



Jenaer Schriften zur Wirtschaftswissenschaft

Heuristische Verfahren

Wolfgang Domschke und Armin Scholl

08/2006

**Arbeits- und Diskussionspapiere
der Wirtschaftswissenschaftlichen Fakultät
der Friedrich-Schiller-Universität Jena**

ISSN 1611-1311

Herausgeber:

Wirtschaftswissenschaftliche Fakultät
Friedrich-Schiller-Universität Jena
Carl-Zeiß-Str. 3, 07743 Jena
www.wiwi.uni-jena.de

Schriftleitung:

Prof. Dr. Hans-Walter Lorenz
h.w.lorenz@wiwi.uni-jena.de
Prof. Dr. Armin Scholl
a.scholl@wiwi.uni-jena.de

Heuristische Verfahren

Prof. Dr. Wolfgang Domschke und **Prof. Dr. Armin Scholl**

Technische Universität Darmstadt
Fachbereich Rechts- und
Wirtschaftswissenschaften
Fachgebiet Operations Research
Hochschulstraße 1, 64289 Darmstadt
e-Mail: domschke@bwl.tu-darmstadt.de

Friedrich-Schiller-Universität Jena
Fakultät für Wirtschaftswissenschaften
Lehrstuhl für Betriebswirtschaftliche
Entscheidungsanalyse
Carl-Zeiß-Straße 3, 07743 Jena
e-Mail: a.scholl@wiwi.uni-jena.de

Zusammenfassung:

Heuristische Verfahren dienen zur näherungsweise Lösung von komplexen Entscheidungs- und Optimierungsproblemen bzw. zugehörigen Optimierungsmodellen. Eröffnungsverfahren konstruieren eine (erste) zulässige Lösung, während Verbesserungsverfahren durch sukzessive Lösungstransformation zu verbesserten (lokal optimalen) Lösungen führen. Metastrategien steuern Verbesserungsverfahren im Hinblick auf die Untersuchung vielversprechender Lösungsbereiche und die Überwindung lokaler Optimalität.

Schlüsselworte:

Heuristisches Verfahren, Heuristik, Optimierungsmodell, Metastrategie

1. Begriffe und Musterbeispiel

Unter **heuristischen Verfahren (Heuristiken)** versteht man Vorgehensweisen zur Ermittlung guter zulässiger Lösungen von Optimierungsmodellen, mit Hilfe derer man reale Entscheidungsprobleme mathematisch abbilden kann. Sie gehen nach bestimmten Regeln zur Lösungsfindung oder -verbesserung vor, die die vorliegende Modellstruktur auf Erfolg versprechende Weise ausnutzen und einen angemessenen Rechenaufwand erfordern.

Während **Optimierungsverfahren** grundsätzlich eine optimale Lösung eines Optimierungsmodells finden und deren Optimalität auch nachweisen müssen, begnügt sich eine Heuristik mit dem Auffinden einer als hinreichend gut eingeschätzten Lösung.

Heuristiken kommen v.a. dort zum Einsatz, wo Optimierungsverfahren wegen des zu hohen Rechenaufwands bei praxisrelevanten Problemgrößen scheitern. Dies gilt grundsätzlich für sogenannte NP-schwere Probleme, wie vielen Reihenfolge-, Zuordnungs- und Gruppierungsproblemen der kombinatorischen Optimierung. Ein bekanntes Beispiel ist das *Traveling Salesman-Problem (TSP)*, bei dem ein Handlungsreisender seine Kunden in streckenminimaler Reihenfolge besuchen möchte.

Anwendung finden Heuristiken auch bei der Bestimmung zulässiger Lösungen, die

- als Startlösungen für Optimierungsverfahren, wie der MODI-Methode für das *klassische Transportproblem (TPP)*, dienen (vgl. z.B. Domschke und Drexl (2005, Kap. 4)) oder
- im Rahmen von (exakten) Branch-and-Bound (B&B)-Verfahren eine globale untere bzw. obere Schranke für den maximalen bzw. minimalen Zielfunktionswert liefern. Je besser derartige Schranken sind, desto stärker dienen sie dem Ausloten von Teilproblemen, um ein vollständiges Enumerieren aller Lösungen zu vermeiden (vgl. z.B. Domschke und Drexl (2005, Kap. 6)).

Heuristiken lassen sich unterteilen in (vgl. zu einer allgemeinen Übersicht Silver (2004)):

- (1) *Eröffnungsverfahren* zur Bestimmung einer (ersten) zulässigen Lösung
- (2) *Lokale Such- bzw. Verbesserungsverfahren* zur Verbesserung einer gegebenen zulässigen Lösung
- (3) *Relaxationsbasierte Verfahren*
- (4) *Unvollständig ausgeführte Optimierungsverfahren*, z.B. vorzeitig beendete B&B-Verfahren.

