

METHODEN ZUR BERECHNUNG DES NUKLEOLUS KOOPERATIVER SPIELE MIT EINER ANWENDUNG FÜR DIE SCHWEIZ

Dissertation

zur Erlangung der Würde
eines Doktors der Wirtschafts- und Sozialwissenschaften

vorgelegt
der Wirtschafts- und Sozialwissenschaftlichen Fakultät
der Universität Freiburg (Schweiz)

von
Yavuz Karagök
aus der Schweiz und der Türkei

Genehmigt von der Wirtschafts- und Sozialwissenschaftlichen Fakultät
am 31. Januar 2006 auf Antrag von
Herrn Prof. Dr. Reiner Wolff (erster Referent) und
Herrn Prof. Dr. Dr. h. c. mult. Wolfgang Eichhorn (zweiter Referent)

Freiburg 2006

„Mit der Annahme einer Dissertation beabsichtigt die Wirtschafts- und Sozialwissenschaftliche Fakultät der Universität Freiburg nicht, zu den darin enthaltenen wissenschaftlichen Meinungen des Verfassers Stellung zu nehmen.“ (Fakultätsbeschluss vom 23. Januar 1990)

Meinen Eltern

Vorwort

Die vorliegende Doktorarbeit ist während meiner Tätigkeit als diplomierter Assistent am Lehrstuhl für Mikroökonomie und Empirische Wirtschaftsforschung der Universität Freiburg (Schweiz) entstanden. An dieser Stelle möchte ich gerne allen Personen danken, die mich bei der Entstehung dieser Arbeit unterstützt haben.

Ein besonderer Dank gilt meinem Doktorvater, Herrn Prof. Dr. Reiner Wolff, für seine wertvolle fachkundige Begleitung und Betreuung. Die wissenschaftlichen Diskussionen mit ihm und seine Anregungen haben die Entstehung meiner Dissertation entscheidend mitgeprägt. Ebenfalls möchte ich Herrn Prof. Dr. Dr. h. c. mult. Wolfgang Eichhorn für die Übernahme des Koreferats und für die hilfreichen Ratschläge danken. Zudem danke ich Herrn Prof. Dr. Heinz Gröflin, der als Mitglied der Examenkommission mitgewirkt hat.

Herzlich danken möchte ich auch meiner grossartigen Bürokollegin Dr. Andrea Graber, mit der stets ein anregender Austausch stattfand. Den Mitgliedern des Departements für Quantitative Wirtschaftsforschung drücke ich für die Bereitschaft zu wissenschaftlichen Diskussionen im Forschungsseminar meinen Dank aus.

Grossen Dank möchte ich meiner Frau Anina aussprechen, die mich während meiner Dissertation ständig liebevoll unterstützt hat und die Doktorarbeit mehrmals unermüdlich gelesen hat. Ebenso danke ich meinen Schwiegereltern Renata und Theodor Martig für ihre wertvolle Unterstützung, insbesondere Theodor für das sorgfältige Korrekturlesen meiner Dissertation.

Besonderen Dank möchte ich an meine Eltern und Geschwister richten. Sie haben mich bei meinen Ausbildungen stets ermutigt und motiviert. Vor allem danke ich meinem Bruder Aydin, der mich während meines Studiums der Wirtschaftswissenschaften finanziell unterstützt hat und somit die Entstehung dieser Arbeit ermöglicht hat.

Bern, im Februar 2006

Yavuz Karagök

Inhaltsverzeichnis

1	Eine elementare Systematik strategischer Spiele	1
2	Kooperative Spiele mit zwei oder mehr Spielern	15
2.1	Grundlagen	15
2.2	Der Kern eines kooperativen Spiels	26
2.3	Balancierte Koalitionssysteme und balancierte Spiele	36
2.4	Der Nukleolus: Ein Konzept zur Lösung kooperativer Spiele	43
2.5	Eigenschaften des Nukleolus	56
2.6	Beweis der Existenz und der Eindeutigkeit des Nukleolus	65
2.7	Zusammenfassung	71
3	Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele	75
3.1	Methoden der ersten Generation	75
3.1.1	Das Verfahren von Kohlberg	75
3.1.2	Das Verfahren von Owen	78
3.2	Methoden der zweiten Generation	81
3.2.1	Das Verfahren von Bruyneel	81
3.2.2	Das Verfahren von Dragan	95
3.3	Methoden der dritten Generation	108
3.3.1	Das Verfahren von Maschler et al.	108
3.3.2	Das Verfahren von Sankaran	119
3.3.3	Das Verfahren von Potters et al.	130
3.3.4	Das Verfahren von Derks/Kuipers	144
3.4	Zusammenfassung	167

4	Ausgewählte spezielle kooperative Spiele	173
4.1	Abstimmungsspiele: Das Verfahren von Wolsey	173
4.2	Zuordnungsspiele und verwandte kooperative Spiele	185
4.3	Zusammenhängende balancierte kooperative Spiele	194
4.4	Bilaterale Marktspele mit komplementären Gütern	196
4.5	Peer-Group- und Veto-Rich-Spiele	197
4.6	Kooperative Spiele über kostenminimale spannende Bäume	202
4.7	Zusammenfassung	208
5	Implementation zweier Verfahren	211
5.1	Wahl der Verfahren	211
5.2	Implementation des Algorithmus von Potters et al.	214
5.3	Implementation des Algorithmus von Wolsey	225
5.4	Vergleich der implementierten Verfahren	239
5.5	Zusammenfassung	249
6	Anwendungsbeispiel: Die Zusammensetzung des schweizerischen Bundesrates	251
6.1	Streit um die Zauberformel	251
6.2	Schweizerische Demokratie als Konsensdemokratie	254
6.3	Spieltheoretische Betrachtung der Eidgenössischen Räte	256
6.4	Der Nukleolus des Anwendungsbeispiels	261
6.5	Zusammenfassung	274
7	Zusammenfassung	277
A	Lineare Optimierung	283
A.1	Grundlagen	283
A.2	Die Simplex-Methode	287
A.3	Die Revidierte Simplex-Methode	292
A.4	Rucksackprobleme	295

A.5 Berechnung der K -besten Lösung eines binären linearen Optimierungsproblems	299
B Minimierung des Maximums linearer Funktionen	303
C Umwandlung eines Sattelpunktproblems in eine lineare Optimierungsaufgabe	305
D Lineare Gleichungs- und Ungleichungssysteme	307
E Einige Begriffe der Graphentheorie	309
F Komplexität von Algorithmen	313
G Aufbau der beigelegten Software	315
H Testspiele zum Vergleich der implementierten Verfahren	317
Literaturverzeichnis	324

Abbildungsverzeichnis

1.1	Eine einfache Systematik strategischer Spiele	9
2.1	Kern des Einführungsbeispiels	30
3.1	Nukleolus des Einführungsbeispiels	117
4.1	Ein Peer-Group-Spiel	198
4.2	Graph eines Veto-Rich-Spiels	201
4.3	Graph eines kostenminimalen spannenden Baumes	204
4.4	Graph eines Standardbaum-Spiels	205
5.1	Ablaufdiagramm des Verfahrens von Potters et al. [105]	215
5.2	Ablaufdiagramm des Verfahrens von Wolsey [150]	227
5.3	Anzahl der Nebenbedingungen	244
5.4	Durchschnittlicher Zeitaufwand der Verfahren (kleine Spiele)	246
5.5	Durchschnittlicher Zeitaufwand der Verfahren (grosse Spiele)	246
A.1	Simplex-Tableau	289
E.1	Ein ungerichteter Graph	310

Tabellenverzeichnis

1.1	Auszahlungsmatrix des Gefangenendilemmas	3
1.2	Die Kosten- und die Einsparungsdaten des Gebührenspiels	5
2.1	Koalitionswerte des Einführungsbeispiels	16
2.2	(0 – 1)-Normierung des Einführungsbeispiels	21
2.3	Modifizierung des Einführungsbeispiels	30
2.4	Geordnete Überschüsse im Einführungsbeispiel für $\mathbf{x} = (9, 14, 19)$. .	45
2.5	Ein Beispiel zur Grundidee des Nukleolus	49
3.1	Koeffizientenmatrix	134
3.2	Nebenbedingungsmatrix des LP' für das Einführungsbeispiel	138
3.3	Tabellarischer Überblick über die Komplexität der Algorithmen . . .	170
4.1	Ein Zuordnungsspiel	188
4.2	Charakteristische Funktion eines Nachbarspiels	194
5.1	Vergleich der Rechenzeiten \bar{t}_{DK} und \bar{t}_{PRA}	212
5.2	Gegenüberstellung der implementierten Verfahren	213
5.3	Simplex-Tableau des linearen Programms LP'	216
5.4	Simplex-Tableau für die Implementation	218
5.5	Variable t wird Basisvariable (Iteration 1)	219
5.6	Aktuelles Simplex-Tableau nach $ps = 3, pz = 5$	219
5.7	Optimales Simplex-Tableau der ersten Iteration	220
5.8	Simplex-Tableau am Ende der ersten Iteration	221
5.9	Simplex-Tableau nach Schritt 2 der ersten Iteration	221
5.10	Simplex-Tableau mit der neuen Spalte t	221

Tabellenverzeichnis

5.11	Variable t wird Basisvariable (Iteration 2)	222
5.12	Anzahl linearer Programme	241
5.13	Anzahl der Nebenbedingungen	243
5.14	Rechenzeit in Sekunden	245
6.1	Länder mit Mehrheits- und Konsensdemokratie	255
6.2	Wahlresultate und Sitzverteilung der Parteien 2003	258
6.3	Zusammensetzung und Sitzverteilung der Fraktionen 2003	260
6.4	Anzahl der Mandate der Parteien zw. 1987-1999	261
6.5	Nukleolus des Spiels Γ_{NR} 2003	263
6.6	Wähleranteile und der Nukleolus des Spiels Γ_{NR} zw. 1987-1999	264
6.7	Nukleolus des Abstimmungsspiels Γ_{VBV}	265
6.8	Verteilung der Bundesratssitze nach Parteien	266
6.9	Theoretische Verteilung der Bundesratssitze nach Parteien	268
6.10	Nukleolus der Abstimmungsspiele für VBV zw. 1987-2003	269
6.11	Sitzverteilung der Parteien im Bundesrat gemäss dem Nukleolus	270
6.12	Nukleolus des Abstimmungsspiels Γ_{F} für 2003	271
6.13	Verteilung der Bundesratssitze nach Fraktionen	271
6.14	Theoretische Verteilung der Bundesratssitze nach Fraktionen	272
6.15	Verteilung der Bundesgerichtssitze nach Parteien für 2003	273
A.1	Alle Werte von $f_n(c)$	298
A.2	Optimale Werte von IP^i für die zweitbeste Lösung	302
D.1	Lösbarkeit linearer Gleichungssysteme	308

1 Eine elementare Systematik strategischer Spiele

Im sozialen und wirtschaftlichen Leben sind die einzelnen Akteure, beispielsweise Individuen, Haushalte oder Unternehmen, mit vielfältigen Entscheidungsproblemen konfrontiert. Dabei ist das Ergebnis einer Entscheidung in der Regel nicht unabhängig vom Verhalten der anderen beteiligten Akteure. Somit kann die Entscheidungsfindung eines Akteurs mit Interessenkonflikten bzw. Koordinationsproblemen verbunden sein. Dieses Problem der strategischen Interdependenz zwischen den Zielen einzelner Akteure führt zu sog. *strategischen Entscheidungssituationen*. Die Analyse dieser Situationen bildet den Gegenstand der sog. *Theorie der strategischen (ökonomischen) Spiele* (im Weiteren kurz: *Spieltheorie*).¹ Eine strategische Entscheidungssituation, die den Akteuren Entscheidungen abverlangt, wird als *Spiel* bezeichnet.² In der Spieltheorie werden die einzelnen Akteure als *Spieler* und die ihnen im Entscheidungsprozess zur Auswahl stehenden Handlungsalternativen als *Strategien* der Spieler bezeichnet. Hierbei wird angenommen, dass jede Strategie eines Spielers in Abhängigkeit von den jeweils gewählten Strategien der übrigen Spieler mit einem Ergebnis verbunden ist. Dieses Ergebnis nennt man *Auszahlung* des Spiels an den Spieler. Wir werden kürzer von der Auszahlung des Spielers sprechen. Die Auszahlung kann im weitesten Sinne als der mit der Strategie verbundene Nutzen interpretiert werden. Es wird angenommen, dass sich jeder Spieler für diejenige Alternative entscheidet, die ihm bei den gegebenen Strategien der übrigen Spieler einen möglichst hohen Nutzen stiftet. In diesem Zusammenhang spricht man von sog. *rationalen* Spielern.

J. von Neumann und O. Morgenstern schufen im Jahre 1944 mit ihrem Werk „*Theory of Games and Economic Behavior*“ [143] den Grundstein der Spieltheorie.

¹Vgl. Holler/Illing [53], S. 1.

²Vgl. Rosenmüller [112], S. 178.

1 Eine elementare Systematik strategischer Spiele

Seither findet eine breite Entwicklung im Bereich der Spieltheorie statt. Im Zuge dieser Entwicklung wurden die strategischen Spiele in zwei Hauptgruppen unterteilt (vgl. Abbildung 1.1): *Nicht-kooperative Spiele* und *kooperative Spiele*. Die Unterscheidung zwischen nicht-kooperativen und kooperativen Spielen findet man bei von Neumann/Morgenstern [143] noch nicht.³ Sie wurde erst von Nash [88] in die Literatur eingeführt.

Nicht-kooperative Spiele sind dadurch gekennzeichnet, dass die Spieler keine Möglichkeit haben, miteinander bindende Vereinbarungen abzuschliessen. Zum Beispiel ist die in der Literatur als *Gefangenendilemma* bekannte Spielsituation ein nicht-kooperatives Spiel. Die Ausgangslage dieses Spiels lässt sich wie folgt beschreiben:⁴ Zwei Gefangene, die gemeinsam ein Verbrechen begangen haben, das man ihnen jedoch nicht nachweisen kann, werden in separaten Zellen verhört. Jeder Gefangene hat zwei Optionen:

1. Er kann das Verbrechen gestehen und damit den anderen verraten.
2. Er kann seine Teilnahme am Verbrechen leugnen.

Wenn beide die Tat leugnen, werden sie beide wegen nachweisbarer minderer Delikte zu einer Gefängnisstrafe von einem Monat verurteilt. Gestehen beide ihre Tat, so erhalten sie eine mittlere Strafe von drei Monaten Haft. Gesteht nur einer, so wird er Kronzeuge und wird deshalb entlassen. In diesem Fall erhält der andere die Höchststrafe von sechs Monaten Haft. In der Tabelle 1.1 ist diese Spielsituation in einer sog. Auszahlungsmatrix des Spiels wiedergegeben. In der Auszahlungsmatrix sind die Gefangenen die Spieler des Spiels, die Gefängnisstrafen die Auszahlungen und das Gestehen bzw. das Leugnen des Verbrechens die möglichen Strategien der Spieler. Leugnet nun der Gefangene 1 die Tat, so stellt sich der Gefangene 2 besser, wenn er das Verbrechen gesteht. Die Strategie „gestehen“ ist aber auch die beste Strategie des Gefangenen 2, falls der Gefangene 1 die Tat gesteht. Somit ist die Strategie „gestehen“ für den Gefangenen 2 die beste Strategie, unabhängig davon, welche

³Für einen Überblick über das spieltheoretische Konzept von von Neumann/Morgenstern und über dessen Unterschiede zur heutigen Spieltheorie vgl. Selten [120].

⁴Vgl. Varian [142], S. 481.

Tabelle 1.1: Auszahlungsmatrix des Gefangenendilemmas

		Gefangener 2	
		<i>Leugnen</i>	<i>Gestehen</i>
Gefangener 1	<i>Leugnen</i>	−1 ; −1	−6 ; 0
	<i>Gestehen</i>	0 ; −6	−3 ; −3

Strategie vom Gefangenen 1 gespielt wird. Man sagt, dass die Strategie „gestehen“ die *dominante Strategie* des Gefangenen 2 ist. Analog lässt sich zeigen, dass „gestehen“ auch für den Gefangenen 1 die dominante Strategie ist. Demnach werden in diesem Spiel die Gefangenen aus Furcht vor dem lohnenden Verrat des anderen das Verbrechen gestehen, obwohl es für beide besser wäre, die Tat zu leugnen. Daher ist diese sog. *Spiellösung* aus der Sicht der Gefangenen nicht optimal. In diesem Spiel wählen die Gefangenen ihre Strategien quasi gleichzeitig, weil kein Gefangener im Zeitpunkt seiner Entscheidung die gewählte Strategie des anderen kennt. Solche Spielsituationen werden häufig in einer Matrix dargestellt (vgl. Tabelle 1.1). Die Darstellungsform eines Spiels in einer Matrix nennt man *Normalform* (oder auch strategische Form bzw. Matrixform) eines Spiels. In vielen Spielsituationen handeln die Spieler nicht simultan. Sie treffen im Verlauf eines Spiels mehrmals Entscheidungen, wobei jeder Spieler die vorausgegangenen Züge des anderen Spielers beobachten kann. Somit ändert sich in diesen Spielsituationen der Informationsstand der Spieler nach jeder Entscheidung des Mitspielers. Solche Spielsituationen werden formal durch einen sog. *Spielbaum* erfasst. Die Darstellungsform eines Spiels in einem Spielbaum bezeichnet man als *extensive Form* (oder auch als sequentielle Form) eines Spiels.⁵

Spielsituationen, in denen zwischen den Spielern Kommunikation möglich ist und die Spieler bindende Abmachungen treffen können, nennt man *kooperative Spiele*. Die vorliegende Arbeit beschränkt sich in der Folge ausschliesslich auf die kooperativen Spiele. Sind an einem kooperativen Spiel insgesamt $n \geq 2$ Spieler beteiligt, so spricht

⁵Vgl. Holler/Illing [53], S. 5.

1 Eine elementare Systematik strategischer Spiele

man in der Literatur von einem *kooperativen Spiel mit n Spielern*. Die Einhaltung von Abmachungen kann durch die Existenz eines externen Mechanismus, wie z. B. eines Rechtssystems, gewährleistet werden. In solchen Spielen kann der erzielte Nutzen im Vergleich zu nicht-kooperativen Spielen durch bindende Verträge, also durch Bildung von Koalitionen, erhöht werden. Am Beispiel des Gefangenendilemmas als eines nicht-kooperativen Spiels wird dies deutlich: Könnten die beiden Gefangenen miteinander kommunizieren und bindende Abmachungen treffen, würden sie sich für die Strategie „leugnen“ entscheiden. Somit würden die beiden Gefangenen zu der für sie niedrigsten Gefängnisstrafe von einem Monat Haft verurteilt.

Auch im Wirtschaftsleben treffen wir oft den Fall an, dass sich einzelne Akteure, wie z. B. Unternehmen, Gemeinden oder auch Personen zusammenschließen, um ein von ihnen erstrebtes gemeinsames Ziel zu erreichen. In der Theorie der kooperativen Spiele nennt man jeden möglichen Zusammenschluss von Spielern eine *Koalition*. Beispielsweise kann sich eine Gruppe von Entscheidungsträgern dazu entschließen, ein Projekt gemeinsam zu realisieren, um auf diesem Wege z. B. auf Grund von entstandenen Synergien die Kosten des Projektes zu verringern. Verschiedene solche Projekte wurden in der Literatur empirisch aus der Sicht der Theorie kooperativer Spiele analysiert. Zum Beispiel untersuchen Dubey [36], Littlechild/Owen [79], Littlechild [77] und Littlechild/Thompson [78] für das Flughafenprojekt von Birmingham die Bestimmung einer optimalen Landebahnlänge und gerechter Nutzungsgebühren spieltheoretisch. Wissler [147] wendet verschiedene spieltheoretische Ansätze auf die Gemeinkostenzuteilung in Unternehmen an. Frank [40] setzt Konzepte der kooperativen Spieltheorie ein, um bei einer betriebsübergreifenden Energieversorgung die durch die Kooperation entstandene Kostenreduktion auf die Koalitionspartner zu verteilen. Faulhaber [39], Knieps [63], Ransmeier [109], Straffin/Heaney [134], Suzuki/Nakayama [138] und Young [154] analysieren mit Hilfe der Theorie kooperativer Spiele verschiedene Aspekte von Wasserversorgungsprojekten.

Zur Illustration beschränken wir uns in diesem Abschnitt auf eine einfache in der Literatur als Gebührenspiel bekannte Situation:⁶ Drei Nachbargemeinden, die wir

⁶Vgl. z. B. Holler/illing [53], S. 276 f., oder Moulin [86], S. 89 f.

Tabelle 1.2: Die Kosten- und die Einsparungsdaten des Gebührenspiels

Koalition	Kosten	Einsparung
$\{1\}$	44	0
$\{2\}$	36	0
$\{3\}$	40	0
$\{1, 2\}$	74	6
$\{1, 3\}$	68	16
$\{2, 3\}$	50	26
$\{1, 2, 3\}$	78	42

im Weiteren als Gemeinde 1, Gemeinde 2 und Gemeinde 3 bezeichnen, planen die Errichtung eines Wasserversorgungssystems. Jede Gemeinde hat einerseits die Möglichkeit, ihr eigenes System zu errichten. Andererseits kann sie mit einer anderen Gemeinde oder mit beiden anderen Gemeinden kooperieren und zusammen ein gemeinsames Wasserversorgungssystem bauen. Wir nehmen an, dass die Baukosten der Errichtung des Wasserversorgungssystems je nach der Kooperation zwischen den Gemeinden wie folgt anfallen: Entwickelt jede Gemeinde ihr eigenes System, so müssen die Gemeinden 1, 2 und 3 für den Bau 44, 36 bzw. 40 Geldeinheiten zahlen. Kooperiert die Gemeinde 1 mit der Gemeinde 2, so erreicht man auf Grund der entstandenen Synergien eine Kosteneinsparung in der Höhe von 6 Geldeinheiten, sodass die Errichtung des Wasserversorgungssystems die Gemeinde 1 und die Gemeinde 2 zusammen nur 74 Geldeinheiten kostet. Realisieren die Gemeinden 1 und 3 das Projekt gemeinsam, so müssen sie mit Gesamtkosten von 68 Geldeinheiten rechnen. Kooperieren die Gemeinden 2 und 3 miteinander und richten eine gemeinsame Wasserversorgung ein, so belaufen sich die Kosten auf 50 Geldeinheiten. Hingegen kostet es die Gemeinden 78 Geldeinheiten, wenn alle Gemeinden ein gemeinsames Wasserversorgungssystem errichten (vgl. Tabelle 1.2). Werden die Gemeinden miteinander kooperieren? Für welche theoretisch mögliche Option werden sie sich entscheiden? Wovon könnte die Entscheidung der Gemeinden abhängen? Vor allem: Wie sollte

1 Eine elementare Systematik strategischer Spiele

im Falle einer Kooperation eine mögliche Aufteilung der aus der Koalition resultierenden Kosten bzw. Einsparungen zwischen den Gemeinden aussehen? Die Theorie der kooperativen Spiele versucht, auf diese Fragen Antworten zu geben und bietet hierfür verschiedene Konzepte an. Diese Fragen wollen wir im Folgenden anhand des Gebührenspiels verdeutlichen.

In einem kooperativen Spiel ist die Bildung einer Koalition für die Spieler (im obigen Beispiel: die Gemeinden) von Interesse, wenn die Koalition ihnen einen höheren Nutzen als der Nutzen bei Nichtkooperation stiftet. Jeder rationale Spieler strebt wie bereits dargelegt nach der Bildung jener Koalition, die dem Spieler einen möglichst hohen Anteil am Gesamtvorteil zusichert. Im Gebührenspiel beträgt die Einsparung der Koalition aus den Gemeinden 1 und 2 sechs Geldeinheiten, während die Bildung der Koalitionen aus den Gemeinden 1 und 3 bzw. 2 und 3 zu einer Einsparung von 16 ($= 84 - 68$) bzw. 26 ($= 76 - 50$) Geldeinheiten führt (vgl. Tabelle 1.2). Die maximale Einsparung erzielen die Gemeinden aber, wenn alle drei Gemeinden das Projekt gemeinsam realisieren. In diesem Fall beträgt die Gesamteinsparung 42 ($= 120 - 78$) Geldeinheiten. Jede durch die Koalitionsbildung realisierte Kosteneinsparung bezeichnet man als *Koalitionsgewinn* oder auch *Koalitionswert*. Steht bei einem kooperativen Spiel die Aufteilung des Koalitionsgewinns im Mittelpunkt, so sprechen wir von einem Problem der *Gewinnaufteilung*. In der vorliegenden Arbeit befassen wir uns vorwiegend mit dem Problem der Gewinnaufteilung. Hierbei betrachten wir Spiele, die ökonomisch von Vorteil sind, bei denen also ein Koalitionsgewinn möglich ist. Solche Spiele heissen *wesentliche* Spiele. Spiele, bei denen keine Koalitionsgewinne erzielt werden können, werden *unwesentliche* kooperative Spiele genannt. In unwesentlichen Spielen ist die Kooperation nicht vorteilhaft.

