sonst} \end{cases} \qquad (i, j = 1, \dots, n)$$

(vgl. (1.5)). Es wird vorausgesetzt, dass $\vec{N}$ keine Zyklen negativer Länge enthält. Wir suchen kürzeste Wege von einem Startknoten a in $\vec{N}$ zu allen übrigen von a aus erreichbaren Knoten j von $\vec{N}$ sowie die entsprechenden Weglängen (Entfernungen).

Sei

$$d_j := \begin{cases} 0, \text{ falls } j = a \\ d\langle a, j\rangle, \text{ falls } j \in \dot{R}(a) \\ \infty, \text{ falls } j \notin R(a) \end{cases} \qquad (j = 1, \dots, n) \tag{3.1}$$

wobei $d\langle a, j\rangle$ die Entfernung von a nach j ist (vgl. Unterkapitel 1.7). Für $j \in \dot{R}(a)$ sei $F_j = \langle a, \dots, k, j\rangle$ ein kürzester Weg von a nach j. $\langle k, j\rangle$ ist also der „letzte Pfeil" auf diesem Weg. Der Teil $F_k = \langle a, \dots, k\rangle$ des Weges F_j muss ebenfalls ein kürzester Weg (und zwar von a nach k) sein. Gäbe es nämlich einen kürzeren Weg von a nach k, etwa F'_k, so würde F'_k zusammen mit dem Pfeil $\langle k, j\rangle$ einen Weg von a nach j liefern, der kürzer als der kürzeste Weg F_j wäre. Es gilt also $d_j = d_k + c_{kj}$.

[. . .]

[1] Die Raute ◇ wird in diesem Modul verwendet, um das Ende eines Beispiels zu kennzeichnen.

Einheit 2
»Standortplanung und Transportoptimierung«

1. Optimale Standortplanung
- 1.1. Standardfragen der Standortplanung
- 1.2. Mediane und Zentren
- 1.3. Überdeckungsprobleme
- 1.4. Warehouse Location Probleme

2. Modellierung von Transportproblemen

3. Primale Verfahren für das Transportproblem
- 3.1. Die Lösung des Transportproblems mit der Simplex-Methode
- 3.2. Die Stepping-Stone-Methode
- 3.3. Bestimmung einer zulässigen Ausgangslösung
- 3.4. Implementierung primaler Methoden
- 3.5. Ganzzahligkeit und vollständige Unimodularität

4. Primal-duale Verfahren für Transport- und Umladeprobleme
- 4.1. Das Umladeproblem
- 4.2. Der LP-Ansatz für das Umladeproblem
- 4.3. Das Out-of-Kilter-Verfahren
- 4.4. Sonderprobleme

5. Die Ungarische Methode
- 5.1. Das Zuordnungsproblem
- 5.2. Duale Zulässigkeit
- 5.3. Flussänderung
- 5.4. Potentialänderung
- 5.5. Netzwerkorientierte Darstellung der Ungarischen Methode
- 5.6. Die Ungarische Methode in Transporttableauform