Auch wenn Heuristiken das Auffinden bzw. Erkennen einer optimalen Lösung nicht garantieren, kann man häufig zumindest eine *Worst-Case-Schranke*, d.h. eine maximale Abweichung einer mit der Heuristik gefundenen von der Optimallösung, angeben. Als *Approximationsverfahren* bezeichnet man Heuristiken mit definierter Worst-Case-Schranke, die für NP-schwere Probleme mit polynomialem Aufwand eine Lösung finden. Ein Beispiel hierfür ist die Heuristik von Christofides für TSP in ungerichteten Graphen, die mit polynomialem Aufwand (der Größenordnung $O(n^3)$ bei n zu besuchenden Orten) eine Lösung findet, deren Länge die einer kürzesten Rundreise um maximal 50 % übersteigt; siehe Domschke (1997, Kap. 3).

Für praktische Anwendungen ist die durchschnittliche bzw. erwartete Abweichung erzielbarer Lösungen vom Optimum interessanter. Hierfür sind experimentelle Rechentests erforderlich; vgl. Berens (1992) oder Rardin und Uzsoy (2001).

Musterbeispiel: Die unten geschilderten Heuristiken verdeutlichen wir anhand eines einfachen binären Optimierungsproblems, des *Knapsackproblems (KP)*:

Ein Wanderer möchte in einem Rucksack eine Auswahl aus p Gütern mitnehmen. Dem Gut $j = 1, \dots, p$ mit Gewicht w_j ordnet er den Nutzen n_j zu. Welche Güter soll er mitnehmen, wenn er bei maximalem Gesamtgewicht b der ausgewählten Güter deren Nutzen maximieren möchte?

Verwenden wir für Gut j die Binärvariable x_j ($= 1$, falls j mitzunehmen ist, 0 sonst), so lässt sich das *allgemeine Modell* sowie eine *Modellinstanz* mit $p = 6$ wie folgt modellieren:

$$\text{Maximiere } N(\mathbf{x}) = \sum_{j=1}^p n_j \cdot x_j = 4x_1 + 6x_2 + 2x_3 + 5x_4 + 6x_5 + 7x_6$$

unter den Nebenbedingungen

$$\sum_{j=1}^p w_j \cdot x_j \leq b \quad \text{bzw.} \quad 5x_1 + 3x_2 + 3x_3 + 3x_4 + 4x_5 + 8x_6 \leq 11$$

$$x_j \in \{0, 1\} \quad \text{für } j = 1, \dots, p (=6)$$

Eine zulässige Lösung ist $\mathbf{x} = (0,1,1,1,0,0)$ mit Gesamtnutzen $N(\mathbf{x}) = 13$.

Im ökonomischen Bereich entstehen KPe z.B. bei der Investitionsplanung bei knappem Budget oder bei Entscheidungen über Annahme oder Ablehnung von Projekten.

2. Grundprinzipien von Heuristiken

2.1 Eröffnungsverfahren

Eröffnungsverfahren dienen der Bestimmung einer (ersten) zulässigen Lösung eines Problems. Häufig wird diese durch sukzessive Festlegung von Lösungselementen konstruiert (daher auch: *Konstruktionsverfahren*). Die Lösungsgüte ist von der „Ausgestaltung“ des Verfahrens und – damit verbunden – vom investierten (Rechen-) Aufwand abhängig (vgl. Klein und Scholl (2004), Kap. 9.4.4):

- **Uninformierte Verfahren** verzichten durch starren Ablauf (weitgehend) auf das Ausnutzen konkreter Problemdaten. Ein Beispiel ist die Nordwesteckenregel für das TPP, die bei geringem Rechenaufwand eher schlechte Lösungen erzielt.
- **Myopische** oder **Greedy-Heuristiken** trachten in jedem Verfahrensschritt nach dem bestmöglichen Zielfunktionswert der bisherigen Teillösung. Beispiele sind die Spaltenminimum-Methode für das TPP oder die Methode des besten Nachfolgers für das TSP. Bei derartigen Verfahren handelt es sich meist um *statische* **Prioritätsregelverfahren**, welche die Lösung unter Beachtung einer zu Verfahrensbeginn erstellten Prioritätsliste der Lösungselemente berechnen.
- **Vorausschauende Verfahren** versuchen Kurzsichtigkeit dadurch zu vermeiden, dass sie in jedem Schritt abschätzen, welche Auswirkungen die aktuelle Zuordnung eines Lösungselements auf die in nachfolgenden Schritten noch erzielbare Lösungsgüte besitzt. Es handelt sich meist um *dynamische* Prioritätsregelverfahren, die in jedem Verfahrensschritt die Prioritäten für das jeweils nächste zu wählende Lösungselement neu ermitteln. Der dadurch erhöhte Aufwand ist häufig durch deutlich bessere Lösungen gerechtfertigt. Beispiele sind die Vogelsche Approximationsmethode für das TPP und die Methode der sukzessiven Einbeziehung für das TSP.