Da im betrachteten Gebührenspiel die höchstmögliche Kostenreduktion entsteht, wenn alle drei Gemeinden gemeinsam ein Wasserversorgungssystem bauen, nehmen wir an, dass die Koalition von allen drei Gemeinden zustande kommt. Geht es bei einem kooperativen Spiel um die Aufteilung von Kosten, so spricht man in der Literatur von einem Problem der *Kostenallokation*.⁷ Es stellt sich nun die Frage, wie die

⁷Vgl. dazu z. B. Littlechild/Thompson [78] oder Young [153], [152].

Kosten von 78 Geldeinheiten dieses gemeinschaftlichen Vorhabens bzw. die erzielten Gesamteinsparungen von 42 Geldeinheiten unter den Gemeinden zu verteilen sind. Werden die Kosten z. B. gleichmässig verteilt, so beträgt der Kostenanteil jeder Gemeinde 26 ($= 78/3$) Geldeinheiten. Die Gemeinden 2 und 3 haben aber in diesem Fall keinen Anreiz, mit der Gemeinde 1 zu kooperieren, denn ihr gemeinsamer Beitrag von 52 ($= 26 + 26$) Geldeinheiten zur Deckung der Gesamtkosten übersteigt die Kosten von 50 Geldeinheiten, die entstehen, wenn die Gemeinden 2 und 3 ein gemeinsames Wasserversorgungssystem errichten würden. Demnach führt die Aufteilung der Projektkosten auf die Spieler zu Interessenkonflikten zwischen den Spielern. Es sollte also ein anderer, von den Gemeinden möglichst als „fair“ oder „gerecht“ akzeptierter Verteilerschlüssel angewendet werden, um die Zusammenarbeit aller Gemeinden nicht zu gefährden. In der Theorie der kooperativen Spiele wurden verschiedene Verteilerschlüssel, also *Lösungskonzepte*, entwickelt, die in wohldefinierter Weise Elemente der „Fairness“ oder der „Gerechtigkeit“ beinhalten. Um beurteilen zu können, ob eine Aufteilung der Kosten bzw. des Gewinns gemäss einem Lösungskonzept alle Mitglieder einer Koalition zur Kooperation motiviert, müssen die den Mitgliedern dieser Koalition zur Auswahl stehenden theoretischen Alternativen überprüft werden. Weil die Anzahl der möglichen Koalitionen im Gebührenspiel $2^3 - 1 = 7$ beträgt und damit nicht gross ist, konnten wir oben ohne erheblichen Aufwand feststellen, dass die Gleichverteilung der Projektkosten zwischen den Spielern den Gemeinden 2 und 3 keinen Anreiz gibt, mit der Gemeinde 1 zu kooperieren. Die Anzahl der möglichen Koalitionen steigt aber mit zunehmender Anzahl der Spieler sehr stark an, sodass die Überprüfung, ob ein Verteilerschlüssel allen Mitgliedern der gebildeten Koalition einen Anreiz zur Zusammenarbeit gibt, spezielle mathematische Methoden erfordert. Beispielsweise beträgt die Anzahl der realisierbaren Koalitionen in einem Spiel mit 10 Spielern $2^{10} - 1 = 1023$ und mit 15 Spielern gar $2^{15} - 1 = 32767$. Wie wir in den folgenden Kapiteln der vorliegenden Arbeit noch erläutern werden, explodiert der Rechenaufwand zur Berechnung einer Spiellösung mit zunehmender Anzahl der Spieler. In diesem Zusammenhang spricht man in der Theorie kooperativer Spiele vom sog. „*Berechnungsproblem*“. Um die Lösung eines kooperativen Spiels gemäss ei-

1 Eine elementare Systematik strategischer Spiele

nem bestimmten Lösungskonzept zu bestimmen, wurden in der Literatur auf Grund dieses Berechnungsproblems verschiedene Verfahren entwickelt, die mathematische Techniken, wie z. B. Methoden der linearen Optimierung, anwenden.

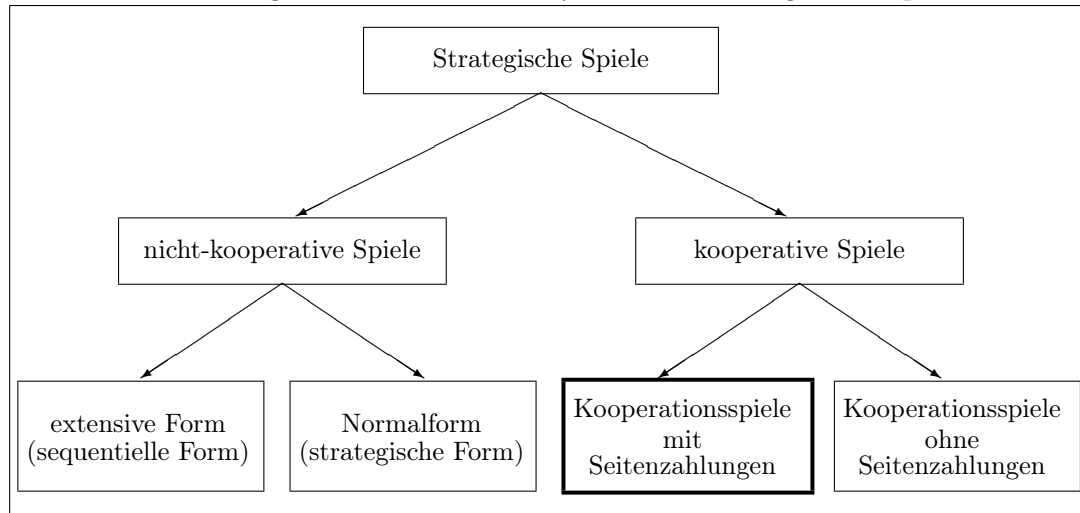
In kooperativen Spielen ist das Verhalten der Spieler insofern „kooperativ“, als sie durch Koalitionsbildung den Gesamtnutzen maximieren möchten, wenn dadurch ihr individueller Nutzen maximal wird. Sobald es um die Aufteilung des durch die Koalition erzielten Gesamtvorteils geht, können zwischen den Spielern Interessenkonflikte entstehen. Kommt im angeführten Beispiel des Gebührenspiels die grosse Koalition mit der Gesamteinsparung von 42 Geldeinheiten zustande, kann z. B. die Gemeinde 1 ihren Vorteil nur auf Kosten mindestens einer der anderen beiden Gemeinden erhöhen. Jede der Gemeinden strebt also bei der Aufteilung des Gesamtgewinns eine möglichst hohe Auszahlung an. In diesem Zusammenhang spricht man von *kooperativen Konflikten*.

Für jedes kooperative Spiel ist es wichtig zu unterscheiden, ob der Nutzen zwischen den Spielern ohne Verlust übertragbar ist, ob also der Nutzen von einem Spieler zum anderen Spieler weitergegeben werden kann. Ein kooperatives Spiel, in dem der Nutzen übertragbar ist, nennt man ein kooperatives Spiel mit *transferierbaren Nutzen* oder mit *Seitenzahlungen*. In solchen Spielen können die Spieler einander kompensieren, um ein gemeinsam erstrebtes Ziel zu erreichen. Also kann der Wert jeder Koalition zwischen den Spielern der Koalition in einer wohldefinierten, von den Spielern akzeptierten Weise aufgeteilt werden. Ist die Übertragung des Nutzens zwischen den Spielern eines kooperativen Spiels nicht möglich, so spricht man von einem kooperativen Spiel mit *nicht-transferierbaren Nutzen* oder von Koalitionsspielen ohne *Seitenzahlungen*. In dieser Arbeit werden wir uns ausschliesslich Koalitionsspielen mit Seitenzahlungen widmen.⁸

Wir beschreiben nun den Inhalt der weiteren Kapitel und nennen sodann die Ziele und die wesentlichen Ergebnisse der Arbeit.

⁸Für eine ausführliche Behandlung der Spiele ohne Seitenzahlungen vgl. z. B. Moulin [86], S. 102 f.

Abbildung 1.1: Eine einfache Systematik strategischer Spiele



Aufbau der Arbeit:

Im zweiten Kapitel dieser Arbeit erläutern wir zunächst einige grundlegende Begriffe der Theorie kooperativer Spiele. Anschliessend befassen wir uns mit zwei wichtigen Konzepten zur Bestimmung kooperativer Konfliktlösungen. Wir stellen zuerst das Konzept des sog. *Kerns* eines kooperativen Spiels vor. Dieses Konzept wurde von Gillies [45] in die Literatur eingeführt. Der Kern enthält jene Elemente, die die Anforderungen der sog. Effizienz und der sog. Anreizkompatibilität (individuellen Rationalität und Gruppenrationalität) erfüllen. Die Effizienz des Kerns verlangt, dass jedes Element des Kerns den Gesamtwert der Koalition aller Spieler vollständig ausschöpft. Die Anreizkompatibilität besagt, dass jedes Element des Kerns jedem Spieler bzw. jeder Koalition von Spielern mindestens den Betrag zuteilt, der seinem bzw. ihrem Koalitionswert entspricht. Daher versprechen die Elemente des Kerns eine gewisse Stabilität der Spiellösung: Wenn alle Spieler übereinstimmend eine durch ein Element des Kerns dargestellte Verteilung des Koalitionsgewinns gewählt haben, so hat weder ein einzelner Spieler noch eine Koalition von Spielern einen Anreiz, diese Aufteilung zu verwerfen. Ein Nachteil des Lösungskonzepts des Kerns besteht darin, dass die Anforderungen des Kerns für ein kooperatives Spiel so restriktiv sein können, dass der Kern des Spiels leer ist. Also sichert der Kern nicht für jedes kooperative

1 Eine elementare Systematik strategischer Spiele

Spiel eine Spiellösung. Umgekehrt hat der Kern aber die Eigenschaft, dass er mehrere Elemente enthalten kann. Neben dem Kern ist der sog. *Nukleolus* ein weiteres Lösungskonzept, das wir im zweiten Kapitel vorstellen. Er wurde von Schmeidler [116] in die Literatur eingeführt. Der Nukleolus stellt diejenige Kostenallokation bzw. Gewinnaufteilung dar, bei der die „Unzufriedenheit“ aller möglichen Koalitionen von Spielern durch Schlichtung minimiert wird. In diesem Zusammenhang spricht man in der Literatur von der Idee der Konsensfindung. Der Nukleolus hat die Eigenschaften, dass er stets nicht-leer und eindeutig ist. Wir zeigen, dass die Berechnung des Nukleolus mit steigender Anzahl der Spieler einen erheblichen Rechenaufwand erfordert und daher die Heranziehung mathematischer Verfahren zur Bestimmung des Nukleolus notwendig ist. Es folgt eine axiomatische Analyse des Konzepts des Nukleolus. Hierfür stellen wir bestimmte Axiome vor und überprüfen anschliessend, ob der Nukleolus das jeweilige Axiom erfüllt. Mit zwei Beweisen der Existenz und der Eindeutigkeit des Nukleolus schliessen wir das zweite Kapitel ab.

Im dritten Kapitel stellen wir die in der Literatur entwickelten Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele vor. Wir unterteilen diese Verfahren in drei Gruppen. Zur ersten Gruppe zählen Verfahren, die den Nukleolus mit Hilfe eines einzigen linearen Programms berechnen. Diese Verfahren nennen wir Methoden der ersten Generation. Zur zweiten Gruppe gehören Verfahren, die für die Berechnung des Nukleolus kooperativer Spiele die Bestimmung der noch zu definierenden sog. „balancierten Koalitionssysteme“ erfordern. Verfahren dieser Gruppe bezeichnen wir als Methoden der zweiten Generation. Die in den letzten Jahren entwickelten neueren Algorithmen beruhen darauf, dass zur Berechnung des Nukleolus eine Folge linearer Programme gelöst wird. Der Algorithmus von Maschler et al. [82] bildet die Grundlage dieser Verfahren. Die Verfahren dieser Gruppe bezeichnen wir als Methoden der dritten Generation.

Im Kapitel 4 geben wir einen Überblick über ausgewählte spezielle kooperative Spiele. Diese Spiele sind dadurch gekennzeichnet, dass sie jeweils eine bestimmte besondere Spielstruktur aufweisen. Deshalb wurden in der Literatur für die Berechnung des Nukleolus dieser Spiele spezielle Algorithmen entwickelt, die die besondere Struk-

tur der Spiele ausnutzen, um eine Einsparung an Rechenaufwand zu erzielen. Wir stellen zunächst exemplarisch den Algorithmus von Wolsey [150] vor. Dieser Algorithmus berechnet den Nukleolus der sog. einfachen kooperativen Spiele, die auch als Abstimmungsspiele bezeichnet werden. In den anschliessenden Abschnitten des Kapitels beschreiben wir weitere spezielle Spiele. Wir erläutern zunächst die jeweilige Spielsituation und zeigen, wie man die noch zu definierende (vgl. S. 15) sog. charakteristische Funktion dieser Spiele aufstellen kann. Anschliessend geben wir einen Literaturüberblick über spezifische Algorithmen zur Berechnung des Nukleolus dieser Spiele.

Die Implementation zweier Algorithmen ist Gegenstand des fünften Kapitels. Wie wir oben erwähnt haben, verfolgen die zur Berechnung des Nukleolus spezieller kooperativer Spiele entwickelten spezifischen Verfahren das Ziel, den Rechenaufwand zu reduzieren. Praktische Rechenbeispiele z. B. aus der Theorie der linearen Optimierung zeigen aber, dass zwar aus der Sicht der Komplexitätsanalyse ein Verfahren theoretisch einen geringeren Rechenaufwand aufweist als das andere, jedoch in der Praxis einen höheren Rechenaufwand in Anspruch nehmen kann.⁹ Um exemplarisch zu untersuchen, ob es sich zur Berechnung des Nukleolus spezieller kooperativer Spiele lohnt, spezifische Verfahren zu entwickeln, implementieren wir im Kapitel 5 zwei Verfahren zur Berechnung des Nukleolus und vergleichen die ausgewählten Verfahren mit Hilfe unserer Implementation. Hierbei überprüfen wir, ob ein zur Berechnung des Nukleolus einfacher kooperativer Spiele entwickeltes Verfahren in der Theorie wie auch in der Praxis einem zur Berechnung allgemeiner kooperativer Spiele entwickelten Verfahren überlegen ist.

Ein interessantes Anwendungsbeispiel präsentieren wir im sechsten Kapitel. Hierbei handelt es sich um die Anwendung des Nukleolus auf die schweizerische Demokratie. Die schweizerische Demokratie wird in der Literatur als eine Konsensde-

⁹In der Literatur wurden sog. Interior Point-Methoden zur Berechnung von Lösungen linearer Optimierungsaufgaben entwickelt, die einen theoretisch geringeren Rechenaufwand aufweisen als die Simplex-Methode. Praktische Rechenbeispiele haben aber gezeigt, dass in der Praxis die Simplex-Methode einen niedrigeren Rechenaufwand erfordert und folglich für die Lösung praktischer Probleme geeigneter ist als die Interior Point-Methoden. Für eine ausführlichere Darstellung und die Erläuterung der in dieser Fussnote verwendeten Begriffe vgl. Anhang A.1.

1 Eine elementare Systematik strategischer Spiele

mokratie bezeichnet. Da die Grundidee des Nukleolus auch auf der Konsensfindung beruht, liegt es nahe, das Lösungskonzept des Nukleolus auf die schweizerische Politik anzuwenden. Wir konzentrieren uns insbesondere auf die parteipolitische Zusammensetzung des schweizerischen Bundesrates. Zu diesem Zweck stellen wir für die Eingensässischen Räte (der Ständerat, der Nationalrat und die Vereinigte Bundesversammlung) jeweils ein kooperatives Spiel auf, wobei wir die Parteien als die Spieler des jeweiligen Spiels betrachten. Wir berechnen eine theoretische Verteilung der Bundesratssitze zwischen den Parteien gemäss dem Konzept des Nukleolus und vergleichen diese rechnerische Sitzverteilung mit der aktuellen parteipolitischen Verteilung der Bundesratssitze.

Ziele und wesentliche Resultate der Arbeit:

Die vorliegende Arbeit verfolgt drei Hauptziele. Ihr erstes Ziel ist es, die in der Literatur entwickelten Verfahren zur Berechnung des Nukleolus kooperativer Spiele vorzustellen und zu klassifizieren. Zudem will sie zwei der vorgestellten Verfahren implementieren und anhand dieser Implementation miteinander vergleichen. Als drittes Ziel strebt die Arbeit an, eine empirische Anwendung des Konzepts des Nukleolus vorzuführen.

Ein erstes Ergebnis der Arbeit besteht darin, dass die Verfahren zur Berechnung des Nukleolus zweckmässigerweise in drei Gruppen eingeteilt werden können. Ferner beinhaltet die Arbeit eine Implementation zweier Verfahren zur Berechnung des Nukleolus und führt anhand dieser Implementation einen Vergleich der Verfahren aus. Aus dem Vergleich gehen unter anderem folgende Ergebnisse hervor: Das ausgewählte Verfahren zur Berechnung des Nukleolus spezieller kooperativer Spiele erfordert im Vergleich zum gewählten Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele zwar die Lösung von mehr linearen Programmen, die aber erheblich weniger Nebenbedingungen enthalten. Aufgrund der niedrigen Anzahl der Nebenbedingungen der linearen Programme benötigt das spezielle Verfahren weniger Speicherplatz und kann somit für die Berechnung des Nukleolus von Spielen mit einer grösseren Anzahl von Spielern eingesetzt werden. Zudem weist der Algo-

rithmus zur Bestimmung des Nukleolus spezieller kooperativer Spiele ein besseres Laufzeitverhalten auf. Demzufolge ziehen wir das Fazit, dass es für die Berechnung des Nukleolus spezieller kooperativer Spiele von Vorteil ist, spezifische Algorithmen zu entwickeln. Der empirische Teil der Arbeit zeigt, dass die schweizerische Politik ein mögliches Anwendungsgebiet für die Theorie der kooperativen Spiele darstellt. Insbesondere lässt dieses letzte Kapitel der Arbeit erkennen, dass der Nukleolus auf die schweizerische Politik angewendet werden kann. Als eine mögliche Anwendung des Nukleolus auf die schweizerische Politik legt die Arbeit dar, dass der Nukleolus zu einer rechnerischen Aufteilung der Bundesratssitze zwischen den politischen Parteien der Schweiz herangezogen werden kann.

2 Kooperative Spiele mit zwei oder mehr Spielern

2.1 Grundlagen

Bei kooperativen Spielen haben die Spieler die Möglichkeit, miteinander Koalitionen einzugehen. Wir gehen in dieser Arbeit von kooperativen Spielen mit insgesamt $n \geq 2$ Spielern aus. Die Menge der Spieler bezeichnen wir als $N = \{1, 2, \dots, n\}$. Eine *Koalition* S ist eine Teilmenge der Spielermenge N . Da eine n -elementige Menge insgesamt 2^n Teilmengen hat, beträgt für die Spielermenge N die Anzahl der theoretisch möglichen Koalitionen 2^n . Wie im vorangegangenen Kapitel erwähnt, wird in der Theorie der kooperativen Spiele vorausgesetzt, dass mit jeder Koalition $S \subseteq N$ bestimmtes Koalitionsergebnis, wie z. B. Kosteneinsparungen durch Kooperation, verbunden ist. In dieser Arbeit beschäftigen wir uns hauptsächlich mit den kooperativen Spielen, bei denen Kooperation stets vorteilhaft ist. Den durch die Kooperation erzielten Gewinn bezeichnet man in der Literatur als den Wert einer Koalition oder kurz als *Koalitionswert*. In der Tabelle 2.1 sind für das Gebührenspiel (im Weiteren als Einführungsbeispiel bezeichnet) die theoretisch denkbaren Koalitionen und deren Koalitionswerte aufgeführt. Da die leere Menge \emptyset auch eine Teilmenge von N ist, nimmt man an, dass für jedes kooperative Spiel der Wert dieser „Koalition“ Null ist. Besteht eine Teilmenge von N aus einem einzigen Spieler, so wird diese Teilmenge in der Theorie der kooperativen Spiele eine „Einerkoalition“ genannt. Die Tabelleneinträge für die Einerkoalitionen geben die Koalitionsgewinne bei Nichtkooperation an. Als *grosse Koalition* bezeichnet man eine Koalition aller Spieler der Spielermenge $N = \{1, 2, \dots, n\}$. Ihren Wert haben wir in der letzten Zeile der Tabelle 2.1 eingetragen. In der Theorie der kooperativen Spiele werden die Werte aller Koalitionen durch die sog. *charakteristische Funktion*

$$v : S \longrightarrow \mathbb{R}_{\geq 0} \quad \forall S \in 2^N \text{ und } v(\emptyset) = 0$$

2 Kooperative Spiele mit zwei oder mehr Spielern

Tabelle 2.1: Koalitionswerte des Einführungsbeispiels

Koalition S	Koalitionswert $v(S)$
$\{1\}$	0
$\{2\}$	0
$\{3\}$	0
$\{1, 2\}$	6
$\{1, 3\}$	16
$\{2, 3\}$	26
$\{1, 2, 3\}$	42

angegeben. Sie ordnet jeder Koalition eines kooperativen Spiels eine nicht-negative reellwertige Zahl zu. Hierbei ist 2^N die Potenzmenge, also die Menge aller Teilmengen, von N . Für jede Koalition $S \subseteq N$ ist $v(S)$ derjenige Wert, den die Koalition S unabhängig von den Handlungen der übrigen Koalitionen erreichen kann. Also ist $v(S)$ der maximale Nutzen, den die Koalition S aus sich selbst heraus verwirklichen kann, ohne mit anderen Koalitionen in $N \setminus S$ zu kooperieren. Somit entspricht $v(S)$ dem Koalitionswert der Koalition S . Das Aufstellen der charakteristischen Funktion eines Spiels ist in der Regel aufwändig. Man kann beispielsweise für die Durchführung eines Projektes durch Schätzung der Kosten die Kosteneinsparungen der Kooperationen, also die Koalitionsgewinne, ermitteln und somit die charakteristische Funktion eines Spiels bestimmen. So haben z. B. Ingenieure und Ökonomen die Projektkosten des in der Literatur als TVA-Kostenspiel (Tennessee Valley Authority-Kostenspiel) bekannten Spiels geschätzt. Die Tennessee Valley Authority wurde in den 30er Jahren in den Vereinigten Staaten gegründet. Das Ziel der TVA bestand darin, die Schifffahrt auf dem Tennessee Valley River zu verbessern, die Anwohner vor Hochwasser zu schützen und Strom zu produzieren. Die drei Pläne konnten kombiniert, aber auch als separate Projekte realisiert werden. Zu diesem Zweck wurden die Kosten sowohl jedes einzelnen Vorhabens wie auch die Kosten der Kombination der drei Teilprojekte geschätzt (vgl. dazu z. B. Ransmeier [109]) und somit die charakteristische Funktion

des Spiels festgelegt.¹ Ein kooperatives Spiel mit n Spielern wird formal wie folgt definiert:

Definition 2.1.1 (kooperatives Spiel) *Ein kooperatives Spiel in charakteristischer Form ist ein Paar $\Gamma = (N, v)$, wobei $N = \{1, 2, \dots, n\}$ die Menge der Spieler ist und v die charakteristische Funktion des Spiels angibt.*

Wir nehmen nun an, dass die charakteristische Funktion eines kooperativen Spiels mit n Spielern gegeben ist. Da in einem solchen Spiel den Spielern mehrere Koalitionsmöglichkeiten zur Verfügung stehen, stellt sich die Frage, welche Koalition von den Spielern am ehesten gebildet wird. In einem kooperativen Spiel hängt es einerseits von der charakteristischen Funktion des Spiels und andererseits von der Verteilung des Koalitionswertes zwischen den Spielern ab, welche Koalition von den Spielern des Spiels möglicherweise gegründet wird.² Betrachten wir das Einführungsbeispiel, so stellen wir fest, dass der Wert einer Koalition umso grösser ist, je höher die Zahl der Mitglieder ist. Während z. B. $v(\{1\}) = 0$ ist, beträgt $v(\{1, 2\}) = 6$, $v(\{1, 3\}) = 16$ und $v(\{1, 2, 3\}) = 42$. Spiele, deren charakteristische Funktion dieses Merkmal besitzt, nennt man *monotone* Spiele. Formal heisst ein kooperatives Spiel $\Gamma = (N, v)$ *monoton*, falls gilt:

$$T \subset S \Rightarrow v(T) \leq v(S) \quad \text{für alle } S, T \subseteq N. \quad (2.1)$$

Bei einem monotonen kooperativen Spiel kann der Wert einer Koalition S nicht niedriger sein als der Wert einer kleineren Koalition, die eine Teilmenge der Koalition S ist.³ Das heisst, dass bei einem monotonen kooperativen Spiel kein neuer Beitritt eines Spielers oder einer Koalition einer bestehenden Koalition schadet. Es

¹Im Kapitel 4 werden wir die Bestimmung der charakteristischen Funktion einiger spezieller kooperativer Spiele vorstellen. Ansonsten beschäftigen wir uns in dieser Arbeit nicht mit der Festlegung der charakteristischen Funktion kooperativer Spiele und setzen voraus, dass diese gegeben ist.