Lösungen zu den Übungsaufgaben

1.2 Mediane und Zentren

1.2.1 Grundbegriffe und Definitionen

Modellgrundlage ist ein bewerteter ungerichteter Graph $G = [V, E, c; b]$, wobei die Bewertung c zunächst o. B. d. A. als Entfernung interpretiert wird. Gewichte b sind den Knoten zugeordnet, und es können etwa vorhandene Angebote oder Bedarfe quantifiziert werden. Die Länge einer kürzesten Kantenfolge mit den Endknoten i, j in einem bewerteten Graphen G heißt *Entfernung* (oder *Distanz*) der Knoten i und j, in Zeichen $d[i, j]$. Enthält G keine Kreise negativer Länge, so existiert für je zwei verschiedene Knoten $i, j \in V$, die miteinander verbunden sind, eine kürzeste Kantenfolge mit den Endknoten i, j und folglich die Entfernung $d[i, j]$. Sind zwei Knoten i, j von G nicht miteinander verbunden, so ist $d[i, j] = \infty$. Für die eingangs formulierten Lokationsprobleme verfolgt man naturgemäß eine knotenorientierte Betrachtung. So ist für ein Unternehmen etwa die Frage von Interesse, wie der Transportaufwand vom Standort aus zu allen Nachfragern bei unterschiedlichen Bedarfen bewertet werden kann. Formal bestimmt man hierzu für einen Knoten i die gewichtete Distanz $\sigma(i)$ zu allen Knoten $j \in V, j \neq i$. Für diese Berechnung ist es erforderlich, dass die kürzesten Entfernungen vom Knoten i zu allen anderen Knoten bekannt sind. Sollten diese Informationen nicht vorliegen, so ist vorab zunächst ein geeigneter Algorithmus zur Erstellung einer vollständigen Entfernungsmatrix anzuwenden (vgl. Unterkapitel 3.2 ff. in Einheit 1).

Entfernung/Distanz

$$\sigma(i) = \sum_{j \in V} d[i,j] \cdot b_j$$

[. . .]

Übungsaufgabe 1.2

Die fünf Dörfer **A**alhus, **B**orscheidt, **C**hurtingen, **D**alenkamp und **E**stringen im Hochsauerland sind durch nur wenige Straßen miteinander verbunden; die Längen der direkten Verbindungen entnehmen Sie bitte folgender Matrix D. (Erinnerung: ∞ bedeutet, es gibt keine direkte Verbindung.)

$$D := \begin{pmatrix} 0 & 2 & \infty & \infty & \infty \\ 2 & 0 & 2 & \infty & 7 \\ \infty & \infty & 0 & \infty & 8 \\ 3 & \infty & 9 & 0 & \infty \\ \infty & 7 & \infty & 3 & 0 \end{pmatrix}$$

a) Zeichnen Sie einen Graphen, durch den die obige Situation visualisiert wird und der alle Angaben enthält. Welche Verbindungen sind nur in einer Richtung befahrbar?

b) Ein Dorf soll durch die Ansiedlung eines Einkaufszentrums aufgewertet werden. Analysen haben ergeben, dass von Aalhus, Borscheidt und Churtingen aus mit je 1000 Einkäufen pro Tag zu rechnen ist. Dalenkamp und Estringen dagegen sind größer; von hier aus werden 2000 Einkäufe täglich erwartet. In welchem Dorf sollte das Einkaufszentrum gebaut werden, damit die Gesamtfahrstrecke für alle Tageseinkäufe minimal ist.

c) Eine weitere Standortfrage beschäftigt die dortige Feuerwehr. In welchem Dorf sollte die neue Feuerwehrstation gebaut werden, damit bei einem Einsatz der Weg zum am weitesten entfernt liegenden Dorf am kürzesten ist.

[. . .]

2
Modellierung von Transportproblemen

[. . .]

Beispiel 2.1
Ein homogenes Gut, das von den Lieferanten L_1, L_2, L_3 in den Mengen a_1, a_2, a_3 [ME] angeboten wird, soll zu den Kunden K_1, K_2, K_3, K_4 mit den Nachfragen b_1, b_2, b_3, b_4 [ME] transportiert werden. Die Kosten des Transportes einer Mengeneinheit vom Lieferanten L_i zum Kunden K_j seien c_{ij} [GE] ($i = 1,\ldots,3$; $j = 1,\ldots,4$). Alle Zahlenwerte sind in Tabelle 2.1 zusammengestellt.

Tabelle 2.1: Ausgangsdaten des Transportproblems

		Kunden				Angebot
		K_1	K_2	K_3	K_4	a_i
Lieferanten	L_1	12	8	11	7	100
	L_2	9	13	10	6	150
	L_3	11	7	9	14	90
Nachfrage	b_j	80	140	70	50	

Wie viele Mengeneinheiten x_{ij} sollen vom Lieferanten L_i zum Kunden K_j transportiert werden, damit die Nachfrage erfüllt ist und die gesamten Transportkosten minimiert werden?