Beispiele:

- Ein uninformiertes Verfahren besteht für unsere KP-Instanz z.B. darin, die Güter in Nummerierungsreihenfolge einzuplanen. Dies ergibt die Lösung $\mathbf{x}^1 = (1,1,1,0,0,0)$ mit Nutzen $N(\mathbf{x}^1) = 12$.
- Sinnvoller ist eine Greedy-Heuristik, die eine Prioritätsliste bildet, in der die Güter nach monoton abnehmenden Nutzen (bei Gleichheit: nach wachsender Nummer) sortiert sind. Plant man die Güter in der dadurch gegebenen Reihenfolge $\langle 6,2,5,4,1,3 \rangle$ ein, so ergibt sich die Lösung $\mathbf{x}^2 = (0,1,0,0,0,1)$ mit Nutzen $N(\mathbf{x}^2) = 13$. Investiert man zusätzlichen Aufwand, indem man die Güter nach monoton abnehmenden „relativen“ Nutzen (Quotient aus Nutzen und Gewicht) sortiert, so erhält man die Prioritätsliste

$\langle 2,4,5,6,1,3 \rangle$ und die Lösung $\mathbf{x}^3 = (0,1,0,1,1,0)$ mit Nutzen $N(\mathbf{x}^3) = 17$. Diese Lösung ist optimal, was mit Hilfe der Heuristik jedoch nicht erkennbar wird.

- Beim KP ist es nicht üblich, vorausschauende Verfahren einzusetzen, da diese bei der Berechnung der Prioritäten zusätzlich beachten müssten, ob sich das jeweilige Gut günstig mit weiteren Gütern kombinieren lässt oder nicht. Dies ist jedoch sehr schwierig und aufwändig.

2.2 Lokale Such- bzw. Verbesserungsverfahren

Sie starten zumeist mit einer zulässigen Lösung des Problems, die man entweder zufällig oder durch Anwendung eines Eröffnungsverfahrens bestimmt. In jeder Iteration wird von der gerade betrachteten Lösung \mathbf{x} zu einer Lösung \mathbf{x}' aus der *Nachbarschaft* $NB(\mathbf{x})$ fortgeschritten. $NB(\mathbf{x})$ enthält alle Lösungen, die sich aus \mathbf{x} durch (einmalige) Anwendung einer zu spezifizierenden *Transformationsvorschrift* (bzw. eines bestimmten **Zuges**) ergeben, z.B.:

- 1) Veränderung einer Lösung an genau einer Stelle, z.B. durch Umschalten einer Position eines binären Lösungsvektors von 0 auf 1 oder umgekehrt
- 2) Vertauschen von Elementen
- 3) Verschieben von Elementen

Beispiele für Züge:

- Bei KPen kann durch einen Zug ein in \mathbf{x} gewähltes Gut aus dem Rucksack entfernt werden oder umgekehrt. Also enthält $NB(\mathbf{x})$ alle Lösungen, die sich von \mathbf{x} in genau einer Komponente unterscheiden (obige Regel 1). Regel 2 erhält man, wenn vorgesehen wird, ein in \mathbf{x} „mitgenommenes“ Gut durch ein anderes zu ersetzen.
- Bei einem r-optimalen Verfahren für TSP werden, ausgehend von einer Rundreise \mathbf{x} , genau r der in \mathbf{x} enthaltenen Verbindungen (im ungerichteten Fall Kanten) gegen r bislang nicht enthaltene ersetzt. $NB(\mathbf{x})$ umfasst also alle Rundreisen, die aus \mathbf{x} durch den Austausch von r Kanten gegen r bislang nicht in der Rundreise enthaltene Kanten entstehen. Zur Anwendung empfohlen werden v.a. „2-opt“ oder „3-opt“ mit 2 bzw. 3 auszutauschenden Kanten oder eine Vorgehensweise von Lin und Kernighan, bei der r nicht von vorneherein fixiert ist.

Denkbar ist auch eine Verschiebung eines Knotens oder mehrerer Knoten an eine andere Stelle der Rundreise. Vgl. z.B. Domschke und Drexl (2005, Kap. 6).

Neben der Nachbarschaftsdefinition lassen sich Verfahren v.a. hinsichtlich der Strategie zur Untersuchung von $NB(\mathbf{x})$ und der Zugauswahl unterscheiden:

- In welcher Reihenfolge werden die Nachbarlösungen $\mathbf{x}' \in NB(\mathbf{x})$ untersucht? Die Reihenfolge kann zufällig oder systematisch gewählt werden.
- Zu welcher der untersuchten Nachbarlösungen wird übergegangen, um von ihr aus in der nächsten Iteration die Suche fortzusetzen? Zwei Extreme sind die *First fit*- und die *Best fit-Strategie*. Bei First fit wird jeweils zur ersten gefundenen besseren Nachbarlösung übergegangen; bei Best fit wird die Nachbarschaft vollständig untersucht und die beste Möglichkeit realisiert.

