

Christoph SCHURY

**Social Choice Methoden zur Lösung von
mehrdimensionalen Rucksackproblemen**

Masterarbeit

zur Erlangung des akademischen Grades

eines Master of Science

der Studienrichtung Betriebswirtschaft

an der Karl-Franzens-Universität Graz

Begutachter: Ao. Univ.-Prof. Dr. Ulrich PFERSCHY

Institut: Statistik und Operations Research

Graz, März 2013

Ehrenwörtliche Erklärung

Ich erkläre ehrenwörtlich, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen nicht benutzt und die den Quellen wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen inländischen oder ausländischen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht. Die vorliegende Fassung entspricht der eingereichten elektronischen Version.

TT.MM.JJJJ

Unterschrift

„Es ist nicht genug, zu wissen, man muss auch anwenden; es ist nicht genug, zu wollen, man muss auch tun“

(Johann Wolfgang von Goethe, Wilhelm Meisters Wanderjahre)

Erinnert man sich an seine Studienzeit und im speziellen an seine Diplomarbeit zurück, treffen die Worte von Johann Wolfgang von Goethe in den meisten Fällen zu. Erstmals muss das über Jahre erlernte Wissen, unter der Prämisse ehest möglich abzuschließen, in Anwendung gebracht werden.

Auch für mich galt Gleiches beim Verfassen der Abschlussarbeit, konnte aber auf die ständige Unterstützung meines Betreuers Ao. Univ.-Prof. Dr. Ulrich Pferschy zählen. Herzlichen Dank dafür!

Inhaltsverzeichnis

Tabellenverzeichnis.....	II
Formelverzeichnis.....	III
Abbildungsverzeichnis.....	III
Abkürzungsverzeichnis.....	IV
Zusammenfassung.....	1
1. Einleitung.....	2
2. Rucksackprobleme.....	3
2.1. Algorithmen zur Lösung von Rucksackproblemen.....	4
2.2. Formen von Rucksackproblemen.....	10
3. Mehrdimensionale Rucksackprobleme.....	12
3.1. Relaxationsverfahren.....	12
3.2. Exakte Verfahren.....	13
3.3. Heuristiken.....	13
4. Social Choice Theory.....	19
4.1. Social Choice Funktion und individuelle Präferenzen.....	19
4.2. Wahlverfahren der Social Choice Theory.....	22
4.3. Lösen von mehrdimensionalen Rucksackproblemen auf Basis der Social Choice Theory.....	27
5. Vergleich von Greedy-Algorithmen und Social Choice Methoden für das mehrdimensionale Rucksackproblem.....	29
5.1. Vergleich der Greedy-Algorithmen.....	30
5.2. Vergleich der Social Choice Methoden.....	39
5.3. Vergleich der Greedy und Social Choice Methoden.....	48
6. Conclusio.....	52
Literaturverzeichnis.....	53
Anhang: Digitales Informationsmaterial.....	55

Tabellenverzeichnis

Tabelle 2-1: Beispiel zur dynamischen Programmierung	8
Tabelle 4-1: Borda-Vote nach Arrow's Axiome	21
Tabelle 4-2: Borda-Vote nach Arrow's verletztem zweiten Axiom	22
Tabelle 4-3: Beispiel zum Single-Vote Verfahren.....	23
Tabelle 4-4: Beispiel zum Double-Vote-Verfahren	23
Tabelle 4-5: Beispiel zum Borda-Vote Verfahren	24
Tabelle 4-6: Beispiel zur Condorcet-Rule	24
Tabelle 4-7: Beispiel zur Stichwahl	25
Tabelle 4-8: Beispiel zum Approval-Vote	26
Tabelle 4-9: Beispiel zum BordaApproval-Vote.....	26
Tabelle 5-1: Ergebnistabelle Greedy 1 nach Dobson (1982).....	31
Tabelle 5-2: Ergebnistabelle Greedy 2 nach Dobson (1982).....	33
Tabelle 5-3: Ergebnistabelle Greedy 3 nach Senju und Toyoda (1968).....	34
Tabelle 5-4: Ergebnistabelle Greedy 4 nach Loulou und Michaelides (1979).....	36
Tabelle 5-5: Ergebnistabelle Häufigkeit der besten Zielfunktion bei den Greedy Verfahren ..	38
Tabelle 5-6: Ergebnistabelle des Single-Vote	40
Tabelle 5-7: Ergebnistabelle des Borda-Vote	42
Tabelle 5-8: Ergebnistabelle des Approval-Vote	43
Tabelle 5-9: Ergebnistabelle des BordaApproval-Vote.....	45
Tabelle 5-10: Ergebnistabelle Häufigkeit der besten Zielfunktion bei den Social Choice Methoden	47
Tabelle 5-11: Zusammenfassung der Zielfunktionswerte aller Verfahren.....	48
Tabelle 5-12: Zusammenfassung der prozentualen Abweichung zum besten Zielfunktionswert	49
Tabelle 5-13: Zusammenfassung der Häufigkeiten des besten Zielfunktionswertes	50

Formelverzeichnis

Formel 2-1: Klassisches Rucksackproblem: Formulierung des Ganzzahligen Linearen Programms.....	4
Formel 2-2: Mathematische Formulierung zum Greedy-Algorithmus	5
Formel 2-3: Sortierung für den Greedy-Algorithmus	5
Formel 2-4: LP-Relaxation: Formulierung des Ganzzahligen Linearen Programmes.....	6
Formel 2-5: Dynamische Programmierung: Formulierung des Ganzzahliges Lineares Programm	7
Formel 2-6: Bellman's Rekursion.....	7
Formel 3-1: Mehrdimensionales Rucksackproblem: Formulierung des Ganzzahligen Linearen Programms.....	12
Formel 3-2: Greedy 1 nach Dobson.....	14
Formel 3-3: Greedy 2 nach Dobson.....	15
Formel 3-4: Greedy 3 nach Senju und Toyoda	15
Formel 3-5: Greedy mit Einbezug des Relevanzwertes	15
Formel 3-6: Relevanzwert nach Fox und Scudder (1985).....	16
Formel 3-7: Relevanzwert nach Toyoda (1975).....	16
Formel 3-8: Greedy 4 nach Loulou und Michaelides (1979)	17
Formel 4-1: Definitionsbereich der Social Choice Funktion.....	20
Formel 4-2: Social Choice Funktion: Mathematische Formulierung	20
Formel 4-3: Mathematische Formulierung zum Greedy-Algorithmus	27

Abbildungsverzeichnis

Abbildung 2-1: Pseudo-Code für den Greedy-Algorithmus	5
Abbildung 2-2: Pseudo-Code für die dynamische Programmierung.....	8
Abbildung 2-3: Darstellung des B&B-Algorithmus.....	10

Abkürzungsverzeichnis

<i>B&B</i>	Branch-and-Bound
<i>b</i>	Anzahl der identischen Kopien eines Gegenstandes
<i>c</i>	Kapazität des Rucksackes unter einer Restriktion
<i>d</i>	Anzahl an Restriktionen für einen Rucksack
<i>e</i>	Effizienzwert eines Gegenstandes
<i>GLP</i>	Ganzzahliges Lineares Programm
<i>i</i>	Index
<i>j</i>	Index
<i>l</i>	Index
<i>LP</i>	Lineares Programm
<i>m</i>	Anzahl an Rucksäcken
<i>n</i>	Anzahl an Gegenständen
<i>NP</i>	nichtdeterministische Polynomialzeit
<i>O</i>	Laufzeit
<i>p</i>	Profit eines Gegenstandes
q^1	verbrauchte Menge einer Restriktion
q^2	verbleibende Menge einer Restriktion
q^3	möglicher zukünftiger Menge einer Restriktion
<i>r</i>	Relevanzwert einer Restriktion
<i>s. t.</i>	unter den Bedingungen
t_1, t_2	Sensitivitätsparameter
<i>v</i>	Pönlwert
<i>w</i>	Gewicht eines Gegenstandes unter einer Restriktion
<i>x</i>	Entscheidungsvariable für das Einpacken eines Gegenstandes
<i>z</i>	Zielfunktionswert
\in	Element von
\forall	für alle
\subseteq	Teilmenge von
\cup	vereinigt

Zusammenfassung

Das mehrdimensionale Rucksackproblem ist ein einfaches Optimierungsmodell der kombinatorischen Optimierung, bei dem auf Basis von mehreren vorgegebenen Restriktionen eine Teilmenge an Gegenständen zur Maximierung des Zielfunktionswertes ausgewählt wird. In der Praxis wird das mehrdimensionale Rucksackproblem beispielsweise bei Transport-, Produktions-, und Portfolio-Problemen angewendet. In der Literatur wurden zur Lösung dieser Probleme bisher eine Vielzahl an approximativen Algorithmen vorgestellt. Bekannte Vertreter dieser Approximationsverfahren waren beispielsweise Dobson (1982), Senju und Toyoda (1968) sowie Loulou und Michaelides (1979). Ihre Greedy-Verfahren ermitteln in überschaubarer Zeit brauchbare Annäherungen an den optimalen Zielfunktionswert. Völlig gegensätzlich hierzu werden auch Social Choice Methoden als Lösungsverfahren für Rucksackprobleme verwendet. Dabei verwaltet ein Individuum jeweils eine Ressource und reiht die Gegenstände anhand der Effizienz bezüglich dieser Ressource. Haben alle Individuen diese Reihung vorgenommen, erfolgt die Stimmenvergabe für jeden Gegenstand und die finale Sortierung für das Einpacken des Rucksacks. Somit wird im Kollektiv und mit einer beliebigen Wahlmethode eine entsprechende Teilmenge ausgewählt. Bekannte Social Choice Methoden sind beispielsweise Single-Vote, Borda-Vota oder auch Approval-Vote.

In der vorliegenden Arbeit wird nun geklärt, welches der vielen Verfahren für das mehrdimensionale Rucksackproblem den höchsten Zielfunktionswert erzielt. Hierfür werden die Testinstanzen von Chu und Beasley (1998) herangezogen und mit den Greedy-Verfahren sowie Social Choice Methoden getestet. Grundsätzlich ist erkennbar, dass die Greedy-Verfahren im Gesamten bessere Ergebnisse liefern, als die Verfahren der Social Choice Methoden. Lediglich Borda-Vota ist konkurrenzfähig und erzielt aufgrund der vollständigen Präferenzordnung der Individuen gute Resultate. Eine eindeutige Tendenz für die verschiedenen Variablen- und Restriktionskombinationen gibt es bei keinen einzigen Verfahren, vielmehr hängen die Zielfunktionswerte und die Häufigkeit der besten Zielfunktionswerte stark von den zu Grunde liegenden Testinstanzen ab, weshalb es kein Universalverfahren gibt, das durchwegs die besten Ergebnisse liefert. In der Praxis ist neben der Abweichung zum Optimum auch die Laufzeit für die Auswahl eines Approximationsverfahrens relevant, weshalb oftmals ein Trade-Off zwischen diesen beiden Eigenschaften zielführend ist.

1. Einleitung

Das mehrdimensionale Rucksackproblem ist ein NP-schweres¹ Problem und kann daher von keinem Verfahren in überschaubarer Zeit exakt gelöst werden. Vielmehr sind unterschiedliche approximative Verfahren notwendig, die innerhalb einer reduzierten Laufzeit eine ökonomisch sinnvolle Annäherung an das Maximum erreichen. (Vgl. Freville, 2004:1-4) Die wohl intuitivsten approximativen Verfahren sind die sogenannten Greedy-Algorithmen, die „gierig“ Gegenstände nach einer vorgegebenen Sortierung einpacken. (Vgl. Neumann und Morlock, 2002:408-410) Dobson (1982), Senju und Toyoda (1968), Loulou und Michaelides (1979) sowie Fox und Scudder (1985) haben zu diesem Zweck bestimmte Effizienzwerte als Kriterium zur Sortierung von Gegenständen entwickelt.

Eine weitere Möglichkeit zur Erzielung einer Annäherung an das Maximum ist die Verwendung der Social Choice Theory. Durch kollektive Entscheidung wird eine sortierte Reihenfolge der einzupackenden Gegenstände festgelegt. (Vgl. Kelly, 1988:1-3)

Fraglich ist nun, welcher dieser Effizienzwerte bzw. welche Sortierung anhand der Social Choice Theory die beste Performance im Sinne eines hohen Zielfunktionswertes – im Vergleich miteinander und untereinander – unabhängig von den Testdaten erreicht. Zusätzlich wird auch ermittelt, welches Verfahren den geringsten Abstand zur LP-Lösung für die Benchmark-Instanzen von Chu und Beasley (1998) aufweist. Ziel der Arbeit ist eine detaillierte Ermittlung aller Zielfunktionswerte inklusive Aufstellung einer entsprechenden Ergebnistabelle, die einen Vergleich der Sortier- und Einpackresultate zulässt.

Methodisch wird die oben beschriebene Fragestellung durch Programmierung in MATLAB mit den Testdaten von Chu und Beasley (1998) und deren Aufbereitung bearbeitet.

Für eine hinreichende Beantwortung der Fragestellung erfolgt in den ersten beiden Kapiteln eine theoretische Einführung in das Rucksack- bzw. in das mehrdimensionale Rucksackproblem auf Basis der aktuellen Literatur, bei der vor allem der Greedy-Algorithmus besondere Berücksichtigung findet. Im vierten Kapitel werden die Ideen und Prämissen der Social Choice Theory dargelegt, in dem vor allem die verschiedenen Wahlverfahren, die für die weitere Analyse relevant sind, besondere Berücksichtigung finden, denn hierauf basiert die spätere Sortierung und das Einpacken der Gegenstände. Schlussendlich wird im fünften Kapitel der Vergleich der Greedy-Algorithmen bzw. der Wahlverfahren der Social Choice Theory für das mehrdimensionale Rucksackproblem beschrieben und eine Bewertung der Performance vorgenommen.