◇

Transportproblem

Allgemein formuliert lautet das *Transportproblem*: Ein homogenes Gut, das an den Orten A_i, ($i = 1,\ldots,m$) in den Mengen a_i [ME] angeboten wird, soll zu den Orten B_j, ($j = 1,\ldots,n$) transportiert werden, an denen eine Nachfrage von b_j [ME] besteht. Die Kosten des Transportes einer Mengeneinheit von Ort A_i zum Ort B_j betragen c_{ij}. Wieviele ME sollen vom Ort A_i zum Ort B_j transportiert werden, so dass die Nachfrage erfüllt wird und die Gesamttransportkosten minimal sind?

[...]
Das dem Beispiel 2.1 analoge Datenschema hat die Form der Tabelle 2.2.

Tabelle 2.2: Daten zum Transportproblem

		Nachfrageorte				Angebot
		B_1	B_2	$\cdots$	B_n	a_i
Angebotsorte	A_1	c_{11}	c_{12}	$\cdots$	c_{1n}	a_1
	A_2	c_{21}	c_{22}	$\cdots$	c_{2n}	a_2
	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$
	A_m	c_{m1}	c_{m2}	$\cdots$	c_{mn}	a_m
Nachfrage	b_j	b_1	b_2	$\cdots$	b_n	

Das Problem der Transportkostenminimierung wird in dieser Form allgemein als das *klassische Transportproblem* bezeichnet. Es handelt sich um den einfachsten Spezialfall eines allgemeinen Transportproblems, das in Kapitel 4 ausführlich behandelt wird.

klassisches Transportproblem

[...]

3.3 Bestimmung einer zulässigen Ausgangslösung

[...]

Gerade die Nord-West-Ecken-Methode zeigt wie man auch bei Berücksichtigung der Zielfunktionskoeffizienten vorzugehen hat, um eine zulässige Basislösung zu erhalten. Kennzeichnend ist, dass mit jeder Zuordnung mindestens eine Nebenbedingung erfüllt wird, d. h. entweder eine Angebots- oder eine Nachfragerestriktion. Die entsprechende Zeile oder Spalte gilt dann für den weiteren Auswahlprozess als „gesperrt“. In den folgenden Beispielen ist die Sperrung durch eine Nummerierung der erfüllten Nebenbedingung vermerkt, woraus sich gleichzeitig ein Hinweis auf die Bearbeitungsfolge ergibt. Wird gleichzeitig ein Angebot und eine Nachfrage befriedigt, so muss im nächsten Schritt eine *Nullzuordnung* vorgenommen werden, um auf die für nachfolgende Verbesserung erforderliche Zahl von $m + n - 1$ Basisfeldern zu kommen. Die Lösung wird dann als *degeneriert* bezeichnet. Die noch nicht erfüllten Zeilen und/oder Spalten sind frei für weitere Zuordnungen.

Nullzuordnung

Degeneration

[...]

5.1 Das Zuordnungsproblem

[...]
Als Sonderfall des klassischen Transportproblems ($a_i = 1,\ i = 1,\ldots,n;\ b_j = 1,\ j = 1,\ldots,n$) lässt sich das Zuordnungsproblem selbstverständlich als Fluss- bzw. als *Zirkulationsflussproblem* darstellen und formulieren.