²In dieser Arbeit gehen wir davon aus, dass jede mögliche Koalition zulässig ist. Kooperative Spiele, bei denen dies nicht der Fall ist, werden z. B. von Aumann/Dreze [3] diskutiert.

³Für zwei beliebige Koalitionen $S, T \subseteq N$ ist die Koalition S grösser als die Koalition T , falls der Koalition S mehr Spieler angehören als der Koalition T .

2 Kooperative Spiele mit zwei oder mehr Spielern

kann jedoch der Fall sein, dass der Wert einer Koalition mit dem neuen Beitritt gesteigert wird. Wenn wir annehmen, dass sich jeder Spieler entscheiden kann, welche Koalition er eingehen möchte, so stellt die Eigenschaft der monotonen Spiele keine Einschränkung der Allgemeinheit dar. Die Überprüfung aller möglichen Varianten bestätigt, dass die charakteristische Funktion des Einführungsbeispiels die Monotonieeigenschaft besitzt. Überdies stellen wir beim Einführungsbeispiel fest, dass der bei einer Kooperation zweier beliebiger disjunkter Koalitionen⁴ erzielbare Gewinn stets mindestens so hoch wie die Summe der Gewinne der beiden Koalitionen ist. Zum Beispiel gilt

$$v(\{1\}) + v(\{2, 3\}) < v(\{1\} \cup \{2, 3\}).$$

Spiele mit dieser Eigenschaft nennt man *superadditiv*. Allgemein heisst ein kooperatives Spiel mit n Spielern *superadditiv* oder auch *proper*, falls für alle Koalitionen $S, T \in 2^N \setminus \emptyset$ gilt, dass

$$v(S) + v(T) \leq v(S \cup T) \quad \text{für } S \cap T = \emptyset. \quad (2.2)$$

Ökonomisch drückt die Superadditivität die Vorteilhaftigkeit der Zusammenarbeit bei der Durchführung eines Projektes aus. Ist die Bedingung (2.2) als Gleichung erfüllt, so erbringt die Kooperation zweier disjunkten Koalitionen keinen zusätzlichen Nutzen, aber hat auch keinen Nachteil. Folglich sollte man erwarten, dass bei einem superadditiven Spiel kein Spieler und keine Koalition von Spielern Vorbehalte gegen die Bildung der grossen Koalition haben. In dieser Arbeit betrachten wir superadditive Spiele und setzen voraus, dass die grosse Koalition $N = \{1, 2, \dots, n\}$ zustande kommt. Die Unterstellung der Superadditivität ist eine plausible Annahme und schränkt somit die Wirklichkeit nicht ein. Denn zwei Koalitionen können immer dadurch „kooperieren“, indem sie ihre eigenen Projekte nebeneinander durchführen (vgl. Wolff [148], S. 1101, und Luce/Raiffa [80], S. 183).

⁴In dieser Arbeit schliesst der Begriff der Koalition auch die Einerkoalitionen ein.

2.1 Grundlagen

Ist bei einem kooperativen Spiel der Grenzbeitrag⁵ eines Spielers umso grösser, je grösser die Koalition ist, der er sich anschliesst, so könnte man ebenfalls erwarten, dass in diesem Spiel eher die grosse Koalition gegründet wird (vgl. Holler/Illing [53], S. 268, und Moulin [86], S. 112). Kooperative Spiele, die dieses Charakteristikum besitzen nennt man *konvexe* Spiele. Formal heisst ein kooperatives Spiel konvex, wenn eine der folgenden Bedingungen erfüllt ist:

- Für alle $i \in N$ und alle $S, T \subset N \setminus \{i\}$ mit $S \subset T$ gilt:

$$v(S \cup \{i\}) - v(S) \leq v(T \cup \{i\}) - v(T). \quad (2.3)$$

- Für alle $S, T \in 2^N \setminus \emptyset$ gilt:

$$v(S) + v(T) \leq v(S \cup T) + v(S \cap T). \quad (2.4)$$

Wir verzichten auf den Beweis der Äquivalenz beider Bedingungen (vgl. Moulin [86], S. 112 f). Ökonomisch drückt die Konvexität eines Spiels steigende Skalenerträge auf Grund der Kooperation aus (vgl. Holler/Illing [53], S. 268, Wissler [147], S. 32). Wir können nun mit Hilfe der Bedingung (2.4) überprüfen, ob das Einführungsbeispiel konvex ist. Nehmen wir beispielsweise an, dass $S = \{1, 2\}$ und $T = \{2, 3\}$ ist:

$$\begin{aligned} v(\{1, 2\}) + v(\{2, 3\}) &= 6 + 26 \\ &= 32, \end{aligned} \quad (2.5)$$

$$\begin{aligned} v(\{1, 2\} \cup \{2, 3\}) + v(\{1, 2\} \cap \{2, 3\}) &= v(\{1, 2, 3\}) + v(\{2\}) \\ &= 42 + 0 \\ &= 42. \end{aligned} \quad (2.6)$$

Aus (2.5) und (2.6) folgt, dass die Bedingung (2.4) für die ausgewählten Koalitionen erfüllt ist. Analog kann man zeigen, dass für alle anderen Koalitionen aus $2^N \setminus \emptyset$ ebenfalls die Bedingung (2.4) gilt. Demnach ist das Einführungsbeispiel ein konvexes Spiel.

⁵Der Grenzbeitrag eines Spielers i zu einer Koalition S entspricht der durch seinen Koalitionsbeitritt entstehenden Steigerung des Koalitionswertes. Formal ist der Grenzbeitrag des Spielers $i \notin S$ zu der Koalition S durch $v(S \cup \{i\}) - v(S)$ definiert.

2 Kooperative Spiele mit zwei oder mehr Spielern

Gemäss der Definition der charakteristischen Funktion kann der Wert einer Koalition eine beliebige nicht-negative reellwertige Zahl sein. Man kann aber die charakteristische Funktion eines superadditiven kooperativen Spiels so umwandeln, dass die Werte der Koalitionen zwischen null und eins liegen. Das ist die sog. $(0 - 1)$ -Normierung eines kooperativen Spiels. Nach der Normierung eines kooperativen Spiels gilt, $v(\{i\}) = 0$ für alle $i \in N$ und $v(\{N\}) = 1$. Ein Vorteil der $(0 - 1)$ -Normierung besteht darin, dass wir die Werte der Koalitionen besser miteinander vergleichen können. Die Definition 2.1.2 gibt an, wie man ein kooperatives Spiel mit n Spielern $(0 - 1)$ -normieren kann (vgl. Friedman [42], S. 193):

Definition 2.1.2 ((0 - 1)-Normierung) *Es sei $\Gamma = (N, v)$ ein kooperatives Spiel mit n Spielern und $v(N) > \sum_{i \in N} v(\{i\})$. Die $(0 - 1)$ -normierte Form dieses Spiels $\Gamma^* = (N, v^*)$ besitzt dann die charakteristische Funktion*

$$v^*(S) := \frac{v(S) - \sum_{i \in S} v(\{i\})}{v(N) - \sum_{i \in N} v(\{i\})} \quad \text{für alle } S \subseteq N. \quad (2.7)$$

Die Superadditivität garantiert, dass $0 \leq v^*(S) \leq 1$ für alle $S \subseteq N$ gilt. Für die Einerkoalitionen ist der Zähler in der charakteristischen Funktion (2.7) immer Null. Gemäss der Annahme $v(N) > \sum_{i \in N} v(\{i\})$ ist der Nenner immer positiv, sodass $v^*(\{i\}) = 0$ für alle $i \in N$ ist. Ferner gilt $v^*(N) = 1$, da im Falle der grossen Koalition der Zähler und der Nenner in (2.7) übereinstimmen. Ein kooperatives Spiel, das die Eigenschaft $v(N) > \sum_{i \in N} v(\{i\})$ besitzt, nennt man *wesentliches Spiel*. Bei wesentlichen Spielen kann die Auszahlung mindestens eines Spielers i durch die Gründung der grossen Koalition im Vergleich zur Einerkoalition $\{i\}$ verbessert werden. Mit Hilfe der Definition 2.1.2 können wir nun das Einführungsbeispiel $(0 - 1)$ -normieren. Da der Wert jeder Einerkoalition Null und der Wert der grossen Koalition Eins ist, wenden wir die Definitionsgleichung (2.7) nur noch auf die Zweierkoalitionen an:

$$S = \{1, 2\} \Rightarrow v^*(\{1, 2\}) = \frac{6 - (0 + 0)}{42 - (0 - 0)} = \frac{6}{42} \cong 0.14,$$

$$S = \{1, 3\} \Rightarrow v^*(\{1, 3\}) = \frac{16 - (0 + 0)}{42 - (0 - 0)} = \frac{16}{42} \cong 0.38,$$

Tabelle 2.2: $(0 - 1)$ -Normierung des Einführungsbeispiels

Koalition S	Koalitionswert $v^*(S)$
$\{1\}$	0
$\{2\}$	0
$\{3\}$	0
$\{1, 2\}$	0.14
$\{1, 3\}$	0.38
$\{2, 3\}$	0.62
$\{1, 2, 3\}$	1

$$S = \{2, 3\} \Rightarrow v^*(\{2, 3\}) = \frac{26 - (0 + 0)}{42 - (0 - 0)} = \frac{26}{42} \cong 0.62.$$

Tabelle 2.2 gibt die charakteristische Funktion des $(0 - 1)$ -normierten Einführungsbeispiels wieder.

Ein kooperatives Spiel und das dazugehörige $(0 - 1)$ -normierte kooperative Spiel repräsentieren dieselbe Spielsituation. Im Allgemeinen können zwei kooperative Spiele mit n Spielern unterschiedliche charakteristische Funktionen haben, aber trotzdem die gleiche Spielsituation beschreiben. Beispielsweise kann durch Veränderung der Masseinheit, in der die Koalitionswerte ausgedrückt werden (z. B. Währungsumrechnung), die charakteristische Funktion eines kooperativen Spiels verändert werden. Dabei spricht man von der *Skalierung* der charakteristischen Funktion. Die so entstandenen Spiele repräsentieren trotz ihrer unterschiedlichen charakteristischen Funktionen dasselbe kooperative Spiel. Zudem ist denkbar, dass die Spieler eines kooperativen Spiels unabhängig von der Spielsituation über bestimmte Auszahlungen verfügen können. In diesem Fall kann man die Koalitionswerte um die Summe dieser Auszahlungen der Koalitionsmitglieder erweitern und somit eine andere charakteristische Funktion erhalten, obwohl sich die Spielsituation nicht geändert hat. In

2 Kooperative Spiele mit zwei oder mehr Spielern

diesem Zusammenhang spricht man in der Literatur von sog. *strategisch äquivalenten* kooperativen Spielen:

Definition 2.1.3 (strategische Äquivalenz) *Zwei kooperative Spiele $\Gamma = (N, v)$ und $\hat{\Gamma} = (N, \hat{v})$ mit n Spielern sind strategisch äquivalent, falls es reellwertige Zahlen $c > 0$ und a_1, a_2, \dots, a_n gibt, so dass gilt:*

$$\hat{v}(S) = cv(S) + \sum_{i \in S} a_i \text{ für alle } S \subseteq N.$$

Die Konstante c stellt die Skalierung der charakteristischen Funktion dar. Die Zahlen $a_1, a_2, \dots, a_n \in \mathbb{R}$ können wir als Auszahlungen an die Spieler so interpretieren, dass die Spieler diese Auszahlungen unabhängig von der charakteristischen Funktion v fest zugeordnet erhalten (vgl. Kaufmann/Tack [58], S. 33). Diese zusätzliche Bereitstellung fester Auszahlungen an die Spieler können wir in der charakteristischen Funktion \hat{v} dadurch berücksichtigen, dass wir den Wert einer Koalition S um die Summe der Zusatzbeträge $a_i \forall i \in S$ erhöhen bzw. reduzieren, falls negative Zusatzbeträge vorliegen. Wenn man feststellt, dass bestimmte Typen kooperativer Spiele strategisch äquivalent sind, kann man diese jeweils in einer gemeinsamen Klasse erfassen und bei der Analyse der Spiele nur ein einziges repräsentatives Spiel aus jeder Klasse betrachten. Üblicherweise wird in der Literatur aus einer solchen Klasse ein $(0 - 1)$ -normiertes kooperatives Spiel ausgewählt (vgl. z. B. Jianhua [56], S. 98). Wenn nämlich $\Gamma^* = (N, v^*)$ das $(0 - 1)$ -normierte Spiel eines kooperativen Spiels $\Gamma = (N, v)$ ist, dann sind die beiden kooperativen Spiele strategisch äquivalent (vgl. Friedman [42], S. 193). Diese Aussage können wir beweisen, indem wir die Gleichung (2.7) folgendermassen umschreiben:

$$\begin{aligned} v^*(S) &= \frac{v(S)}{v(N) - \sum_{i \in N} v(\{i\})} - \frac{\sum_{i \in S} v(\{i\})}{v(N) - \sum_{i \in N} v(\{i\})} \\ &= \frac{1}{v(N) - \sum_{i \in N} v(\{i\})} v(S) - \sum_{i \in S} \left[\frac{v(\{i\})}{v(N) - \sum_{i \in N} v(\{i\})} \right] \end{aligned}$$

$$\begin{aligned}
&= \underbrace{\frac{1}{v(N) - \sum_{i \in N} v(\{i\})}}_{=: c} v(S) + \sum_{i \in S} \underbrace{\left[\frac{-v(\{i\})}{v(N) - \sum_{i \in N} v(\{i\})} \right]}_{=: a_i} \\
&= cv(S) + \sum_{i \in S} a_i.
\end{aligned}$$

□

Satz 2.1.1 *Zu jedem kooperativen Spiel $\Gamma = (N, v)$ mit $v(N) > \sum_{i \in N} v(\{i\})$ ist genau ein $(0-1)$ -normiertes Spiel strategisch äquivalent.*

Beweis: Vgl. Jianhua [56], S. 99.

Dieser Satz impliziert, dass für eine Spielermenge N jede Klasse strategisch äquivalenter kooperativer Spiele durch ihr $(0-1)$ -normiertes kooperatives Spiel repräsentiert werden kann.

Wie wir im vorangegangenen Kapitel ausgeführt haben, besteht ein wesentliches Merkmal der kooperativen Spiele in der Annahme, dass die Spieler den Wert einer Koalition maximieren, indem sie miteinander kooperieren. Einigen sich die Spieler auf die Kooperation und gründen die grosse Koalition, so muss entschieden werden, wie der Koalitionsgewinn unter den Spielern dieser Koalition aufgeteilt werden soll. In dieser Phase beginnen die Verhandlungen zwischen den miteinander kooperierenden Spielern. Um die zustande gekommene Koalition nicht zu gefährden, ist eine für alle an der Koalition beteiligten Spieler akzeptable Lösung dieser Konfliktsituation wichtig. Für ein kooperatives Spiel $\Gamma = (N, v)$ nennen wir eine Aufteilung des Koalitionsgewinns zwischen den Spielern *Auszahlungs-* bzw. *Allokationsvektor*. Wir bezeichnen einen Auszahlungsvektor als $\mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbb{R}_+^n$, wobei x_i die Auszahlung für Spieler $i \in N$ bedeutet und n die Anzahl der Spieler ist. Werden die Koalitionsgewinne bzw. die Koalitionswerte in Geldeinheiten ausgedrückt, wie wir es im vorangegangenen Kapitel vorgenommen haben, können die Auszahlungen x_i für alle $i \in N$ als Geldbeträge betrachtet werden. Ein von den Spielern realisierter Koalitionsgewinn kann in unterschiedlicher Weise unter den Spielern aufgeteilt werden

2 Kooperative Spiele mit zwei oder mehr Spielern

(z. B. Gleichverteilung). Insbesondere sollte ein Auszahlungsvektor den gesamten Koalitionsgewinn ohne Rest auf die Spieler verteilen, denn die Spieler wollen ihre Auszahlung maximieren. Wenn wir annehmen, dass der Nutzen jedes Spielers mit steigender Auszahlung zunimmt, würde ein verbleibender Teil des Koalitionsgewinns bedeuten, dass nicht alle Möglichkeiten der Nutzenmaximierung erschöpft worden sind. Ein Auszahlungsvektor, bei dem die Summe der Auszahlungen für alle Spieler mit dem Wert der grossen Koalition N übereinstimmt, heisst *effizient*. In diesem Fall kann die Auszahlung keines Spielers verbessert werden, ohne mindestens einen anderen Spieler schlechter zu stellen. Daher spricht man auch von der Eigenschaft der *Pareto-Optimalität* eines Auszahlungsvektors. Ein Auszahlungsvektor $\mathbf{x} \in \mathbb{R}^n$ ist also effizient, falls er die Gleichung $\sum_{i \in N} x_i = v(N)$ erfüllt. In dieser Arbeit definieren wir die Summe der Auszahlungen einer Koalition S bezüglich eines Auszahlungsvektors $\mathbf{x} \in \mathbb{R}^n$ durch

$$x(S) := \sum_{i \in S} x_i \text{ für alle } S \subseteq N.$$

Demzufolge ist ein Auszahlungsvektor $\mathbf{x} \in \mathbb{R}^n$ effizient, falls $x(N) = v(N)$ gilt. Die Menge der effizienten Auszahlungsvektoren heisst *Preimputationsmenge*:

Definition 2.1.4 (Preimputationsmenge) Die Preimputationsmenge $\mathcal{PI}(N, v)$, \mathcal{PI} bzw. $\mathcal{PI}(\Gamma)$ eines kooperativen Spiels $\Gamma = (N, v)$ ist die Menge aller effizienten Auszahlungsvektoren.

Neben der Effizienzanforderung wird jeder Spieler erwarten, dass sein Anteil am Gewinn der grossen Koalition mindestens so hoch ist, wie er auch allein, ohne die Mitgliedschaft in einer Mehrspielerkoalition, realisieren würde. Ein Auszahlungsvektor mit dieser Eigenschaft heisst *individuell rational*. Die individuelle Rationalität impliziert die Annahme, dass sich kein Spieler einer Koalition anschliessen wird, in der seine Auszahlung niedriger ist, als die Auszahlung, die er allein erzielen würde. Ansonsten würde ein Spieler die freiwillige Kooperation verweigern. Mit Hilfe dieser beiden Kriterien, nämlich der Effizienz und der individuellen Rationalität, kann nun die Menge der möglichen Auszahlungsvektoren zur Aufteilung des Gewinns der grossen Koalition eingegrenzt werden:

Definition 2.1.5 (Imputation) Ein Auszahlungsvektor, der individuell rational und Pareto-optimal (effizient) ist, heisst Imputation oder Zurechnung. Für kooperative Spiele $\Gamma = (N, v)$ mit Seitenzahlungen erfüllt eine Imputation $\mathbf{x} \in \mathbb{R}^n$ die folgenden Bedingungen:

$$\bullet \quad x(N) = v(N) \quad (\text{Effizienz}) \quad (2.8)$$

$$\bullet \quad x_i \geq v(\{i\}), \quad i = 1, 2, \dots, n \quad (\text{individuelle Rationalität}). \quad (2.9)$$

Definition 2.1.6 (Imputationsmenge) Die Imputationsmenge \mathcal{I} , $\mathcal{I}(N, v)$ bzw. $\mathcal{I}(\Gamma)$ eines kooperativen Spiels $\Gamma = (N, v)$ ist die Menge aller Imputationen.

Durch Aufsummierung der Ungleichungen (2.9) und anschliessende Anwendung der Gleichung (2.8) erhalten wir:

$$\sum_{i=1}^n x(\{i\}) \geq \sum_{i=1}^n v(\{i\}) \Rightarrow v(N) \geq \sum_{i=1}^n v(\{i\}). \quad (2.10)$$

Erfüllt die charakteristische Funktion eines kooperativen Spiels die Ungleichung (2.10), so ist die Imputationsmenge dieses Spiels nicht-leer. In diesem Fall existiert mindestens ein Auszahlungsvektor, der individuell rational und effizient ist. Gilt in (2.10) das Gleichheitszeichen, so hat keiner der Spieler einen Anreiz zur Bildung der grossen Koalition N . Es gibt aber auch keinen Anreiz, die grosse Koalition zu verhindern. Ist die Bedingung (2.10) als Ungleichung erfüllt, das Spiel also wesentlich, dann kann durch die Bildung der grossen Koalition die Auszahlung mindestens eines Spielers im Vergleich zur Einerkoalition erhöht werden. In dieser Arbeit sind für uns kooperative Spiele, deren Imputationsmenge nicht-leer ist und deren Spieler einen Anreiz haben, die grosse Koalition zu bilden, von Interesse. Deswegen gehen wir von wesentlichen kooperativen Spielen aus.

Wir bestimmen nun die Imputationsmenge des Einführungsbeispiels. Diese Menge besteht aus den Auszahlungsvektoren $\mathbf{x} \in \mathbb{R}^3$, die folgende Bedingungen erfüllen:

$$\text{Effizienz:} \quad x_1 + x_2 + x_3 = 42,$$

$$\text{Individuelle Rationalität:} \quad x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

2 Kooperative Spiele mit zwei oder mehr Spielern

Somit ist jeder Auszahlungsvektor $\mathbf{x} \in \mathbb{R}^3$, dessen Komponenten nicht-negativ sind und die sich zu dem Wert der grossen Koalition 42 addieren, ein Element der Imputationsmenge des Einführungsbeispiels.

2.2 Der Kern eines kooperativen Spiels

Die Imputationsmenge eines kooperativen Spiels grenzt die Menge aller möglichen Auszahlungsvektoren zur Aufteilung eines Koalitionsgewinns auf die Menge der effizienten und individuell rationalen Auszahlungsvektoren ein. Während die Effizienzanforderung garantiert, dass der Wert der grossen Koalition vollständig ausgeschöpft wird, gewährleistet die Bedingung der individuellen Rationalität, dass sich keine Einerkoalition durch Nichtkooperation gegenüber der grossen Koalition verbessern kann. Garantiert aber jede Imputation, dass die Spieler die grosse Koalition bilden würden? In unserem Einführungsbeispiel ist der Auszahlungsvektor $(0, 0, 42)$ ein Element der Imputationsmenge, da er effizient und individuell rational ist. Diese Imputation gibt aber der Koalition $\{1, 2\}$ keinen Anreiz zur Kooperation mit dem Spieler 3. Denn die Koalition $\{1, 2\}$ kann ohne Kooperation mit der Koalition $\{3\}$ den Wert $v(\{1, 2\}) = 6$ realisieren. Durch z. B. Gleichverteilung des Koalitionswertes 6 könnten sich die Spieler der Koalition $\{1, 2\}$ jeweils eine höhere Auszahlung sichern als die Auszahlung von Null bei der Imputation $(0, 0, 42)$. Das Zustandekommen der grossen Koalition hängt also davon ab, wie der Koalitionsgewinn unter den Spielern verteilt wird. Möchte man garantieren, dass sich keine Koalition veranlasst fühlt, die grosse Koalition zu verlassen, müsste man zur Aufteilung des Koalitionsgewinns möglichst diejenigen Imputationen in Betracht ziehen, bei denen die Mitglieder jeder Koalition gemeinsam mindestens ihren Koalitionswert erhalten. Diese Überlegung führt uns zum Begriff der Dominanz von Auszahlungsvektoren:

Definition 2.2.1 (schwache Dominanz) *Es sei $\Gamma = (N, v)$ ein kooperatives Spiel mit Seitenzahlungen. Eine Imputation \mathbf{y} heisst schwach dominiert durch eine andere Imputation \mathbf{x} in Bezug auf eine Koalition $S \subset N$, wenn die Bedingungen*

$$(i) \quad x_i \geq y_i \quad \forall i \in S \quad \text{und} \quad \exists k \in S \quad \text{mit} \quad x_k > y_k$$

2.2 Der Kern eines kooperativen Spiels

$$(ii) \quad x(S) \leq v(S).$$

erfüllt sind.