¹ siehe Seite 3

2. Rucksackprobleme

Das Rucksackproblem (engl.: knapsack problem) beschreibt im Wesentlichen ein einfaches Optimierungsmodell aus dem Bereich der kombinatorischen Optimierung, bei dem Entscheidungen auf Basis von numerischen Werten – beispielsweise Kosten, Profit oder Nutzen – getroffen werden, welche in weiterer Folge den Vergleich zwischen einzelnen Entscheidungen zulassen. Beim Rucksackproblem wird dazu jene Teilmenge an Gegenständen gesucht, die einerseits bestimmte Restriktionen erfüllt und andererseits hinsichtlich des numerischen Wertes optimal ist. (Vgl. Martello und Toth, 1990:13-14) Viele Probleme, die in der kombinatorischen Optimierung gelöst werden und auch das Rucksackproblem, sind als NP-schwer charakterisiert. Dies sind Probleme, die bei steigender Variablenanzahl n von keinem Algorithmus in polynomieller Laufzeit² $O(n^k)$ für die Konstante k gelöst werden können. (Vgl. Kellerer/Pferschy/Pisinger, 2004:13-14)

Neumann und Morlock (2002) beschreiben das Rucksackproblem mit einer einfachen Metapher: Ein Wanderer hat einen Rucksack mit begrenzter Kapazität c und kann diesen mit n Gegenständen füllen. Jeder Gegenstand j generiert für den Wanderer sowohl einen Nutzen p_j als auch ein dazugehöriges Gewicht w_j . Nun muss der Wanderer entscheiden, welche Teilmenge an Gegenstände er in seinen Rucksack einpackt und welche er nicht einpackt, damit der Wanderer sein Nutzenmaximum erreicht. Voraussetzung für eine sinnvolle Lösung ist hierbei $c \geq \max(w_j) \forall j$. (Vgl. Neumann und Morlock, 2002:406-407) In der Praxis wird das Rucksackproblem bei zahlreichen Fragestellungen eingesetzt, beispielsweise wenn es darum geht einen Frachttransporter optimal zu beladen, das optimale Produktionsprogramm festzulegen oder auch das optimale Finanzportfolio zu erwerben. (Vgl. Martello und Toth, 1990:10)

Die einfachste Form – auch bekannt unter 0-1 Rucksackproblem – zur Entscheidungsfindung des Wanderers ist die binäre Entscheidung, die zwischen zwei Alternativen anhand einer Binärvariable $x_j \in \{0,1\}$ auswählt. Dementsprechend gibt es die Möglichkeit zwischen „Gegenstand einpacken“ $x_j = 1$ und „Gegenstand nicht einpacken“ $x_j = 0$. Daraus resultiert, dass das Rucksackproblem aus n Binärvariablen besteht, die für jeden Gegenstand j bestimmen, ob dieser eingepackt wird oder auch nicht. (Vgl. Kellerer/Pferschy/Pisinger, 2004:2)

² unter Laufzeit eines Algorithmus werden die zur Lösung maximal notwendigen Rechenschritte bezeichnet. (Vgl. Kellerer/Pferschy/Pisinger, 2004:13-14)

Mathematisch kann das vorliegende Problem durch ein Ganzzahliges Lineares Programm³ dargestellt werden. Darin gilt es jene Kombination von Gegenständen zu finden, die ein Profitmaximum unter Einhaltung der Gewichtskapazität erzielt. (Vgl. Kellerer/Pferschy/Pisinger, 2004:3)

$$\max \sum_{j=1}^n p_j x_j \quad (2-1.1)$$

$$s. t. \sum_{j=1}^n w_j x_j \leq c \quad (2-1.2)$$

$$x_j \in \{0,1\} \quad \forall j \quad (2-1.3)$$

Formel 2-1: Klassisches Rucksackproblem: Formulierung des Ganzzahligen Linearen Programms

(Quelle: Kellerer/Pferschy/Pisinger, 2004:3)

Im Laufe der Jahre wurde das Rucksackproblem in der Literatur weitgreifend behandelt, wobei vor allem die Vielzahl an verschiedenen Formen und Lösungsverfahren besondere Berücksichtigung fanden. Bellman beispielsweise hat erstmals in den 50er Jahren einen exakten Algorithmus zur Lösung des 0-1 Rucksackproblems vorgestellt. Nur einige Jahre später entwickelte Dantzig die LP-Relaxationslösung, die elegant und effizient eine obere Schranke zum optimalen Zielfunktionswert berechnet. 1967 haben Gilmore und Gomory den Branch-and-Bound Algorithmus erstmals für Rucksackprobleme angewandt und akzeptable Ergebnisse erzielt. (Vgl. Martello und Toth, 1990:14-15) All diese erwähnten Entwicklungen zur Lösung des Rucksackproblems sowie deren Verschiedenartigkeit werden nun in weiterer Folge näher beschrieben, da sie teilweise zur Beantwortung der Fragestellung von Relevanz sind und zur Steigerung des allgemeinen Verständnisses beitragen.

2.1. Algorithmen zur Lösung von Rucksackproblemen

Wie bereits anfänglich skizziert, gibt es eine Fülle an verschiedenartigen Verfahren zur Lösung von Rucksackproblemen. Im Übrigen sind aufgrund der Tatsache der NP-Schwere alle in weiterer Folge vorgestellten Algorithmen lediglich Verfahren mit reduzierter Laufzeit, die eine Annäherung an das Optimum in akzeptabler Zeit garantieren. Alle Approximationsverfahren sind als Eröffnungsverfahren deklariert und können durch ein entsprechendes Verbesserungsverfahren erweitert werden.

³ Bei dem Ganzzahligen Linearen Programm (GLP) dürfen die Variablen anstatt reeler Werte nur ganzzahlige Werte für die Optimierung annehmen. (Vgl. Kellerer/Pferschy/Pisinger, 2004:6)

Greedy-Algorithmus

Der erste und wohl intuitivste ist der sogenannte Greedy-Algorithmus (dt. gieriger Algorithmus), der ein Eröffnungsverfahren ist und die „profitabelsten“ Gegenstände j in den Rucksack packt, sofern noch ausreichend Kapazität dafür vorhanden ist. (Vgl. Neumann und Morlock, 2002:408-410) Die „Profitabilität“ selbst lässt sich mittels der Effizienz eines Gegenstandes e_j messen, die das Verhältnis von Profit zu Gewicht ausdrückt. Mathematisch ist dieser Zusammenhang in Formel 2–2 beschrieben. (Vgl. Kellerer/Pferschy/Pisinger, 2004:15)

$$e_j = \frac{p_j}{w_j} \quad \forall j \quad (2-2.1)$$

Formel 2-2: Mathematische Formulierung zum Greedy-Algorithmus

(Quelle: Kellerer/Pferschy/Pisinger, 2004:15)

Zum Einpacken selbst werden nun alle n Gegenstände absteigend nach e_j sortiert, sodass gilt:

$$\frac{p_1}{w_1} \geq \frac{p_2}{w_2} \geq \dots \geq \frac{p_n}{w_n} \quad (2-3.1)$$

Formel 2-3: Sortierung für den Greedy-Algorithmus

(Quelle: Kellerer/Pferschy/Pisinger, 2004:15)

Die nach Formel 2–3 sortierten Gegenstände werden sukzessiv, und solange die Restkapazität vorhanden ist, in einen anfänglich leeren Rucksack eingepackt. Der Profit aller tatsächlich eingepackten Gegenstände j ergibt summiert den gesuchten Zielfunktionswert. Je besser die Güte der Sortierung ist, desto besser ist schlussendlich auch die erzielte Annäherung an das Optimum. Eine explizite Beschreibung des Greedy-Verfahrens in Pseudocode-Form bietet Abbildung 2–1. Ein Pseudocode beschreibt allgemein die einzelnen Verfahrensschritte eines Algorithmus in detaillierter, jedoch nicht programmspezifischer, Art und Weise. (Vgl. Kellerer/Pferschy/Pisinger, 2004:16)

```
 $\bar{w} := 0$   $\bar{w}$  ist das Gesamtgewicht der bisher eingepackten Gegenstände  
 $z^G := 0$   $z^G$  ist der Profit der bisher eingepackten Gegenstände  
for  $j := 1$  to  $n$  do  
    if  $\bar{w} + w_j \leq c$  then  
         $x_j := 1$  Gegenstand  $i$  wird eingepackt  
         $\bar{w} := \bar{w} + w_j$   
         $z^G := z^G + p_j$   
    else  $x_j := 0$ 
```

Abbildung 2-1: Pseudo-Code für den Greedy-Algorithmus

(Quelle: Kellerer/Pferschy/Pisinger, 2004:16)

Linear Programming Relaxation

Der zweite Algorithmus zur approximativen Lösung von Rucksackproblemen ist die Linear Programming (LP) Relaxation, bei der die binäre Bedingung der Entscheidungsvariable x_j „relaxiert“ wird, wodurch das Problem mit Hilfe der linearen Optimierung lösbar ist. Durch die Relaxation verliert das Rucksackproblem den NP-schweren Charakter und lässt sich in polynomieller Zeit lösen. Formel 2-4 stellt das Ganzzahlige Lineare Programm einschließlich der LP-Relaxation mathematisch dar. (Vgl. Matousek und Gärtner, 2007:29-31)

$$\max \sum_{j=1}^n p_j x_j \quad (2-4.1)$$

$$\text{s. t. } \sum_{j=1}^n w_j x_j \leq c \quad (2-4.2)$$

$$0 \leq x_j \leq 1 \quad \forall j \quad (2-4.3)$$

Formel 2-4: LP-Relaxation: Formulierung des Ganzzahligen Linearen Programmes

(Quelle: Matousek und Gärtner, 2007:30)

Dementsprechend findet das LP-Relaxation bei Rucksackproblemen ein Optimum, das stets einen höheren Zielfunktionswert als jenes mit Integralbedingung aufweist, da eine Optimierung mit reellen Zahlen durchgeführt wird. Selbst wenn es nicht nachvollziehbar erscheint reelle Zahlen zu verwenden – impliziert teilweise eingepackte Gegenstände –, wird an dieser Stelle aber nochmals auf die Festlegung einer oberen Schranke für den Zielfunktionswert hingewiesen. (Vgl. Matousek und Gärtner, 2007:32-39)

Neben der linearen Optimierung ist zur Lösung auch die nichtlineare Optimierung und das Verfahren der Lagrange-Relaxierung geeignet. Die Grundidee ist jene, dass störende Ungleichungen mittels positiven Lagrange-Multiplikator in die Zielfunktion einfließen und das Ganzzahlige Lineare Programm trivial lösbar macht. Die Schwierigkeit der Lagrange-Relaxierung ist aber die optimale Festlegung bzw. auch die Fixierung des Multiplikators. Für das 0-1 Rucksackproblem wird diese Variante des Lösungswegs aber eher selten verwendet. (Vgl. Martello und Toth, 1990:19)

Dynamische Programmierung

Die oben beschriebenen Verfahren sind sehr einfache und intuitive Approximationen an den optimalen Zielfunktionswert. Die LP-Relaxation zeichnet sich aber lediglich durch die Benennung einer oberen Schranke und das Greedy-Verfahren durch die Benennung einer unteren Schranke aus. Das dynamische Programmieren von Bellman (1957) hingegen ist ein Exaktverfahren, welches zuerst kleine Subprobleme löst und diese in weiterer Folge iterativ auf

das komplette Problem ausweitet.⁴ Vor allem das Rucksackproblem erfüllt diese Voraussetzung, denn reduziert man die einzupackenden Gegenstände n um den Gegenstand j , reduziert sich auch die Kapazität $c - w_j$ und stellt ein Subproblem dar. (Vgl. Freville, 2004:12-14)

Die dynamische Programmierung zur Lösung des Rucksackproblems kann folgendermaßen beschrieben werden: Es gibt bereits eine optimale Lösung für ein bestimmtes Subproblem. Nun wird iterativ jeweils ein weiterer Gegenstand zur bestehenden Lösung hinzugefügt und überprüft, ob der optimale Zielfunktionswert einer Anpassung bedarf. Diese Prozedur wird solange durchgeführt, bis alle relevanten Gegenstände eingepackt und das Optimum gefunden wurde. (Vgl. Freville, 2004:14-15)

Mathematisch kann die dynamische Programmierung durch Formel 2–5 dargestellt werden. Eine Menge an Gegenständen $\{1 \dots j\}$ wird in einen Rucksack mit reduzierter Kapazität $d \leq c$ eingepackt. (Vgl. Kellerer/Pferschy/Pisinger, 2004:20)

$$\max \sum_{l=1}^j p_l x_l \quad (2-5.1)$$

$$s. t. \sum_{l=1}^j w_l x_l \leq d \quad (2-5.2)$$

$$x_l \in \{0,1\} \quad \forall j \quad (2-5.3)$$

Formel 2-5: Dynamische Programmierung: Formulierung des Ganzzahligen Lineares Programm

(Quelle: Kellerer/Pferschy/Pisinger, 2004:20)

Zusätzlich zu dem oben beschriebenen Ganzzahligen Linearen Programm hat Bellman (1957) sowohl den optimalen Zielfunktionswert $z_j(d)$ als auch die Rekursion erstmalig dargestellt. Rekursion bedeutet im Allgemeinen, dass eine Funktion durch sich selbst definiert wird und dementsprechend zu sich selbst zurückläuft. Bellman's Rekursion ist in Formel 2–6 festgehalten und beschreibt $z_j(d)$ in Abhängigkeit der vorherigen Iteration $z_{j-1}(d)$. (Vgl. Kellerer/Pferschy/Pisinger, 2004:21)

$$z_j(d) = \begin{cases} z_{j-1}(d) & \text{if } d < w_j \\ \max\{z_{j-1}(d), z_{j-1}(d - w_j) + p_j\} & \text{if } d \geq w_j \end{cases} \quad (2-6.1)$$

Formel 2-6: Bellman's Rekursion

(Quelle: Kellerer/Pferschy/Pisinger, 2004:21)

⁴ diese Eigenschaft ist auch als „Bellman's Principle of Optimality“ bekannt. (Vgl. Bellman, 1957)

Im Falle von $d < w_j$ ist der Rucksack für den Gegenstand j zu klein, weshalb der Zielfunktionswert unverändert mit $z_{j-1}(d)$ bleibt. Passt der Gegenstand j in den Rucksack, gibt es zwei Möglichkeiten. Entweder Gegenstand j wird nicht eingepackt und der Zielfunktionswert ändert sich mit $z_{j-1}(d)$ nicht, oder er wird eingepackt, wodurch der Zielfunktionswert um p_j steigt, aber um die verbleibende Kapazität w_j sinkt. Das Maximum dieser beiden Alternativen führt zum Optimum der folgenden Iteration. (Vgl. Kellerer/Pferschy/Pisinger, 2004:21-22)

Anhand eines selbstgewählten Beispiels in Abbildung 2–3, dessen Pseudo-Code in Abbildung 2–2 beschrieben ist, wird die Vorgehensweise der dynamischen Programmierung beschrieben. Der erste Schritt ist die Initialisierung des Zielfunktionswertes, auf dem aufbauend die Rekursion durchgeführt wird.

```

for d := 0 to c do
    z_0(d) := 0 Initialisierung
for j := 1 to n do
    for d := 0 to w_j - 1 do
        z_j(d) := z_{j-1}(d)
    for d := w_j to c do
        if z_{j-1}(d - w_j) + p_j > z_{j-1}(d) then
            z_j(d) := z_{j-1}(d - w_j) + p_j
        else z_j(d) := z_{j-1}(d)
z* := z_n(c)

```

Abbildung 2-2: Pseudo-Code für die dynamische Programmierung

(Quelle: Kellerer/Pferschy/Pisinger, 2004:22)

Ein Beispiel zum eindimensionalen Rucksackproblem mit folgenden Ausgangsdaten ist gegeben:

$$\max 4p_1 + 3p_2 + 3p_3 + 2p_4$$

$$\text{u. d. N. } 5w_1 + 4w_2 + 2w_3 + 1w_4 \leq 6$$

d	$z_0(d)$	$z_1(d)$	x_1	$z_2(d)$	x_2	$z_3(d)$	x_3	$z_4(d)$	x_4
0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0,2	1
2	0	0	0	0	0	0,3	1	3,2	0
3	0	0	0	0	0	0,3	1	3,5	1
4	0	0	0	0,3	1	3,3	0/1	3,5	1
5	0	0,4	1	4,3	0	4,3	0	4,5	1
6	0	0,4	1	4,3	0	4,6	1	6,6	0/1

Tabelle 2-1: Beispiel zur dynamischen Programmierung

(Quelle: eigene Darstellung)

Für das selbstgewählte Beispiel gibt es zwei Lösungsmöglichkeiten. Entweder kann Gegenstand 4 gemeinsam mit Gegenstand 1, oder Gegenstand 3 gemeinsam mit Gegenstand 2 eingepackt werden. Beide führen zum selben Zielfunktionswert. Ein Spezialfall bei diesem Verfahren ist jener, bei dem die Rekursion zu keinem eindeutigen Maximum führt, wie es beispielsweise im Punkt $z_4(6)$ der Fall ist. An dieser Stelle gibt es eine Indifferenz zwischen den beiden Entscheidungsmöglichkeiten bzw. eine beliebige Auswahl zwischen Gegenstand einpacken oder Gegenstand nicht einpacken. Die Lösung des dynamischen Programmierens gelingt in $O(nc)$ und ist ein pseudopolynomieller Algorithmus, dessen Laufzeit ein Polynom im numerischen Wert des Parameters ist. (Vgl. Kellerer/Pferschy/Pisinger, 2004:487)

Branch-and-Bound (B&B)

Einen vollkommen anderen, aber ebenso exakten, Lösungsweg bietet der vierte Algorithmus, der Branch-and-Bound-Algorithmus (B&B), der durch intelligente Enumeration⁵ alle brauchbaren Lösungen anhand eines Baumes auflistet und jene mit dem höchsten Zielfunktionswert auswählt. Problematisch wird Branch-and-Bound nur bei einer hohen Gegenstandsanzahl n , denn die Anzahl der möglichen Lösungen entspricht 2^n . Dieses Problem wird aber durch die intelligente Enumeration behoben, da ausschließlich bessere Verzweigungen weiterverfolgt werden und garantiert ist, dass im nicht berücksichtigten Lösungsraum keine Optimallösung vorhanden ist. Wie der Name des B&B bereits verrät, besteht der Algorithmus aus zwei wichtigen Bestandteilen. Beim Branching wird der Lösungsraum in disjunkte (überschneidungsfreie) Teilmengen zerlegt und beim Bounding wird für diese disjunkten Teilmengen eine obere bzw. untere Schranke definiert. Unterschreitet bzw. Überschreitet ein Zielfunktionswert die Schranke, wird sie auf den entsprechend neuen Zielfunktionswert angepasst. Am Ende des B&B ist der Zielfunktionswert gleich dem letztgültigen Schrankenwert und das Optimum ist gefunden. Zum Durchsuchen des Baumes wählt man entweder die Tiefensuche – ein Ast wird von Anfang bis zum Ende durchsucht und entspricht einer rekursiven Suchstruktur – oder auch die Breitensuche – alle Äste einer Ebene werden durchsucht, bevor man in die Tiefe geht. (Vgl. Domschke und Drexl, 2007:133-137)

Zur besseren Veranschaulichung des B&B ist in Abbildung 2–3 ein Wertebaum skizziert. In den Knoten befindet sich der obere bzw. untere Schrankenwert. Der „fett“ markierte Knotenwert entspricht der Startlösung. Nach dem eben vorgestellten Verfahren wird bei der Minimierung des Zielfunktionswertes lediglich der rechte Ast (= 13) weiterverfolgt, der linke Ast wird gestrichen. (Vgl. Domschke und Drexl, 2007:139)

⁵ Enumeration ist die Aufzählung eines endlichen Wertebereichs (Vgl. Domschke und Drexl, 2007:134)

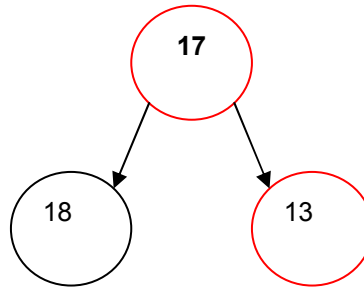


Abbildung 2-3: Darstellung des B&B-Algorithmus

(Quelle: eigene Darstellung, i.A.a. Domschke und Drexl, 2007:139)

Zusätzlich zu den hier vorgestellten approximativen sowie exakten Verfahren haben Kellerer/Pferschy/Pisinger (2004) in ihrem Buch unter Kapitel 3 noch eine weitere Auswahl von komplexeren Algorithmen zur Lösung des Rucksackproblems vorgestellt. Für den Zweck dieser Arbeit sind die bisherigen Ausführungen ausreichend, denn der Fokus liegt auf der Erarbeitung des Greedy-Verfahrens.