Zirkulations-problem

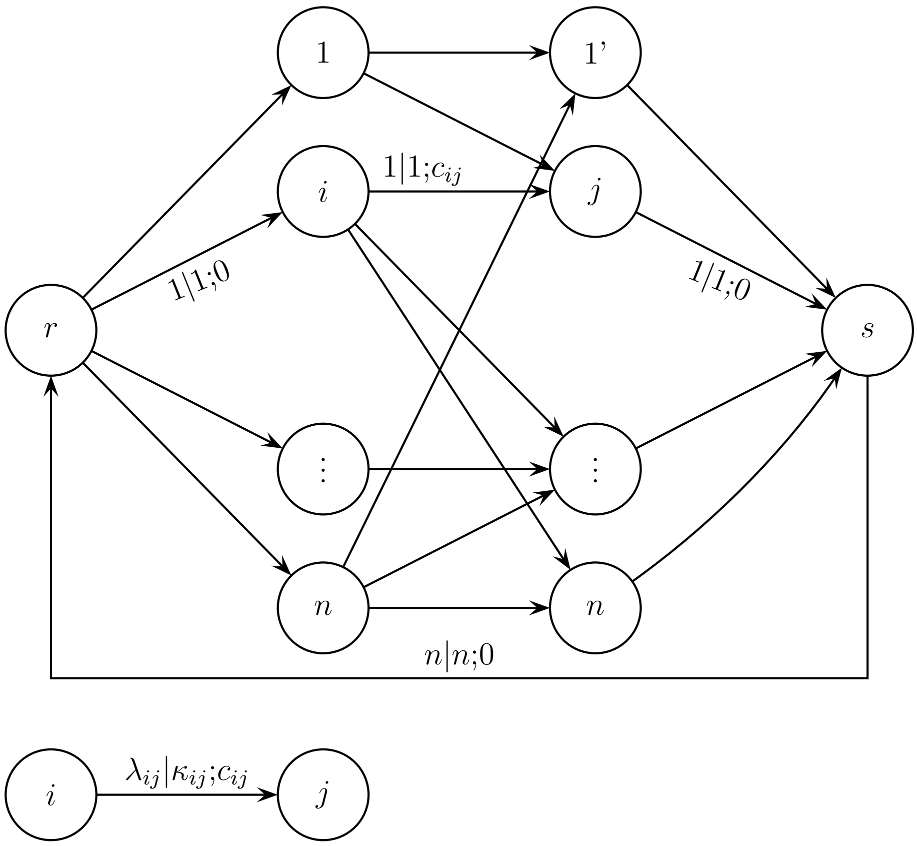

Abbildung 5.2: Darstellung des Zuordnungsproblems als Zirkulationsproblem in $\vec{N} = \langle V, E; \lambda,\kappa; c\rangle$

Lösungen der Übungsaufgaben

Beachten Sie auch die Übungsangebote des Lehrstuhls für Betriebswirtschaftslehre, insb. Quantitative Methoden und Wirtschaftsmathematik unter:
http://www.fernuni-hagen.de/bwlquam/studium/uebungen.shtml

[...]

Lösungshinweis zu Übungsaufgabe 1.2

a)

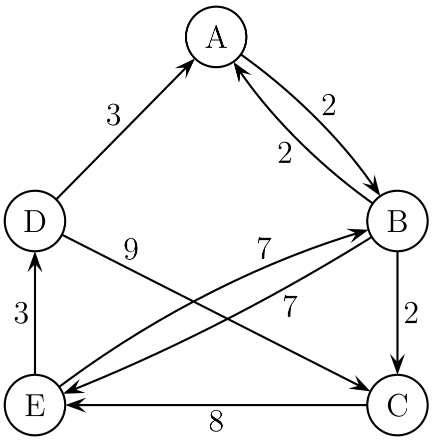

Abbildung L.1: Gerichteter Graph $\vec{G}$ zum Straßennetz

Dem Graphen $\vec{G}$ in Abbildung L.1 ist zu entnehmen, dass die Verbindungen von $\langle B, C\rangle$, $\langle C, E\rangle$, $\langle D, A\rangle$, $\langle D, C\rangle$ und $\langle E, D\rangle$ nur in einer Richtung befahrbar sind.

b) Für den Einkauf sind jeweils die Hin- und Rückfahrt auf dem kürzesten Weg zu betrachten. Aus dem in a) erstellten Graphen sind deshalb entsprechend die kürzesten Entfernungen zwischen allen Knoten abzulesen und in einer Distanzmatrix zusammenzustellen. Für einen komplexeren Graphen liefert der Tripelalgorithmus das gewünschte Ergebnis.