Die Bedingung (i) besagt, dass die Auszahlung jedes Mitglieds der Koalition S bezüglich der Imputation \mathbf{x} mindestens so hoch ist, wie die Auszahlung bezüglich der Imputation \mathbf{y} . Darüber hinaus ist zumindest ein Spieler $k \in S$ durch die Imputation \mathbf{x} besser gestellt als durch \mathbf{y} . Die Bedingung (ii) hingegen gewährleistet, dass die Koalition S in der Lage ist, jedem Mitglied $i \in S$ den Wert x_i ausbezahlen, dass also die Imputation \mathbf{x} durchführbar ist. Werden einer Koalition S zwei Imputationen \mathbf{x} und \mathbf{y} angeboten, die die Bedingungen (i) und (ii) erfüllen, wird die Koalition S die Imputation \mathbf{y} nicht vorziehen.

Wenn wir die Bedingung (i) durch $x_i > y_i \quad \forall i \in S$ ersetzen, dann sagen wir, dass die Imputation \mathbf{x} die Imputation \mathbf{y} in Bezug auf die Koalition S *strikt dominiert*. Diese Eigenschaft werden wir mit $\mathbf{x} \succ_S \mathbf{y}$ ausdrücken. In kooperativen Spielen mit Seitenzahlungen ist diese Unterscheidung in schwache und strikte Dominanz irrelevant. Wegen der Möglichkeit der Seitenzahlungen zwischen den Spielern können die Auszahlungen an die Mitglieder der Koalition S so umverteilt werden, dass $x_i > y_i$ für alle $i \in S$ erfüllt wird (vgl. Holler/Illing [53], S. 271).⁶ In dieser Arbeit gehen wir im Weiteren immer von strikter Dominanz aus, wenn nichts anderes explizit angenommen wird. Vergleichen wir nun in unserem Einführungsbeispiel die Imputation $\mathbf{y} = (0, 0, 42)$ mit der Imputation $\mathbf{x} = (2, 3, 37)$, so stellen wir fest, dass die Imputation $\mathbf{y} = (0, 0, 42)$ durch die Imputation $\mathbf{x} = (2, 3, 37)$ in Bezug auf die Koalition $S = \{1, 2\}$ dominiert ist. Denn

$$(i) \quad x_1 = 2 \geq y_1 = 0, \quad x_2 = 3 \geq y_2 = 0,$$

$$(ii) \quad x(S) := x_1 + x_2 = 5 \leq v(S) = 6.$$

⁶Erfüllen zwei Imputationen \mathbf{x} und \mathbf{y} die Bedingung (i), so ist $x(S)$ mindestens um den Betrag $x_k - y_k > 0$ höher als $y(S)$. Da bei den kooperativen Spielen mit Seitenzahlungen die Mitglieder einer Koalition S innerhalb der Koalition die Auszahlungen beliebig umverteilen können, kann der Betrag $x_k - y_k$ zwischen den Spielern beliebig geteilt werden, sodass $x_i > y_i$ für alle $i \in S$ erfüllt ist.

2 Kooperative Spiele mit zwei oder mehr Spielern

Stellen wir nun die Imputation $\mathbf{x} = (2, 3, 37)$ einer anderen Imputation $\mathbf{z} = (3, 3, 36)$ gegenüber, so erkennen wir durch Überprüfung der Bedingungen (i) und (ii), dass die Imputation $\mathbf{x} = (2, 3, 37)$ durch die Imputation $\mathbf{z} = (3, 3, 36)$ bezüglich der Koalition $S = \{1, 2\}$ dominiert ist. Wenn sich also die Koalition $S = \{1, 2\}$ zwischen den beiden Imputationen \mathbf{x} und \mathbf{z} entscheiden muss, wird sie die Imputation \mathbf{z} gegenüber der Imputation \mathbf{x} vorziehen. Zur Aufteilung des Koalitionsgewinns der grossen Koalition unter den Mitgliedern der grossen Koalition könnten wir also diejenigen Imputationen, die durch mindestens eine andere Imputation dominiert sind, ausser Acht lassen und somit nur die nicht-dominierten Imputationen in Betracht ziehen:

Definition 2.2.2 (Kern) *Gegeben sei ein kooperatives Spiel $\Gamma = (N, v)$. Der Kern $\mathcal{C}(N, v)$ des kooperativen Spiels $\Gamma = (N, v)$ ist die Menge aller nicht-dominierten Imputationen.*

Der Kern ist eines der bekanntesten Konzepte zur Lösung kooperativer Spiele mit n Spielern. Er wurde von Gillies [45] in die Literatur eingeführt. Der folgende Satz verdeutlicht den Unterschied zwischen der Imputationsmenge und dem Kern eines kooperativen Spiels:

Satz 2.2.1 *Der Kern eines kooperativen Spiels $\Gamma = (N, v)$ besteht aus den Imputationen \mathbf{x} , die die folgenden Eigenschaften erfüllen:*

$$\mathcal{C}(N, v) := \{\mathbf{x} \in \mathbb{R}^n \mid x(N) = v(N), \quad x(S) \geq v(S) \quad \forall S \in 2^N\}.$$

Beweis: Vgl. Owen [97], S. 219.

Im Vergleich zur Imputationsmenge verlangt die Kernbedingung

$$x(S) \geq v(S) \quad \forall S \in 2^N,$$

dass nicht nur jeder Spieler, sondern auch jede Koalition von zwei oder mehr Spielern mindestens ihren Koalitionswert zugeteilt bekommen sollte. Zusätzlich zu den Anforderungen der Imputationsmenge verlangt also der Kern eines kooperativen Spiels die sog. *Gruppenrationalität*, die voraussetzt, dass keine Mehrspielerkoalition $S \subset N$ eine Auszahlung erhalten sollte, die geringer als der Koalitionswert $v(S)$ ist. Folglich

2.2 Der Kern eines kooperativen Spiels

besteht der Unterschied zwischen der Imputationsmenge und dem Kern darin, dass der Kern neben der Effizienz und der individuellen Rationalität einer Auszahlung \mathbf{x} zusätzlich auch ihre Gruppenrationalität verlangt. Somit ist der Kern immer eine Teilmenge der Imputationsmenge. Auf Grund der individuellen Rationalität und Gruppenrationalität der Kernelemente kann kein Spieler und keine Koalition von Spielern glaubwürdig damit drohen, die grosse Koalition zu verlassen. In diesem Sinne ist der Kern ein stabiles Lösungskonzept in der Theorie der kooperativen Spiele: Einigen sich die Spieler eines kooperativen Spiels auf einen Auszahlungsvektor im Kern des Spiels, so wird sich keiner der Spieler veranlasst fühlen, die grosse Koalition zu verlassen. Nun wollen wir den Kern des Einführungsbeispiels bestimmen. Er besteht aus allen Allokationsvektoren $\mathbf{x} = (x_1, x_2, x_3) \in \mathbb{R}^3$, die gemäss dem Satz 2.2.1 die folgenden Bedingungen erfüllen:

$$\begin{array}{rcl} x_1 + x_2 + x_3 & = & 42 \quad \text{Effizienz} \\ \left. \begin{array}{rcl} x_1 + x_2 & \geq & 6 \\ x_1 + x_3 & \geq & 16 \\ x_2 + x_3 & \geq & 26 \end{array} \right\} & & \text{Gruppenrationalität} \\ x_1, x_2, x_3 & \geq & 0. \quad \text{Individuelle Rationalität} \end{array}$$

Beispielsweise sind die Auszahlungsvektoren $(3, 3, 36)$, $(8, 6, 28)$ und $(10, 20, 12)$ Elemente des Kerns des Einführungsbeispiels, da sie die Kernbedingungen erfüllen. Geometrisch können wir den Kern des Einführungsbeispiels im zweidimensionalen Simplex

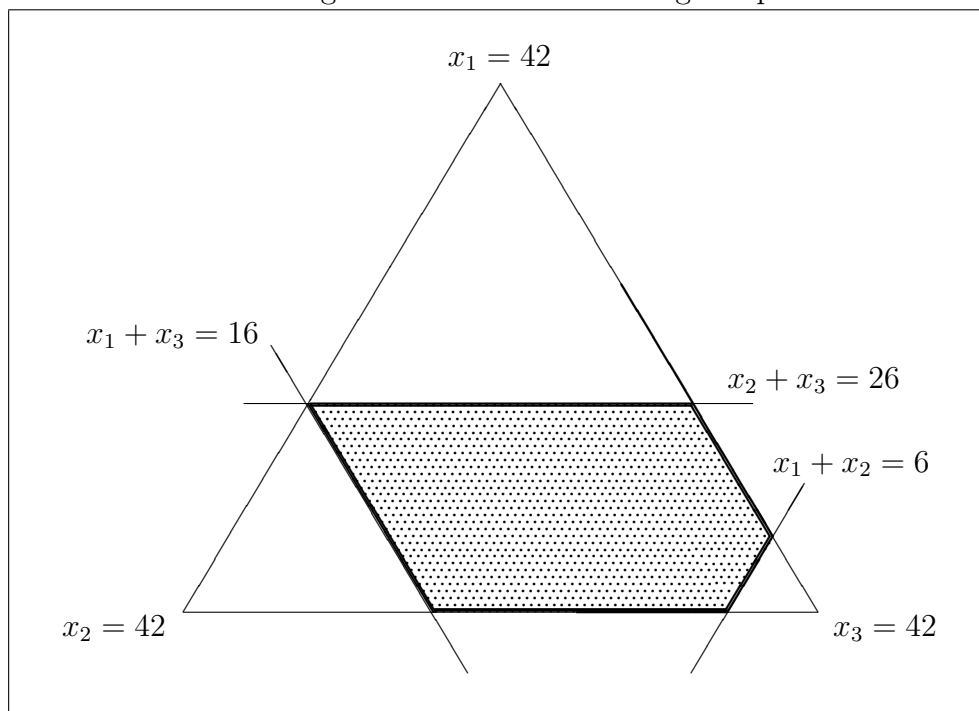
$$s^{(3)} := \{x_1, x_2, x_3 \geq 0 \mid x_1 + x_2 + x_3 = 42\}$$

darstellen (vgl. Abbildung 2.1). In der Abbildung 2.1 entspricht jeder Punkt des Simplex genau einem Auszahlungsvektor, der den Wert 42 der grossen Koalition auf die Spieler verteilt. Die Ecken des Dreiecks repräsentieren Auszahlungsvektoren, die jeweils einem Spieler den gesamten Koalitionsgewinn von 42 Geldeinheiten zuteilen. Die Punkte auf einer Kante des Dreiecks repräsentieren diejenigen Auszahlungsvektoren, bei denen jeweils einem Spieler nichts zugeteilt wird. Jeder Punkt innerhalb des Dreiecks entspricht einem Allokationsvektor, bei dem jeder Spieler einen posi-

2 Kooperative Spiele mit zwei oder mehr Spielern

tiven Betrag erhält. Während die Imputationsmenge des Einführungsbeispiels aus allen Auszahlungsvektoren des Dreiecks besteht, wird der Kern durch die gepunktete Fläche illustriert.

Abbildung 2.1: Kern des Einführungsbeispiels



Der Kern eines kooperativen Spiels beinhaltet, wie das Einführungsbeispiel darlegt, potentiell mehrere Allokationsvektoren. Es kann jedoch vorkommen, dass der Kern eines kooperativen Spiels leer ist. Um dies zu zeigen, modifizieren wir das Einführungsbeispiel, indem wir den Wert der grossen Koalition von 42 auf 28 reduzieren und den Wert der Koalition $\{1, 2\}$ von 6 auf 16 erhöhen (vgl. Tabelle 2.3).

Tabelle 2.3: Modifizierung des Einführungsbeispiels

Koalition S	$\{1\}$	$\{2\}$	$\{3\}$	$\{1, 2\}$	$\{1, 3\}$	$\{2, 3\}$	$\{1, 2, 3\}$
$v(S)$	0	0	0	16	16	26	28

2.2 Der Kern eines kooperativen Spiels

Wir bemerken, dass dieses kooperative Spiel superadditiv ist, da es die Bedingung (2.2) (vgl. S. 18) erfüllt. Hingegen ist das Spiel nicht-konvex, da z. B. für die Koalitionen $S = \{1, 2\}$ und $T = \{1, 3\}$ die Konvexitätsbedingung (2.4) (vgl. S. 19) verletzt ist:

$$\begin{aligned} v(\{1, 2\}) + v(\{1, 3\}) &\not\geq v(\{1, 2\} \cup \{1, 3\}) + v(\{1, 2\} \cap \{1, 3\}), \\ 16 + 16 &\not\geq 28 + 0. \end{aligned}$$

Um zu überprüfen, ob der Kern des modifizierten Einführungsbeispiels leer ist, schreiben wir zunächst die Kernbedingungen auf:

$$\begin{array}{ll} x_1 + x_2 + x_3 = 28 & \text{Effizienz} \\ \left. \begin{array}{l} x_1 + x_2 \geq 16 \\ x_1 + x_3 \geq 16 \\ x_2 + x_3 \geq 26 \end{array} \right\} & \text{Gruppenrationalität} \\ x_1, x_2, x_3 \geq 0. & \text{Individuelle Rationalität} \end{array}$$

Durch Addition der Bedingungen der Gruppenrationalität und anschließende Division durch 2 erhalten wir

$$x_1 + x_2 + x_3 \geq 29. \quad (2.11)$$

Die Anforderung (2.11) verletzt die Effizienzeigenschaft. Folglich ist der Kern des Spiels leer.

Wie wir bereits festgestellt haben, kann der Kern eines Spiels mehrere Auszahlungsvektoren beinhalten oder auch leer sein. Es ist ebenfalls möglich, dass der Kern eines kooperativen Spiels nur aus einem einzigen Auszahlungsvektor besteht. Um dies zu demonstrieren, betrachten wir das folgende kooperative Spiel mit drei Spielern:

Beispiel 2.2.1 Die charakteristische Funktion des Spiels sei gegeben durch

$$\begin{aligned} v(\{1\}) &= 0, & v(\{2\}) &= 0, & v(\{3\}) &= 0, \\ v(\{1, 2\}) &= 0, & v(\{1, 3\}) &= 1, & v(\{2, 3\}) &= 1, \\ v(\{1, 2, 3\}) &= 1. \end{aligned}$$

2 Kooperative Spiele mit zwei oder mehr Spielern

In diesem Spiel ist der Wert jeder Koalition $S \subseteq N$ entweder Null oder Eins. Kooperative Spiele, deren charakteristische Funktionen diese Eigenschaft besitzen, nennt man *einfache kooperative Spiele*.⁷ Zudem weist das angegebene Beispiel die Besonderheit auf, dass der Wert einer beliebigen Koalition nur dann Eins ist, wenn der Spieler 3 ein Mitglied dieser Koalition ist. Daher nennt man den Spieler 3 *Vetospieler*.⁸ Jeder Allokationsvektor, der im Kern des Spiels liegt, muss die folgenden Kernbedingungen erfüllen:

$$x_1 + x_2 + x_3 = 1 \quad (2.12)$$

$$x_1 + x_2 \geq 0 \quad (2.13)$$

$$x_1 + x_3 \geq 1 \quad (2.14)$$

$$x_2 + x_3 \geq 1 \quad (2.15)$$

$$x_1, x_2, x_3 \geq 0. \quad (2.16)$$

Wegen der Beschränkungen (2.12) und (2.16) müssen die Ungleichungen (2.14) und (2.15) als Gleichung gelten. Weiterhin ist die Anforderung (2.13) aufgrund der Bedingung (2.16) überflüssig und kann vernachlässigt werden. Somit reduzieren sich die Kernbedingungen auf das folgende Gleichungssystem:

$$x_1 + x_2 + x_3 = 1 \quad (2.17)$$

$$x_1 + x_3 = 1 \quad (2.18)$$

$$x_2 + x_3 = 1 \quad (2.19)$$

$$x_1, x_2, x_3 \geq 0. \quad (2.20)$$

Lösen wir die Gleichungen (2.18) und (2.19) nach x_1 bzw. nach x_2 auf und setzen in die Gleichung (2.17) ein, erhalten wir $x_3 = 1$. Anschliessend gewinnen wir anhand der

⁷Wie wir später sehen werden, sind *einfache kooperative Spiele* für uns von besonderer Bedeutung.

⁸Allgemein heisst in einem einfachen kooperativen Spiel ein Spieler $i \in N$ *Vetospieler*, falls $v(N \setminus \{i\}) = 0$ gilt.

2.2 Der Kern eines kooperativen Spiels

Gleichung (2.18) bzw. (2.19) die Werte $x_1 = 0$ und $x_2 = 0$. Der Kern des Spiels besitzt also einen einzigen Allokationsvektor, nämlich den Vektor $\mathbf{x} = (x_1, x_2, x_3) = (0, 0, 1)$.⁹

□

Enthält der Kern eines kooperativen Spiels einen einzigen Allokationsvektor, so kann man diesen als Lösung des Spiels betrachten. Falls der Kern hingegen mehrere Elemente beinhaltet, stellt sich die Frage, welcher Allokationsvektor als Lösung zustande kommen wird. Ist umgekehrt der Kern eines kooperativen Spiels leer, so kann man keinen Allokationsvektor finden, der effizient, individuell rational und gruppenrational ist. Auf Grund des Tatbestands, dass der Kern leer sein kann und ein nicht-leerer Kern in der Regel mehrere Auszahlungsvektoren beinhaltet, sind in der Literatur Verallgemeinerungen des Lösungskonzepts des Kerns zu finden. Eine davon stellt der sog. ε -Kern dar, welcher von Shapley/Shubik [122] in die Literatur eingeführt worden ist:

Definition 2.2.3 (ε -Kern) *Es sei $\Gamma = (N, v)$ ein kooperatives Spiel mit Seitenzahlungen und ε eine reellwertige Zahl. Dann heisst*

$$\mathcal{C}_\varepsilon(N, v) := \{\mathbf{x} \in \mathcal{PI}(N, v) \mid x(S) \geq v(S) - \varepsilon \text{ für alle } S \subset N, S \neq \emptyset\}$$

der ε -Kern des Spiels.

Aus der Definition folgt, dass für $\varepsilon = 0$ der ε -Kern und der Kern des Spiels übereinstimmen. Den ε -Kern kann man wie folgt interpretieren: Ist der Kern eines kooperativen Spiels leer, so gibt es mindestens eine Koalition $S \in N$, für die die Bedingung $x(S) \geq v(S)$ für alle Preimputationen $\mathbf{x} \in \mathcal{PI}(N, v)$ verletzt ist. In diesem Fall kann man nach den Preimputationen fragen, welche die um eine reellwertige positive Zahl ε gelockerten Bedingungen für die individuelle Rationalität und die Gruppenrationalität einer Spiellösung erfüllen. Schreiben wir die Ungleichungen in der Definition 2.2.3 des ε -Kerns in der Form

$$x(S) + \varepsilon \geq v(S) \quad \text{für alle } S \subset N, S \neq \emptyset,$$

⁹Man kann zeigen, dass ein einfaches kooperatives Spiel nur dann einen nicht-leeren Kern besitzt, wenn es in diesem Spiel einen Vetospieler gibt. Für den Beweis dieser Aussage vgl. z. B. Fargó et al. [38], S. 231.

2 Kooperative Spiele mit zwei oder mehr Spielern

so können wir die reellwertige Zahl ε als finanzielle Unterstützung oder Subvention an alle Koalitionen $S \subset N$, $S \neq \emptyset$, interpretieren, sodass diese Subvention die Existenz des Kerns des derart modifizierten Spiels gewährleistet (vgl. Shubik [127], S. 86, und [126], S. 154). Wählen wir beispielsweise für unser modifiziertes Einführungsbeispiel (vgl. Tabelle 2.3, S. 30) $\varepsilon = 1$, so erhalten wir die folgenden ε -Kernbedingungen

$$\begin{aligned}x_1 + x_2 + x_3 &= 28 \\x_1 + x_2 &\geq 16 - 1 = 15 \\x_1 + x_3 &\geq 16 - 1 = 15 \\x_2 + x_3 &\geq 26 - 1 = 25 \\x_1, x_2, x_3 &\geq 0 - 1 = -1,\end{aligned}$$

die z. B. für die Imputationen $(3, 12, 13)$ und $(2, 13, 13)$ erfüllt sind. Ist hingegen der Kern eines kooperativen Spiels nicht-leer und enthält mehrere Imputationen, so kann man die Bedingungen der individuellen Rationalität und der Gruppenrationalität um eine reellwertige negative Zahl ε verschärfen, um eine Auswahl aus den Imputationen im Kern des Spiels zu treffen. Wie wir in den nächsten Kapiteln sehen werden, ist z. B. der Nukleolus in der Theorie der kooperativen Spiele ein Lösungskonzept, das durch Verschärfung bzw. Lockerung der Kernbedingungen zu einer eindeutigen Spiellösung führt.

Die Definition des ε -Kerns beinhaltet, dass dieser nicht für jede beliebig gewählte reellwertige Zahl ε existiert:

Definition 2.2.4 (Kleinst-Kern) *Es sei $\Gamma = (N, v)$ ein kooperatives Spiel und $\varepsilon_0(N, v)$ der kleinste Wert, den ε annehmen kann, so dass der ε -Kern nicht-leer ist. Der Kleinst-Kern (engl. least core) $\mathcal{LC}(N, v)$ des Spiels ist dann die Menge $C_{\varepsilon_0}(N, v)$.*

Der Kleinst-Kern hat die Eigenschaft, dass er nie leer ist und stets eine Teilmenge des Kerns ist, falls dieser nicht-leer ist. Diese Charakteristika folgen direkt aus der Definition 2.2.4 des Kleinst-Kerns.

2.2 Der Kern eines kooperativen Spiels

Nun wollen wir zeigen, wie man den Wert $\varepsilon_0(N, v)$ bestimmen kann. Aus der Definition des ε -Kerns folgt, dass dieser aus den Auszahlungsvektoren \mathbf{x} besteht, die für alle Koalitionen $S \subset N$, $S \neq \emptyset$ die Bedingungen

$$\varepsilon \geq v(S) - x(S)$$

erfüllen. Definieren wir nun

$$\varepsilon_{\max}(\mathbf{x}) := \max_{S \neq \emptyset, N} \{v(S) - x(S)\},$$

so muss $\varepsilon_0(N, v)$ gerade dem kleinstmöglichen Wert von $\varepsilon_{\max}(\mathbf{x})$ entsprechen. Folglich kann der Wert $\varepsilon_0(N, v)$ wie folgt ausgedrückt werden (Maschler et al. [82], S. 306).

$$\varepsilon_0(N, v) := \min_{\mathbf{x} \in \mathcal{PI}(N, v)} \max_{S \neq \emptyset, N} [v(S) - x(S)]. \quad (2.21)$$

Hierbei nennt man den Ausdruck

$$e(S, \mathbf{x}) := v(S) - x(S) = v(S) - \sum_{i \in S} x_i \quad (2.22)$$

Überschuss der Koalition $S \in 2^N$ bezüglich eines Auszahlungsvektors $\mathbf{x} \in \mathcal{I}(N, v)$. Ist eine Imputation \mathbf{x} ein Element des Kerns, so folgt aus der Definition des Kerns, dass der Überschuss jeder Koalition $S \subset N$ bezüglich der Imputation \mathbf{x} nicht-positiv ist, also $e(S, \mathbf{x}) \leq 0$ für alle Koalitionen $S \subset N$ gilt. Falls die entsprechenden Spieler die Auszahlung \mathbf{x} verwerfen und stattdessen sich in der Koalition S zusammenfinden, so gibt der Koalitionsüberschuss den Gewinn (bei negativem Vorzeichen den Verlust) an, den eine Koalition S realisiert (vgl. Maschler et al. [82], S. 305). Der Koalitionsüberschuss kann als ein „Mass der Unzufriedenheit“ einer Koalition interpretiert werden. Denn $v(S)$ ist der Wert der Koalition, den sich die Koalition S alleine sichern kann. Führt ein Auszahlungsvektor $\mathbf{x} = (x_1, x_2, \dots, x_n)$ zu $e(S, \mathbf{x}) < 0$, dann bedeutet das, dass die Koalition S mit dem Auszahlungsvektor \mathbf{x} besser gestellt ist als mit ihrem eigenen Koalitionswert $v(S)$. Umgekehrt ist sie schlechter gestellt, falls $e(S, \mathbf{x}) > 0$ ist. Demzufolge ist die Koalition S mit dem Auszahlungsvektor \mathbf{x} umso weniger zufrieden, je grösser der Koalitionsüberschuss $e(S, \mathbf{x})$ ist. Somit ist der Kleinst-Kern die Menge aller Imputationen, die den maximalen Koalitionsüberschuss minimieren.