2.2. Formen von Rucksackproblemen

Neben dem klassischen binären gibt es in der Literatur auch weitere wichtige Formen des Rucksackproblems, die nachfolgend zur Vollständigkeit in einigen Sätzen genannt werden.

Bounded Rucksackproblem

Beim bounded Rucksackproblem gibt es eine endliche Menge an identischen Kopien b_j von Gegenstand j , die eingepackt werden können. Durch die Kopien eines Gegenstandes verändert sich zum klassischen Problem lediglich Bedingung (2–1.3) aus Formel 2–1 zu $0 \leq x_j \leq b_j$. Dementsprechend kann x_j jeden ganzzahligen Wert zwischen 0 und b_j annehmen. (Vgl. Martello und Toth, 1990:81-82)

Unbounded Rucksackproblem

Im Gegensatz dazu ist beim unbounded Rucksackproblem eine unendliche Menge an identischen Kopien b_j von Gegenstand j vorhanden, wodurch sich Bedingung (2–1.3) aus Formel 2–1 zu $x_j \geq 0$ ändert und jeden ganzzahligen Wert größer gleich 0 annimmt. Sowohl das bounded als auch das unbounded Rucksackproblem können in das klassische binäre Problem transformiert werden, wodurch alle in Kapitel 2 vorgestellten Lösungsverfahren gültig werden. (Vgl. Kellerer/Pferschy/Pisinger, 2004:211-213)

Mehrdimensionales Rucksackproblem

Um der Komplexität der realen Welt gerecht zu werden, sind beim mehrdimensionalen Rucksackproblem zusätzliche Restriktionen für Gegenstand j im Ganzzahligen Linearen Programm enthalten. (Vgl. Domschke und Drexl, 2007:141) Kapitel 3 der vorliegenden Arbeit beschäftigt sich eingehend mit dieser Form von Rucksackproblemen.

Multiples Rucksackproblem

Die letzte hier beschriebene Ausprägung ist das multiple Rucksackproblem, bei dem anstatt eines einzelnen Rucksackes mehrere Rucksäcke m mit unterschiedlichen Kapazitäten zur Verfügung stehen. Ziel ist es, alle Gegenstände derart in den Rucksäcken zu verteilen, dass keine Restriktion verletzt und insgesamt das Profitmaximum erreicht wird. Der große Unterschied zum klassischen Rucksackproblem liegt in der Binärvariablenmatrix, welche beschreibt, welcher Gegenstand j in welchen Rucksack i eingepackt wird. (Vgl. Martello und Toth, 1990:157-159)

Neben den hier beschriebenen Formen haben Kellerer/Pferschy/Pisinger (2004) weitere Ausprägungen von Rucksackproblemen in ihrem Buch unter Kapitel 11-13 dargelegt.

3. Mehrdimensionale Rucksackprobleme

Für die Beschreibung von realen Problemen ist eine einzelne Restriktion zur Auswahl von Gegenständen für das Einpacken nicht ausreichend. Vielmehr ist es notwendig, mehrere beeinflussende Faktoren zu berücksichtigen, um die passende Entscheidung zu treffen. Genau diese Idee verfolgt das mehrdimensionale Rucksackproblem durch das Einbeziehen von d unterschiedlichen Restriktionen. (Vgl. Domschke und Drexl, 2007:141-143) Das dazugehörige Ganzzahlige Lineare Programm ist in Formel 3–1 abgebildet.

$$\max \sum_{j=1}^n p_j x_j \quad (3-1.1)$$

$$\text{s. t. } \sum_{j=1}^n w_{ij} x_j \leq c_i \quad \forall i \quad (3-1.2)$$

$$x_j \in \{0,1\} \quad \forall j \quad (3-1.3)$$

Formel 3-1: Mehrdimensionales Rucksackproblem: Formulierung des Ganzzahligen Linearen Programms
(Quelle: Akcay/Li/Xu, 2007:2)

Um triviale Ergebnisse schon anfänglich auszuschließen, muss einerseits $w_{ij} \leq c_i \forall i, j$ angenommen werden, wodurch jede Gewichtsrestriktion eines Gegenstandes einzeln betrachtet kleiner gleich der dazugehörige Gewichtskapazität der Restriktion ist und andererseits muss $\sum_{j=1}^n w_{ij} \geq c_i \forall i$ angenommen werden, wodurch das Gesamtgewicht aller Gegenstände unter Restriktion i größer gleich der dazugehörigen Gewichtskapazität ist. (Vgl. Freville, 2004:1-2)

Wie schon beim klassischen Rucksackproblem, kann auch das mehrdimensionale Rucksackproblem mit verschiedenen approximativen bzw. exakten Verfahren gelöst werden. Dabei fassen Kellerer/Pferschy/Pisinger (2004) alle bekannten Lösungswege in drei Gruppen zusammen. Die erste Gruppe behandelt Relaxationsverfahren, die zweite Gruppe exakte Verfahren und die dritte Gruppe umfasst Heuristiken. (Vgl. Kellerer/Pferschy/Pisinger, 2004:256)

3.1. Relaxationsverfahren

Beim klassischen Rucksackproblem konnte mittels LP-Relaxation eine obere Schranke für den Zielfunktionswert ermittelt werden. Beim mehrdimensionalen Rucksackproblem sind verschiedene Relaxationsverfahren (LP-Relaxation, Lagrange-Relaxation, Surrogate-Relaxation und Composite-Relaxation) für die Benennung eines Schrankenwertes relevant und lösen die Integralbedingung der Entscheidungsvariable auf. Diese Verfahren sind von der Anzahl an Restriktionen sowie von der Anzahl an Gegenständen stark abhängig. Je grö-

ßer die Anzahl, desto komplexer wird das Verfahren bzw. desto ungenauer wird die Berechnung des Schrankenwertes. (Vgl. Kellerer/Pferschy/Pisinger, 2004: 238-246)

3.2. Exakte Verfahren

Das erste Verfahren zur exakten Lösung von mehrdimensionalen Rucksackproblemen war das Branch-and-Bound Verfahren in den 1960er Jahren. Wenig später gab es aber auch einen Beitrag zur dynamischen Programmierung. (Vgl. Domschke und Drexl, 2007:139)

Branch-and-Bound

In erster Linie soll an diesem Punkt angemerkt werden, dass der B&B Algorithmus aus Kapitel 2 auch für das mehrdimensionale Rucksackproblem Anwendung findet. Lediglich zur Berechnung der notwendigen oberen Schranke wird eine Familie von d binären Rucksackproblemen mit der dazugehörigen i -ten Restriktion benötigt, wobei gilt: $i \in \{1, \dots, d\}$. Für diese Familie wird jeweils der optimale Zielfunktionswert z_i^* berechnet und die obere Schranke $z^* \leq U := \min\{z_i^* | i = 1, \dots, d\}$ definiert. Jedes z_i^* stellt nämlich eine obere Schranke für z^* dar. (Vgl. Kellerer/Pferschy/Pisinger, 2004:246-247) Diese Vorgehensweise ist sehr rechen- und zeitintensiv, weshalb Garvish und Pirkul (1985) eine Möglichkeit fanden den Wertebaum zu reduzieren und nur relevante Lösungen mit einzubeziehen. Zusätzlich ist es vorteilhaft, dass nur ein einzelnes universelles Verfahren für verschiedenartige Rucksackprobleme notwendig ist. (Vgl. Garvish und Pirkul, 1985:78-105)

Dynamische Programmierung

Im Gegensatz zum klassischen Rucksackproblem ist die Berechnung des optimalen Zielfunktionswertes mittels dynamischer Programmierung beim mehrdimensionalen Rucksackproblem problematisch. Eine Wertetabelle der Größe $n * c_1 * \dots * c_d$ mit Laufzeit $O(nc_{max}^d)$ bewirkt rechen- und zeitintensive Algorithmusschritte. Anfängliche Rechenversuche waren daher mit zwei Restriktionen limitiert. (Vgl. Kellerer/Pferschy/Pisinger, 2004:248-249)

3.3. Heuristiken

Aufgrund der Komplexität von mehrdimensionalen Rucksackproblemen, wird in der Literatur vor allem der Bereich der Heuristiken intensiv erforscht, da diese Algorithmen brauchbare Lösungen in guter Qualität und annehmbar geringer Laufzeit erzeugen. Innerhalb der Heuristiken werden Greedy-Verfahren, relaxationsbasierte Verfahren, approximierete dynamische Programmierung und Metaheuristiken unterschieden. (Vgl. Kellerer/Pferschy/Pisinger, 2004:256)

Greedy-Verfahren

Ähnlich zum Greedy-Algorithmus für das „0–1“-Rucksackproblem, werden beim mehrdimensionalen Rucksackproblem gleichermaßen Gegenstände nach einer bestimmten Sortierung eingepackt, sofern keine der d Restriktionen verletzt wird. Dieser Vorgang wird auch als primales Greedy-Verfahren bezeichnet, bei dem mit einem leeren Rucksack begonnen und dieser sukzessiv befüllt wird. Dabei durchläuft der primale Greedy alle n Gegenstände. Im Gegensatz dazu gibt es aber auch das duale Greedy-Verfahren, bei dem es zur schrittweisen Entleerung eines komplett gepackten Rucksackes kommt und die erste mögliche Lösung zum Abbruch führt. Beim dualen Greedy werden also nur all jene Gegenstände durchlaufen, die aufgrund der Gewichtskapazität nicht in den Rucksack passen. Grundsätzlich lässt sich zeigen, dass der primale Greedy bessere Ergebnisse als der duale Greedy erzielt, da beim primalen Greedy deutlich mehr Gegenstände – und zwar n Gegenstände – für das Einpacken relevant sind und große Kapazitätslücken innerhalb der d Restriktionen gewissermaßen gefüllt werden. Mit einer erneuten Verbesserungsroutine, die alle bisher nicht eingepackten Gegenstände nochmals untersucht, lassen sich die Abweichungen beim dualen Greedy reduzieren. Unabhängig von den beiden Einpackstrategien ist die Frage der Sortierung der Gegenstände von zentraler Bedeutung. Dafür ist es notwendig die „Attraktivität“ bzw. die „Profitabilität“ des Gegenstandes j mit einem passenden Wert auszudrücken. (Vgl. Kellerer/Pferschy/Pisinger, 2004:256-257)

Dobson (1982), Senju und Toyoda (1968), Loulou und Michaelides (1979) sowie Fox und Scutter (1985) haben dazu den Effizienzwert e_j entwickelt. Zur Unterscheidung sind die verschiedenen Effizienzwerte in dieser Arbeit von 1-4 nummeriert. Der erste Effizienzwert ist in Formel 3–2 dargestellt und setzt den Profit eines Gegenstandes j mit dem Gesamtgewicht aller Restriktionen eines Gegenstandes j ins Verhältnis, wodurch eine „absolute“ Effizienz für jeden Gegenstand j generiert wird. (Vgl. Dobson, 1982:515-521)

$$e1_j := \frac{p_j}{\sum_{i=1}^d w_{ij}} \quad \forall j \quad (3-2.1)$$

Formel 3-2: Greedy 1 nach Dobson

(Quelle: Kellerer/Pferschy/Pisinger, 2004:257)

Der „absolute“ Effizienzwert aus Formel 3–2 ist ein sehr naiver Ansatz zur Ermittlung der „Attraktivität“ eines Gegenstandes, da auf die Skalierung der Restriktionen nicht geachtet wird. Vielmehr verzerren sehr dominante Restriktionswerte das Ergebnis und führen zur Unbrauchbarkeit des Ergebnisses. Beispielsweise ist es vorstellbar, dass Millimeter mit Stunden und Gewicht summiert wird. Um dem entgegenzuwirken, hat Dobson (1982) einen „relativen“

Effizienzwert entworfen, der den Profit mit dem relativen skalierteren Gesamtgewicht ins Verhältnis setzt. Hierzu wird das Gewicht mit der dazugehörigen Kapazität in Verbindung gebracht. Dadurch ist gewährleistet, dass etwaige hohe Gewichte auch durch entsprechend hohe Kapazitäten ausgeglichen werden. Somit lassen sich die Restriktionen schlussendlich problemlos addieren ohne das Resultat zu verzerren. Formel 3–3 stellt diesen Zusammenhang mathematisch dar. (Vgl. Dobson, 1982:515-521)

$$e2_j := \frac{p_j}{\sum_{i=1}^d \frac{w_{ij}}{c_i}} \quad \forall j \quad (3-3.1)$$

Formel 3-3: Greedy 2 nach Dobson

(Quelle: Kellerer/Pferschy/Pisinger, 2004:257)

Einen anderen, jedoch zu Dobson (1982) ähnlichen, Ansatz zur Ermittlung der „relativen“ Effizienz eines Gegenstandes j haben Senju und Toyoda (1968) entwickelt. Im Gegensatz zu Dobson (1982) werden die Nachfrageüberschüsse der einzelnen Restriktionen mit dem Gewicht von Gegenstand j unter der entsprechenden Restriktion multipliziert. Mathematisch ist dieser Zusammenhang in Formel 3–4 abgebildet. (Vgl. Senju und Toyoda, 1968:B200-B202)

$$e3_j := \frac{p_j}{\sum_{i=1}^d w_{ij} (\sum_{l=1}^n w_{il} - c_i)} \quad \forall j \quad (3-4.1)$$

Formel 3-4: Greedy 3 nach Senju und Toyoda

(Quelle: Senju und Toyoda, 1968:B201)

Eine allgemein gültige Darstellung mit entsprechendem Relevanzwert r_i einer Restriktion lässt sich durch Formel 3–5 zeigen. Je höher r_i ist, desto höher ist die Knappheit und damit die Wichtigkeit der Restriktion, wobei damit einhergeht, dass das Einpacken von „schweren“⁶ Gegenstände unattraktiv wird. (Vgl. Kellerer/Pferschy/Pisinger, 2004:257)

$$e_j := \frac{p_j}{\sum_{i=1}^d w_{ij} r_i} \quad \forall j \quad (3-5.1)$$

Formel 3-5: Greedy mit Einbezug des Relevanzwertes

(Quelle: Kellerer/Pferschy/Pisinger, 2004:257)

Demzufolge ist schnell zu erkennen, dass in Formel 3–3 sowie in Formel 3–4 ein Teil des Nenners durch r_i substituierbar ist.