$$D^* := \begin{pmatrix} 0 & 2 & 4 & 12 & 9 \\ 2 & 0 & 2 & 10 & 7 \\ 14 & 15 & 0 & 11 & 8 \\ 3 & 5 & 7 & 0 & 12 \\ 6 & 7 & 9 & 3 & 0 \end{pmatrix}$$

Da die Anzahl der zu erwartenden Einkäufe je Dorf unterschiedlich ist, muss eine zusätzliche Gewichtung vorgenommen werden. Die Ergebnisse (in 1000 km) sind in der nachfolgenden Matrix zusammengefasst:

$$\begin{array}{cc}\begin{bmatrix} 0 & 2 & 4 & 12 & 9 \\ 2 & 0 & 2 & 10 & 7 \\ 14 & 15 & 0 & 11 & 8 \\ 3 & 5 & 7 & 0 & 12 \\ 6 & 7 & 9 & 3 & 0 \end{bmatrix} & \begin{array}{c} 48 \\ 38 \\ 67 \\ 39 \\ 28 \end{array} \\ \begin{array}{ccccc} 34 & 41 & 38 & 39 & 48 \end{array} & \end{array}$$

Somit ergibt sich beispielsweise bei einer Wahl des Standorts „Aalhus“, dass von allen Einkaufenden aus den verschiedenen Dörfern insgesamt zunächst einmal 34.000 km zurückzulegen sind, um zum Kaufhaus zu gelangen, und dann 48.000 km, um wieder nach Hause zu kommen. In der Summe berechnet sich für A: 82.000 km, für B: 79.000 km, für C: 105.000 km, für D: 78.000 km und für E: 76.000 km. Das Kaufhaus sollte somit unter den gegebenen Annahmen in Estringen gebaut werden.

c) Für den Bau des Feuerwehrhauses ergibt sich folgende Situation. Hier muss nur der zurückzulegende Weg zum Einsatzort betrachtet werden. Da die weiteste Entfernung minimal sein soll, ist allerdings auch hier der Standort Estringen zu wählen. In folgender Übersicht ist die weiteste Entfernung vom Standort aus rechts neben der Matrix notiert. Für Estringen ist der Wert mit 9 km minimal.

$$\begin{bmatrix} 0 & 2 & 4 & 12 & 9 \\ 2 & 0 & 2 & 10 & 7 \\ 14 & 15 & 0 & 11 & 8 \\ 3 & 5 & 7 & 0 & 12 \\ 6 & 7 & 9 & 3 & 0 \end{bmatrix} \begin{array}{c} 12 \\ 10 \\ 15 \\ 12 \\ 9 \end{array}$$

Einheit 3
»Optimierung mit Intelligenten Strategien«

1. Komplexität
- 1.1. Allgemeine Komplexitätsbetrachtungen
- 1.2. Die Komplexität spezieller Algorithmen

2. Exakte Methoden
- 2.1. Das Branch & Bound-Verfahren
- 2.2. Das A*-Verfahren als Bestensuchverfahren
- 2.3. Vergleich von A*- und Branch & Bound-Verfahren
- 2.4. Beispiel zu Suchstrategien

3. Klassische Heuristiken
- 3.1. Klassifizierung
- 3.2. Eröffnungsverfahren
- 3.3. Verbesserungsverfahren
- 3.4. Kritische Beurteilung klassischer Heuristiken
- 3.5. Praxisbeispiel Bandbreitennutzung

4. Simulated Annealing
- 4.1. Physikalischer Hintergrund und Grundidee
- 4.2. Algorithmische Realisierung
- 4.3. Beispiele zum Simulated Annealing
- 4.4. Übungen zur Anwendung von Simulated Annealing

5. Tabu Search
- 5.1. Die Grundidee des Tabu-Search
- 5.2. Vermeidung von Zyklen
- 5.3. Berechnung des TSP-Beispiels
- 5.4. Ein Praxisbeispiel: Digitaler Hörfunk
- 5.5. Übungen zur Anwendung von Tabu Search