Außerdem lassen sich Eröffnungs- und lokale Such- bzw. Verbesserungsverfahren wie folgt unterteilen:

- *Deterministische Verfahren* ermitteln bei mehrfacher Anwendung auf ein Problem unter identischen Startbedingungen stets dieselbe Lösung. Die oben geschilderten Eröffnungsverfahren sind deterministisch.
- *Stochastische (oder randomisierte) Verfahren* enthalten demgegenüber eine zufällige Komponente, die bei wiederholter Anwendung des Algorithmus auf ein Problem zu unterschiedlichen Lösungen führen kann. Z.B. führt eine zufällige Untersuchungsreihenfolge der Nachbarschaft bei First fit - Strategie zu einem stochastischen Verfahren.

Reine Verbesserungsverfahren enden, sobald in einer Iteration keine bessere Nachbarlösung existiert bzw. gefunden wird. Die beste erhaltene Lösung stellt für die gewählte Nachbarschaftsdefinition ein lokales Optimum dar, dessen Zielfunktionswert deutlich schlechter als der eines globalen Optimums sein kann.

Um ein solches lokales Optimum wieder verlassen zu können, müssen Züge erlaubt werden, die zwischenzeitlich zu Verschlechterungen des Zielfunktionswertes führen. Heuristiken, die diese Möglichkeit vorsehen, nennen wir **lokale Suchverfahren**. Hierzu zählen Simulated Annealing, Tabu Search und Genetische Algorithmen (vgl. Abschnitt 3.1 bis 3.3).

Der Unterschied zwischen Verbesserungs- und lokalen Suchverfahren lässt sich mittels Abb. 1 verdeutlichen. Sie enthält den Verlauf einer zu minimierenden Funktion $F(y)$ in Abhängigkeit von einer Variablen y . Startet ein Verbesserungsverfahren z.B. mit der Lösung y^1 , so gelangt es durch ein oder mehrere Züge zum lokalen Minimum in y^2 . Es gelingt ihm jedoch nicht, das globale Minimum y^* zu finden.

Ein lokales Suchverfahren gelangt u.U. aus y^2 wieder heraus, indem es einen oder mehrere Züge zur schlechteren Lösung y^3 (lokales Maximum) durchführt. Ausgehend von y^3 kann es gelingen, y^* zu erreichen.

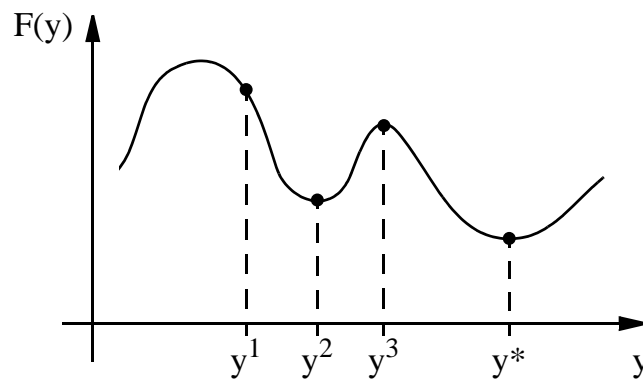


Abb. 1: Lokale und globale Optima

2.3 Relaxationsbasierte Verfahren

Im Rahmen der Lösung NP-schwerer Optimierungsprobleme mittels B&B spielen neben globalen unteren bzw. oberen Schranken für den maximalen bzw. minimalen Zielfunktionswert, die man anhand von zulässigen Lösungen erhält (Abschnitt 1), lokale obere bzw. untere Schranken eine wichtige Rolle. Man gewinnt eine lokale obere bzw. untere Schranke für den maximalen bzw. minimalen Zielfunktionswert durch Lösen eines *relaxierten* (d.h. weniger restringierten) Problems.

Wichtige Relaxationsmöglichkeiten bestehen in der Eliminierung von Nebenbedingungen (z.B. Zyklenbedingungen beim TSP), der zusätzlichen Berücksichtigung ihrer Verletzung in der Zielfunktion (*Lagrange-Relaxation*) sowie der Vereinfachung des Typs der Variablen (*LP-Relaxation*: Vernachlässigung der Ganzzahligkeit von Variablen in einem linearen Modell).

Aus der Lösung einer Relaxation lässt sich häufig eine gute zulässige Lösung des Ausgangsproblems ableiten.

Beispiel: Eine optimale Lösung der LP-Relaxation des Knapsackproblems besitzt maximal eine Variable mit einem Wert ungleich 0 oder 1 (vgl. Domschke et al. (2005, Aufg. 6.7)). Setzt man sie gleich 0, so ergibt sich eine i.d.R. gute Lösung des ursprünglichen Problems. Für unser obiges Musterbeispiel erhalten wir dadurch die Lösung \mathbf{x}^3 ; in der optimalen Lösung der LP-Relaxation besitzt x_6 den Wert $1/8$.