⁶ unter den Begriff „schwer“ wird in diesem Zusammenhang ein hoher Restriktionswert verstanden. (Vgl. Kellerer/Pferschy/Pisinger, 2004:257)

Neuere Entwicklungen des Greedy-Verfahrens aktualisieren nicht nur vor dem Einpacken, sondern sogar nach jedem einzelnen Einpackvorgang den Relevanzwert, um möglicherweise eine noch bessere Optimierung bzw. einen noch besseren Zielfunktionswert zu erhalten. Die auf Toyota (1975) basierende Idee von Fox und Scudder (1985) ist in Formel 3–6 abgebildet. Sie definieren den Relevanzwert r_i als das Produkt aus dem Kehrwert der Kapazität von Restriktion i und der Summe des relativen skalierten Gesamtgewichts der bereits eingepackten Gegenstände von Restriktion i und soll ausdrücken, ob eine Restriktion i kritisch ist oder nicht. X_j beschreibt alle bereits in den Rucksack eingepackten Gegenstände. Nach jedem zusätzlich eingepackten Gegenstand kann sich der Relevanzwert entsprechend verändern, da die Menge X_j je Iteration möglicherweise um einen Gegenstand erweitert wird. (Vgl. Fox und Scudder, 1985:615-620; Toyota, 1975:1420-1425)

$$r_i := \frac{1}{c_i} \sum_{k \in X} \frac{w_{ik}}{c_i} \quad \forall i \quad (3-6.1)$$

Formel 3-6: Relevanzwert nach Fox und Scudder (1985)

(Quelle: Fox und Scudder, 1985:615-617)

Toyoda's (1975) eigene Entwicklungen, auf denen das Modell aus Formel 3–6 fußt, ist in Formel 3–7 abgebildet. Wichtig war ihm dabei zwischen kritischen und unkritischen Ressourcen zu unterscheiden. Restriktionen, die nahezu aufgebraucht sind, erhalten einen hohen Relevanzwert, jene die kaum genutzt wurden, erhalten einen niedrigen Relevanzwert. (Vgl. Toyoda, 1975:1420-1423)

$$r_i := \max \left\{ 0, \frac{1}{c_i} \left(\sum_{k \in X} \frac{w_{ik}}{c_i} - t_1 \left(\max_{l=1}^d \left\{ \sum_{k \in X} \frac{w_{lk}}{c_l} \right\} \right)^{t_2} \right) \right\} \quad \forall i \quad (3-7.1)$$

Formel 3-7: Relevanzwert nach Toyoda (1975)

(Quelle: Kellerer/Pferschy/Pisinger, 2004:259)

Der Relevanzwert entspricht bei niedriger Restriktionsnachfrage 0, bei hoher Restriktionsnachfrage wird ein entsprechender Term von Formel 3–6 subtrahiert. t_1 und t_2 sind Sensitivitätsparameter, die behutsam angepasst werden, um ein bestmögliches Resultat zu erzielen. (Vgl. Toyoda, 1975:1425-1426)

Obwohl die dynamische Idee von Fox und Scudder (1985) sowie Toyoda (1975) vielversprechend erscheint, führt sie nicht zwingend zu besseren Ergebnissen als der Greedy von Dobson (1982) oder von Senju und Toyoda (1968). Vielmehr handelt es sich wiederum um Heuristiken, die abhängig von den Testdaten einen höheren oder niedrigeren Zielfunktionswert erreichen.

Anstatt den Relevanzwert nach jedem Einpackvorgang zu aktualisieren, haben Loulou und Michaelides (1979) direkt den Effizienzwert und daraus resultierend die Sortierung der Gegenstände aktualisiert. Dazu beschrieben sie den Pönalwert v_j , der den potentiellen Effekt des Einpackens von Gegenstand j illustriert. Das Ergebnis von Loulou und Michaelides (1979) ist in Formel 3–8 dargestellt. q_{ij}^1 ist die bereits konsumierte Menge an Restriktion i . q_{ij}^2 ist die noch verbleibende Menge von Restriktion i für die weiteren Gegenstände. q_{ij}^3 ist der maximal mögliche zukünftige Verbrauch von Restriktion i . v_j ist das Maximum des Produkts aller drei q_{ij} über alle Restriktionen. (Vgl. Loulou und Michaelides, 1979:1102-1107)

$$e_{4j} := \frac{p_j}{v_j} \quad \forall j \quad (3-8.1)$$

$$v_j := \max_{i=1}^d \left\{ \frac{q_{ij}^1 q_{ij}^3}{c_i q_{ij}^2} \right\} \quad \forall j \quad (3-8.2)$$

$$q_{ij}^1 := \sum_{k \in X_{j-1}} w_{ik} + w_{ij} \quad \forall i, j \quad (3-8.3)$$

$$q_{ij}^2 := c_i - \left(\sum_{k \in X_{j-1}} w_{ik} + w_{ij} \right) \quad \forall i, j \quad (3-8.4)$$

$$q_{ij}^3 := \sum_{k=j+1}^n w_{ik} \quad \forall i, j \quad (3-8.5)$$

Formel 3-8: Greedy 4 nach Loulou und Michaelides (1979)

(Quelle: Loulou und Michaelides, 1979:1104)

Loulou und Michaelides (1979) bemerkten im Rahmen ihrer Forschung zwei wesentliche Verbesserungen ihres Effizienzwertes. Wird die Wurzel für q_{ij}^1 , q_{ij}^2 und für q_{ij}^3 berechnet, wird der Zielfunktionswert in Abhängigkeit der Testdaten erhöht. Weiters führt eine zusätzliche Prozedur, die alle Gegenstände j mit negativen q_{ij}^2 verbietet, möglicherweise zu einem verbesserten Resultat. (Vgl. Loulou und Michaelides, 1979:1106-1107)

Für die Heuristik von Loulou und Michaelides (1979) gilt, dass ein erhöhter Zielfunktionswert gegenüber den bisher beschriebenen Greedy-Verfahren nicht zwingend eintritt, sondern das Ergebnis wiederum von den Testdaten abhängt.

Relaxationsbasierte Heuristiken

Die Idee hinter relaxationsbasierten Heuristiken ist jene, dass bisher bekannte Verfahren (z.B. primaler und dualer Greedy, dynamische Programmierung, etc.) mittels Relaxationsverfahren erweitert werden, um zulässige Lösungen in überschaubarer Laufzeit zu erhalten. (Vgl. Bertsimas und Demir, 2002:550)

Bertsimas und Demir (2002) beispielsweise haben eine schnelle und einfache Heuristik für das Einpacken auf Basis der LP-Relaxation erarbeitet. Dabei startet das Verfahren ähnlich zur dynamischen Programmierung mit der Lösung von einzelnen Subproblemen, die schlussendlich iterativ zum finalen Resultat zusammengefügt werden. (Vgl. Bertsimas und Demir, 2002:551-553)

Magazine und Oguz (1984) verbanden die Lagrange Multiplikation mit dem dualen Greedy. Gegenstände werden anhand ihrer skalierten Nachfrage gereiht und mit dem Lagrange Multiplikator derart verändert, sodass sie entweder aus dem Rucksack entfernt werden oder im Rucksack bleiben. Sobald ein brauchbarer Schrankenwert erreicht ist, stoppt der Algorithmus. (Vgl. Magazine und Oguz, 1984:322-323)

Approximierte dynamische Programmierung

Bertsimas und Demir (2002) entwickelten, aufgrund des enorm hohen Rechen- und Zeitaufwandes der dynamischen Programmierung, ein Verfahren zur Annäherung an den optimalen Zielfunktionswert. Basierend auf Kapitel 2–1 verfolgen sie einen partiellen Branch-and-Bound Lösungsweg von der Wurzel bis zum Blatt und ermitteln für jeden Gegenstand j approximativ, ob er eingepackt wird oder nicht. (Vgl. Bertsimas und Demir, 2002:556-565)

Metaheuristiken

Die hohen Laufzeiten bei exakten Verfahren führt zur Anwendung einer Vielzahl an unterschiedlichen Metaheuristiken zur Lösung von mehrdimensionalen Rucksackproblemen. Eine Metaheuristik ist die Weiterentwicklung einer Heuristik, die nicht problemspezifisch, sondern vielmehr universell auf verschiedenste Probleme anwendbar ist und eine Heuristik gewissermaßen leitet. Dabei soll die Metaheuristik robust – unabhängig von der Testinstanz und Testgröße – verfahren und ein globales Optimum erzielen. Bekannte Vertreter der Metaheuristik sind der Tabu-Search, der Genetic-Algorithm und der Scatter-Search. Auch exotische Verfahren, wie beispielsweise der Ameisenalgorithmus, können beim mehrdimensionalen Rucksackproblem genutzt werden. (Vgl. Kellerer/Pferschy/Pisinger, 2004:268)

4. Social Choice Theory

Die Social Choice Theory beschäftigt sich im Allgemeinen mit der Zusammenführung von individuellen Präferenzen und Vorlieben zu einer kollektiven Entscheidung. Dabei behilft man sich im Entscheidungsprozess mit Abstimmverfahren oder Wahlen, die eine gewisse Stimmenmehrheit ausdrücken und eine Auswahl zwischen verschiedenen Alternativen zulassen. (Vgl. Kelly, 1988:1-3) Die Social Choice Theory ist heutzutage sehr eng mit der Wohlfahrtsökonomie verbunden, die Wohlfahrtsgewinne bzw. Wohlfahrtsverluste von unterschiedlichen Handlungsalternativen untersucht und die Auswirkungen auf eine Gruppe oder auch ein einzelnes Individuum darstellt, denn sobald mehrere Individuen an einer Abstimmung teilnehmen, kann nicht jeder Einzelne vollkommen zufriedengestellt sein. Demzufolge umfasst die Social Choice Theory zwei Forschungsschwerpunkte. Dies ist einerseits die Erforschung verschiedener Wahlverfahren mit all ihren Stärken und Schwächen und andererseits die Auswirkungen von kollektiven Entscheidungen auf das Individuum selbst. (Vgl. Suzumura, 2002:1-2)

Die uns heute bekannten Pioniere der Social Choice Theory waren zwei französische Gelehrte, Marie-Jean de Condorcet und Jean-Charles de Borda, die erstmals in Zeiten der französischen Revolution und dem Aufkommen von Menschenrechten soziale Entscheidungsprozesse untersuchten. Beide traten sehr mathematisch an das Problem heran und entwickelten daher auch sogenannte Scoring-Verfahren, die in weiterer Folge in Kapitel 4.2 detailliert beschrieben werden. Im 19. Jahrhundert widmeten sich einige weitere Forscher dieser Thematik, die Black (1958) in einem Sammelwerk bündelte und erstmals der Öffentlichkeit zugänglich machte. Darunter fiel beispielsweise die Entwicklung des Single-Vote, des Double-Vote, der Stichwahl und des Approval-Voting Verfahrens. (Vgl. Suzumura, 2002:3-6)

4.1. Social Choice Funktion und individuelle Präferenzen

Die individuellen Präferenzen und die damit einhergehenden Nutzenfunktionen sind zur Social Choice Funktion oftmals divergent. Durch kollektive Entscheidung besteht die Möglichkeit einer drastischen Entfernung eines Individuums von dessen Nutzenmaximum – siehe hierzu Arrow's (1963) Axiome. Um dem vorzubeugen, ist eine Abschätzung des Verhaltens der anderen Wahlbeteiligten vorteilhaft, damit Entscheidungen zum eigenen Gunsten beeinflusst werden können und kein Nachteil entsteht. (Vgl. Brams und Fishburn, 2002:179)

Um die Social Choice Funktion mathematisch darzustellen, sind vorab einige Voraussetzungen notwendig: Zu allererst muss von mehr als zwei Wahlbeteiligten mit unterschiedlichen aber transitiven sowie vollständigen Präferenzen ausgegangen werden, die auch als Voter Profile bezeichnet werden. (Vgl. Brams und Fishburn, 2002:180)

Zwei Inputfaktoren beeinflussen den Definitionsbereich der Social Choice Funktion wesentlich: Der erste Faktor ist ein nichtleeres Subset $A \subseteq X$ von Kandidaten – X ist die Menge aller möglichen Kandidaten – und der zweite Faktor ist das Set D_A – Rückmeldung von individuellen Präferenzen mittels Stimmenabgabe – bei dem der Wähler ausgehend von der zu Grunde liegenden Situation aus unterschiedlichen Präferenzprofilen $d = (d_1, d_2, \dots, d_n)$ auswählt. Somit ist gewährleistet, dass das Wahlergebnis von äußeren Einflüssen abhängt. Dieser Zusammenhang ist nochmals in Formel 4–1 dargelegt. (Vgl. Brams und Fishburn, 2002:180-181)

$$D = \bigcup_{A \subseteq X} \{(A, d) : d \in D_A\}$$

Formel 4-1: Definitionsbereich der Social Choice Funktion

(Quelle: Brams und Fishburn, 2002:181)

Dabei beschreibt nun der Definitionsbereich in Formel 4–1 das Set an teilnehmenden Kandidaten und deren dazugehörigen Stimmen. Darauf aufbauend ist in Formel 4–2 nun die endgültige Social Choice Funktion skizziert. Diese kann derart interpretiert werden, dass bei jeder Entscheidung eine bestimmte Teilmenge an Wotern aus der Gesamtmenge teilnimmt, die entsprechend ihrer Präferenzprofile ausgewählt werden. (Vgl. Brams und Fishburn, 2002:181)

$$F(A, d) \subseteq A$$

Formel 4-2: Social Choice Funktion: Mathematische Formulierung

(Quelle: Brams und Fishburn, 2002:181)

Speziell Arrow (1963) hat während seiner Forschung mehrere Axiome zu dieser Thematik erarbeitet, um alle bis dahin bekannten Social Choice Funktionen nach ethischen und methodischen Kriterien zu bewerten. Dabei hat er nachstehend fünf essentielle Axiome formuliert: (Vgl. Arrow, 1963:22-33)

- **Unbeschränkter Definitionsbereich**

Jeder Wahlbeteiligte muss die Freiheit haben sein eigenes Präferenzprofil gestalten zu können und die Social Choice Funktion ist als deterministisch charakterisiert. (Vgl. Arrow, 1963:22-26)

- **Unabhängigkeit gegenüber irrelevanter Alternativen**

Sind für die kollektive Entscheidung nur bestimmte Alternativen relevant, so werden auch nur diese bei allen Beteiligten berücksichtigt, die anderen werden nicht in Betracht gezogen. (Vgl. Arrow, 1963:26-28)

- **Transitivität**

Die Präferenzordnung muss transitiv sein. Ist Alternative A besser als B und Alternative B besser als C, so kann daraus schlussgefolgert werden, dass Alternative A besser als C ist. (Vgl. Arrow, 1963:28-30)

- **Keine Diktatur**

Es gibt keinen Diktator, der eine kollektive Präferenzordnung aufhebt, überstimmt oder außer Kraft setzt. (Vgl. Arrow, 1963:30-31)

- **Pareto Effizienz**

Kein einzelnes Individuum sollte durch eine Gruppenentscheidung ein geringeres Nutzenniveau als zuvor genießen. Haben beispielsweise alle Individuen Alternative A vor Alternative B geordnet, so sollte dies bei der kollektiven Ordnung ebenfalls so sein. Die Pareto Effizienz ist ein schwaches Axiom, denn durch gemeinsame Entscheidungen kann ein Individuum durchaus benachteiligt sein. (Vgl. Arrow, 1963:30-33)