6. Genetische Algorithmen
- 6.1. Einführung in die Genetischen Algorithmen
- 6.2. Basiswissen zum Genetischen Algorithmus
- 6.3. Beispiele zum Genetischen Algorithmus
- 6.4. Genetische Algorithmen in der Praxis
- 6.5. Übungen zu Genetischen Algorithmen

Lösungen zu den Übungsaufgaben

6.2 Basiswissen zum Genetischen Algorithmus

6.2.1 Der Genetische Algorithmus im Überblick

Der Algorithmus 6.1 ist als Pseudocode notiert. So können Sie die nachfolgend beschriebenen einzelnen Schritte der Vorgehensweise beim genetischen Algorithmus auch formal in ihrem Ablauf nachvollziehen.

Algorithmus 6.1: Genetischer Algorithmus

Schritt 1:

Erzeuge eine Ausgangspopulation G^0 von Individuen $I_1^0, \ldots, I_n^0$;
Bestimme Fitnesswerte $f(I_j^0)$ aller Individuen $(j = 1, \ldots, n)$;
$k := 0$; % Iterationszähler

Schritt 2:

($\mathrm{IT_{GA}}$) WENN Abbruchkriterium erfüllt, STOPP;
SONST
$k := k + 1$;
Erzeuge Folgegeneration G^k
durch Reproduktion und
Rekombination
unter Anwendung der genetischen Operatoren
Crossover und Mutation;
Bestimme Fitnesswerte $f(I_j^k)$ aller Individuen;
Selektion;
GEHE ZU ($\mathrm{IT_{GA}}$);

Output: Ausgabe der global besten Lösung

[. . .]

Die zentralen Begriffe »Individuen«, »Generationen«, »Fitness«, »Selektion«, »Crossover« und »Mutation« werden nun in ihrer Bedeutung für das Verfahren beschrieben.

[. . .]

6.2.3 Fitness und Selektion

Nach Abschluss jeder Reproduktions- und Rekombinationsphase wird die Population wieder auf ihre Ursprungsgröße reduziert. Dazu ermittelt man die Güte oder *Fitness* jedes Individuums. In Abbildung 6.2 wird der Fitness-Wert durch die »Größe des Individuums« dargestellt.

Fitness

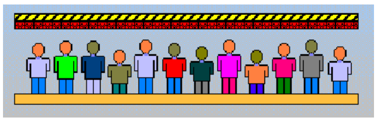

Abbildung 6.2: Der Fitnesswert als Größe eines Individuums

Im einfachsten Fall werden die n »Besten« einer Generation in die nächste übertragen. Man sortiert also die dargestellten Individuen und wählt, wie in der Abbildung 6.3 beispielhaft gezeigt, die sieben Besten für die nächste Generation.

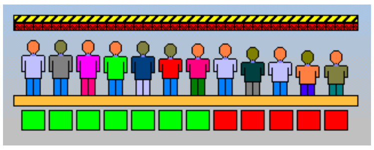

Abbildung 6.3: Sortierung und Besten-Auswahl

Alternativ kann ein Individuum auch mit einer zu seiner Fitness proportionalen Wahrscheinlichkeit ausgewählt werden. Häufig wird zum besseren Verständnis solcher zufallsgesteuerten Auswahl das Bild eines Roulettes gewählt.