2.3 Balancierte Koalitionssysteme und balancierte Spiele

Wir haben oben festgestellt, dass der Kern eines kooperativen Spiels leer sein kann. Ist der Kern eines Spiels leer, kann kein Auszahlungsvektor gefunden werden, der die Kernbedingungen für alle Koalitionen $S \subset N$ erfüllt. In diesem Fall ist also zumindest für eine Koalition $S \subset N$ wenigstens eine der Kernbedingungen verletzt. Ist z. B. für eine Koalition $S \subset N$ die Gruppenrationalität verletzt, so besteht für die Koalition S kein Anreiz, sich der grossen Koalition N anzuschliessen. Daher ist die Existenz des Kerns eines kooperativen Spiels von grosser Bedeutung. In diesem Zusammenhang stellt sich die Frage, unter welchen Bedingungen der Kern eines kooperativen Spiels nicht-leer ist. Shapley [123] und Bondavera [12]¹⁰ geben eine notwendige und hinreichende Bedingung an, anhand derer wir die Existenz des Kerns eines kooperativen Spiels überprüfen können. Hierzu verwenden Shapley und Bondavera das Konzept der sog. *balancierten Spiele*. Zunächst führen wir einige Begriffe ein, die für die Definition der balancierten Spiele zweckmässig sind.

Definition 2.3.1 *Ein $(0, 1)$ -Vektor $\mathbf{1}_S := (1_S(1), 1_S(2), \dots, 1_S(n))$ heisst Inzidenzvektor der Koalition $S \subset N$, falls*

$$1_S(i) := \begin{cases} 1 & : \text{für } i \in S, \\ 0 & : \text{für } i \in N \setminus S. \end{cases}$$

Ist zum Beispiel $N = \{1, 2, 3, 4\}$ und $S = \{1, 3, 4\}$, so gilt $\mathbf{1}_S = (1, 0, 1, 1)$. Man kann eine Koalition $S \subset N$ auch durch ihren Inzidenzvektor repräsentieren. Ist also der Inzidenzvektor einer Koalition $S \subset N$ gegeben, kann aus dem Inzidenzvektor die dazu gehörige Koalition bestimmt werden. Beispielsweise zeigt $\mathbf{1}_S = (1, 0, 0, 1)$ die Koalition $S = \{1, 4\}$ an.

Definition 2.3.2 (balanciertes Koalitionssystem) *Eine Menge von nicht-leeren Koalitionen wird als Koalitionssystem bezeichnet. Ein Koalitionssystem $\mathcal{B} = \{S_1, S_2,$*

¹⁰Die Literaturstelle Bondavera [12] ist zitiert nach Moulin [86], S. 96.

2.3 Balancierte Koalitionssysteme und balancierte Spiele

$\dots, S_m\} \subset 2^N \setminus \emptyset$ heisst *balanciert (ausgewogen)*, wenn es für jeden Spieler $i \in N$ reellwertige Koeffizienten $0 \leq \alpha_j \leq 1$, $j = 1, 2, \dots, m$, gibt, so dass gilt:

$$\sum_{j=1}^m \alpha_j 1_{S_j}(i) = 1 \quad \text{für alle } S_j \in \mathcal{B}.^{11} \quad (2.23)$$

Um ein balanciertes Koalitionssystem zu interpretieren,¹² nehmen wir an, dass jeder Spieler $i \in N$ über bestimmte Ressourcen, z. B. Arbeitszeit, verfügt, die er beliebig verwenden kann. Insbesondere kann er mit unterschiedlichen Koalitionen zusammenarbeiten, wobei jede Koalition S_j , der der Spieler i angehört, den Anteil α_j seiner Ressourcen in Anspruch nimmt. Besitzen diese Anteile α_j , $0 \leq \alpha_j \leq 1$, $j = 1, 2, \dots, m$, die Eigenschaft (2.23), so bedeutet dies, dass jeder Spieler $i \in N$ seine Gesamtressourcen ohne Abstriche auf die Koalitionen $S \in \mathcal{B}$ verteilt. Daher bezeichnet man ein balanciertes Koalitionssystem \mathcal{B} auch als eine *ausgewogene* Gruppe von Koalitionen. Wie wir am Beispiel 2.3.1 zeigen werden, könnte man hingegen ein nicht-balanciertes Koalitionssystem so interpretieren, dass mindestens ein Spieler i nicht alle ihm zur Verfügung stehenden Ressourcen gebraucht. Dem Spieler i verbleibt also ein Teil seiner Ressourcen. Das würde bedeuten, dass dem Spieler i die Verwendung dieses Rests keinen zusätzlichen Nutzen stiftet. Wäre dies nicht der Fall, würde er z. B. durch Bildung der trivialen Koalition $\{i\}$ seine übriggebliebenen Ressourcen verwendet.¹³

Eine andere in der Literatur übliche Schreibweise für die Gleichung (2.23) lautet

$$\sum_{S: i \in S} \alpha_S = 1 \quad \text{für alle } S \in \mathcal{B}. \quad (2.24)$$

Hierbei wird die Summation für alle Koalitionen S ausgeführt, die den Spieler i enthalten. Die Koeffizienten $\alpha_1, \alpha_2, \dots, \alpha_m$, (bzw. α_S , $\forall S \in \mathcal{B}$) heissen *Gewichte* der Koalitionen S_j , $j = 1, 2, \dots, m$. Man sagt, sie balancieren das Koalitionssystem \mathcal{B} aus (Fargó et al. [38], S. 227). Wir werden im Weiteren den Vektor $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_m)$

¹¹In der Literatur spricht man mitunter von *schwach* balancierten Koalitionssystemen, wenn für die Koeffizienten α_j , $j = 1, 2, \dots, m$, der Wert Null zulässig ist. Vgl. z. B. Maschler [81], S. 612.

¹²Vgl. dazu Shubik [126], S. 165, Moulin [87], S. 414, und Rosenmüller [112], S. 188.

¹³Hierbei nehmen wir an, dass sich jeder Spieler $i \in N$ entscheiden kann, ob er und mit welchen der anderen Spieler er gegebenenfalls kooperiert.

2 Kooperative Spiele mit zwei oder mehr Spielern

als *Gewichtsvektor* des jeweiligen Koalitionssystems und α_S als das Gewicht der Koalition $S \in \mathcal{B}$ bezeichnen.

Beispiel 2.3.1 Es sei $N = \{1, 2, 3, 4\}$. In diesem Fall ist das Koalitionssystem

$$\mathcal{B} = \{\{1, 2\}, \{1, 3\}, \{1, 4\}, \{2, 3, 4\}, \{3, 4\}\}$$

balanciert für den Gewichtsvektor $\alpha = (2/4, 1/4, 1/4, 2/4, 1/4)$. Für den Beweis dieser Behauptung prüfen wir, ob für alle Spieler $i = 1, 2, 3, 4$ die Bedingung (2.23) erfüllt ist:

- Spieler 1: $\sum_{j=1}^5 \alpha_j 1_{S_j}(1) = \frac{2}{4}1 + \frac{1}{4}1 + \frac{1}{4}1 + \frac{2}{4}0 + \frac{1}{4}0 = 1, \checkmark$
- Spieler 2: $\sum_{j=1}^5 \alpha_j 1_{S_j}(2) = \frac{2}{4}1 + \frac{1}{4}0 + \frac{1}{4}0 + \frac{2}{4}1 + \frac{1}{4}0 = 1, \checkmark$
- Spieler 3: $\sum_{j=1}^5 \alpha_j 1_{S_j}(3) = \frac{2}{4}0 + \frac{1}{4}1 + \frac{1}{4}0 + \frac{2}{4}1 + \frac{1}{4}1 = 1, \checkmark$
- Spieler 4: $\sum_{j=1}^5 \alpha_j 1_{S_j}(4) = \frac{2}{4}0 + \frac{1}{4}0 + \frac{1}{4}1 + \frac{2}{4}1 + \frac{1}{4}1 = 1, \checkmark$

Folglich ist das Koalitionssystem \mathcal{B} für den angegebenen Gewichtsvektor α balanciert. Um die Interpretation eines nicht-balancierten Koalitionssystems zu verdeutlichen (vgl. S. 37), nehmen wir an, dass $\sum_{j=1}^5 \alpha_j 1_{S_j}(1) = 0.8 \neq 1$ gilt, also die Bedingung (2.23) für den Spieler 1 verletzt ist. In diesem Fall könnte der Spieler 1 die Einerkoalition $\{1\}$ mit dem Gewicht 0.2 gründen, so dass nun für das Koalitionssystem $\mathcal{B} \cup \{1\}$ die Gleichung (2.23) erfüllt ist.

Ist ein balanciertes Koalitionssystem \mathcal{B} eine Aufspaltung¹⁴ von N , so ist jeder Spieler $i \in N$ Mitglied nur einer einzigen Koalition. Damit in diesem Fall das Koalitionssystem \mathcal{B} balanciert ist, muss das Gewicht jeder Koalition eins sein. Für $N = \{1, 2, 3, 4\}$ ist beispielsweise das Koalitionssystem $\mathcal{B} = \{\{1\}, \{2, 3\}, \{4\}\}$ eine Aufspaltung von N . Analog wie im obigen Beispiel kann man zeigen, dass für den

¹⁴Ein Koalitionssystem $\mathcal{B} = \{S_1, \dots, S_m\}$ ist eine Aufspaltung (auch Partition genannt) der Menge N , falls $\cup_{j=1}^m S_j = N$, mit $S_j \cap S_k = \emptyset$ für alle $1 \leq j, k \leq m, j \neq k$.

2.3 Balancierte Koalitionssysteme und balancierte Spiele

Gewichtsvektor $\alpha = (1, 1, 1)$ das Koalitionssystem $\mathcal{B} = \{\{1\}, \{2, 3\}, \{4\}\}$ balanciert ist.

Entsprechend zeigt man, dass das Koalitionssystem im Beispiel 2.3.1 für den Gewichtsvektor $\alpha = (3/5, 1/5, 1/5, 2/5, 2/5)$ ebenfalls balanciert ist. Ein balanciertes Koalitionssystem kann also für mehrere Gewichtsvektoren α balanciert sein. Man kann zeigen,¹⁵ dass ein balanciertes Koalitionssystem \mathcal{B} dann und nur dann einen eindeutigen Gewichtsvektor α besitzt, wenn keine echte Teilmenge von \mathcal{B} balanciert ist. In diesem Zusammenhang spricht man in der Literatur von *minimal balancierten Koalitionssystemen*:

Definition 2.3.3 (minimal balanciertes Koalitionssystem) *Wenn in einem balancierten Koalitionssystem keine echte Teilmenge balanciert ist, nennen wir es minimal balanciertes System.*

Beispiel 2.3.2 *Das Koalitionssystem $\mathcal{B} = \{\{1, 2\}, \{1, 3\}, \{2\}, \{3\}\}$ ist balanciert für $\alpha = (1/2, 1/2, 1/2, 1/2)$, aber ist kein minimal balanciertes System. Denn die Teilmenge $\mathcal{T} = \{\{1, 2\}, \{2\}, \{3\}\} \subset \mathcal{B}$ ist ein balanciertes Koalitionssystem für $\alpha = (1, 0, 1)$.*

Beispiel 2.3.3 (Owen [97], S. 230) *Für $N = \{1, 2, 3, 4\}$ sind die unten angegebenen Koalitionssysteme minimal balanciert:*

- $\mathcal{B}_1 = \{\{1, 2, 3\}, \{1, 2, 4\}, \{1, 3, 4\}, \{2, 3, 4\}\}, \quad \text{für } \alpha = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}),$
- $\mathcal{B}_2 = \{\{1, 2\}, \{1, 3\}, \{1, 4\}, \{2, 3, 4\}\}, \quad \text{für } \alpha = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{2}{3}),$
- $\mathcal{B}_3 = \{\{1, 2\}, \{1, 3\}, \{2, 3\}, \{4\}\}, \quad \text{für } \alpha = (\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 1),$
- $\mathcal{B}_4 = \{\{1, 2\}, \{1, 3, 4\}, \{2, 3, 4\}\}, \quad \text{für } \alpha = (\frac{1}{2}, \frac{1}{2}, \frac{1}{2}).$

Wir zeigen exemplarisch, dass keine echte Teilmenge von \mathcal{B}_4 balanciert ist. Die Teilmenge $\mathcal{T} = \{\{1, 2\}\}$ enthält die beiden Spieler 3 und 4 nicht. Deswegen kann

¹⁵Für den Beweis vgl. Owen [97], S. 228.

2 Kooperative Spiele mit zwei oder mehr Spielern

kein Gewichtsvektor die Bedingung (2.23) für diese Spieler erfüllen. Dasselbe Argument gilt auch für die Teilmenge $\mathcal{T} = \{\{1, 3, 4\}\}$ bezüglich des Spielers 2 und für die Teilmenge $\mathcal{T} = \{\{2, 3, 4\}\}$ bezüglich des Spielers 1. Nun überprüfen wir die Teilmenge $\mathcal{T} = \{\{1, 2\}, \{1, 3, 4\}\}$. Damit auf der einen Seite die Bedingung (2.23) für den Spieler 2 erfüllt ist, müsste das Gewicht der Koalition $\{1, 2\}$ eins sein, weil der Spieler 2 nur dieser Koalition angehört. Auf der anderen Seite müsste das Gewicht der Koalition $\{1, 3, 4\}$ ebenfalls eins sein, da die Spieler 3 und 4 nur zu dieser Koalition zählen. Dies würde aber dazu führen, dass die Bedingung (2.23) für den Spieler 1 nicht erfüllt werden kann, weil er Mitglied beider Koalitionen ist. Mit Hilfe derselben Überlegung kann man zeigen, dass sowohl $\mathcal{T} = \{\{1, 2\}, \{2, 3, 4\}\}$ als auch $\mathcal{T} = \{\{1, 3, 4\}, \{2, 3, 4\}\}$ für keinen Gewichtsvektor balanciert sein kann. Wir haben also gezeigt, dass keine der Teilmengen von \mathcal{B}_4 balanciert ist und somit das Koalitionssystem \mathcal{B}_4 minimal balanciert ist. \square

Nun können wir mit Hilfe von balancierten Koalitionssystemen das Konzept der balancierten Spiele einführen:

Definition 2.3.4 (balancierte Spiele) *Ein kooperatives Spiel $\Gamma = (N, v)$ heisst balanciert, wenn für jedes balancierte System $\mathcal{B} \subset 2^N \setminus \emptyset$ und für die zugehörigen Koeffizienten gilt, dass*

$$\sum_{j=1}^m \alpha_j v(S_j) \leq v(N) \quad (2.25)$$

oder äquivalent

$$\sum_{S \in \mathcal{B}} \alpha_S v(S) \leq v(N), \quad (2.26)$$

wobei $\sum_{S: i \in S} \alpha_S = 1$ für alle $i \in N$.

Verbal heisst dies, dass jede Koalition $S_j \in \mathcal{B}$, die mit der Intensität α_j , $0 \leq \alpha_j \leq 1$, betrieben wird, im Vergleich zu $v(S_j)$ den verminderten Nutzen $\alpha_j v(S_j)$ stiftet.¹⁶

¹⁶Wie wir bei der Interpretation der balancierten Koalitionssysteme erläutert haben, bedeutet das Gewicht α_j einer Koalition S_j , dass jedes Mitglied dieser Koalition nur den Anteil α_j seiner gesamten Ressourcen für diese Koalition verwenden muss. Als Konsequenz stiftet jede Koalition $S_j \in \mathcal{B}$ dem balancierten Koalitionssystem \mathcal{B} den verminderten Nutzen $\alpha_j v(S_j)$, $0 \leq \alpha_j \leq 1$. Vgl. dazu Osborne/Rubinstein [95], S. 262.

2.3 Balancierte Koalitionssysteme und balancierte Spiele

Wenn wir alle diese verminderten Nutzen der Koalitionen S_j addieren, dann erhalten wir auf der linken Seite der Ungleichungen (2.25) bzw. (2.26) den Gesamtnutzen des balancierten Systems \mathcal{B} . Die rechte Seite der Ungleichungen (2.25) bzw. (2.26) besagt dann, dass es in einem balancierten Spiel für alle Koalitionen von Vorteil ist, in die grosse Koalition N einzutreten. Also besagt die Bedingung (2.25) bzw. (2.26), dass in einem balancierten Spiel für alle Gewichtsvektoren α die Werte $v(S)$ aller Koalitionen $S \subset N$ in Bezug auf den Wert der grossen Koalition $v(N)$ relativ klein sind (vgl. Moulin [86], S. 96, und Rosenmüller [112], S. 188).

Beispiel 2.3.4 (vgl. Jianhua [56], S. 148) *Wir betrachten das folgende kooperative Spiel*

$$\begin{aligned} v(\{1\}) &= 0, \quad v(\{2\}) = 0, \quad v(\{3\}) = 0 \\ v(\{1, 2\}) &= \frac{1}{3}, \quad v(\{1, 3\}) = \frac{1}{6}, \quad v(\{2, 3\}) = \frac{5}{6} \\ v(\{1, 2, 3\}) &= 1. \end{aligned}$$

Dieses kooperative Spiel ist ein balanciertes Spiel. Zur Verdeutlichung betrachten wir exemplarisch das System $\mathcal{B} = \{\{1, 2\}, \{1, 3\}, \{2\}, \{3\}\}$, welches für den Gewichtsvektor $\alpha = (3/5, 2/5, 2/5, 3/5)$ balanciert ist. Wir prüfen, ob das Koalitionssystem \mathcal{B} die Bedingung (2.25) erfüllt:

$$\begin{aligned} \sum_{j=1}^4 \alpha_j v(S_j) &= \alpha_1 v(S_1) + \alpha_2 v(S_2) + \alpha_3 v(S_3) + \alpha_4 v(S_4) \\ &= \frac{3}{5} \frac{1}{3} + \frac{2}{5} \frac{1}{6} + \frac{2}{5} 0 + \frac{3}{5} 0 \\ &= \frac{4}{15} \leq v(N) = 1. \quad \checkmark \end{aligned}$$

Mit Hilfe des Konzepts der balancierten Spiele geben Shapley [123] und Bondavera [12] eine notwendige und hinreichende Bedingung für die Existenz des Kerns eines kooperativen Spiels an. Um diese Bedingung darzulegen, nehmen wir an, dass $\Gamma = (N, v)$ ein kooperatives Spiel ist, dessen Kern nicht-leer ist. Wir zeigen zunächst,

2 Kooperative Spiele mit zwei oder mehr Spielern

dass der Kern eines Spiels dann und nur dann nicht-leer ist, wenn die Zielfunktion des linearen Programms (vgl. Owen [97], S. 224 f.)¹⁷

$$\begin{aligned}
 (\text{LP}_C) : \quad & \min z := \sum_{i=1}^n x_i \\
 & \text{unter den Nebenbedingungen} \\
 & \sum_{i \in S} x_i \geq v(S) \text{ für alle } S \subset N, \text{ Dualvariable } =: \delta_S \quad (2.27)
 \end{aligned}$$

im Optimum die Bedingung $z^* \leq v(N)$ erfüllt. Denn wenn $z^* \leq v(N)$ gilt, dann ist jede optimale Lösung $\mathbf{x}^* = (x_1, x_2, \dots, x_n)$ ein Element des Kerns. Falls umgekehrt $\mathbf{x} \in C(N, v)$ ist, erfüllt \mathbf{x} die Nebenbedingungen (2.27), wobei auf Grund der Definition des Kerns zusätzlich $\sum_{i \in N} x_i = v(N)$ gelten muss. Somit muss im Optimum $z^* \leq v(N)$ gelten. Das zu (LP_C) duale lineare Programm lautet wie folgt:

$$\begin{aligned}
 (\text{DLP}_C) : \quad & \max q := \sum_{S \subset N} \delta_S v(S) \\
 & \text{unter den Nebenbedingungen} \\
 & \sum_{\substack{S \subset N \\ i \in S}} \delta_S = 1, \quad i = 1, 2, \dots, n \quad (2.28)
 \end{aligned}$$

$$\delta_S \geq 0, \quad \forall S \subset N. \quad (2.29)$$

Da die Bedingungen (2.28) und (2.29) genau diejenigen Anforderungen sind, die von einem Gewichtsvektor eines balancierten Koalitionssystems \mathcal{B} erfüllt werden müssen, können die Variablen $\delta_i, i = 1, 2, \dots, n$, als Koeffizienten eines balancierten Koalitionssystems interpretiert werden.¹⁸ Es sei q^* der optimale Wert der Zielfunktion von (DLP_C) . Gemäss dem starken Dualitätssatz¹⁹ muss $z^* = q^*$ gelten. Somit ist der Kern eines Spiels $\Gamma = (N, v)$ dann und nur dann nicht-leer, wenn $q^* \leq v(N)$. Dieses Resultat ist gleichzeitig der Beweis des folgenden Satzes:

¹⁷Für die Darstellung linearer primaler und dualer Programme vgl. Anhang A.1.

¹⁸Demnach hat das lineare Programm (DLP_C) eine eindeutige Lösung, wenn \mathcal{B} ein minimal balanciertes Koalitionssystem ist. Denn die minimal balancierten Koalitionssysteme besitzen einen eindeutigen Gewichtsvektor α . Vgl. dazu S. 39.

¹⁹Für die Definition des starken Dualitätssatzes vgl. im Anhang S. 287.

2.4 Der Nukleolus: Ein Konzept zur Lösung kooperativer Spiele

Satz 2.3.1 (Bondavera-Shapley-Satz, schwache Form) *Notwendig und hinreichend für einen nicht-leeren Kern eines kooperativen Spiels $\Gamma = (N, v)$ ist, dass das Spiel balanciert ist.*

Folgendes Theorem ist äquivalent zu dem Satz 2.3.1. Dieses Theorem geht nur von minimal balancierten Koalitionssystemen aus:

Satz 2.3.2 (Bondavera-Shapley-Satz, starke Form) *Notwendig und hinreichend für einen nicht-leeren Kern eines kooperativen Spiels $\Gamma = (N, v)$ ist, dass für jedes minimal balancierte Koalitionssystem $\mathcal{B} = \{S_1, S_2, \dots, S_m\} \subset 2^N \setminus \emptyset$ mit dem Gewichtsvektor $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_m)$ die Ungleichung*

$$\sum_{S \in \mathcal{B}} \alpha_S v(S) \leq v(N) \quad (2.30)$$

erfüllt ist.

Beweis: Vgl. Owen [97], S. 230, und Peleg/Sudhölter [102], S. 40–41.

Der Satz 2.3.2 gibt uns ein Ungleichungssystem an, welches erfüllt werden muss, damit der Kern eines kooperativen Spiels $\Gamma = (N, v)$ nicht-leer ist.

Beispielsweise können wir mit Hilfe des Satzes 2.3.2 zeigen, dass das modifizierte Einführungsbeispiel (vgl. Tabelle 2.3, S. 30) einen leeren Kern besitzt: Das Koalitionssystem $\mathcal{B} = \{\{1, 2\}, \{1, 3\}, \{2, 3\}\}$ ist für den Gewichtsvektor $\alpha = (1/2, 1/2, 1/2)$ minimal balanciert. Dies kann man analog zu dem Beispiel 2.3.3, S. 39, nachweisen. Nun überprüfen wir, ob die Bedingung (2.30) für das Koalitionssystem \mathcal{B} erfüllt ist:

$$\sum_{S \in \mathcal{B}} \alpha_S v(S) = \frac{1}{2}16 + \frac{1}{2}16 + \frac{1}{2}26 = 29 \not\leq v(N) = 28.$$

Da das minimal balancierte Koalitionssystem \mathcal{B} die Anforderung (2.30) nicht erfüllt, ist der Kern des modifizierten Einführungsbeispiels leer.