Mit diesen Axiomen hat Arrow (1963) Regeln wie Single-Vote, Double-Vote, Borda-Vote, Approval-Voting, die Stichwahl und die Condorcet-Rule (auch bekannt unter paarweisem Vergleich) bewertet. Beispielsweise erfüllt die Condorcet-Rule trotz der Einfachheit nahezu alle Axiome – Transitivität wird nicht garantiert – bravourös. (Vgl. Arrow, 1963:35-37) Borda-Vote hingegen verletzt die Bedingung der Unabhängigkeit gegenüber irrelevanter Alternativen. Arrow (1963) zeigt dies an einem einfachen Beispiel: 11 Wahlbeteiligte können über drei unterschiedliche Handlungsalternativen A,B und C entscheiden. Fünf Wähler haben die Präferenzordnung ABC und sechs Wähler haben die Präferenzordnung BAC. Verteilt man nun entsprechend die Werte von 2 bis 0 für die einzelnen Alternativen ergibt sich folgendes Resultat in Tabelle 4–1. (Vgl. Arrow, 1963:39-40)

#	Ordnung	A	B	C
5	ABC	10	5	0
6	BAC	6	12	0
	Total	16	17	0

Tabelle 4-1: Borda-Vote nach Arrow's Axiome

(Quelle: Arrow, 1963:39-40)

Ändern nun die ersten fünf Voter ihre Präferenz zu ACB – lediglich A und B sind relevante Alternativen lt. Tabelle 4–1 – ergibt sich eine neue Ordnung in Tabelle 4–2, die das zweite Axiom verletzt und das Ergebnis verändert. (Vgl. Arrow, 1963:39-40)

#	Ordnung	A	B	C
5	ACB	10	0	5
6	BAC	6	12	0
	Total	16	12	5

Tabelle 4-2: Borda-Vote nach Arrow's verletztem zweiten Axiom

(Quelle: Arrow, 1963:39-40)

4.2. Wahlverfahren der Social Choice Theory

Die Forschungsentwicklung in der Social Choice Theory hat gezeigt, dass mittlerweile eine Vielzahl an unterschiedlichen Verfahren vorgestellt wurden, die jedoch alle einer ähnlichen Logik folgen. Die heute als „die Wichtigsten“ deklarierten Wahlmethoden sind Single-Vote, Double-Vote, Borda-Vote, die Condorcet-Rule, die Stichwahl und Approval-Vote, die nachstehend auch im Detail beschrieben werden. Neben den klassischen Wahlverfahren gibt es zusätzlich noch Mischformen von Verfahren wie beispielsweise das Borda-Approval-Vote. Für den späteren Zielfunktionswertevergleich im Rahmen des mehrdimensionalen Rucksackproblems sind nicht alle dieser Wahlmethoden von Relevanz. (Vgl. Kelly, 1988:1-2; Suzumura, 2002:1-3) Existieren nur zwei Alternativen ist die Wahl des Verfahrens nebensächlich, da alle zwingend zum gleichen Ergebnis führen müssen. Bei Gleichstand bzw. Indifferenz zwischen zwei oder mehreren Alternativen entscheidet ein Zusatzkriterium – beispielsweise ist dies beim Rucksackproblem der Profit p_j eines Gegenstandes – über den weiteren Wahlausgang. (Vgl. Laux und Liermann, 2005:88-89)

Single-Vote

Das Single-Vote Verfahren gehört zu den einfacheren Wahlverfahren. Hierbei wird lediglich die erste Präferenz eines jeden Individuums für die kollektive Entscheidung herangezogen. Jene Alternative mit dem höchsten Stimmenwert erhält den Zuschlag. Tabelle 4–3 enthält ein Beispiel mit fünf Individuen und drei Handlungsalternativen zur Illustration des Single-Vote Verfahrens. (Vgl. Laux und Liermann, 2005:87-88)

Im selbstgewählten Beispiel wird als kollektive Entscheidung Handlungsalternative B mit drei Stimmen ausgewählt, obwohl Individuum 1 und Individuum 4 dieses an letzter Stelle gereiht haben.

Individuum	Präferenz	A	B	C
Ind. 1	ACB	x		
Ind. 2	BAC		x	
Ind. 3	BAC		x	
Ind. 4	CAB			x
Ind. 5	BCA		x	
	Total	1	3	1

Tabelle 4-3: Beispiel zum Single-Vote Verfahren

(Quelle: eigene Darstellung, i.A.a. Laux und Liermann, 2005:89)

Double-Vote

Double-Vote ist als Weiterentwicklung zum Single-Vote zu verstehen, bei dem die ersten beiden Präferenzen eines jeden Individuums für die kollektive Entscheidung relevant sind. Wiederum gewinnt jene Alternative mit dem höchsten Stimmenwert. Tabelle 4–4 zeigt das Ergebnis des Double-Vote-Verfahrens beim selbstgewählten Beispiel. (Vgl. Thiele, 2008:379-381)

Anhand des Double-Vote Verfahrens haben sich die fünf Individuen auf Alternative A geeinigt, wobei es ein knapper Wahlsieg ist.

Individuum	Präferenz	A	B	C
Ind. 1	ACB	x		x
Ind. 2	BAC	x	x	
Ind. 3	BAC	x	x	
Ind. 4	CAB	x		x
Ind. 5	BCA		x	x
	Total	4	3	3

Tabelle 4-4: Beispiel zum Double-Vote-Verfahren

(Quelle: eigene Darstellung, i.A.a. Thiele, 2008:380)

Borda-Vote

Borda-Vote ist ein klassisches Scoring-Verfahren. Hierbei vergibt das Individuum anhand der Präferenzordnung den Alternativen entsprechend Punkte. Bei n Alternativen erhält die wichtigste Präferenz n Punkte, aller weiteren um einen weniger. Am Ende wird für jede Alternative die Gesamtsumme gebildet und jene Wahlmöglichkeit mit dem höchsten Scoring gewinnt die Abstimmung. Tabelle 4–5 zeigt das Ergebnis des Borda-Vote Verfahrens, bei dem Alternative B als Konsensentscheidung von den fünf Individuen ausgewählt wird. (Vgl. Brams und Taylor, 1998:209-212)

Individuum	Präferenz	A	B	C
Ind. 1	ACB	3	1	2
Ind. 2	BAC	2	3	1
Ind. 3	BAC	2	3	1
Ind. 4	CAB	2	1	3
Ind. 5	BCA	1	3	2
	Total	10	11	9

Tabelle 4-5: Beispiel zum Borda-Vote Verfahren

(Quelle: eigene Darstellung, i.A.a. Brams und Taylor, 1998:209-212)

Condorcet-Rule

Bei der Condorcet-Rule werden zwei Alternativen paarweise miteinander verglichen. Dabei wird lediglich darauf geachtet, welche Alternative öfters eine höhere Präferenzposition einnimmt. Mittlerweile sind zwei Arten der Condorcet-Rule bekannt. Die erste Variante vernachlässigt das Axiom der Transitivität und vergleicht alle Alternativen untereinander. Problematisch ist aber hierbei jener Fall, wenn A vor B, B vor C, aber C vor A liegt. Dementsprechend sind alle drei Alternativen gleichwertig. Bei der zweiten Variante wird auf die Erfüllung der Transitivität geachtet. Im obigen Fall liegt A vor B, wodurch B für weitere Vergleiche ausscheidet. Somit wird letztlich nur mehr A mit C verglichen, bei dem C gewinnt bzw. ausgewählt wird. Die Condorcet-Rule wird in Tabelle 4–6 anhand des Beispiels nochmals erläutert. (Vgl. Moulin, 2003:110-115)

Individuum	Präferenz	A vs. B		B vs. C		C vs. A	
		A	B	B	C	C	A
Ind. 1	ACB	1	0	0	1	0	1
Ind. 2	BAC	0	1	1	0	0	1
Ind. 3	BAC	0	1	1	0	0	1
Ind. 4	CAB	1	0	0	1	1	0
Ind. 5	BCA	0	1	1	0	1	0
	Total	2	3	3	2	2	3

Tabelle 4-6: Beispiel zur Condorcet-Rule

(Quelle: eigene Darstellung, i.A.a. Moulin, 2003:110-115)

Handlungsalternative B wird im selbstgewählten Beispiel bei beiden Varianten ausgewählt, wobei für die zweite Variante der Vergleich zwischen Alternative C und Alternative A nicht mehr relevant ist, da A bereits gegen B ausgeschieden ist.

Stichwahl

Die Stichwahl vereint Single-Vote als auch die Condorcet-Rule miteinander. Im ersten Schritt gibt jedes Individuum nur die beste Präferenz preis. Erhält eine Alternative die absolute Mehrheit (über 50% der Stimmen), so wird diese Alternative final ausgewählt. Liegt keine absolute Mehrheit vor, werden die zwei bestbewerteten Wahlmöglichkeiten der gemeinsamen Entscheidung paarweise verglichen. (Vgl. Rothe et. al, 2012:138-140)

Im selbstgewählten Beispiel in Tabelle 4–7 erzielt keine Alternative die absolute Mehrheit, vielmehr liegen A und B gleich auf und treten in einen paarweisen Vergleich, bei dem Alternative A schlussendlich gewinnt.

Individuum	Präferenz	A	B	C	A vs. B	
					A	B
Ind. 1	ACB	x			1	0
Ind. 2	ABC	x			1	0
Ind. 3	BAC		x		0	1
Ind. 4	CAB			x	1	0
Ind. 5	BCA		x		0	1
	Total	2	2	1	3	2

Tabelle 4-7: Beispiel zur Stichwahl

(Quelle: eigene Darstellung, i.A.a. Rothe et. al, 2012:138-140)

Approval-Vote

Approval-Vote ist ein Wahlverfahren, bei dem ein Individuum für eine bestimmte Teilmenge an Präferenzen die Zustimmung geben kann. Die Präferenzen müssen dabei nicht zwingend die Besten sein und nicht jedes Individuum muss für die gleichen Alternativen zustimmen, vielmehr wird jene Wahlmöglichkeit akzeptiert, bei der der größte Nutzenzuwachs erwartet wird. Jene Alternative wird kollektiv ausgewählt, die die meisten Zustimmung findet. Die Bewertung der Präferenzen erfolgt durch zwei Möglichkeiten: Entweder gibt ein Voter die Zustimmung (=1) oder der Voter gibt keine Zustimmung (=0) ab. (Vgl. Brams und Fishburn, 2007:4-6)

In Tabelle 4–8 ist das Ergebnis vom Approval-Vote dargelegt. Individuum 1 stimmt für Teilmenge an Alternative A und C, Individuum 2 stimmt nur für B, Individuum 3 stimmt für alle drei Alternativen, Individuum 4 stimmt für C und Individuum 5 stimmt für B und C. Die kollektive Entscheidung wird zu Gunsten von Alternative C gefällt, da hierfür die meiste Zustimmung abgegeben wird.

Individuum	Präferenz	A	B	C
Ind. 1	ACB	x		x
Ind. 2	BAC		x	
Ind. 3	BAC	x	x	x
Ind. 4	CAB			x
Ind. 5	BCA		x	x
	Total	2	3	4

Tabelle 4-8: Beispiel zum Approval-Vote

(Quelle: eigene Darstellung, i.A.a. Brams und Fishburn, 2007:4-6)

BordaApproval-Vote

BordaApproval ist ein Mischverfahren, bei dem das Individuum für eine bestimmte Teilmenge an Präferenzen die Zustimmung geben kann und dafür entsprechend das Borda-Scoring vergibt. Die Präferenzen müssen dabei nicht zwingend die Besten sein und nicht jedes Individuum muss für die gleichen Alternativen zustimmen, vielmehr wird jene Wahlmöglichkeit akzeptiert, bei der der größte Nutzenzuwachs erwartet wird. Jene Alternative mit dem höchsten Scoring gewinnt die Wahl.

In Tabelle 4–9 ist das Ergebnis vom BordaApproval-Vote dargelegt. Individuum 1 stimmt für die Teilmenge an Alternative A und C, Individuum 2 stimmt nur für B, Individuum 3 stimmt für alle drei Alternativen, Individuum 4 stimmt für C und Individuum 5 stimmt für B und C.

Individuum	Präferenz	A	B	C
		Score	Score	Score
Ind. 1	ACB	x 2		x 1
Ind. 2	BAC		x 1	
Ind. 3	BAC	x 2	x 3	x 1
Ind. 4	CAB			x 1
Ind. 5	BCA		x 2	x 1
	Total	4	6	4

Tabelle 4-9: Beispiel zum BordaApproval-Vote

(Quelle: eigene Darstellung, i.A.a. Brams und Fishburn, 2007:4-6)

Die kollektive Entscheidung wird zu Gunsten von Alternative B getroffen, da hierfür der höchste Score verteilt wurde.

4.3. Lösen von mehrdimensionalen Rucksackproblemen auf Basis der Social Choice Theory

Dieser Abschnitt klärt die Frage, wie die oben beschriebenen unterschiedlichen Forschungsgebiete miteinander verknüpft werden können und eine neue approximative Lösung für mehrdimensionale Rucksackprobleme bieten. Unabhängig von dem zu Grunde liegenden Wahlverfahren wird eine Menge an einzupackenden Gegenständen n mit entsprechenden Profiten p_j und Gewichten w_{ij} vorausgesetzt. Zusätzlich unterliegt jeder Gegenstand j einer Menge an Restriktionen d . Anhand dieser Basisdaten wurden bereits in Kapitel 3.3. Einpacklösungen mittels Greedy-Verfahren vorgestellt. Eine durch Effizienzwerte definierte Sortierung der Gegenstände wird iterativ, und solange Kapazität vorhanden ist, in einen Rucksack eingepackt. (Vgl. Neumann und Morlock, 2002:408-410)

Bei der Social Choice Theory werden nun die d Restriktionen jeweils auf d Individuen aufgeteilt. Ein Individuum verwaltet demzufolge eine Restriktion i und kann diese gemäß einem klassischen Rucksackproblem handhaben. Dazu wird im ersten Schritt für jeden Voter die Effizienz der n Gegenstände für die dazugehörige Restriktion i als Verhältnis von Profit zu Gewicht ermittelt und die Gegenstände absteigend nach dieser sortiert. (Vgl. Akcay/Li/Xu, 2007:19) Formel 4–3 stellt diesen Zusammenhang mathematisch dar und ist äquivalent zu Formel 2–2 bzw. Formel 2–3, wobei i der Index der Individuen ist.

$$e_{ij} = \frac{p_j}{w_{ij}} \quad \forall i \quad (4-3.1)$$

$$\frac{p_1}{w_{i1}} \geq \frac{p_2}{w_{i2}} \geq \dots \geq \frac{p_n}{w_{in}} \quad \forall i \quad (4-3.2)$$

$$p_1 \geq p_2 \geq \dots \geq p_n \quad (4-3.3)$$

Formel 4-3: Mathematische Formulierung zum Greedy-Algorithmus

(Quelle: Akcay/Li/Xu, 2007:19)

Auf Basis der d unterschiedlichen Sortierungen der n Gegenstände nach Effizienzwerten erfolgt nun im zweiten Schritt die Stimmenvergabe bzw. die Zuteilung von Zustimmungspunkten innerhalb der einzelnen Wahlverfahren. Im dritten Schritt werden die Gegenstände absteigend anhand der Zustimmungspunkte final für das Einpacken mit dem primalen Greedy sortiert. Haben zwei oder mehrere Gegenstände denselben Effizienzwert erfolgt eine zusätzliche absteigende Sortierung nach den Profiten, damit sichergestellt wird, dass der theoretisch höchste Zielfunktionswert erreicht wird.

Single-Vote

Da beim Single-Vote jedes Individuum nur eine Stimme abgeben darf, erhält der Gegenstand mit dem höchsten Effizienzwert einen Zustimmungspunkt. Haben alle Individuen ihre Präfe-

renz verteilt, werden die Zustimmungspunkte kumuliert und alle Gegenstände erneut absteigend nach Zustimmungspunkten sowie Profiten sortiert. Beispielsweise können bei fünf Individuen maximal fünf verschiedene Gegenstände einen Zustimmungspunkt erhalten, alle weiteren Gegenstände erhalten keine Zustimmung und werden lediglich durch deren Profit miteinander verglichen. Mit dieser Sortierung wird der Rucksack mittels primalen Greedy gepackt und der Zielfunktionswert errechnet.

Borda-Vote

Mit der Vorab-Sortierung aus Formel 4–3 werden die n Gegenstände mit einem Scoring bewertet. Jedes Individuum verteilt Punkte von n bis 1. Die Gegenstände werden final mit dem kumulierten Borda-Wert je Gegenstand und anschließend mit deren Profit sortiert und mittels primalen Greedy in den Rucksack gepackt.