2.4 Unvollständig ausgeführte Optimierungsverfahren

Bei der Ausführung von B&B-Verfahren besteht die Möglichkeit, vor dem Auffinden einer optimalen Lösung (oder dem Erkennen ihrer Optimalitätseigenschaft) das Verfahren zu beenden. Die bis dahin ermittelten unteren und oberen Schranken bieten dann den Vorteil, dass Aussagen über die Lösungsgüte der Lösung möglich sind.

Eng begrenzte Rechenzeiten sind v.a. im Rahmen der Online-Optimierung üblich, bei der aufgrund von Datenänderungen (z.B. Eintreffen neuer Aufträge bei Transport- oder Tourenplanung) ständig neue Optimierungsprobleme in Echtzeit zu lösen sind.

3. Heuristische Metastrategien

Um das Problem der lokalen Optimalität zu überwinden, hat man verschiedene **Metastrategien** bzw. **Metaheuristiken** entwickelt, die Verbesserungsheuristiken durch Anwendung allgemeiner Prinzipien so steuern, dass sie in lokalen Optima nicht "stecken bleiben" und/oder viel versprechende Lösungsbereiche untersuchen. Dabei werden verschiedene Vorgehensweisen verwendet, von denen wir die wichtigsten beschreiben (vgl. Domschke 1997, Kap. 1.3.2; Klein und Scholl 2004; Kap. 9.4.6).

3.1 Simulated Annealing

Der Name **Simulated Annealing (SA)** kennzeichnet die Analogie des Verfahrensprinzips mit einem physikalischen Abkühlungsvorgang in der Thermodynamik. Annealing bezeichnet dort den gesteuerten Erstarrungsvorgang in einem Molekülgitter, der einen Zustand minimaler freier Gitterenergie im Festkörper zum Ziel hat.

Ausgehend von einer zulässigen Lösung \mathbf{x} , bestimmt man eine zulässige Lösung $\mathbf{x}' \in \text{NB}(\mathbf{x})$. Ist \mathbf{x}' besser als \mathbf{x} , so wird der Zug zu \mathbf{x}' durchgeführt (first fit). Andernfalls lehnt man diesen Zug nicht generell ab, sondern erlaubt ihn zufällig mit Wahrscheinlichkeit $P(\Delta, \alpha) := e^{-\Delta/\alpha}$.

$P(\Delta, \alpha)$ ist abhängig von der Höhe Δ der Verschlechterung und von einem vorzugebenden Temperaturparameter α . Dieser Parameter wird zu Beginn des Verfahrens so gewählt, dass durchaus Verschlechterungen des Zielfunktionswertes z.B. um 15 oder 20% möglich sind. Im Laufe des Verfahrens wird er durch Multiplikation mit einem Parameter $\beta \in (0, 1)$ sukzessive reduziert, die Temperatur des Systems also gesenkt, so dass am Ende des Verfahrens nur noch Verbesserungen erlaubt werden.

Aufgrund vielfältiger Möglichkeiten der Vorgabe von Parametern zur Steuerung des Abkühlungsprozesses ist es u.U. schwierig und aufwändig, günstige Kombinationen zu finden, die bei geringer Rechenzeit zu guten Lösungen führen. Vgl. hierzu Aarts und Korst (1989) oder Domschke (1997).

Eine vereinfachte Variante von SA ist **Threshold Accepting**. Hierbei wird jede Lösung akzeptiert, die den Zielfunktionswert höchstens um einen vorzugebenden Wert Δ verschlechtert. Im Laufe des Verfahrens wird Δ sukzessive auf 0 reduziert. Beim ähnlichen *Sintflutalgorithmus* wird das erlaubte Zielfunktionsniveau (eines Maximierungsproblems) wie ein Wasserspiegel sukzessive erhöht.

3.2 Tabu Search

In der einfachsten Version von **Tabu Search (TS)** wird in jeder Iteration die Nachbarschaft $NB(x)$ vollständig untersucht (best fit). Unter allen (nicht verbotenen) Nachbarn wird derjenige mit dem besten Zielfunktionswert – auch wenn er eine Verschlechterung darstellt – ausgewählt und als Ausgangspunkt für die nächste Iteration verwendet (Prinzip des *steepest ascent / mildest descent* bei Maximierungs-, *steepest descent / mildest ascent* bei Minimierungsproblemen). Im Gegensatz zum stochastischen Simulated Annealing wird also eine schlechtere Lösung nicht zufällig, sondern nur dann akzeptiert, wenn es keine Verbesserungsmöglichkeit gibt.

Um jedoch nach Verschlechterungen nicht wieder zu zuvor besuchten, besseren Lösungen zurückzukehren, was zu einem Kreisen des Verfahrens führen würde, müssen derartige Lösungen verboten – *tabu gesetzt* – werden. Wenn das Speichern und Überprüfen von vollständigen Lösungen zu aufwändig ist, kann man eine *Tabuliste* führen, die Informationen über zuvor ausgeführte Züge speichert.