2.4 Der Nukleolus: Ein Konzept zur Lösung kooperativer Spiele

In der Theorie der kooperativen Spiele ist der Kern als Lösungskonzept eine wichtige Methode zur Aufteilung der Koalitionsgewinne. Er besteht aus Allokationsvektoren,

2 Kooperative Spiele mit zwei oder mehr Spielern

die wünschenswerte Anforderungen, nämlich sowohl die Effizienzanforderung als auch die Rationalitätsanforderungen, erfüllen und dadurch den Spielern Anreize zur Kooperation geben. Da aber der Kern eines kooperativen Spiels, wenn er existiert, in der Regel keine eindeutige Lösung liefert, steht man vor einem Auswahlproblem: Welches Element des Kerns soll als Spiellösung herangezogen werden? Ein möglicher Weg zur Beantwortung dieser Frage besteht darin, an die Kernelemente zusätzliche Anforderungen zu stellen, die von den Spielern und deren Koalitionen akzeptiert und insoweit als „fair“ betrachtet werden. Ist hingegen der Kern eines Spiels leer, so stellt sich die Frage nach einer anderen möglichen Spiellösung. Ein Konzept, das in diesem Zusammenhang in der Theorie der kooperativen Spiele betrachtet wird, ist der sog. *Nukleolus*, der von Schmeidler [116] in die Literatur eingeführt wurde. Für das Verständnis des Nukleolus spielt der Begriff des *Koalitionsüberschusses* eine wichtige Rolle. Wie wir im vorangegangenen Abschnitt festgehalten haben (vgl. S. 35), ist in einem kooperativen Spiel $\Gamma = (N, v)$ der Überschuss einer Koalition S als die Differenz zwischen dem Koalitionswert und der Auszahlung dieser Koalition bezüglich des Auszahlungsvektors \mathbf{x} definiert:

$$e(S, \mathbf{x}) := v(S) - x(S) = v(S) - \sum_{i \in S} x_i. \quad (2.31)$$

Aus der Definition des Koalitionsüberschusses folgt, dass eine Koalition $S \subset N$ bezüglich eines Auszahlungsvektors \mathbf{x} umso schlechter gestellt ist, je grösser der Koalitionsüberschuss $e(S, \mathbf{x})$ ist. Daher wird der Koalitionsüberschuss $e(S, \mathbf{x})$ in der Literatur als „Mass der Unzufriedenheit“ der Koalition S bezüglich des Auszahlungsvektors \mathbf{x} interpretiert. Folglich wird eine Koalition S einen Allokationsvektor $\mathbf{x} \in \mathcal{I}(N, v)$ gegenüber einem anderen Allokationsvektor $\mathbf{y} \in \mathcal{I}(N, v)$ bevorzugen, wenn $e(S, \mathbf{x}) < e(S, \mathbf{y})$ gilt. Beispielsweise würde in unserem Einführungsbeispiel die Koalition $S = \{2, 3\}$ die Imputation $\mathbf{x} = (10, 20, 12)$ der Imputation $\mathbf{y} = (13, 20, 9)$ vorziehen, denn

$$e(S, \mathbf{x}) = 26 - (20 + 12) = -6 < e(S, \mathbf{y}) = 26 - (20 + 9) = -3.$$

Will man nun feststellen, welche Koalition $S \subseteq N$ in Bezug auf einen Allokationsvektor $\mathbf{x} \in \mathcal{I}(N, v)$ am schlechtesten gestellt ist, kann man die Koalitionsüberschüsse

2.4 Der Nukleolus: Ein Konzept zur Lösung kooperativer Spiele

aller möglichen Koalitionen $S \subseteq N$ vom grössten zum kleinsten ordnen. Es sei $\Theta(\mathbf{x})$ der Vektor, der alle Koalitionsüberschüsse der Koalitionen $S \in 2^N$ bezüglich des Auszahlungsvektors $\mathbf{x} \in \mathcal{I}(N, v)$ in absteigender Reihenfolge ordnet. Dieser Vektor besteht aus 2^n Komponenten, weil es für die Spielermenge $N = \{1, 2, \dots, n\}$ insgesamt 2^n Koalitionen $S \subseteq N$ gibt. Formal können wir den Vektor $\Theta(\mathbf{x})$ wie folgt darstellen:

$$\begin{aligned}\Theta(\mathbf{x}) &= (\theta_1(\mathbf{x}), \theta_2(\mathbf{x}), \dots, \theta_{2^n}(\mathbf{x})) \\ &:= (e(S_1, \mathbf{x}), e(S_2, \mathbf{x}), \dots, e(S_{2^n}, \mathbf{x})) \in \mathbb{R}^{2^n} \text{ mit} \\ &\quad e(S_i, \mathbf{x}) \geq e(S_j, \mathbf{x}) \quad \text{wobei} \quad 1 \leq i \leq j \leq 2^n.\end{aligned}$$

Den so definierten Vektor $\Theta(\mathbf{x})$ nennen wir *Vektor der geordneten Koalitionsüberschüsse*. Da die erste Komponente des Vektors $\Theta(\mathbf{x})$ der grösste Wert ist, ist die Koalition S_1 bezüglich des Auszahlungsvektors \mathbf{x} „am wenigsten zufrieden“, die Koalition S_2 am „am zweitwenigsten zufrieden“ usw. Umgekehrt kann die Koalition S_{2^n} bezüglich des Auszahlungsvektors \mathbf{x} als die zufriedenste Koalition interpretiert werden. Beispielsweise stellt sich für die Imputation $\mathbf{x} = (9, 14, 19)$ des Einführungsbeispiels der Vektor der geordneten Überschüsse $\Theta(\mathbf{x})$ wie folgt ein:²⁰

Tabelle 2.4: Geordnete Überschüsse im Einführungsbeispiel für $\mathbf{x} = (9, 14, 19)$

Koalition S	$\{2, 3\}$	$\{1\}$	$\{1, 3\}$	$\{2\}$	$\{1, 2\}$	$\{3\}$
$\Theta(\mathbf{x})$	-7	-9	-12	-14	-17	-19

Aus der Tabelle 2.4 ist ersichtlich, dass die Koalition $\{2, 3\}$, die bezüglich des Allokationsvektors $\mathbf{x} = (9, 14, 19)$ den höchsten Koalitionsüberschuss -7 hat, am wenigsten zufrieden ist. Betrachten wir nun einen anderen Allokationsvektor $\mathbf{y} = (8, 16, 18)$ des Einführungsbeispiels und den dazugehörigen Vektor des geordneten Koalitionsüberschüsse

$$\begin{aligned}\Theta(\mathbf{y}) &= (e(\{1\}, \mathbf{y}), e(\{2, 3\}, \mathbf{y}), e(\{1, 3\}, \mathbf{y}), e(\{2\}, \mathbf{y}), e(\{3\}, \mathbf{y}), e(\{1, 2\}, \mathbf{y})) \\ &= (-8, -8, -10, -16, -18, -18),\end{aligned}$$

²⁰Dabei lassen wir die grosse Koalitionen N und die leere Menge \emptyset ausser Acht, weil deren Koalitionsüberschüsse für alle Allokationsvektoren $\mathbf{x} \in \mathcal{I}(N, v)$ null sind.

2 Kooperative Spiele mit zwei oder mehr Spielern

so stellen wir fest, dass der höchste Koalitionsüberschuss von -7 auf -8 gesunken ist. Nehmen wir nun an, dass die Spieler bei der Aufteilung des Wertes der grossen Koalition das Ziel verfolgen, aus den Kernallokationen eine zu wählen, bei der die Auszahlung der unter allen möglichen Imputationen am wenigsten zufriedenen Koalition möglichst hoch ist, damit diese Koalition das Bestehen der grossen Koalition nicht gefährdet. In diesem Fall würden die Spieler den Allokationsvektor \mathbf{y} dem Allokationsvektor \mathbf{x} vorziehen. Nun kann es aber mehrere solche Kernallokationen geben. Um aus diesen Kernallokationen eine zu bestimmen, könnten wir das Auswahlkriterium erweitern, indem wir nun die zweithöchsten Koalitionsüberschüsse miteinander vergleichen und den Allokationsvektor mit einem möglichst niedrigeren zweithöchsten Koalitionsüberschuss auswählen usw. Diese Überlegung, die den Grundgedanken des Konzepts des Nukleolus wiedergibt, führt zur folgenden Definition:

Definition 2.4.1 *Ein Vektor $\mathbf{y} \in \mathbb{R}^n$ ist lexikographisch kleiner als der Vektor $\mathbf{z} \in \mathbb{R}^n$, falls es eine Indexzahl $m \leq n$ gibt, so dass gilt:*

- $y_i = z_i$ für $i = 1, 2, \dots, m-1$ und
- $y_m < z_m$.

Man schreibt $\mathbf{y} <_L \mathbf{z}$. Analog bedeutet $\mathbf{y} \leq_L \mathbf{z}$, dass entweder $\mathbf{y} <_L \mathbf{z}$ oder $\mathbf{y} = \mathbf{z}$.

Definition 2.4.2 (Nukleolus) *Der Nukleolus $\mathcal{N} = \mathcal{N}(\Gamma) = \mathcal{N}(N, v)$ eines kooperativen Spiels $\Gamma = (N, v)$ ist die Menge aller Imputationen, die $\Theta(\mathbf{x})$ lexikographisch minimieren:*

$$\mathcal{N}(N, v) := \{\mathbf{x} \in \mathcal{I}(N, v) \mid \Theta(\mathbf{x}) \leq_L \Theta(\mathbf{y}) \quad \forall \mathbf{y} \in \mathcal{I}(N, v)\}. \quad (2.32)$$

Ersetzt man in (2.32) die Bedingung $\mathbf{x}, \mathbf{y} \in \mathcal{I}(N, v)$ durch $\mathbf{x}, \mathbf{y} \in \mathcal{PI}(N, v)$, so spricht man in der Literatur vom *Prenukleolus* des kooperativen Spiels $\Gamma = (N, v)$. Bei superadditiven Spielen stimmen der Nukleolus und der Prenukleolus des Spiels stets überein (vgl. Moulin [86], S. 122, und Young [153], S. 27).²¹ Des Weiteren kann

²¹Man verwendet in der Literatur deswegen auch die Bezeichnung (Pre)Nukleolus anstatt Nukleolus.

2.4 Der Nukleolus: Ein Konzept zur Lösung kooperativer Spiele

man zeigen, dass der Nukleolus eine Teilmenge des Kerns ist, falls der Kern nicht-leer ist (vgl. z. B. Moulin [86], S. 122, und Shubik [126], S. 339–340). Zudem existiert der Nukleolus kooperativer Spiele immer und besteht jeweils aus nur einem Auszahlungsvektor. Die Beweise der soeben gemachten Existenz- und der Eindeutigkeitsaussagen werden wir im Abschnitt 2.6 führen. Wegen der Eindeutigkeit des Nukleolus werden wir im Weiteren nicht mehr von einer Menge sprechen, sondern nur von einem einzigen Punkt $\mathcal{N}(\Gamma)$ als dem Nukleolus(-Punkt).

Maschler et al. [82], S. 337 f., motivieren das Konzept des Nukleolus als ein Fairnesskonzept, das auf folgender Idee beruht. Es sei $\Gamma = (N, v)$ ein kooperatives Spiel, wobei $N = \{1, 2, \dots, n\}$ wie bisher die Spielermenge ist. Der Wert der grossen Koalition ist nun zwischen den Mitgliedern der grossen Koalition aufzuteilen. Zu diesem Zweck soll ein „Schlichter“ oder „Schiedsrichter“ Vorschläge unterbreiten. Dabei muss er gewährleisten, dass die grosse Koalition nicht auseinander fällt. Er muss also alle denkbaren Koalitionen, die aus der grossen Koalition austreten können, in Betracht ziehen. Im ersten Schritt schlägt der Schiedsrichter einen Auszahlungsvektor $\mathbf{x} \in \mathcal{I}(N, v)$ als eine mögliche Aufteilung des Koalitionsgewinns vor. Die Koalition mit der grössten Unzufriedenheit, also mit dem höchsten Koalitionsüberschuss, ist am ehesten geneigt, die grosse Koalition zu verlassen. Deswegen wird der Schlichter versuchen, im ersten Schritt Imputationen zu finden, die die maximale Unzufriedenheit (maximalen Koalitionsüberschuss) minimieren:

$$\min\{\max\{e(S, \mathbf{x}) : S \subset N\} : \mathbf{x} \in \mathcal{I}(N, v)\} =: \epsilon_1.$$

Es sei X_1 die Menge der Auszahlungsvektoren $\mathbf{x} \in \mathcal{I}(N, v)$, die diese Eigenschaft erfüllen.²² Besteht die Menge X_1 aus mehreren Elementen, so wird Schlichter jeder Koalition $S \subset N$, für die $e(S, \mathbf{x}) = \epsilon_1$ für alle $\mathbf{x} \in X_1$ gilt, mitteilen, dass er ihre Lage nicht weiter verbessern kann. Im nächsten Schritt versucht er mit einem neuen Auszahlungsvorschlag das Maximum der Koalitionsüberschüsse der übrigen Koalitionen zu minimieren, indem er Auszahlungsvektoren aus $X_1 \subseteq \mathcal{I}(N, v)$ wählt. Er versucht also den zweitgrössten Koalitionsüberschuss zu minimieren. Dieses Verfahren wird so

²²Diese Menge entspricht genau dem Kleinst-Kern des kooperativen Spiels, den wir auf S. 35 definiert haben.

2 Kooperative Spiele mit zwei oder mehr Spielern

lange fortgesetzt, bis in einem Schritt i die Menge X_i nur noch aus einem einzigen Auszahlungsvektor besteht. In diesem Fall ist der verbleibende Auszahlungsvektor der Nukleolus des kooperativen Spiels.

Wenn wir die Koalitionsüberschüsse als die Einwandspotentiale der Koalitionen bezüglich des Auszahlungsvorschlags interpretieren, so geht es bei der Berechnung des Nukleolus darum, die Einwandspotentiale der Koalitionen möglichst gering zu halten. Man möchte also einen minimalen Konsens finden. Aus diesem Grund spricht man auch von der Idee des sog. „Minimalkonsenses“ als dem Grundgedanken des Nukleolus. Zudem kann das Konzept des Nukleolus insofern als „fair“ oder „gerecht“ betrachtet werden, als es die „Unzufriedenheiten“ der Koalitionen lexikographisch minimiert. In diesem Sinne steht das Lösungskonzept des Nukleolus in engem Zusammenhang mit dem sog. *Rawlsschen Wohlfahrtskriterium*. Das Rawlssche Wohlfahrtskriterium ist ein Ansatz zur Definition von sozialer Gerechtigkeit in einer Gesellschaft. Gemäss diesem Kriterium²³ steigt die gesellschaftliche Wohlfahrt nur dann, wenn der Nutzen des Ärmsten in der Gesellschaft steigt.²⁴

An Hand der oben erläuterten Idee der „Schlichtermethode“ berechnen wir nunmehr den Nukleolus des Einführungsbeispiels. Zur Erinnerung geben wir zunächst nochmals die charakteristische Funktion des Spiels an (vgl. Tabelle 2.1, S. 16):

$$v(\{1\}) = 0, \quad v(\{2\}) = 0, \quad v(\{3\}) = 0$$

$$v(\{1, 2\}) = 6, \quad v(\{1, 3\}) = 16, \quad v(\{2, 3\}) = 26$$

$$v(\{1, 2, 3\}) = 42.$$

²³Young [155], S. 10, beschreibt die Grundidee des Rawlsschen Wohlfahrtskriteriums wie folgt: „... the least well-off group in society should be made as well off as possible“.

²⁴Es sei $u_i(y_i)$ der Nutzen eines Individuums i , wobei y_i alle nutzenstiftenden Einflüsse auf das Individuum i erfasst. Diese Einflüsse können sowohl als Güterbündel wie auch als Einkommen interpretiert werden. Für eine Gesellschaft mit $n \geq 2$ Individuen ist die gesellschaftliche Wohlfahrtsfunktion W vom sog. Bergson-Typ als eine Funktion der individuellen Nutzen definiert, d. h. $W(u_1, u_2, \dots, u_n)$. Dabei kommen in der gesellschaftlichen Wohlfahrtsfunktion die Gerechtigkeitsvorstellungen der Gesellschaftsmitglieder zum Ausdruck. So wird z. B. bei der nach J. Rawls [110] benannten Rawlsschen Wohlfahrtsfunktion die gesellschaftliche Wohlfahrt durch den niedrigsten Nutzen eines Individuums in der Gesellschaft bestimmt: $W(u_1, u_2, \dots, u_n) = \min\{u_1, u_2, \dots, u_n\}$. Vgl. dazu Lachmann [71], S. 178 f.

2.4 Der Nukleolus: Ein Konzept zur Lösung kooperativer Spiele

Tabelle 2.5: Ein Beispiel zur Grundidee des Nukleolus

Koalition	$e(S, \mathbf{x})$	(4, 32, 6)	(10, 20, 12)	(8, 16, 18)	(8, 13, 21)
$\{1\}$	$0 - x_1 =$	-4	-10	-8	-8
$\{2\}$	$0 - x_2 =$	-32	-20	-16	-13
$\{3\}$	$0 - x_3 =$	-6	-12	-18	-21
$\{1, 2\}$	$6 - (x_1 + x_2) =$	-30	-24	-18	-15
$\{1, 3\}$	$16 - (x_1 + x_3) =$	6	-6	-10	-13
$\{2, 3\}$	$26 - (x_2 + x_3) =$	-12	-6	-8	-8

Um den Nukleolus des Spiels zu bestimmen, schlägt der Schlichter zuerst einen effizienten und individuell rationalen Auszahlungsvektor vor, wie z. B. $\hat{\mathbf{x}} = (4, 32, 6)$ (vgl. Tabelle 2.5). Dann ermittelt er die Koalitionsüberschüsse.

Er stellt fest, dass bei dem Auszahlungsvorschlag $\hat{\mathbf{x}}$ die Koalition $\{1, 3\}$ mit ihrem Koalitionsüberschuss 6 am unzufriedensten ist. Sie tendiert also am ehesten dazu, die grosse Koalition zu verlassen. Deswegen wird der Schlichter versuchen, diese Unzufriedenheit zu reduzieren, indem er mit einem neuen Vorschlag die Auszahlung der Koalition $\{1, 3\}$ erhöht, wobei dies auf Grund der Effizienz der Auszahlungsvektoren nur auf Kosten der Koalition $\{2\}$ geschehen kann. Er muss also die Auszahlung der Koalition $\{2\}$ reduzieren. Zu diesem Zweck kann der Schlichter den Auszahlungsvektor $\bar{\mathbf{x}} = (10, 20, 12)$ vorschlagen. Die maximale Unzufriedenheit (der maximale Koalitionsüberschuss) bezüglich des zweiten Vorschlags ist -6 und damit kleiner als bei dem Auszahlungsvektor $\hat{\mathbf{x}}$. Die Koalitionen mit dem Koalitionsüberschuss -6 sind $\{1, 3\}$ und $\{2, 3\}$. Nun überprüft der Schlichter, ob es noch möglich ist, einen anderen Auszahlungsvektor zu finden, der einen kleineren maximalen Koalitionsüberschuss als -6 aufweist. Er stellt fest, dass z. B. der Auszahlungsvektor $\tilde{\mathbf{x}} = (8, 16, 18)$ diese Bedingung erfüllt. Der maximale Koalitionsüberschuss beträgt -8 und ist somit kleiner als -6 . In dieser Phase wird jeder Versuch des Schlichters zur Senkung des maximalen Koalitionsüberschusses unter -8 fehlschlagen. Denn es ist nicht mehr möglich, die Unzufriedenheiten der Koalitionen $\{2, 3\}$ und $\{1\}$, deren Koalitionsüberschuss

2 Kooperative Spiele mit zwei oder mehr Spielern

je -8 beträgt, gleichzeitig zu reduzieren. Eine Erhöhung der Koalitionsauszahlung $x(\{2, 3\})$ bedeutet eine Senkung der Koalitionsauszahlung $x(\{1\})$ und umgekehrt. Folglich besteht keine Möglichkeit, einen individuell rationalen und effizienten Auszahlungsvektor zu finden, der mit einem niedrigeren maximalen Koalitionsüberschuss als -8 verbunden ist. Das heisst, dass für alle Auszahlungsvorschläge der maximale Koalitionsüberschuss mindestens einer der beiden Koalitionen den Wert -8 nicht unterschreiten wird. Somit ist der erste Schritt beendet:

$$\epsilon_1 := \min\{\max\{e(S, \mathbf{x}) : S \subset N\} : \mathbf{x} \in \mathcal{I}(N, v)\} = -8,$$

$$\begin{aligned} X_1 &:= \{\mathbf{x} = (x_1, x_2, x_3) \mid e(S, \mathbf{x}) = -8, S \in \{\{1\}, \{2, 3\}\}, \mathbf{x} \in \mathcal{I}(N, v)\} \\ &= \{\mathbf{x} = (x_1, x_2, x_3) \mid x_1 = 8, x_2 + x_3 = 34, \mathbf{x} \in \mathcal{I}(N, v)\}. \end{aligned}$$

Die nun auf X_1 beschränkte Menge der Imputationen impliziert, dass der Schlichter für die Koalitionen $\{1\}$ und $\{2, 3\}$ nichts mehr tun kann, weil die Unzufriedenheiten dieser beiden Koalitionen bezüglich der Auszahlungsvektoren in der reduzierten Imputationsmenge X_1 bereits minimal sind.

Im zweiten Schritt versucht der Schlichter, die Situation mindestens einer der restlichen Koalitionen soweit wie möglich zu verbessern, indem er geeignete Auszahlungsvektoren aus der Menge $X_1 \subseteq \mathcal{I}(N, v)$ vorschlägt. Er sucht also das Minimum der maximalen Koalitionsüberschüsse für die Auszahlungsvektoren in der reduzierten Imputationsmenge X_1 . Er beginnt diesmal mit derjenigen Koalition, deren Unzufriedenheit nach den Koalitionen $\{1\}$ und $\{2, 3\}$ am höchsten ist. Das ist die Koalition $\{1, 3\}$ mit dem Koalitionsüberschuss in Höhe von -10 (vgl. Tabelle 2.5). Die Senkung der Unzufriedenheit der Koalition $\{1, 3\}$ ist nur möglich, wenn der Schlichter die Auszahlung x_3 und/oder die Auszahlung x_1 erhöht. Da aber in der aktuellen Imputationsmenge X_1 der Wert der Auszahlung x_1 auf 8 fixiert ist, kann er nur die Auszahlung x_3 erhöhen. Auf der anderen Seite darf gemäss der Definition der Menge X_1 die Bedingung $x_2 + x_3 = 34$ nicht verletzt werden. Deshalb muss der Schlichter die Auszahlung x_2 um den selben Betrag reduzieren, um den er die Auszahlung x_3 erhöht. Folglich müssen die Überschüsse der Koalitionen $\{1, 3\}$ und $\{2\}$ zuletzt übereinstimmen. Das ist der Fall, wenn der Schlichter die Auszahlung x_3 um 3 erhöht

2.4 Der Nukleolus: Ein Konzept zur Lösung kooperativer Spiele

und gleichzeitig die Auszahlung x_2 um 3 reduziert. Somit schlägt der Schlichter den Auszahlungsvektor $\mathbf{x}^* = (8, 13, 21)$ vor, welcher als einziger Auszahlungsvektor die Unzufriedenheiten aller Koalitionen lexikographisch minimiert. Dieser Auszahlungsvektor ist der Nukleolus des Spiels.