Approval-Vote

Zur Ermittlung der Teilmenge an Zustimmungen wurde beim Approval-Vote eine Lösung mittels Greedy Verfahren gewählt. Dabei wurde mit der Sortierung aus Formel 4–3 ein primaler bzw. dualer Greedy durchlaufen, der die Anzahl an eingepackten Gegenständen je Restriktion i zurückgibt und die besten „Approvals“ definiert. Jeder „Approval“ erhält einen Zustimmungspunkt. Die nach den kumulierten Zustimmungspunkten sowie Profiten sortierten Gegenstände werden mit dem primalen Greedy eingepackt.

BordaApproval-Vote

Als Mischverfahren wird mittels primalen und dualen Greedy die Anzahl an „Approvals“ festgelegt, die anschließend einen passenden Borda-Wert erhalten. Absteigend nach dem Borda-Wert und Profit werden die Gegenstände mittels primalen Greedy eingepackt.

5. Vergleich von Greedy-Algorithmen und Social Choice Methoden für das mehrdimensionale Rucksackproblem

Wie schon in den vorangegangenen Kapiteln gezeigt, gibt es in der Literatur verschiedene Ansätze zur Messung der „Attraktivität“ eines Gegenstandes für das Einpacken in den Rucksack. Fraglich ist nun, welcher Effizienzwert den tatsächlich höchsten Zielfunktionswert generiert. Zur Gegenüberstellung werden deshalb die vier obig beschriebenen Effizienzwerte des Greedy-Verfahrens und vier Social Choice Methoden (Single-Vote, Borda-Vote, Approval-Vote und BordaApproval-Vote) herangezogen und anhand gleicher Testdaten miteinander verglichen und untereinander ausgewertet. Die Testdaten für die quantitative Analyse selbst stammen von Chu und Beasley (1998) und können auf J.E. Beasley's Operations Research Library nachgelesen werden.⁷ Insgesamt werden neun Dateien – mit der Bezeichnung „mknapcb1.txt“ bis „mknapcb9.txt“ – zu je 30 Testproblemen sowie eine Ergebnisdatei mit der Bezeichnung „mkcbres.txt“ bearbeitet. Diese neun Dateien unterscheiden sich sowohl hinsichtlich Profit-, Restriktions- und Kapazitätswerte als auch hinsichtlich einer Knappheitsrate des Rucksacks. Im Grunde haben sie aber stets denselben Aufbau mit je fünf, zehn und 30 Restriktionen, mit je 100, 250 und 500 Gegenständen und mit einem Alpha⁸ von je 0.25, 0.50 und 0.75. (Vgl. Chu und Beasley, 1998:78-80) Für die vier Greedy Verfahren, Single-Vote und Borda-Vote werden die Zielfunktionswerte mittels primalen Einpacken generiert, beim Approval-Vote und BordaApproval-Vote wurde sowohl das primale als auch das duale Einpacken durchgeführt. Grundsätzlich ist es für den Vergleich zwischen den Effizienzwerten zwar unwesentlich, welcher Greedy tatsächlich verwendet wird, jedoch ist für die Analyse mit den LP-Werten von Chu und Beasley's (1998) der primale Greedy notwendig und unumgänglich für plausible Ergebnisse. Die LP-Werte stellen im Allgemeinen die obere Schranke zum optimalen Zielfunktionswert dar. Je kleiner also die Abweichung eines Zielfunktionswertes zur LP-Lösung ist, desto wahrscheinlicher ist es, dass der Wert nahe am Optimum liegt.

Der erste Schritt der Untersuchung ist nun die Errechnung der Effizienzwerte sowie die entsprechende absteigende Sortierung der Gegenstände für jede der 270 Testinstanzen. Im zweiten Schritt werden die Zielfunktionswerte jedes Verfahrens anhand der Knappheitsrate mittels aggregierter Durchschnittsbetrachtung zusammengefasst und die prozentuale Abweichung zur LP-Lösung von Chu und Beasley (1998) gebildet. Zusätzlich wird innerhalb einer Verfahrensgruppe (Greedy Methoden und Social Choice Methoden) die Häufigkeit der besten Lösung je Methode und Testinstanz dargestellt als auch die prozentuale Abweichung zur besten Lösung ermittelt.

⁷ siehe dazu <http://people.brunel.ac.uk/~mastjjb/jeb/info.html>, abgerufen am 17.05.2012

⁸ Alpha ist der Knappheitsgrad der Rucksackkapazität. (Vgl. Chu und Beasley, 1998:79)

5.1. Vergleich der Greedy-Algorithmen

Für den bevorstehende Vergleich werden die vier nummerierten Greedy Algorithmen aus Kapitel 3.3. herangezogen. Vor allem bei Greedy 4 von Loulou und Michaelides (1979) sind programmtechnisch auch die beiden Möglichkeiten des „fine-tunings“ berücksichtigt worden, wobei die Ergebnisse mit der zusätzlichen Prozedur, die alle Gegenstände j mit negativen q_{ij}^2 verbietet, am vielversprechendsten sind und damit auch als Referenz dienen. Separiert nach der Anzahl an Gegenständen sind in den folgenden Tabellen die Zielfunktionswerte inkl. Abweichung zur LP-Lösung je Greedy abgebildet.

Greedy 1 nach Dobson (1982)

GREEDY1_DOBSON 1982 (n=100)									
m/α	0,25	0,5	0,75		m/α	0,25	0,5	0,75	
5	20.025,9	38.474,3	57.097,9	38.532,7	5	18,06%	11,45%	5,88%	11,79%
10	18.981,2	39.170,1	56.905,6	38.352,3	10	17,33%	8,91%	4,91%	10,38%
30	19.336,8	39.022,6	57.260,3	38.539,9	30	13,31%	7,08%	4,08%	8,15%
	19.448,0	38.889,0	57.087,9			16,23%	9,15%	4,95%	

GREEDY1 _ DOBSON 1982 (n=250)

m/α	0,25	0,5	0,75		m/α	0,25	0,5	0,75	
5	49.783,2	97.652,7	143.648,7	97.028,2	5	17,78%	10,75%	5,29%	11,27%
10	50.942,2	100.880,3	146.379,0	99.400,5	10	14,08%	7,43%	3,42%	8,31%
30	51.891,9	100.584,6	146.570,4	99.682,3	30	9,84%	6,20%	2,87%	6,30%
	50.872,4	99.705,9	145.532,7			13,90%	8,13%	3,86%	

GREEDY1 _ DOBSON 1982 (n=500)

m/α	0,25	0,5	0,75		m/α	0,25	0,5	0,75	
5	99.095,6	200.002,1	287.513,2	195.537,0	5	17,91%	8,92%	4,93%	10,59%
10	103.183,1	203.534,3	292.684,5	199.800,6	10	13,17%	6,42%	3,33%	7,64%
30	104.538,3	207.624,1	295.715,6	202.626,0	30	10,02%	4,20%	2,36%	5,53%
	102.272,3	203.720,2	291.971,1			13,70%	6,52%	3,54%	

Tabelle 5-1: Ergebnistabelle Greedy 1 nach Dobson (1982)

(Quelle: eigene Darstellung)

Greedy 2 nach Dobson (1982)

GREEDY2 _ DOBSON 1982 (n=100)

m/α	0,25	0,5	0,75	
5	22.627,4	42.017,6	59.622,7	41.422,6
10	20.904,2	41.269,8	57.990,0	40.054,7
30	19.735,7	39.487,1	57.712,0	38.978,3
	21.089,1	40.924,8	58.441,6	

m/α	0,25	0,5	0,75	
5	7,41%	3,30%	1,72%	4,14%
10	8,96%	4,03%	3,10%	5,36%
30	11,52%	5,97%	3,32%	6,94%
	9,30%	4,43%	2,71%	

GREEDY2 _ DOBSON 1982 (n=250)

m/α	0,25	0,5	0,75	
5	57.722,1	107.020,0	150.095,9	104.946,0
10	55.751,1	105.226,0	149.395,9	103.457,7
30	53.367,3	102.720,2	147.563,8	101.217,1
	55.613,5	104.988,7	149.018,5	

m/α	0,25	0,5	0,75	
5	4,67%	2,19%	1,04%	2,63%
10	5,97%	3,45%	1,43%	3,61%
30	7,27%	4,21%	2,21%	4,56%
	5,97%	3,28%	1,56%	

GREEDY2 _ DOBSON 1982 (n=500)

m/α	0,25	0,5	0,75		m/α	0,25	0,5	0,75	
5	116.950,1	215.150,3	299.448,7	210.516,4	5	3,12%	2,02%	0,99%	2,04%
10	114.813,9	212.345,4	298.999,6	208.719,6	10	3,38%	2,37%	1,25%	2,33%
30	109.776,4	210.540,9	298.234,7	206.184,0	30	5,52%	2,86%	1,53%	3,30%
	113.846,8	212.678,9	298.894,3			4,01%	2,42%	1,25%	

Tabelle 5-2: Ergebnistabelle Greedy 2 nach Dobson (1982)

(Quelle: eigene Darstellung)

Greedy 3 nach Senju und Toyoda (1968)

GREEDY3 _ SENJU&TOYODA 1968 (n=100)

m/α	0,25	0,5	0,75		m/α	0,25	0,5	0,75	
5	22.434,1	42.053,2	59.588,7	41.358,7	5	8,20%	3,21%	1,77%	4,40%
10	20.957,3	41.011,7	57.943,0	39.970,7	10	8,72%	4,63%	3,18%	5,51%
30	20.055,7	39.257,7	57.814,3	39.042,6	30	10,09%	6,52%	3,15%	6,58%
	21.149,0	40.774,2	58.448,7			9,00%	4,79%	2,70%	

GREEDY3 _ SENJU&TOYODA 1968 (n=250)

m/α	0,25	0,5	0,75		m/α	0,25	0,5	0,75	
5	57.965,6	106.659,6	149.749,6	104.791,6	5	4,26%	2,52%	1,27%	2,68%
10	55.775,5	104.828,2	149.117,8	103.240,5	10	5,93%	3,81%	1,61%	3,78%
30	53.612,1	102.622,6	147.625,0	101.286,6	30	6,85%	4,30%	2,17%	4,44%
	55.784,4	104.703,5	148.830,8			5,68%	3,54%	1,68%	

GREEDY3 _ SENJU&TOYODA 1968 (n=500)

m/α	0,25	0,5	0,75		m/α	0,25	0,5	0,75	
5	117.142,1	215.310,0	299.803,2	210.751,8	5	2,96%	1,95%	0,87%	1,93%
10	114.307,8	212.226,3	299.187,6	208.573,9	10	3,81%	2,43%	1,19%	2,47%
30	109.416,6	210.631,1	298.077,8	206.041,8	30	5,83%	2,81%	1,58%	3,41%
	113.622,2	212.722,5	299.022,9			4,20%	2,40%	1,21%	

Tabelle 5-3: Ergebnistabelle Greedy 3 nach Senju und Toyota (1968)

(Quelle: eigene Darstellung)

Greedy 4 nach Loulou und Michaelides (1979)

GREEDY4 _ LOULOU&MICHAELIDES 1979 - inkl. Prozedur für $q_2 < 0$ (n=100)

m/ α	0,25	0,5	0,75	
5	21.461,3	40.944,6	58.741,5	40.382,5
10	20.108,9	40.625,1	57.809,2	39.514,4
30	19.607,6	39.425,6	57.644,2	38.892,5
	20.392,6	40.331,8	58.065,0	

m/ α	0,25	0,5	0,75	
5	12,18%	5,77%	3,17%	7,04%
10	12,42%	5,52%	3,40%	7,11%
30	12,09%	6,12%	3,43%	7,22%
	12,23%	5,80%	3,33%	

GREEDY4 _ LOULOU&MICHAELIDES 1979 - inkl. Prozedur für $q_2 < 0$ (n=250)

m/ α	0,25	0,5	0,75	
5	57.692,1	107.000,3	150.094,3	104.928,9
10	55.724,4	105.149,8	149.428,3	103.434,2
30	53.402,3	102.866,1	147.725,1	101.331,2
	55.606,3	105.005,4	149.082,6	

m/ α	0,25	0,5	0,75	
5	4,72%	2,20%	1,04%	2,65%
10	6,01%	3,52%	1,41%	3,65%
30	7,21%	4,07%	2,11%	4,46%
	5,98%	3,26%	1,52%	

GREEDY4 _ LOULOU&MICHAELIDES 1979 - inkl. Prozedur für $q_2 < 0$ (n=500)

m/ α	0,25	0,5	0,75	
5	117.230,2	215.480,2	299.825,5	210.845,3
10	115.237,7	212.597,4	299.292,4	209.042,5
30	110.068,6	210.957,5	298.616,7	206.547,6
	114.178,8	213.011,7	299.244,9	

m/ α	0,25	0,5	0,75	
5	2,89%	1,87%	0,86%	1,88%
10	3,03%	2,26%	1,15%	2,14%
30	5,26%	2,66%	1,40%	3,11%
	3,73%	2,26%	1,14%	

Tabelle 5-4: Ergebnistabelle Greedy 4 nach Loulou und Michaelides (1979)

(Quelle: eigene Darstellung)

Interpretation der Ergebnisse

Wie schon in Kapitel 3 angedeutet, zeigen auch die vorliegenden Ergebnisse ganz klar, dass der „absolute“ Greedy 1 nach Dobson (1982) deutlich geringere Zielfunktionswerte aufweist als es bei den anderen der Fall ist. Begründet ist dies durch dominante Restriktionswerte, die das Resultat verfälschen. Weiters ist der Unterschied zwischen Greedy 2 nach Dobson (1982) und Greedy 3 nach Senju und Toyoda (1968) nur unwesentlich, abhängig von der jeweiligen Testinstanz erzielt Greedy 2 oder Greedy 3 einen höheren Zielfunktionswert. Greedy 4 nach Loulou und Michaelides (1979) erzielt mit steigender Variablenanzahl bessere Ergebnisse als Greedy 2 und Greedy 3, da bei einer höheren Gegenstandsanzahl mehr „ausgewogene“ – unter ausgewogen wird das Verhältnis von Profit zu Gewicht verstanden – Gegenstände anfänglich eingepackt werden, anstatt jene Gegenstände zu packen, die sowohl einen hohen Profit als auch ein hohes Gewicht besitzen.

Neben den Zielfunktionsmittelwerten ist zusätzlich die Varianz der einzelnen Testinstanzen von Interesse. Eine hohe Streuung der Abweichung zur LP-Lösung von Chu und Beasley (1998) führt zu einem hohen Varianzwert. Beispielsweise ist die Streuung beim Greedy 1 deutlich höher als bei den restlichen Greedy-Verfahren, was das bereits bestehende Bild von geringeren Zielfunktionswerten bei Greedy 1 bestätigt.

Auffallend sind in der Ergebnisstabelle auch die teilweise annähernd identen durchschnittlichen Zielfunktionswerte der Greedy-Verfahren, die auf identische Einpackvorgänge einzelner Testinstanzen schließen lassen.

Letztlich soll noch auf einen wichtigen Aspekt hingewiesen werden, dass bei gleichbleibender Variablenanzahl und steigender Restriktionsanzahl der Testdaten die prozentuale Abweichung der Zielfunktionswert bei Greedy 2-4 zur LP-Lösung von Chu und Beasley (1998) steigt. Der Grund hierfür liegt in der limitierten Möglichkeit Gegenstände einzupacken. Bleibt die Restriktionsanzahl unverändert, sinkt die prozentuale Abweichung zur LP-Lösung bei steigender Variablenanzahl bei allen Greedy, da die Auswahl zum Einpacken größer wird.