Ein wichtiger Parameter ist die Anzahl der Iterationen (*Tabudauer*), über die man zuletzt ausgeführte Züge tabu setzt. Grundsätzlich gilt: Je kürzer die Tabudauer gewählt wird, umso eher gerät das Verfahren ins Kreisen. Je länger sie gewählt wird, umso größer ist die Gefahr, dass es man Lösungen unnötig verbietet. Die geeignete Tabudauer ist abhängig von der Art und Größe des betrachteten Problems. Beim *reactive TS* wird die Dauer im Laufe des Verfahrens in bestimmten Intervallgrenzen dynamisch verändert.

Zu vielfältigen weiteren Variationen von TS vgl. Glover und Laguna (1997).

Beispiel: Wir veranschaulichen TS anhand des KPs aus Abschnitt 1 und erlauben nur Züge, die eine einzige Position des Vektors \mathbf{x} umdrehen. Ersetzen wir in \mathbf{x} an Position q eine 0 durch eine 1 oder umgekehrt, so symbolisieren wir dies durch den Eintrag q in der Tabuliste. Wählen wir die Tabudauer $TD=2$, so darf an einer tabu gesetzten Position q jeweils in den beiden folgenden Iterationen keine Veränderung vorgenommen werden.

Iter.	Lösung	Tabu- liste	Zf.- wert	Rest- kapaz.
0	(0,1,0,0,0,1)	<]	13	0
1	(0,0,0,0,0,1)	<2]	7	3
2	(0,0,0,1,0,1)	<2, 4]	12	0
3	(0,0,0,1,0,0)	<4, 6]	5	8
4	(0,1,0,1,0,0)	<6, 2]	11	5
5	(0,1,0,1,1,0)	<2, 5]	17	1
6	(0,1,0,0,1,0)	<5, 4]	12	4

Tab. 1

Starten wir mit $\mathbf{x} = (0,1,0,0,0,1)$, so ergibt sich der in Tab. 1 wiedergegebene Lösungsgang. In der 5. Iteration finden wir die optimale Lösung des Problems (ohne dass TS dies bemerkt).

3.3 Genetische Algorithmen

Genetische Algorithmen (GA) sind stochastische Heuristiken, die den Suchraum simultan an mehreren Stellen untersuchen. Sie greifen Ideen aus der Genetik auf und versuchen, die Prinzipien der biologischen Evolution zur Optimierung mathematisch-technischer Systeme heranzuziehen.

Ein GA arbeitet auf Populationen von Individuen, bezogen auf ein Optimierungsproblem also auf Mengen zulässiger (oder vorübergehend auch unzulässiger) Lösungen. Jedes Individuum (Lösung) wird durch seinen genetischen Code (*String*, *Vektor*) fester Länge repräsentiert, in dem die Werte von Entscheidungsvariablen kodiert sind.

Bei einem KP entspricht dieser String dem binären Lösungsvektor. Bei anderen Optimierungsproblemen wie dem TSP bietet sich eine entsprechend einfache *Kodierung* nicht unmittelbar an. Stattdessen kann z.B. die Folge der in einer Rundreise besuchten Orte als Kodierung dienen (vgl. Domschke 1997, S. 39 f.).

Die *prinzipielle Vorgehensweise* eines GA besteht darin, so lange Generationen von Populationen zu erzeugen und zu analysieren, die durch die *genetischen Operatoren* Selektion, Rekombination und/oder Mutation aus der jeweils vorhergehenden entstehen, bis ein Abbruchkriterium erfüllt ist.

- *Selektion*: Aus den Lösungen einer Population wird ein sogenannter *Genpool* erzeugt, in den einzelne Lösungen mit einer Wahrscheinlichkeit eingehen, die proportional zur Lösungsgüte (Fitness) ist. Von sehr guten Lösungen enthält der Pool u.U. mehrere, von schlechten keine Kopien.
- *Rekombination*: Es werden jeweils zwei Lösungen dem Genpool entnommen und mit einer vorzugebenden Wahrscheinlichkeit gekreuzt (rekombiniert) oder aber unverändert in die neue Generation übernommen.
Beim sogenannten 1-Punkt Crossover wird mittels einer Gleichverteilung zufällig eine Kreuzungsposition in den beiden Strings ermittelt; die Kreuzung selbst geschieht durch kreuzweises Vertauschen der Teilstrings.
Beispiel: Kreuzt man bei unserem KP die Lösungen (0,1,0;0,1,0) mit Nutzen $N=12$ und (0,1,1;1,0,0) mit $N = 13$ nach Position 3, so entstehen die „Nachkommen“ (0,1,0;1,0,0) mit $N = 11$ und (0,1,1;0,1,0) mit $N = 14$. Beide Nachkommen sind zulässig, was beim Crossover nicht immer der Fall sein muss.
- *Mutation*: Durch Rekombination entstandene Lösungen werden mit einer kleinen Wahrscheinlichkeit zufällig verändert und der neuen Population hinzugefügt. Eine alternative „Mutationsmöglichkeit“ besteht darin, die Nachkommen durch ein Verbesserungsverfahren nachzuoptimieren.