Da das Konzept des Nukleolus bei der Lösung kooperativer Spiele die Koalitionsüberschüsse aller Koalitionen $S \in 2^N \setminus \emptyset$ lexikographisch minimiert, muss man bei der Berechnung des Nukleolus alle diese Koalitionen berücksichtigen. Sind an einem Spiel nur drei Spieler beteiligt, wie im obigen Beispiel, so sind insgesamt $2^n - 1 = 7$ Koalitionen zu überprüfen, während bei einem Spiel mit z. B. $n = 10$ Spielern schon $2^n - 1 = 1023$ Koalitionsmöglichkeiten bestehen. Die Berechnung des Nukleolus wird aus diesem Grund mit steigender Anzahl der Spieler aufwändiger. Möchte man den Nukleolus kooperativer Spiele mit mehr als drei Spielern berechnen, müssten dafür bereits eigens entwickelte Methoden angewendet werden. In der Literatur werden verschiedene Verfahren vorgeschlagen, die zur Überwindung dieses Berechnungsproblems beitragen. Diese Methoden werden wir im Kapitel 3 vorstellen.

Oben haben wir anhand des Einführungsbeispiels gezeigt, wie der Nukleolus eines kooperativen Spiels $\Gamma = (N, v)$ mit Hilfe der Schlichtermethode bestimmt werden kann. Umgekehrt besteht auch die Möglichkeit zu überprüfen, ob eine gegebene Imputation $\mathbf{x} \in \mathcal{I}(N, v)$ den Nukleolus eines kooperativen Spiels $\Gamma = (N, v)$ darstellt. Kohlberg [64] gibt eine notwendige und hinreichende Bedingung an, anhand derer geprüft werden kann, ob eine Imputation $\mathbf{x} \in \mathcal{I}(N, v)$ der Nukleolus des Spiels ist. Diese Bedingung nennt man in der Literatur *Kohlberg-Kriterium*. Wir formulieren nun das Kohlberg-Kriterium nach Owen [97], S. 328 f. Anschliessend wenden wir es auf unser Einführungsbeispiel an. Gegeben sei ein kooperatives Spiel mit n Spielern $\Gamma = (N, v)$. Für eine Imputation $\mathbf{x} \in \mathcal{I}(N, v)$ definieren wir eine Folge von Koalitionssystemen $\mathcal{B}_1, \mathcal{B}_2, \dots, \mathcal{B}_q$, so dass \mathcal{B}_1 die Menge der Koalitionen mit maxima-

2 Kooperative Spiele mit zwei oder mehr Spielern

lem Koalitionsüberschuss ϵ_1 , \mathcal{B}_2 die Menge der Koalitionen mit dem zweithöchsten Koalitionsüberschuss ϵ_2 usw. ist. In diesem Fall gibt es insgesamt q Zahlen,²⁵

$$\epsilon_1 > \epsilon_2 > \cdots > \epsilon_q,$$

so dass für alle $S \in \mathcal{B}_k$

$$e(S, \mathbf{x}) = \epsilon_k$$

gilt und $\mathcal{B}_1, \mathcal{B}_2, \dots, \mathcal{B}_q$ eine Partition von 2^N ist:

$$\bigcup_{k=1}^q \mathcal{B}_k = 2^N, \quad \mathcal{B}_i \cap \mathcal{B}_j = \emptyset, \quad 1 \leq i, j \leq q, \quad i \neq j.$$

Beispielsweise erhalten wir in unserem Beispiel für die Imputation $\mathbf{x} = (8, 13, 21)$ folgende Mengen von Koalitionen (vgl. Tabelle 2.5, S. 49):²⁶

$$\begin{aligned} \mathcal{B}_1 &= \{\{1\}, \{2, 3\}\} \quad \text{mit } \epsilon_1 = -8, \\ \mathcal{B}_2 &= \{\{2\}, \{1, 3\}\} \quad \text{mit } \epsilon_2 = -13, \\ \mathcal{B}_3 &= \{\{1, 2\}\} \quad \text{mit } \epsilon_3 = -15, \\ \mathcal{B}_4 &= \{\{3\}\} \quad \text{mit } \epsilon_4 = -21. \end{aligned} \tag{2.33}$$

Für $k = 2$ beinhaltet die Menge \mathcal{B}_2 zum Beispiel diejenigen Koalitionen, deren Koalitionsüberschüsse am zweithöchsten sind. Mit Hilfe der obigen Notation geben wir nun das Kohlberg-Kriterium an:

Satz 2.4.1 (Kohlberg-Kriterium) *Es sei $\Gamma = (N, v)$ ein kooperatives Spiel mit n Spielern und \mathbf{x} eine Imputation, die zu den Koalitionssystemen $\mathcal{B}_0 := \{\{i\} \mid x_i = v(\{i\})\}$ und $\mathcal{D}_k := \bigcup_{j=1}^k \mathcal{B}_j$ für $k = 1, 2, \dots, q$ führt. Die Imputation \mathbf{x} ist dann und nur dann der Nukleolus des Spiels, wenn jede Menge*

$$\mathcal{B}_0 \cup \mathcal{D}_k, \quad k = 1, 2, \dots, q, \tag{2.34}$$

balanciert ist, wobei die Gewichte aller Koalitionen $S \in \mathcal{D}_k$, $k = 1, 2, \dots, q$, positiv sein müssen.

²⁵Sind die Überschüsse der Koalitionen für die Imputation $\mathbf{x} \in \mathcal{I}(N, v)$ alle gleich, so gibt es nur ein Koalitionssystem und $q = 1$.

²⁶Dabei lassen wir die Koalitionen N und \emptyset wiederum ausser Acht, weil für sie der Wert von ϵ immer Null ist.

2.4 Der Nukleolus: Ein Konzept zur Lösung kooperativer Spiele

Beweis: Vgl. Kohlberg [64].

Hierbei beinhaltet die Menge \mathcal{B}_0 diejenigen trivialen Einerkoalitionen, für die die individuelle Rationalität bezüglich der Imputation \mathbf{x} als Gleichung erfüllt ist. Jede dieser Einerkoalitionen realisiert also mit der Imputation \mathbf{x} ihren maximal möglichen Überschuss null. Die Bedingung, dass die Koalitionssysteme $\mathcal{B}_0 \cup \mathcal{D}_k$, $k = 1, 2, \dots, q$, balanciert sein müssen, kann wie folgt interpretiert werden: Wir nehmen an, dass $k = 1$ ist. In diesem Fall gilt $\mathcal{D}_1 = \mathcal{B}_1$. Die Menge \mathcal{D}_1 enthält also diejenigen Koalitionen, deren Unzufriedenheit am höchsten ist. Wenn aber die Imputation \mathbf{x} der Nukleolus des Spiels ist, so müsste sie die Unzufriedenheiten dieser Koalitionen, soweit es möglich ist, minimieren. Zudem sollten die Überschüsse der Einerkoalitionen $\{i\} \in \mathcal{B}_0$ bezüglich \mathbf{x} ebenfalls minimiert worden sein. Ist jedoch die Menge $\mathcal{B}_0 \cup \mathcal{D}_1$ nicht-balanciert, so würde dies gemäss der Interpretation der balancierten Koalitionssysteme bedeuten (vgl. S. 37), dass die Menge der Koalitionen $\mathcal{B}_0 \cup \mathcal{D}_1$ nicht alle Ressourcen jedes Spielers $i \in N$ in Anspruch nimmt. Das heisst, dass nicht alle Möglichkeiten zur Minimierung der Unzufriedenheiten aller Koalitionen ausgeschöpft worden sind. Also könnte die Situation mindestens einer Koalition in diesem Fall noch verbessert werden. Demnach kann die Imputation \mathbf{x} nicht der Nukleolus des Spiels sein.²⁷ Ist das Koalitionssystem $\mathcal{B}_0 \cup \mathcal{D}_1 = \mathcal{B}_0 \cup \mathcal{B}_1$ balanciert, so wäre es anschliessend erforderlich zu überprüfen, ob dabei ebenfalls die Unzufriedenheiten der Koalitionen mit zweithöchstem Koalitionsüberschuss minimiert worden sind. Wenn dies der Fall ist, so müsste die Menge der Koalitionen $\mathcal{B}_0 \cup \mathcal{D}_2 = \mathcal{B}_0 \cup \mathcal{B}_1 \cup \mathcal{B}_2$ ebenfalls balanciert sein usw. Mit dem Satz 2.4.1 zeigt Kohlberg [64], dass eine Imputation $\mathbf{x} \in \mathcal{I}(N, v)$, für die die Koalitionssysteme $\mathcal{B}_0 \cup \mathcal{D}_k$, $k = 1, 2, \dots, q$, stets balanciert sind, den Vektor der geordneten Koalitionsüberschüsse lexikographisch minimiert und somit der Nukleolus des Spiels ist. Umgekehrt beweist er auch, dass die Koalitionssysteme $\mathcal{B}_0 \cup \mathcal{D}_k$, $k = 1, 2, \dots, q$, balanciert sind, wenn \mathbf{x} der Nukleolus eines kooperativen Spiels ist.

²⁷Ist \mathcal{D}_1 balanciert, so ist das Koalitionssystem $\mathcal{B}_0 \cup \mathcal{D}_1$ ebenfalls balanciert, z. B. für die Gewichte $\alpha_{\{i\}} = 0$, $\{i\} \in \mathcal{B}_0$, wobei die Gewichte der Koalitionen $S \in \mathcal{D}_1$ nicht geändert werden müssen. Ist hingegen das Koalitionssystem \mathcal{D}_1 nicht-balanciert, jedoch $\mathcal{B}_0 \cup \mathcal{D}_1$ balanciert, so bedeutet das, dass die Unzufriedenheit jeder Einerkoalition $\{i\} \in \mathcal{B}_0$, der mit der Imputation \mathbf{x} gerade ihren Koalitionswert zugeteilt wird, minimal ist.

2 Kooperative Spiele mit zwei oder mehr Spielern

Wir wenden nun das Kohlberg-Kriterium auf das Einführungsbeispiel an. Hierbei betrachten wir die Imputation $\mathbf{x} = (8, 13, 21)$. Für diese Imputation gilt $\mathcal{B}_0 := \{\{i\} \mid x_i = v(\{i\})\} = \emptyset$. Analog zum Beispiel 2.3.1 (vgl. S. 38) kann man zeigen, dass die Koalitionssysteme $\mathcal{B}_0 \cup \mathcal{D}_k$, $k = 1, 2, \dots, q$, für z. B. folgende Gewichtsvektoren balanciert sind (die Koalitionssysteme $\mathcal{D}_k = \mathcal{B}_1 \cup \dots \cup \mathcal{B}_k$, $k = 1, 2, \dots, q$, sind aus den Gleichungen (2.33) herzuleiten):

$$\begin{aligned} \mathcal{B}_0 \cup \mathcal{D}_1 &= \{\{1\}, \{2, 3\}\} && \text{für } \boldsymbol{\alpha} = (1, 1), \\ \mathcal{B}_0 \cup \mathcal{D}_2 &= \{\{1\}, \{2, 3\}, \{2\}, \{1, 3\}\} && \text{für } \boldsymbol{\alpha} = (\tfrac{1}{2}, \tfrac{1}{2}, \tfrac{1}{2}, \tfrac{1}{2}), \\ \mathcal{B}_0 \cup \mathcal{D}_3 &= \{\{1\}, \{2, 3\}, \{2\}, \{1, 3\}, \{1, 2\}\} && \text{für } \boldsymbol{\alpha} = (\tfrac{2}{5}, \tfrac{3}{5}, \tfrac{1}{5}, \tfrac{2}{5}, \tfrac{1}{5}), \\ \mathcal{B}_0 \cup \mathcal{D}_4 &= \{\{1\}, \{2, 3\}, \{2\}, \{1, 3\}, \{1, 2\}, \{3\}\} && \text{für } \boldsymbol{\alpha} = (\tfrac{1}{3}, \tfrac{1}{3}, \tfrac{1}{3}, \tfrac{1}{3}, \tfrac{1}{3}, \tfrac{1}{3}). \end{aligned}$$

Somit ist die Imputation \mathbf{x} tatsächlich der Nukleolus des Einführungsbeispiels, da die Bedingungen des Satzes 2.4.1 bezüglich der Imputation \mathbf{x} erfüllt sind. Betrachten wir hingegen die Imputation $\mathbf{x} = (8, 16, 18)$ des Einführungsbeispiels, so kommen folgende Koalitionssysteme zustande (vgl. Tabelle 2.5, S. 49):

$$\begin{aligned} \mathcal{B}_1 &= \{\{1\}, \{2, 3\}\} \text{ mit } \epsilon_1 = -8, \\ \mathcal{B}_2 &= \{\{1, 3\}\} \text{ mit } \epsilon_2 = -10, \\ \mathcal{B}_3 &= \{\{2\}\} \text{ mit } \epsilon_3 = -16, \\ \mathcal{B}_4 &= \{\{3\}, \{1, 2\}\} \text{ mit } \epsilon_4 = -18. \end{aligned} \tag{2.35}$$

Diese Imputation $\mathbf{x} = (8, 16, 18)$ impliziert ebenfalls $\mathcal{B}_0 := \{\{i\} \mid x_i = v(\{i\})\} = \emptyset$. Somit erhalten wir für $k = 1$ das Koalitionssystem $\mathcal{B}_0 \cup \mathcal{D}_1 = \mathcal{B}_0 \cup \mathcal{B}_1 = \{\{1\}, \{2, 3\}\}$, welches für den Gewichtsvektor $\boldsymbol{\alpha} = (1, 1)$ balanciert ist. Demgegenüber ist das Koalitionssystem $\mathcal{B}_0 \cup \mathcal{D}_2 = \mathcal{B}_0 \cup \mathcal{B}_1 \cup \mathcal{B}_2 = \{\{1\}, \{2, 3\}, \{1, 3\}\}$ für keinen Gewichtsvektor mit positiven Gewichten für die Koalitionen der Menge \mathcal{D}_2 balanciert. Denn der Spieler 2 gehört nur der Koalition $\{2, 3\}$ an. Folglich sollte das Gewicht dieser Koalition eins sein, damit das Koalitionssystem $\mathcal{B}_0 \cup \mathcal{D}_2$ balanciert ist. In diesem Fall müsste aber das Gewicht der Koalition $\{1, 3\}$ null betragen, da der Spieler 3 zu beiden Koalitionen gehört. Nach dem Satz 2.4.1 darf aber das Gewicht keiner Koalition einer Menge \mathcal{D}_k , $k = 1, 2, \dots, q$, null sein. Daher kann die Imputation $\mathbf{x} = (8, 16, 18)$ nicht der Nukleolus des Einführungsbeispiels sein.

2.4 Der Nukleolus: Ein Konzept zur Lösung kooperativer Spiele

In der Literatur finden wir unterschiedliche Modifikationen des Nukleolus.²⁸ Beispielsweise definiert Wallmeier [144] in seiner Dissertation den f -Nukleolus. Dieser Ansatz berücksichtigt die Kritik, dass bei der Berechnung des Nukleolus die Grösse der Koalitionen, d. h. die Anzahl der Spieler in der jeweiligen Koalition, nicht betrachtet wird. Stimmen beispielsweise für ein kooperatives Spiel $\Gamma = (N, v)$ die Überschüsse zweier Koalitionen $S_1 \in 2^N$ und $S_2 \in 2^N$ bezüglich eines Auszahlungsvektors \mathbf{x} überein, so nimmt man gemäss dem Konzept des Nukleolus an, dass die beiden Koalitionen S_1 und S_2 die gleiche Anziehungskraft auf die Spieler des kooperativen Spiels ausüben.²⁹ Bei dieser Betrachtung spielt also die Anzahl der Mitglieder der jeweiligen Koalitionen keine Rolle. Wallmeier [144], S. 28, kritisiert, dass aus den Überschüssen potentieller Koalitionen nicht ohne weiteres auf deren Anziehungskraft geschlossen werden kann. Um die tatsächliche Anziehungskraft auszudrücken, müsste man nach Wallmeier zumindest den mittleren Zugewinn der einzelnen Koalitionspartner berücksichtigen. Um dies zu verdeutlichen nehmen wir an, dass der Koalition S_1 zwei und der Koalition S_2 deutlich mehr, z. B. 15, Mitglieder angehören. Ferner sei \mathbf{x} ein Auszahlungsvektor mit $e(S_1, \mathbf{x}) = e(S_2, \mathbf{x}) = -30$. Für die Koalition S_1 würde in diesem Fall der mittlere Zugewinn der einzelnen Mitglieder 15 betragen, aber für die Mitglieder der Koalition S_2 merklich weniger, nämlich 2. Aus diesem Grund schlägt Wallmeier vor, dass man bei der Berechnung des Nukleolus die Koalitionsüberschüsse $e(S, \mathbf{x})$ beispielsweise im Verhältnis zu der Anzahl der Koalitionsmitglieder betrachten sollte. In der Literatur spricht man vom *per-capita-Nukleolus* oder auch vom *equal division nucleolus*, wenn bei der Definition des Nukleolus die Koalitionsüberschüsse $e(S, \mathbf{x})$, $S \subset N$, durch die „durchschnittlichen“ Koalitionsüberschüsse $e(S, \mathbf{x})/|S|$ ersetzt werden, wobei hier $|S|$ die Anzahl der Mitglieder der Koalition S repräsentiert.

²⁸An dieser Stelle geben wir zwei dieser Modifikationen an. Für einen Überblick über die in der Literatur vorhandenen Modifikationen des Nukleolus verweisen wir auf Krus/Bronisz [68], S. 265.

²⁹Vgl. dazu Wallmeier [144], S. 28 f.

2 Kooperative Spiele mit zwei oder mehr Spielern

Allgemein definiert Wallmeier den Koalitionsüberschuss derart, dass er anstelle von $e(S, \mathbf{x})$ die folgende Funktion verwendet:

$$e_f(S, \mathbf{x}) = \begin{cases} \frac{e(S, \mathbf{x})}{f(|S|)} & , \quad \text{falls } S \neq \emptyset \\ 0 & , \quad \text{falls } S = \emptyset, \end{cases}$$

wobei $f : 2^N \rightarrow \mathbb{R}_{>0}$ eine monotone, nicht-abnehmende Funktion ist. Er dividiert also den Koalitionsüberschuss bezüglich eines Auszahlungsvektors \mathbf{x} durch eine Funktion, die von der Anzahl der Spieler der Koalition S abhängt.

Sudhölter [136] betrachtet hingegen die Differenz $e(S, \mathbf{x}) - e(T, \mathbf{x})$, $S, T \subseteq N$, der Koalitionsüberschüsse. Diese Differenz interpretiert er als den Neid (engl. envy) der Koalition S gegenüber der Koalition T . Den so definierten modifizierten Nukleolus nennt man den *Modiclus*.

Der Nukleolus hat plausibel begründbare, wünschenswerte Eigenschaften, die wir im nächsten Abschnitt erläutern. Die oben vorgestellten Modifikationen des Nukleolus erfüllen nicht alle dieser Eigenschaften. In dieser Arbeit beschäftigen wir uns im Weiteren mit dem Nukleolus.³⁰

2.5 Eigenschaften des Nukleolus

Damit eine Methode als ein Lösungskonzept zur Aufteilung eines Koalitionsgewinns in Frage kommt, sollte sie bestimmte, wünschenswerte Eigenschaften besitzen. Möchte man den aus einer Koalitionsbildung resultierten Gewinn z. B. gemäss dem Konzept des Kerns aufteilen, so weiss man a priori, welche Anforderungen die Elemente eines nicht-leeren Kerns erfüllen: Sie sind durchführbar und effizient. Im Sinne der individuellen Rationalität und Gruppenrationalität sind sie aber auch „fair“ und anreizkompatibel. Diese Eigenschaften werden ebenfalls vom Nukleolus eines kooperativen Spiels mit nicht-leerem Kern erfüllt, da der Nukleolus im nicht-leeren Kern liegt. Zudem beinhaltet der Nukleolus eines kooperativen Spiels einen einzigen Auszahlungsvektor und beruht, wie in den vorangegangenen Abschnitten erläutert, auf der Idee

³⁰Für weitergehende Ausführungen hierzu vgl. Maschler [81], S. 611, Wallmeier [144], S. 76 f., und Sudhölter [136], S. 166 ff.

der Konsensfindung. Gibt es aber andere plausibel begründbare Eigenschaften, die von einem Lösungskonzept gefordert werden können? Werden diese Anforderungen vom Nukleolus eines kooperativen Spiels unabhängig von der jeweiligen Spielsituation a priori erfüllt? Im Folgenden werden wir plausibel begründbare, wünschenswerte Eigenschaften vorstellen und zeigen, ob der Nukleolus die jeweils vorgestellte Eigenschaft besitzt. Falls dies der Fall ist, begründen wir den Nukleolus axiomatisch. Im Besonderen werden wir auf die noch zu definierenden Eigenschaften der Konsistenz, Kovarianz, Anonymität und Monotonie eingehen.³¹ Zu diesem Zweck legen wir zunächst formal fest, wie eine Gewinnaufteilungsmethode oder ein Lösungskonzept in der Theorie der kooperativen Spiele allgemein definiert wird (vgl. Curiel [28], S. 5, und Peleg/Sudhölter [102], S. 24):

Definition 2.5.1 (Lösungskonzept) *Ein Lösungskonzept f ordnet jedem kooperativen Spiel mit n Spielern $\Gamma = (N, v)$ eine Teilmenge $f(N, v)$ bzw. $f(v)$ des \mathbb{R}^n zu, wobei diese Teilmenge leer sein kann.*

Enthält ein Lösungskonzept f für alle kooperative Spiele nur ein Element des \mathbb{R}^n , so schreiben wir $f(N, v) = \mathbf{x}$. In der Theorie der kooperativen Spiele gibt ein Lösungskonzept Antwort darauf, wie ein Koalitionsgewinn zwischen den Spielern aufgeteilt werden soll. Der Kern und der Nukleolus sind Beispiele hierfür.

Eine wichtige in der Literatur diskutierte Eigenschaft eines Lösungskonzepts f ist die sog. *Konsistenzeigenschaft*: Es sei $\Gamma = (N, v)$ ein kooperatives Spiel mit n Spielern und $\mathbf{x} = (x_1^*, x_2^*, \dots, x_n^*)$ ein gemäss einem Lösungskonzept f bestimmter Auszahlungsvektor. Wir fixieren nun für eine beliebige Koalition $S' \subset N$ die Auszahlungen x_i^* , $i \in S'$, und wenden das Lösungskonzept f auf das kooperative Spiel mit der Spielermenge $N \setminus S'$ an. Erhält in diesem Fall jeder Spieler j der Koalition $N \setminus S'$ wiederum die Auszahlung x_j^* , $j \in N \setminus S'$, so heisst das Lösungskonzept f konsistent. Um die Konsistenzeigenschaft formal darzustellen, benötigen wir die Definition der sog. *reduzierten Spiele*:

³¹Die in dieser Arbeit diskutierten Eigenschaften von Lösungskonzepten sind keineswegs abschliessend. Für eine ausführlichere Behandlung der Anforderungen vgl. z. B. Snijders [128], Potters [104] oder Young [152].

2 Kooperative Spiele mit zwei oder mehr Spielern

Definition 2.5.2 (Reduzierte Spiele) Es sei $\Gamma = (N, v)$ ein kooperatives Spiel mit Seitenzahlungen. Des Weiteren sei \mathbf{x} ein effizienter Allokationsvektor. Für jede Koalition $S \subset N$, $S \neq \emptyset$, ist das auf S reduzierte Spiel bezüglich \mathbf{x} definiert als (vgl. Moulin [86], S. 135):

$$v_{\mathbf{x}}^S(T) = \begin{cases} 0, & \text{falls } T = \emptyset, \\ v(N) - \sum_{i \notin S} x_i, & \text{falls } S = T, \\ \max_{R \subset N \setminus S} \{v(R \cup T) - \sum_{i \in R} x_i\}, & \text{falls } \emptyset \subset T \subset S. \end{cases}$$

Die Idee der reduzierten Spiele können wir wie folgt erläutern: Der Wert $x(S)$ ist der Anteil der Koalition S an $x(N)$. Jeder Spieler i , der kein Mitglied der Koalition S ist, erhält in jedem Fall die Auszahlungen x_i , $i \notin S$. Folglich bleibt der Koalition S die Differenz $v(N) - \sum_{i \notin S} x_i$. Somit muss $x(S) = v_{\mathbf{x}}^S(S) = v(N) - \sum_{i \notin S} x_i$ sein. Jede andere nicht-leere Koalition $T \subset S$ hat die Möglichkeit, mit einer Koalition $R \subset N \setminus S$ zu kooperieren. Zusammen können die Koalitionen T und R den Koalitionswert $v(T \cup R)$ realisieren, wobei die Auszahlung der Koalition R durch $\sum_{i \in R} x_i$ gegeben ist. Somit kommt für die Koalition T als Auszahlung nur die Differenz $v(T \cup R) - \sum_{i \in R} x_i$ in Frage. Die Koalition T wird mit derjenigen Koalition $R \subset N \setminus S$ koalieren wollen, die den Wert der Koalition T maximiert.