All diese Beobachtung sind nochmals mittels Häufigkeit des höchsten Zielfunktionswertes in Tabelle 5–5 skizziert. Jede Kombination von Variablenanzahl, Restriktionsanzahl und Knappheitsgrad besteht aus zehn Testinstanzen. Beispielsweise hat bei der Kombination 100 Variablen, fünf Restriktionen und Knappheitsgrad von 0.25 Greedy 1 nach Dobson (1982) ein Mal, Greedy 2 nach Dobson (1982) sechs Mal und Greedy 3 nach Senju und Toyoda (1968) vier Mal den höchsten Zielfunktionswert. Da die Summe der Häufigkeiten größer zehn ist, kann auf zwei ident hohe beste Zielfunktionswerte rückgeschlossen werden.

n	m	α	Greedy1	Greedy2	Greedy3	Greedy4
100	5	0,25	1	6	4	0
		0,50	0	6	7	0
		0,75	1	6	7	0
	10	0,25	0	4	7	1
		0,50	0	7	4	0
		0,75	1	4	4	2
	30	0,25	2	3	6	0
		0,50	1	6	3	2
		0,75	2	5	7	1
Summe			8	47	49	6
250	5	0,25	0	0	8	2
		0,50	0	2	5	3
		0,75	0	3	3	4
	10	0,25	0	2	4	4
		0,50	0	3	4	3
		0,75	0	4	1	5
	30	0,25	1	1	5	3
		0,50	1	2	3	4
		0,75	1	1	4	5
Summe			3	18	37	33
500	5	0,25	0	1	3	6
		0,50	0	0	4	6
		0,75	0	0	5	5
	10	0,25	0	1	1	9
		0,50	0	1	3	6
		0,75	0	0	4	6
	30	0,25	0	1	4	5
		0,50	0	1	3	6
		0,75	0	2	2	7
Summe			0	7	29	56

Tabelle 5-5: Ergebnistabelle Häufigkeit der besten Zielfunktion bei den Greedy Verfahren

(Quelle: eigene Darstellung)

5.2. Vergleich der Social Choice Methoden

Für den Vergleich wurden lediglich vier verschiedene Wahlverfahren, Single-Vote, Borda-Vote, Approval-Vote und BordaApproval-Vote, ausgewählt, deren Ergebnisse als Referenz auch für andere Methoden gelten. Obwohl beim Approval-Vote und beim BordaApproval-Vote der primale und duale Greedy für die Bestimmung der Anzahl an Zustimmung herangezogen wurde, sind in weiterer Folge nur die Resultate des primalen Greedy von Relevanz, da beim primalen Greedy die Anzahl an Zustimmung mindestens so hoch wie beim dualen Greedy ist und somit ein besseres Ergebnis wahrscheinlicher – aber nicht zwingend – ist. Separiert nach der Anzahl an Gegenständen sind in den folgenden Tabellen die Zielfunktionswerte inkl. Abweichung zur LP-Lösung je ausgewählter Social Choice Methode abgebildet.

Single-Vote

SINGLE VOTE (n=100)									
m/α	0,25	0,5	0,75		m/α	0,25	0,5	0,75	
5	20.038,6	38.327,4	57.256,2	38.540,7	5	18,00%	11,79%	5,62%	11,80%
10	18.558,8	39.027,8	56.991,7	38.192,8	10	19,17%	9,24%	4,77%	11,06%
30	17.327,6	37.307,0	56.725,5	37.120,0	30	22,32%	11,16%	4,97%	12,82%
	18.641,7	38.220,7	56.991,1			19,83%	10,73%	5,12%	

SINGLE VOTE (n=250)

m/α	0,25	0,5	0,75		m/α	0,25	0,5	0,75	
5	49.517,4	97.291,2	144.540,0	97.116,2	5	18,22%	11,08%	4,70%	11,33%
10	50.358,1	100.845,3	146.462,6	99.222,0	10	15,06%	7,47%	3,36%	8,63%
30	48.009,7	99.420,3	145.999,5	97.809,8	30	16,58%	7,28%	3,25%	9,04%
	49.295,1	99.185,6	145.667,4			16,62%	8,61%	3,77%	

SINGLE VOTE (n=500)

m/α	0,25	0,5	0,75		m/α	0,25	0,5	0,75	
5	99.254,1	200.063,6	287.488,5	195.602,1	5	17,78%	8,89%	4,94%	10,54%
10	102.555,9	203.852,7	292.064,6	199.491,1	10	13,70%	6,28%	3,54%	7,84%
30	102.449,7	206.491,0	295.639,0	201.526,6	30	11,82%	4,72%	2,38%	6,31%
	101.419,9	203.469,1	291.730,7			14,43%	6,63%	3,62%	

Tabelle 5-6: Ergebnistabelle des Single-Vote

(Quelle: eigene Darstellung)

Borda-Vote

BORDA VOTE (n=100)

m/α	0,25	0,5	0,75	
5	22.562,9	41.956,2	59.408,4	41.309,2
10	20.514,1	41.203,0	58.264,6	39.993,9
30	20.032,2	39.425,1	57.478,1	38.978,5
	21.036,4	40.861,4	58.383,7	

m/α	0,25	0,5	0,75	
5	7,67%	3,44%	2,07%	4,39%
10	10,65%	4,18%	2,64%	5,83%
30	10,19%	6,12%	3,71%	6,67%
	9,51%	4,58%	2,81%	

BORDA VOTE (n=250)

m/α	0,25	0,5	0,75	
5	57.998,6	106.633,7	149.365,6	104.666,0
10	55.872,7	105.098,5	149.316,1	103.429,1
30	53.233,8	102.640,5	147.625,5	101.166,6
	55.701,7	104.790,9	148.769,1	

m/α	0,25	0,5	0,75	
5	4,21%	2,54%	1,52%	2,76%
10	5,76%	3,56%	1,48%	3,60%
30	7,51%	4,28%	2,17%	4,65%
	5,83%	3,46%	1,73%	

BORDA VOTE (n=500)

m/α	0,25	0,5	0,75		m/α	0,25	0,5	0,75	
5	116.952,1	214.139,5	299.265,3	210.119,0	5	3,12%	2,48%	1,05%	2,22%
10	114.150,6	212.469,6	298.768,9	208.463,0	10	3,94%	2,31%	1,32%	2,53%
30	109.871,3	210.395,6	298.076,7	206.114,5	30	5,43%	2,92%	1,58%	3,31%
	113.658,0	212.334,9	298.703,6			4,17%	2,57%	1,32%	

Tabelle 5-7: Ergebnistabelle des Borda-Vote

(Quelle: eigene Darstellung)

Approval-Vote

APPROVAL VOTE GREEDY1 (n=100)

m/α	0,25	0,5	0,75		m/α	0,25	0,5	0,75	
5	21.882,5	40.883,7	58.297,6	40.354,6	5	10,46%	5,91%	3,90%	6,75%
10	20.055,2	40.360,5	57.614,5	39.343,4	10	12,65%	6,14%	3,73%	7,51%
30	19.531,7	39.442,3	57.774,5	38.916,2	30	12,43%	6,08%	3,21%	7,24%
	20.489,8	40.228,8	57.895,5			11,85%	6,04%	3,61%	

APPROVAL VOTE GREEDY1 (n=250)

m/α	0,25	0,5	0,75		m/α	0,25	0,5	0,75	
5	55.454,8	104.098,4	147.490,2	102.347,8	5	8,41%	4,86%	2,76%	5,34%
10	54.864,5	103.138,4	147.987,6	101.996,8	10	7,46%	5,36%	2,36%	5,06%
30	52.120,7	101.874,5	147.324,6	100.439,9	30	9,44%	4,99%	2,37%	5,60%
	54.146,7	103.037,1	147.600,8			8,44%	5,07%	2,50%	

APPROVAL VOTE GREEDY1 (n=500)

m/α	0,25	0,5	0,75		m/α	0,25	0,5	0,75	
5	111.713,2	210.979,5	295.150,0	205.947,6	5	7,46%	3,92%	2,41%	4,60%
10	111.480,7	209.428,4	297.155,2	206.021,4	10	6,19%	3,71%	1,86%	3,92%
30	107.226,3	209.478,0	297.394,1	204.699,5	30	7,71%	3,35%	1,80%	4,29%
	110.140,1	209.962,0	296.566,4			7,12%	3,66%	2,02%	

Tabelle 5-8: Ergebnistabelle des Approval-Vote

(Quelle: eigene Darstellung)

BordaApproval-Vote

BORDA APPROVAL VOTE GREEDY1 (n=100)

m/α	0,25	0,5	0,75	
5	20.797,2	40.711,1	59.263,3	40.257,2
10	19.621,6	40.440,2	57.955,8	39.339,2
30	19.075,4	39.639,3	57.586,8	38.767,2
	19.831,4	40.263,5	58.268,6	

m/α	0,25	0,5	0,75	
5	14,90%	6,30%	2,31%	7,84%
10	14,54%	5,95%	3,16%	7,88%
30	14,48%	5,61%	3,53%	7,87%
	14,64%	5,96%	3,00%	

BORDA APPROVAL VOTE GREEDY1 (n=250)

m/α	0,25	0,5	0,75	
5	53.149,9	104.147,5	149.457,7	102.251,7
10	53.300,8	103.669,4	148.552,5	101.840,9
30	51.915,5	102.728,4	147.482,6	100.708,8
	52.788,7	103.515,1	148.497,6	

m/α	0,25	0,5	0,75	
5	12,22%	4,81%	1,46%	6,16%
10	10,10%	4,87%	1,98%	5,65%
30	9,80%	4,20%	2,27%	5,42%
	10,71%	4,63%	1,90%	

BORDA APPROVAL VOTE GREEDY1 (n=500)

m/α	0,25	0,5	0,75		m/α	0,25	0,5	0,75	
5	107.704,0	210.565,2	298.232,5	205.500,6	5	10,78%	4,11%	1,39%	5,43%
10	108.225,0	208.613,7	297.959,0	204.932,6	10	8,93%	4,09%	1,59%	4,87%
30	105.797,3	208.337,2	296.610,7	203.581,7	30	8,94%	3,87%	2,06%	4,96%
	107.242,1	209.172,0	297.600,7			9,55%	4,02%	1,68%	

Tabelle 5-9: Ergebnistabelle des BordaApproval-Vote

(Quelle: eigene Darstellung)

Interpretation der Ergebnisse

Im Gegensatz zu den Greedy-Verfahren sind die Resultat bei den Social Choice Methoden eindeutiger zu interpretieren. Borda-Vote erzielt, bis auf einige Ausnahmen, sehr oft den höchsten Zielfunktionswert. Der Grund hierfür liegt in der vollständigen Berücksichtigung aller Gegenstände und der entsprechenden Bewertung der Präferenz. Beispielsweise lässt sich dies mit dem Approval-Vote oder dem BordaApproval-Vote einfach vergleichen, bei denen nur für zugestimmte Gegenstände eine Präferenzordnung existiert. Das Single-Vote hat durchweg und aufgrund einer einzelnen Präferenz den geringsten Zielfunktionswert. Zwischen Approval-Vote und BordaApproval gibt es keine eindeutige Tendenz, vielmehr unterscheiden sich die Ergebnisse der einzelnen Testinstanzen nur geringwertig bzw. sind ident.

Auffallend ist die hohe Streuung – ausgenommen vom Borda-Vote – bei den Social Choice Methoden, die sich mit der Unstimmigkeit der abstimmenden Individuen und der eingeschränkten Präferenzordnung erklären lässt. Vor allem Gegenstände, die zwischen Profit und Gewicht ausgewogen sind aber zu denen divergente Präferenzen bestehen, werden tendenziell später gereiht bzw. möglicherweise gar nicht eingepackt.

Steigt bei gleichbleibender Restriktionsanzahl die Variablenanzahl, so wird die prozentuale Abweichung zur LP-Lösung von Chu und Beasley (1998) bei allen Methoden geringer. Der Grund hierfür liegt in der größeren Auswahlmöglichkeit von einzupackenden Gegenstände. Bei gleichbleibender Variablenanzahl und steigender Restriktionsanzahl gibt es keine eindeutig beobachtbare Tendenz.

All diese Beobachtung sind nochmals mittels Häufigkeit des höchsten Zielfunktionswertes in Tabelle 5–10 skizziert. Jede Kombination von Variablenanzahl, Restriktionsanzahl und Knappheitsgrad besteht aus zehn Testinstanzen. Beispielsweise hat bei der Kombination von 100 Variablen, fünf Restriktionen und Knappheitsgrad von 0.25 Borda-Vota sieben Mal den höchsten Zielfunktionswert und Approval-Vote drei Mal den höchsten Zielfunktionswert.

n	m	α	Borda			
			Single	Borda	Approval	Appro.
100	5	0,25	0	7	3	0
		0,50	0	6	3	1
		0,75	0	6	0	5
	10	0,25	0	6	2	2
		0,50	0	6	2	2
		0,75	0	5	2	3
	30	0,25	0	8	1	1
		0,50	0	4	1	5
		0,75	0	2	6	2
Summe			0	50	20	21
250	5	0,25	0	10	0	0
		0,50	0	10	0	0
		0,75	0	7	0	4
	10	0,25	0	8	2	0
		0,50	0	8	0	2
		0,75	0	8	0	2
	30	0,25	0	8	2	0
		0,50	0	5	1	4
		0,75	0	4	5	1
Summe			0	68	10	13
500	5	0,25	0	10	0	0
		0,50	0	8	2	0
		0,75	0	9	0	1
	10	0,25	0	9	1	0
		0,50	0	8	1	1
		0,75	0	7	2	1
	30	0,25	0	9	1	0
		0,50	0	5	4	1
		0,75	0	8	2	0
Summe			0	73	13	4

Tabelle 5-10: Ergebnistabelle Häufigkeit der besten Zielfunktion bei den Social Choice Methoden

(Quelle: eigene Darstellung)

5.3. Vergleich der Greedy und Social Choice Methoden

n	m	α					Borda			
			Greedy1	Greedy2	Greedy3	Greedy4	Single	Borda	Approval	Approval
100	5	0,25	20.025,9	22.627,4	22.434,1	21.461,3	20.038,6	22.562,9	21.882,5	20.797,2
		0,50	38.474,3	42.017,6	42.053,2	40.944,6	38.327,4	41.956,2	40.883,7	40.711,1
		0,75	57.097,9	59.622,7	59.588,7	58.741,5	57.256,2	59.408,4	58.297,6	59.263,3
	10	0,25	18.981,2	20.904,2	20.957,3	20.108,9	18.558,8	20.514,1	20.055,2	19.621,6
		0,50	39.170,1	41.269,8	41.011,7	40.625,1	39.027,8	41.203,0	40.360,5	40.440,2
		0,75	56.905,6	57.990,0	57.943,0	57.809,2	56.991,7	58.264,6	57.614,5	57.955,8
	30	0,25	19.336,8	19.735,7	20.055,7	19.607,6	17.327,6	20.032,2	19.531,7	19.075,4
		0,50	39.022,6	39.487,1	39.257,7	39.425,6	37.307,0	39.425,1	39.442,3	39.639,3
		0,75	57.260,3	57.712,0	57.814,3	57.644,2	56.725,5	57.478,1	57.774,5	57.586,8
250	5	0,25	49.783,2	57.722,1	57.965,6	57.692,1	49.517,4	57.998,6	55.454,8	53.149,9
		0,50	97.652,7	107.020,0	106.659,6	107.000,3	97.291,2	106.633,7	104.098,4	104.147,5
		0,75	143.648,7	150.095,9	149.749,6	150.094,3	144.540,0	149.365,6	147.490,2	149.457,7
	10	0,25	50.942,2	55.751,1	55.775,5	55.724,4	50.358,1	55.872,7	54.864,5	53.300,8
		0,50	100.880,3	105.226,0	104.828,2	105.149,8	100.845,3	105.098,5	103.138,4	103.669,4
		0,75	146.379,0	149.395,9	149.117,8	149.428,3	146.462,6	149.316,1	147.987,6	148.552,5
	30	0,25	51.891,9	53.367,3	53.612,1	53.402,3	48.009,7	53.233,8	52.120,7	51.915,5
		0,50	100.584,6	102.720,2	102.622,6	102.866,1	99.420,3	102.640,5	101.874,5	102.728,4
		0,75	146.570,4	147.563,8	147.625,0	147.725,1	145.999,5	147.625,5	147.324,6	147.482,6
500	5	0,25	99.095,6	116.950,1	117.142,1	117.230,2	99.254,1	116.952,1	111.713,2	107.704,0
		0,50	200.002,1	215.150,3	215.310,0	215.480,2	200.063,6	214.139,5	210.979,5	210.565,2
		0,75	287.513,2	299.448,7	299.803,2	299.825,5	287.488,5	299.265,3	295.150,0	298.232,5
	10	0,25	103.183,1	114.813,9	114.307,8	115.237,7	102.555,9	114.150,6	111.480,7	108.225,0
		0,50	203.534,3	212.345,4	212.226,3	212.597,4	203.852,7	212.469,6	209.428,4	208.613,7
		0,75	292.684,5	298.999,6	299.187,6	299.292,4	292.064,6	298.768,9	297.155,2	297.959,0
	30	0,25	104.538,3	109.776,4	109.416,6	110.068,6	102.449,7	109.871,3	107.226,3	105.797,3
		0,50	207.624,1	210.540,9	210.631,1	210.957,5	206.491,0	210.395,6	209.478,0	208.337,2
		0,75	295.715,6	298.234,7	298.077,8	298.616,7	295.639,0	298.076,7	297.394,1	296.610,7