Zu vielfältigen Variationen von GA und weiteren *evolutionären Verfahren* vgl. u.a. Kolen und Pesch (1994), Nissen (1997) oder Michalewicz und Fogel (2002).

3.4 Ameisenalgorithmen

Sie beziehen ihren Namen und ihre Lösungsidee von Ameisen und ihrem Verhalten bei der Futtersuche. Die Tiere markieren Wege zwischen Nest und einer Futterquelle mit ihren Pheromonen. Nachfolgende Ameisen wählen i.d.R. denjenigen Weg mit der größten Pheromonkonzentration. Je kürzer ein Weg zwischen Nest und Futterquelle ist, umso häufiger kann er in einem bestimmten Zeitraum zurückgelegt werden; je mehr Tiere ihn benutzen, umso stärker wird die Pheromonspur und umso eher wird er jeweils gewählt werden.

Dieses Grundprinzip wurde in den letzten Jahren zur Lösung verschiedener Probleme der kombinatorischen Optimierung eingesetzt und verfeinert, so dass es durchaus mit anderen Metastrategien konkurrieren kann. Vgl. Dorigo und Stützle (2004) oder Boysen (2005).

3.5 Scatter Search

In den letzten Jahren wird v.a. versucht, die Vorteile der verschiedenen Metastrategien miteinander zu verbinden. Eine derartige kombinierte Vorgehensweise wird als **Scatter Search** bezeichnet. Bei dieser Metastrategie wird ebenfalls auf einer Menge von Lösungen

(Referenzmenge) gearbeitet, die jedoch auf systematischere Weise und zielgerichteter kombiniert werden als bei GA. Außerdem werden entstehende Lösungen mit Hilfe von weiteren Heuristiken verbessert, bevor die besten von ihnen (Elite-Lösungen) in die Referenzmenge eingehen. Dabei wird neben der Lösungsgüte v.a. auch auf die Diversität der Referenzmenge im Hinblick auf die Lösungseigenschaften geachtet, um den Lösungsraum großräumig abzusuchen. Zu einer Darstellung von Scatter Search vgl. Glover et al. (2003).

4. Literaturhinweise

- Aarts, E. und J. Korst (1989): *Simulated annealing and Boltzmann machines*. Wiley, Chichester.
- Berens, W. (1992): *Beurteilung von Heuristiken: Neuorientierung und Vertiefung am Beispiel logistischer Probleme*. Gabler, Wiesbaden.
- Boysen, N. (2005): Ameisenalgorithmen. *WiSt* 21, S. 607 - 612.
- Domschke, W. (1997): *Logistik: Rundreisen und Touren*. 4. Aufl., Oldenbourg, München.
- Domschke, W. und A. Drexl (2005): *Einführung in Operations Research*. 6. Aufl., Springer, Berlin.
- Domschke, W.; A. Drexl, R. Klein, A. Scholl und S. Voß (2005): *Übungen und Fallbeispiele zum Operations Research*. 5. Aufl., Springer, Berlin.
- Dorigo, M. und T. Stützle (2004): *Ant colony optimization*. MIT-Press, London.
- Glover, F. und M. Laguna (1997): *Tabu search*. Kluwer, Boston.
- Glover, F.; M. Laguna und R. Marti (2003): Scatter search. In: Ghosh, A. und S. Tsutsui (Hrsg.): *Advances in evolutionary computing: Theory and applications*, Springer, New York, S. 519-537.
- Klein, R. und A. Scholl (2004): *Planung und Entscheidung*. Vahlen, München.
- Kolen, A. und E. Pesch (1994): Genetic local search in combinatorial optimization. *Discrete Applied Mathematics* 48, S. 273 - 284.
- Michalewicz, Z. und D.B. Fogel (2002): *How to solve it: Modern heuristics*. 3. Aufl., Springer, Berlin.
- Nissen, V. (1997): *Einführung in Evolutionäre Algorithmen: Optimierung nach dem Vorbild der Evolution*. Vieweg, Wiesbaden.
- Rardin, R.L. und R. Uzsoy (2001): Experimental evaluation of heuristic optimization algorithms: A tutorial. *Journal of Heuristics* 7, S. 261 - 304.
- Silver, E.A. (2004): An overview of heuristic solution methods. *Journal of the OR Society* 55, S. 936 - 956.