Indiv.	Fitness
A	4
B	6
C	2
D	1
E	3
F	1
G	8
H	4
I	**5**
J	2

Zufallsgradzahl: 310

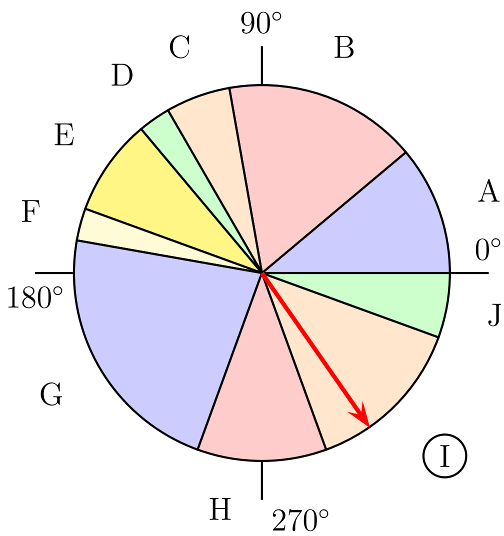

Abbildung 6.4: Proportionalauswahl eines Individuums

Hat man z. B. – wie in Abbildung 6.4 – die Fitnesswerte der Individuen A bis J ermittelt, so können diese den Feldern eines Rouletterades zugeordnet werden, wobei die Größe eines solchen Feldes proportional zum Fitnesswert gewählt ist. Nach Drehen des Rades stellt sich nach Stillstand eine Zufallszahl zwischen 0 und 360 ein; ein Individuum ist bestimmt. Das so ausgewählte Individuum muss nicht notwendigerweise die größte Fitness haben, aber seine *Auswahlwahrscheinlichkeit* p_i ist proportional zum Fitnesswert bezogen auf die Summe der Fitnesswerte aller Individuen.

Auswahl-wahrscheinlichkeit

[. . .]

6.5 Übungen zum Genetischen Algorithmus

Übungsaufgabe 6.3

Gesucht sei eine Rundreise minimaler Länge durch die Städte **A**achen, **B**erlin, **C**elle, **D**resden, **E**mden, **F**rankfurt und **G**reifswald (vgl. Abbildung 6.5). Jede Stadt muss in der Rundreise enthalten sein.

Mit der (zufälligen) Wahl eines Crossoverpunktes $1 \leq r < 7$ entstehen für jedes Elternteil zwei Teilstrings, die überkreuz unverändert wieder zusammengeführt werden (*Ein-Punkt-Crossover*).

Ein-Punkt-Crossover

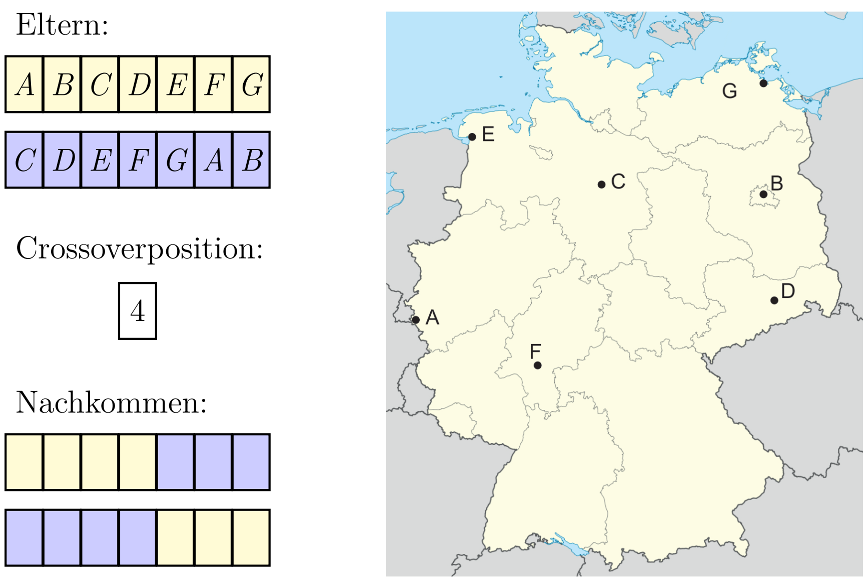

Abbildung 6.5: Ein-Punkt-Crossover für das TSP

Prüfen Sie nun empirisch, ob durch Anwendung des Operators die gewünschten Eigenschaften einer Rundreise erhalten bleiben. Führt die Anwendung dieses Operators in jedem Fall wieder zu einer zulässigen Tour?

(10/21)

31801