Tabelle 5-11: Zusammenfassung der Zielfunktionswerte aller Verfahren

(Quelle: eigene Darstellung)

n	m	α					Borda			
			Greedy1	Greedy2	Greedy3	Greedy4	Single	Borda	Approval	Approval
100	5	0,25	13,15%	2,13%	2,94%	7,05%	13,09%	2,33%	5,23%	9,93%
		0,50	9,25%	0,91%	0,81%	3,46%	9,59%	1,04%	3,60%	3,97%
		0,75	4,55%	0,33%	0,39%	1,80%	4,29%	0,69%	2,53%	0,92%
	10	0,25	10,19%	1,17%	0,89%	4,91%	12,24%	3,04%	5,15%	7,20%
		0,50	5,81%	0,77%	1,39%	2,32%	6,14%	0,95%	2,97%	2,74%
		0,75	2,80%	0,93%	1,01%	1,23%	2,65%	0,47%	1,58%	0,99%
	30	0,25	4,65%	2,71%	1,12%	3,34%	14,55%	1,26%	3,70%	5,93%
		0,50	2,38%	1,22%	1,79%	1,37%	6,65%	1,37%	1,32%	0,83%
		0,75	1,52%	0,75%	0,57%	0,86%	2,43%	1,15%	0,64%	0,96%
250	5	0,25	14,95%	1,46%	1,03%	1,50%	15,40%	0,98%	5,30%	9,29%
		0,50	9,12%	0,40%	0,75%	0,42%	9,45%	0,76%	3,12%	3,08%
		0,75	4,49%	0,21%	0,45%	0,21%	3,90%	0,70%	1,93%	0,63%
	10	0,25	9,57%	1,08%	1,03%	1,13%	10,58%	0,86%	2,63%	5,40%
		0,50	4,62%	0,51%	0,89%	0,59%	4,67%	0,64%	2,49%	1,99%
		0,75	2,18%	0,17%	0,35%	0,15%	2,13%	0,22%	1,11%	0,73%
	30	0,25	3,66%	0,94%	0,49%	0,88%	10,86%	1,20%	3,23%	3,62%
		0,50	2,84%	0,78%	0,88%	0,64%	3,96%	0,86%	1,60%	0,77%
		0,75	1,16%	0,48%	0,44%	0,37%	1,54%	0,44%	0,64%	0,54%
500	5	0,25	16,04%	0,92%	0,75%	0,68%	15,90%	0,92%	5,35%	8,76%
		0,50	7,44%	0,44%	0,37%	0,29%	7,42%	0,91%	2,36%	2,55%
		0,75	4,26%	0,29%	0,17%	0,16%	4,27%	0,35%	1,72%	0,69%
	10	0,25	10,62%	0,57%	1,01%	0,21%	11,17%	1,15%	3,44%	6,27%
		0,50	4,57%	0,44%	0,50%	0,33%	4,43%	0,39%	1,81%	2,19%
		0,75	2,38%	0,28%	0,21%	0,18%	2,59%	0,35%	0,89%	0,62%
	30	0,25	5,28%	0,54%	0,86%	0,27%	7,16%	0,45%	2,83%	4,13%
		0,50	1,75%	0,38%	0,33%	0,18%	2,29%	0,44%	0,88%	1,42%
		0,75	1,05%	0,20%	0,26%	0,08%	1,08%	0,26%	0,49%	0,75%

Tabelle 5-12: Zusammenfassung der prozentualen Abweichung zum besten Zielfunktionswert

(Quelle: eigene Darstellung)

n	m	α					Borda				
			Greedy1	Greedy2	Greedy3	Greedy4	Single	Borda	Approval	Approval	
100	5	0,25	1	3	2	0	0	3	1	0	
		0,50	0	2	5	0	0	2	1	1	
		0,75	0	6	6	0	0	2	0	0	
	10	0,25	0	3	6	1	0	2	0	0	
		0,50	0	4	0	0	0	4	1	1	
		0,75	0	1	2	1	0	4	1	2	
	30	0,25	2	2	4	0	0	6	0	0	
		0,50	0	2	1	1	0	3	0	5	
		0,75	0	2	3	1	0	0	4	2	
Summe			3	25	29	4	0	26	8	11	
250	5	0,25	0	0	4	1	0	5	0	0	
		0,50	0	0	3	3	0	4	0	0	
		0,75	0	2	3	4	0	1	0	0	
	10	0,25	0	2	3	1	0	3	1	0	
		0,50	0	1	2	2	0	3	0	2	
		0,75	0	3	1	4	0	2	0	0	
	30	0,25	0	0	3	3	0	3	1	0	
		0,50	1	0	1	0	0	4	0	4	
		0,75	0	0	2	3	0	2	2	1	
	Summe			1	8	22	21	0	27	4	7
	500	5	0,25	0	0	3	4	0	3	0	0
			0,50	0	0	2	6	0	1	1	0
			0,75	0	0	5	4	0	1	0	0
		10	0,25	0	0	0	8	0	1	1	0
			0,50	0	0	2	4	0	3	1	0
0,75			0	0	1	5	0	2	1	1	
30		0,25	0	1	2	2	0	5	0	0	
		0,50	0	0	2	3	0	2	3	0	
		0,75	0	1	1	5	0	2	1	0	
Summe			0	2	18	41	0	20	8	1	

Tabelle 5-13: Zusammenfassung der Häufigkeiten des besten Zielfunktionswertes

(Quelle: eigene Darstellung)

Letztlich wird in diesem Kapitel der bisher noch ausstehende übergreifende Methodenvergleich behandelt. Die in Tabelle 5–11 abgebildeten aggregierten Zielfunktionswerte und die in Tabelle 5–12 abgebildeten aggregierten prozentualen Abweichungen zum besten Zielfunktionswert zeigen recht deutlich, dass die Greedy-Verfahren im Allgemeinen bessere Ergebnisse erzielen als Verfahren der Social Choice Methode. Lediglich Borda-Vote und in Einzelfällen auch BordaApproval-Vote sind „konkurrenzfähig“, eine klare Tendenz hinsichtlich Variablen- bzw. Restriktionskombination ist aber nicht erkennbar.

Vergleicht man die beiden „schlechtesten“ Verfahren der Gruppen – Greedy 1 nach Dobson (1982) und Single-Vote – miteinander, ist beim Single-Vote die Abweichung zum besten Zielfunktionswert je Knappheitsgrad relativ konstant. Greedy 1 verbessert sich hingegen mit steigender Variablenanzahl deutlich.

Auffallend ist die Entwicklung bei Greedy 4 nach Loulou und Michaelides (1979), der bei steigender Variablenanzahl zunehmend den höchsten aggregierten Zielfunktionswert aufweist.

Tabelle 5–13 stellt die Häufigkeit des besten Zielfunktionswertes je Testinstanz dar. Im Wesentlichen sind die Ergebnisse von Greedy 3 nach Senju und Toyoda (1968) sowie Borda-Vote nahezu immer sehr hoch. Bei geringer Variablenanzahl ist Greedy 2 nach Dobson (1982) und bei hoher Variablenanzahl Greedy 4 nach Loulou und Michaelides (1979) entsprechend zu berücksichtigen. In der Praxis sind Greedy 2 und Greedy 4 nur bedingt brauchbar, da sie nur bei einer bestimmten Kombination überhaupt zielführend sind.

Die Laufzeit spielt im obigen Vergleich keine wesentliche Rolle, da alle behandelten Verfahren bei geeigneter Implementierung schnelle Ergebnisse liefern. Lediglich bei besonders zeitkritischen Anwendungen muss dieser Aspekt gesondert berücksichtigt werden.

6. Conclusio

Auf die Fragestellung in Kapitel 1, welches der vorgestellten Verfahren die besten Resultate unabhängig von den Testinstanzen zur Lösung von mehrdimensionalen Rucksackproblemen liefert, kann keine eindeutige Antwort gegeben werden. Abhängig von den Testinstanzen von Chu und Beasley (1998) maximieren unterschiedliche Greedy- oder Social Choice Methoden den Zielfunktionswert. Auffallend ist aber, dass die Greedy-Verfahren im Allgemeinen bessere Resultate als die Social Choice Methoden – mit Ausnahme des Borda-Vota – bieten. Grund hierfür ist einerseits die eingeschränkte Beachtung der Präferenzordnung und andererseits die Auswahl von nicht ausgewogenen – hoher Profit, aber auch hohes Gewicht – Gegenständen. Aufgrund der Divergenz der Ergebnisse fällt es schwer ein Universalverfahren für die Lösung von mehrdimensionalen Rucksackproblemen zu benennen. Muss man dies aber dennoch tun, ist meine Wahl Greedy 3 nach Senju und Toyoda (1968), da dieser Greedy unabhängig von den zu Grunde liegenden Testinstanzen in reduzierter Laufzeit gute bzw. die besten Resultate erzielt. Dicht gefolgt wird Greedy 3 vom Borda-Vota, denn durch die umfassende Beachtung aller Präferenzen der Individuen ist die Kollektiventscheidung entsprechend gut aber zeitintensiver.

Relativ problematisch ist der Umgang mit jenen Verfahren, die nur bei bestimmten Variablen- und Restriktionskombinationen gute Ergebnisse liefern. Greedy 2 nach Dobson (1982) oder auch Greedy 4 nach Loulou und Michaelides (1979) sind nur bei niedriger bzw. hoher Variablenanzahl wirklich brauchbar. Ist die Variablen- und Restriktionskombination aber vorab bekannt, können diese beiden Verfahren gezielt eingesetzt werden.

Für die Auswahl der passenden Methode zur Lösung eines mehrdimensionalen Rucksackproblems wurde bisher ein wesentlicher Aspekt nicht beachtet: die Laufzeit. Für jedes Verfahren gilt der Kompromiss aus Genauigkeit und Schnelligkeit. Ein exaktes Verfahren, welches mehrere Wochen Laufzeit hat, ist genauso unbrauchbar wie ein Greedy, der sofort ein stark verfälschtes Ergebnis liefert. Weder das eine, noch das andere Verfahren wird in der Praxis Anwendung finden, denn beides führt im Endeffekt zur Erhöhung der Kosten. Im vorliegenden Methodenvergleich ist die Laufzeit nicht relevant, da die behandelten Verfahren durchwegs schnell sind.

Literaturverzeichnis

- Akcay, Y./Li, H./Xu, S.H. (2007): *Greedy algorithm for the general multidimensional knapsack problem*. Annals of Operations Research, Vol. 150, No. 1, S. 17-29.
- Arrow, K.J. (1963): *Social Choice and Individual Values*. 2nd Edition, New York: Wiley.
- Bellman, R.E. (1957): *Dynamic Programming*. Princeton University Press.
- Bertsimas, D./Demir, R. (2002): *An approximate dynamic programming approach to multidimensional knapsack problems*. Management Science, Vol. 48, No.4, S. 550-565.
- Brams, S.J./Taylor, A.D. (1998): *Fair division: From cake-cutting to dispute resolution*. Cambridge: University Press.
- Brams, S.J./Fishburn, P.C. (2002): *Voting Procedures*. in: Arrow, K.J./Amartya, K.S./Suzumura, K. (2002): *Handbook of Social Choice and Welfare – Volume 1*. Amsterdam: Elsevier.
- Brams, S.J./Fishburn, P.C. (2007): *Approval Voting*. 2nd Edition, New York: Springer.
- Chu, P.C./Beasley, J.E. (1998): *A Genetic Algorithm for the Multidimensional Knapsack Problem*. Journal of Heuristics, Vol. 4, No. 1, S. 63-86.
- Dobson, G. (1982): *Worst-Case Analysis of Greedy Heuristics for Integer Programming with Nonnegative Data*. Mathematics of Operations Research, Vol. 7, No. 4, S. 515-531.
- Domschke, W./Drexl, A. (2007): *Einführung in Operations-Research*. 7. überarb. Aufl., Berlin: Springer.
- Fox, G.E./Scudder, G.D. (1985): *A heuristic with tie breaking for certain 0-1 integer programming models*. Naval Research Logistics Quarterly, Vol. 32, No. 1, S. 613-623.
- Freville, A. (2004): *The multidimensional 0-1 knapsack problem: An overview*. European Journal of Operations Research, Vol. 155, No. 1, S. 1-21.
- Garvish, B./Pirkul, H. (1985): *Efficient Algorithms for solving multiconstraint 0-1 knapsack problems to optimality*. Mathematical Programming, Vol. 31, No. 1, S. 78-105.
- Kelly, J.S. (1988): *Social Choice Theory: An Introduction*. Berlin: Springer.
- Laux, H./Liermann, F. (2005): *Grundlagen der Organisation: Die Steuerung von Entscheidungen als Grundproblem der Betriebswirtschaftslehre*. 6. Aufl., Berlin: Springer.

- Loulou, R./Michaelides, E. (1979): *New Greedy-like Heuristics for the Multidimensional 0-1 Knapsack Problem*. Operations Research, Vol. 27, No. 6, S. 1101-1114.
- Magazine, M.J./Oguz, O. (1984): *A heuristic algorithm for the multidimensional zero-one knapsack problem*. European Journal of Operational Research, Vol. 16, No. 3, S. 319-326.
- Martello, S./Toth, P. (1990): *Knapsack problems: algorithms and computer implementations*. UK: John Wiley and Sons Ltd.
- Matousek, J./Gärtner, B. (2007): *Understanding and using linear programming*. Berlin: Springer.
- Moulin, H.J. (2003): *Fair division and collective welfare*. MIT: University Press.
- Neumann, K./Morlock, M. (2002): *Operations Research*. 2. Aufl., München: Carl Hanser Verlag.
- Kellerer, H./Pferschy, U./Pisinger, D. (2004): *Knapsack Problems*. Berlin: Springer.
- Rothe, J./Baumeister, D./Lindner, C./Rothe, I. (2012): *Einführung in Computational Social Choice*. Heidelberg: Spektrum Verlag.
- Senju, S./Toyoda, Y. (1968): *An approach to linear programming with 0-1 variables*. Management Science, Vol. 15, No. 4, S. B196-B207.
- Suzumura, K. (2002): *Introduction*. in: Arrow, K.J./Amartya, K.S./Suzumura, K. (2002): *Handbook of Social Choice and Welfare – Volume 1*. Amsterdam: Elsevier.
- Thiele, C. (2008): *Regeln und Verfahren der Entscheidungsfindung innerhalb von Staaten und Staatenverbindungen*. Berlin: Springer.
- Toyoda, Y. (1975): *A simplified Algorithm for obtaining approximate solutions to zero-one programming problems*. Management Science, Vol. 21, No. 12, S. 1417-1427.

Anhang: Digitales Informationsmaterial

1. Excel- und Text-Dateien:

- Testinstanzen: mknapcb1.txt – mknapcb9.txt
- LP-Lösung von Chu und Beasley: mkcbres.txt
- Zielfunktionswerte aller Methoden: Rohdaten_Zielfunktionswerte_GESAMT.xlsx
- Aggregierte Zielfunktionswerte je Knappheitsgrad aller Methoden: Auswertung_Zielfunktionswert_GESAMT.xlsx
- Aggregierte prozentuale Abweichung je Knappheitsgrad aller Methoden: Auswertung_prozAbweichung_GESAMT.xlsx
- Aggregierte Tabellen lt. Kapitel 5: Auswertungen_konsolidiert_GESAMT.xlsx

2. Programmcode (Matlab)