Jenaer Schriften zur Wirtschaftswissenschaft

2006

- 1 Roland **Helm** und Michael **Steiner**: Nutzung von Eigenschaftsarten im Rahmen der Präferenzanalyse - Eine Meta-Studie, Diskussion und Empfehlungen.
- 2 Uwe **Cantner** and Jens J. **Krüger**: Micro-Heterogeneity and Aggregate Productivity Development in the German Manufacturing Sector.
- 3 Roland **Helm**: Implication from Cue Utilization Theory and Signalling Theory for Firm Reputation and the Marketing of New Products.
- 4 Simon **Renaud**: Betriebsräte und Strukturwandel.
- 5 Wolfgang **Schultze**: Anreizkompatible Entlohnung mithilfe von Bonusbanken auf Basis des Residualen Ökonomischen Gewinns.
- 6 Susanne **Büchner**, Andreas **Freytag**, Luis G. **González**, Werner **Güth**: Bribery and Public Procurement - An Experimental Study.
- 7 Reinhard **Haupt**, Martin **Kloyer** und Marcus **Lange**: Patent indicators of the evolution of technology life cycles.
- 8 Wolfgang **Domschke** und Armin **Scholl**: Heuristische Verfahren.
- 9 Wolfgang **Schultze** und Ruth-Caroline **Zimmermann**: Unternehmensbewertung und Halbeinkünfteverfahren: Der Werteeinfluss des steuerlichen Eigenkapitals.

2005

- 1 Reinhard **Haupt**: Patent analysis of a company's technology strength.
- 2 Axel **Braßler**, Christoph **Grau**, Herfried **Schneider**: Wissenslabor Betriebswirtschaft - Eine Lehr-, Lern- und Kommunikationsumgebung für die universitäre und betriebliche Aus- und Weiterbildung.
- 3 Wolfgang **Kürsten**: Risikomanagement und aktionärsorientierte Unternehmenssteuerung - Mehr Fragen als Antworten.
- 4 Roland **Helm**, Reinhard **Meckl**, Nicole **Sodeik**: Wissensmanagement - Ein Überblick zum Stand der empirischen Forschung.
- 5 Uwe **Cantner**, Kristina **Drefßler**, Jens J. **Krüger**: Knowledge and Creative Destruction over the Industry Life Cycle - The Case of the German Automobile Industry.
- 6 Reinhard **Meckl**, Robert **Schramm**: Empirical evidence for a theory of international new ventures.
- 7 Andreas **Freytag**, Dirk **Schiereck**, Thomas W. **Thomas**: Consolidation and Market Power of Energy Utilities - The case of US-American and German Utility Takeovers.
- 8 Roland **Helm** und Oliver **Mauroner**: New Firms from Research-based Spin-offs.
- 9 Werner **Jammerneegg** und Peter **Kischka**: A Decision Rule Based on the Conditional Value at Risk.
- 10 Roland **Helm** und Wolfgang **Stölzle**: Out-of-Stocks im Handel: Einflussfaktoren und Kundenreaktionsmuster.
- 11 Uwe **Cantner**, Kristina **Drefßler**, Jens J. **Krüger**: Knowledge Compensation in the German Automobile Industry.
- 12 Volkmar **Botta**, Martin **Köhler**: Zur Wertaufhellungskonzeption nach IAS 10.
- 13 Roland **Helm** und Michael **Gehrer**: Zum Aufbau von Vertrauen in interaktiven Entscheidungsprozessen.
- 14 Andreas **Feytag** and Donato **Masciandaro**: Financial Supervision Fragmentation and Central Bank Independence: The Two Sides of the Same Coin?
- 15 Volkmar **Botta** und Adrian A. **Weinaug**: Behandlung von Investitionszulagen für Sachanlagen gemäß Investitionszulagengesetz bei freiwilliger Offenlegung des Einzelabschlusses nach § 325 Abs. 2a HGB.

- 16 Volkmar **Botta** und Martin **Köhler**: Implikationen der Bestimmung des Cashflows aus betrieblicher Tätigkeit nach der direkten Methode.
- 17 Uwe **Cantner**, Andreas **Nicklisch** und Torsten **Weiland**: Innovation races: An experimental study on strategic research activities.
- 18 Markus **Pasche**: (Self-)Regulation of a Natural Monopoly via Complementary Goods -the Case of F/OSS Business Models.
- 19 Markus **Pasche**: Das Vertrauensspiel – eine verhaltenensorientierte Erklärung.
- 20 Reinhard **Haupt** und Matthias **Korgel**: A Comparison between US and International Patent Classification as Input Data of a Technology Competition-Oriented Cluster Analysis.
- 21 Wolfgang **Kürsten**, Reinhard **Meckl** und Andreas **Krostewitz**: Value-Based M&A-Management – der M&A-Prozess im Lichte des Shareholder Value-Prinzips.
- 22 Simone **Martin**: Risikominimierung bei der Arbeitgeberwahl.
- 23 Wolfgang **Kürsten**: Neoklassische Finanzierungstheorie - Eine didaktisch motivierte Einführung.
- 24 Karsten **Korsa** und Simone **Martin**: Die Glaubwürdigkeit personalpolitischer Maßnahmen zur Signalisierung von Arbeitgeberattraktivität.