Bei der Berechnung von $v_{\mathbf{x}}^S(T)$ gehen die Mitglieder der Koalition $T \neq \emptyset$, $T \subset S$, davon aus, dass sie jeden Spieler $i \notin S$ in ihre Koalition einwerben können, wenn sie ihm seine Auszahlung x_i , $i \notin S$ garantieren (vgl. dazu Güth [49], S. 246). An dieser Stelle wollen wir einen Kritikpunkt, der in der Literatur diskutiert wird, erwähnen. Es könnte nämlich sein, dass eine Koalition P , die für T der beste Partner ist, gleichzeitig jedoch auch für die Koalition $T' \subset S$, $T \neq T'$, die beste Partnerkoalition ist. Die beiden Koalitionen $T \cup P$ und $T' \cup P$ können aber nicht gleichzeitig gebildet werden. Die Definition der reduzierten Spiele berücksichtigt also alle möglichen Alternativen, ohne dabei zu beachten, dass nicht alle davon realisiert werden können. (vgl. dazu Maschler [81], S. 617–618, und Güth [49], S. 245–246). Maschler [81], S. 618, relativiert aber diesen Kritikpunkt, indem er hervorhebt, dass die Aufstellung eines reduzierten Spiels in Bezug auf den jeweiligen Kontext vorgenom-

2.5 Eigenschaften des Nukleolus

men werden sollte. Hierfür finden wir in der Literatur unterschiedliche Definitionen der reduzierten Spiele.³²

Zur Illustration betrachten wir das Einführungsbeispiel und leiten das auf $S = \{1, 2\}$ reduzierte Spiel bezüglich des Auszahlungsvektors $\mathbf{x} = (8, 13, 21)$, welcher der Nukleolus des Einführungsbeispiels ist, her:

$$\begin{aligned}
 v_{\mathbf{x}}^S(\emptyset) &= 0, \\
 v_{\mathbf{x}}^S(\{1, 2\}) &= 42 - (8 + 13) = 21, \\
 v_{\mathbf{x}}^S(\{1\}) &= \max_{R \subset N \setminus S} \{v(\{3\} \cup \{1\}) - x_3 ; v(\emptyset \cup \{1\})\} \\
 &= \max\{16 - 21 ; 0\} = 0, \\
 v_{\mathbf{x}}^S(\{2\}) &= \max_{R \subset N \setminus S} \{v(\{3\} \cup \{2\}) - x_3 ; v(\emptyset \cup \{2\})\} \\
 &= \max\{26 - 21 ; 0\} = 5.
 \end{aligned}$$

Es sei nun $\Gamma = (N, v)$ ein kooperatives Spiel und $\mathbf{x} = (x_1, x_2, \dots, x_n)$ eine Impu-tation, die gemäss einem Lösungskonzept f eine Lösung des Spiels ist. Kann keine der nicht-leeren Koalitionen $S \subset N$ die Auszahlung ihrer Mitglieder gemäss dem Lö-sungskonzept f verbessern, indem sie das Spiel $\Gamma = (N, v)$ verlässt und ein eigenes „kooperatives Spiel“ $\Gamma = (S, v_{\mathbf{x}}^S)$ etabliert, so heisst das Lösungskonzept f *konsistent*. Formal nennt man ein Lösungskonzept f konsistent (reduzierbar, stabil), falls für je-des kooperative Spiel $\Gamma = (N, v)$ mit Seitenzahlungen und jede Koalition $S \subset N$ gilt, dass

$$\mathbf{x} = f(N, v) \Rightarrow f(S, v_{\mathbf{x}}^S) = \mathbf{x}_S := (x_i)_{i \in S}. \quad (2.36)$$

Die Konsistenzeigenschaft impliziert somit eine Art „Stabilität“ eines Lösungskon-zepts. Erfüllt nämlich ein Lösungskonzept die Konsistenzeigenschaft, so kann keine Koalition $S \subset N$ durch „Einwerbung“ anderer Spieler in diese Koalition ihr Droh-potenzial und somit die Auszahlung ihrer Mitglieder erhöhen. Hierbei wird voraus-gesetzt, dass die „Einwerbung“ dadurch erfolgt, dass die Koalition S den Spielern $j \in N \setminus S$ ihre fixe Auszahlungen (x_j) garantiert.

³²Für ausführliche Diskussion dieser Problematik vgl. Hart/Mass-Colell [51], S. 601 f., und Maschler et al. [83].

2 Kooperative Spiele mit zwei oder mehr Spielern

Um beispielhaft zu überprüfen, ob der Nukleolus ein konsistentes Lösungskonzept ist, berechnen wir nun den Nukleolus des obigen kooperativen Spiels $\Gamma = (S, v_{\mathbf{x}}^S)$. Wir müssen also einen Auszahlungsvektor $\mathbf{x} \in \mathbb{R}^2$ finden, der den Vektor der geordneten Überschüsse $\Theta(\mathbf{x})$ lexikographisch minimiert. Für die Imputation $\mathbf{x} = (8, 13)$ erhalten wir beispielsweise

$$\begin{aligned} e(\{1\}, \mathbf{x}) &= v_{\mathbf{x}}^S(\{1\}) - x_1 = 0 - 8 = -8, \\ e(\{2\}, \mathbf{x}) &= v_{\mathbf{x}}^S(\{2\}) - x_2 = 5 - 13 = -8. \end{aligned}$$

Somit ist $\Theta(\mathbf{x}) = (-8, -8)$. Da die beiden Komponenten des Vektors der geordneten Überschüsse den gleichen Wert haben, wird jede andere Imputation \mathbf{y} , $\mathbf{y} \neq \mathbf{x}$, zu einem lexikographisch grösseren Vektor der geordneten Überschüsse führen. Folglich ist der Auszahlungsvektor $\mathbf{x} = (8, 13)$ der Nukleolus des Spiels $\Gamma = (S, v_{\mathbf{x}}^S)$. Die Auszahlungen der Spieler der Koalition $S = \{1, 2\}$ bleiben also gemäss dem Lösungskonzept des Nukleolus unverändert, wenn die Koalition S das Spiel $\Gamma = (N, v)$ verlässt und das kooperative Spiel $\Gamma = (S, v_{\mathbf{x}}^S)$ bildet. Analog kann man zeigen, dass für das Einführungsbeispiel diese Eigenschaft bezüglich des Nukleolus für alle $S \subset N$ gilt. Das Lösungskonzept des Nukleolus erfüllt also die Konsistenzeigenschaft für das Einführungsbeispiel. Ferner kann man allgemein beweisen, dass der Nukleolus konsistent ist.³³

Eine andere in der Literatur diskutierte Anforderung an ein Lösungskonzept f ist die sog. *Kovarianzeigenschaft*. Gemäss dieser Anforderung sollte ein Lösungskonzept f davon unabhängig sein, in welcher Einheit, also beispielsweise in welcher Währung, die Koalitionswerte ausgedrückt werden. Verfügen zudem die einzelnen Spieler, und folglich auch deren Koalitionen, unabhängig von der charakteristischen Funktion des Spiels über feste Auszahlungen (z. B. Vermögen), so sollte dies ebenfalls keine Wirkung auf die Lösung des Spiels haben. Ein Lösungskonzept sollte also bei der Aufteilung eines Koalitionsgewinns ausschliesslich die ursprünglichen Koalitionswerte $v(S)$ in Betracht ziehen. Die Kovarianzeigenschaft wird formal wie folgt definiert:

³³Für den Beweis vgl. z. B. Snijders [128] oder Peleg/Sudhölter [102], S. 154.

Definition 2.5.3 (Kovarianz) Es sei $\Gamma = (N, v)$ ein kooperatives Spiel mit n Spielern. Ein Lösungskonzept f ist kovariant, wenn gilt:

$$f(cv(S) + \sum_{i \in S \subseteq N} b_i) = cf(v) + \mathbf{b}, \quad c \in \mathbb{R}_+ \text{ und } \mathbf{b} = (b_1, b_2, \dots, b_n) \in \mathbb{R}^n.$$

Wir können die Kovarianzeigenschaft so interpretieren, dass für $w := cv(S) + \sum_{i \in S \subseteq N} b_i$ die kooperativen Spiele $\Gamma = (N, w)$ und $\Gamma = (N, v)$ strategisch äquivalent sind. Gemäss der Kovarianzeigenschaft sollte dann die Lösung des Spiels $\Gamma = (N, w)$ der gleichen Transformation wie die charakteristische Funktion v unterliegen. Man kann zeigen, dass der Nukleolus ein kovariantes Lösungskonzept ist. Einen Beweis dieser Aussage findet man in Sobolev [129].³⁴

Anhand des Einführungsbeispiels $\Gamma = (N, v)$ illustrieren wir nun die Kovarianzeigenschaft des Nukleolus. Wir nehmen an, dass $c = 2$ und $\mathbf{b} = (2, 3, 5)$ ist. In diesem Fall können wir die charakteristische Funktion $w := cv(S) + \sum_{i \in S \subseteq N} b_i$ des Spiels $\Gamma = (N, w)$ wie folgt bestimmen:

$$\begin{aligned} w(\{1\}) &= 2v(\{1\}) + b_1 = 2, \\ w(\{2\}) &= 2v(\{2\}) + b_2 = 3, \\ w(\{3\}) &= 2v(\{3\}) + b_3 = 5, \\ w(\{1, 2\}) &= 2v(\{1, 2\}) + (b_1 + b_2) = 12 + (2 + 3) = 17, \\ w(\{1, 3\}) &= 2v(\{1, 3\}) + (b_1 + b_3) = 32 + (2 + 5) = 39, \\ w(\{2, 3\}) &= 2v(\{2, 3\}) + (b_2 + b_3) = 52 + (3 + 5) = 60, \\ w(\{1, 2, 3\}) &= 2v(\{1, 2, 3\}) + (b_1 + b_2 + b_3) = 84 + (2 + 3 + 5) = 94. \end{aligned}$$

Der Nukleolus des Einführungsbeispiels $\Gamma = (N, v)$ ist $\mathcal{N}(N, v) = (8, 13, 21)$, während der Nukleolus des Spiels $\Gamma = (N, w)$ dem Auszahlungsvektor $\mathcal{N}(N, w) = (18, 29, 47)$ entspricht.³⁵ Somit stellen wir fest, dass die Gleichung $\mathcal{N}(N, w) =$

³⁴Zitiert nach Maschler [81].

³⁵In dieser Arbeit wurden die Berechnungen der Nukleoli mit dem von uns programmierten Algorithmus zur Berechnung des Nukleolus kooperativer Spiele von Potters et al. [105] durchgeführt. Vgl. dazu Kapitel 5.

2 Kooperative Spiele mit zwei oder mehr Spielern

$2\mathcal{N}(N, v) + (2, 3, 5)$ erfüllt ist und daher der Nukleolus die Bedingung der Kovarianz erfüllt.

Ein weiteres wünschenswertes Kriterium, das von einem Lösungskonzept f erfüllt werden sollte, ist die sog. *Anonymität* bzw. *Symmetrie* einer Spiellösung. Gemäss dieser Eigenschaft sollte eine Lösung unabhängig von der Benennung oder der Nummerierung der Spieler sein. Ändert man also die Nummerierung bzw. Reihenfolge der Spieler, so sollte ein Lösungskonzept f die Auszahlungen der Spieler entsprechend dieser Reihenfolge austauschen, so dass die Auszahlung jedes Spielers gleich bleibt.³⁶ Betrachten wir wieder das Einführungsbeispiel. Für die Spielermenge $N = \{1, 2, 3\}$, die gleichzeitig die Nummerierung der Spieler repräsentiert, wird der Nukleolus dieses Spiels durch den Auszahlungsvektor $\mathcal{N} = (x_1, x_2, x_3) = (8, 13, 21)$ angegeben. Ändern wir nun die Reihenfolge der Spieler, so dass $N = \{3, 1, 2\}$ ist, so wird man feststellen, dass der Nukleolus des neuen Spiels dem Auszahlungsvektor $\mathcal{N} = (x_3, x_1, x_2) = (21, 8, 13)$ entspricht. Allgemein kann man beweisen, dass das Konzept des Nukleolus die Anforderung der Symmetrie erfüllt.³⁷ Die oben diskutierten Eigenschaften begründen den Nukleolus axiomatisch:

Satz 2.5.1 *Der Nukleolus ist das einzige Lösungskonzept, das eindeutig, konsistent, kovariant und anonym (symmetrisch) ist.*

Beweis: Vgl. z. B. Snijders [128], S. 190 ff.

Neben diesen oben dargestellten Eigenschaften, kann man andere Anforderungen formulieren, die aber der Nukleolus nicht in jedem Fall erfüllt. Zu diesen Anforderungen zählt z. B. die sog. *Monotonieeigenschaft*. Bisher haben wir angenommen, dass die charakteristische Funktion eines kooperativen Spiels unter vollständiger Information ermittelt worden ist und sich daher nicht ändert. In der Praxis kommt es aber

³⁶Diese Eigenschaft wird in der Literatur unterschiedlich formuliert. Beispielsweise ersetzt Orshan [94] das Axiom der Anonymität durch das *Axiom der Gleichverteilung für symmetrische (substituierbare) Spieler*. Für ein Spiel (N, v) sind zwei Spieler i und $j \in N$, $i \neq j$, symmetrisch (oder substituierbar) in (N, v) , wenn $v(S \cup \{i\}) = v(S \cup \{j\})$ für alle $S \subseteq N \setminus \{i, j\}$ gilt. Ein Lösungskonzept f erfüllt das Axiom der Gleichverteilung für symmetrische Spieler, wenn das Lösungskonzept f zwei symmetrischen Spielern i und j in einem Spiel (N, v) die gleiche Auszahlung zuordnet.

³⁷Für den Beweis vgl. z. B. Snijders [128], S. 190 ff.

2.5 Eigenschaften des Nukleolus

nicht selten vor, dass die durch Schätzung gewonnenen Daten, wie z. B. Erlös- oder Gewinnziffern, aufgrund von neu gewonnenen Informationen modifiziert werden. Die Monotonieeigenschaft besagt, dass sich die Lösung des Spiels stets in die „gleiche Richtung“ anpasst wie die Änderung der Koalitionswerte. Ändert sich nur der Wert der grossen Koalition, nämlich $v(N)$, so spricht man von der *Monotonie im Aggregat*:

Definition 2.5.4 (Monotonie im Aggregat) *Ein Lösungskonzept f ist monoton im Aggregat, wenn für alle charakteristischen Funktionen v, \hat{v} und die Spielermenge N gilt (Young [152], S. 16):*

$$\begin{aligned} &v(N) \geq \hat{v}(N) \text{ und } v(S) = \hat{v}(S) \text{ für alle } S \subset N \\ \Rightarrow &f_i(N, v) \geq f_i(N, \hat{v}) \text{ für alle } i \in N. \end{aligned}$$

Hierbei ist $f_i(N, v)$ die Auszahlung des Spielers i .

Um die Eigenschaft der Monotonie im Aggregat zu interpretieren nehmen wir an, dass alle Mitglieder der Spielermenge $N = \{1, 2, \dots, n\}$ der Zuteilung des erwarteten Koalitionswerts $\hat{v}(N)$ gemäss dem Lösungskonzept f zugestimmt haben. Nun aber stellen die Spieler durch z. B. neu gewonnenen Informationen fest, dass der Wert der grossen Koalition grösser ist als der anfänglich erwartete Wert $\hat{v}(N)$. Die Monotonie im Aggregat beinhaltet, dass in einem solchen Fall die Auszahlungen der einzelnen Spieler nicht sinken und ferner die Auszahlung mindestens eines Spielers steigt. Ist dies nicht der Fall, sinkt also die Auszahlung eines Spielers $i \in N$ mit steigendem Wert der grossen Koalition, so hat dieser Spieler keinen Anreiz, zur Erhöhung des Wertes der grossen Koalition beizutragen. Umgekehrt wird ein Spieler gegen einen Effizienzverlust, also gegen die Abnahme des Wertes der grossen Koalition, nicht entgegen wirken, falls seine Auszahlung mit sinkendem Wert von $v(N)$ steigt. Das Konzept der Monotonie im Aggregat können wir verallgemeinern, indem wir eine Erhöhung des Wertes irgendeiner Koalition $T \subset N$ betrachten:

2 Kooperative Spiele mit zwei oder mehr Spielern

Definition 2.5.5 (Monotonie) Ein Lösungskonzept f ist monoton, wenn für alle v, \hat{v} und für beliebige $T \subseteq N$ gilt:

$$\begin{aligned} v(T) &\geq \hat{v}(T) \text{ und } v(S) = \hat{v}(S) \text{ für alle } S \neq T \\ \Rightarrow f_i(N, v) &\geq f_i(N, \hat{v}) \text{ für alle } i \in T. \end{aligned}$$

Die Interpretation der Monotonieeigenschaft ist analog zu der Interpretation der Monotonie im Aggregat: Nimmt der Wert einer Koalition $T \subseteq N$ zu bei nicht veränderten Werten aller anderen Koalitionen, dann sollten die Auszahlungen der Mitglieder der Koalition T nicht abnehmen.

Dass der Nukleolus kein monotonen Lösungskonzept ist, können wir bereits anhand eines Beispiels zeigen:³⁸ Es sei $\Gamma = (N, v)$ ein kooperatives Spiel mit der charakteristischen Funktion

$$\begin{aligned} v(\{i\}) &= 0, \text{ für alle } i \in N = \{1, 2, 3, 4\}, \\ v(\{1, 2\}) &= v(\{3, 4\}) = 0, \quad v(\{1, 3\}) = v(\{2, 4\}) = 9, \quad v(\{1, 4\}) = v(\{2, 3\}) = 7, \\ v(S) &= 9 \text{ für alle } |S| = 3, \quad S \subset N, \\ v(N) &= 18. \end{aligned}$$

Der Nukleolus dieses Spiels ist $\mathcal{N} = (\frac{9}{2}, \frac{9}{2}, \frac{9}{2}, \frac{9}{2})$. Nun erhöhen wir den Wert der Koalition $S = \{1, 2, 3\}$ um 9 Einheiten, so dass $v(\{1, 2, 3\}) = 18$ wird, wobei die Werte aller anderen Koalitionen unverändert bleiben. Der Nukleolus dieses modifizierten kooperativen Spiels ist $\mathcal{N} = (8, 9, 1, 0)$. Obwohl der Wert der Koalition $v(\{1, 2, 3\})$ gestiegen ist, sinkt die Auszahlung des Spielers 3, der ein Mitglied dieser Koalition ist, von 4.5 auf 1.

Wir fassen die Eigenschaften des Nukleolus zusammen. Der Nukleolus eines kooperativen Spiels mit Seitenzahlungen existiert immer und ist eindeutig, wie wir im anschließenden Abschnitt beweisen werden. Er ist konsistent, kovariant, symmetrisch und effizient (d. h. Pareto-optimal). Er ist ein Kern-Element, falls der Kern des Spiels nicht-leer ist. Er besitzt die Monotonieeigenschaft nicht.

³⁸Für einen allgemeinen Beweis vgl. Megiddo [84].

2.6 Beweis der Existenz und der Eindeutigkeit des Nukleolus

In diesem Abschnitt geben wir die Beweise für die Existenz und die Eindeutigkeit des Nukleolus eines kooperativen Spiels mit n Spielern an. Die hier vorgeführten Beweise unterscheiden sich nicht von der in der Literatur üblichen Darstellung. Daher richtet sich dieser Abschnitt besonders an den an der Beweisführung interessierten Leser. Andere Leser können diesen Abschnitt überspringen.

Es sei zunächst bemerkt, dass die Imputationsmenge kooperativer Spiele mit n Spielern kompakt³⁹ und konvex ist. Diese Eigenschaften folgen direkt aus der Definition 2.1.6 der Imputationsmenge. Mit Hilfe der Eigenschaft der Kompaktheit zeigen wir nachfolgend zuerst, dass der Nukleolus eines kooperativen Spiels nicht-leer ist. Anschliessend beweisen wir an Hand der Konvexität der kompakten Imputationsmenge, dass der Nukleolus höchstens ein Element enthält.

Satz 2.6.1 (Existenz des Nukleolus) *Der Nukleolus eines kooperativen Spiels $\Gamma = (N, v)$, $N = \{1, 2, \dots, n\}$, mit einer kompakten und nicht-leeren Imputationsmenge $\mathcal{I} = \mathcal{I}(N, v)$ ist ebenfalls kompakt und nicht-leer.*

Beweis (Owen [97], S. 323 f., Jianhua [56], S. 139 f.):

Es sei $X_1 \subset \mathcal{I}(N, v)$ die Menge derjenigen Imputationen \mathbf{x} , die den maximalen Koalitionsüberschuss minimieren, X_2 die Menge der Imputationen $\mathbf{x} \in X_1$, die den zweithöchsten Koalitionsüberschuss minimieren usw. (vgl. dazu S. 50). Sind θ_i , $i = 1, 2, \dots, q$, die Koalitionsüberschüsse bezüglich des jeweiligen Auszahlungsvektors, so können die Mengen X_i , $i = 1, 2, \dots, 2^n$, wie folgt dargestellt werden:⁴⁰

$$X_1 := \{\mathbf{x} \in \mathcal{I}(N, v) \mid \theta_1(\mathbf{x}) \leq \theta_1(\mathbf{y}) \text{ für alle } \mathbf{y} \in \mathcal{I}(N, v)\},$$

$$X_i := \{\mathbf{x} \in X_{i-1} \mid \theta_i(\mathbf{x}) \leq \theta_i(\mathbf{y}) \text{ für alle } \mathbf{y} \in X_{i-1}\}, \quad i = 2, 3, \dots, 2^n.$$

Um zu zeigen, dass der Nukleolus des kooperativen Spiels kompakt und nicht-leer ist, wollen wir beweisen, dass die hier definierten Mengen X_1, X_2, \dots, X_{2^n} ebenfalls

³⁹Eine Menge nennen wir kompakt, wenn sie abgeschlossen und beschränkt ist.

⁴⁰Bei dieser Darstellung der Mengen X_i , $i = 1, 2, \dots, 2^n$, kann der Fall vorliegen, dass nach einem bestimmten Schritt $q < 2^n$ die Mengen X_i , $i = q, q + 1, \dots, 2^n$, übereinstimmen.

2 Kooperative Spiele mit zwei oder mehr Spielern

kompakt und nicht-leer sind. Eine hinreichende Bedingung dafür ist, dass die Koalitionsüberschüsse $\theta_i(\mathbf{x})$, $i = 1, 2, \dots, 2^n$, bezüglich der Auszahlungsvektoren $\mathbf{x} \in \mathcal{I}(N, v)$ stetig sind. Sind nämlich die Koalitionsüberschüsse $\theta_i(\mathbf{x})$, $i = 1, 2, \dots, 2^n$, stetige Funktionen und ist die Imputationsmenge des Spiels kompakt und nicht-leer, so ist die oben definierte Menge X_1 ebenfalls kompakt und nicht-leer.⁴¹ Denn X_1 beinhaltet diejenigen Auszahlungsvektoren \mathbf{x} , die den Koalitionsüberschuss θ_1 über der Imputationsmenge $\mathcal{I}(N, v)$ minimieren. Aus dem selben Grund ist die Menge X_2 auch kompakt und nicht-leer, falls X_1 kompakt und nicht-leer ist und die Funktionen $\theta_i(\mathbf{x})$, $i = 1, 2, \dots, 2^n$, stetig sind usw. Folglich sind X_1, X_2, \dots, X_{2^n} kompakt und nicht-leer, falls die Überschüsse $\theta_i(\mathbf{x})$, $i = 1, 2, \dots, 2^n$, stetig sind und die Imputationsmenge $\mathcal{I}(N, v)$ kompakt und nicht-leer ist. Aus der Definition des Nukleolus folgt nun aber, dass $X_{2^n} = \mathcal{N}(\Gamma)$ (für einen Beweis vgl. Owen [97], S. 324). Somit ist der Nukleolus eines kooperativen Spiels kompakt und nicht-leer.

Nun wollen wir zeigen, dass die Koalitionsüberschüsse $\theta_i(\mathbf{x})$ für $i = 1, 2, \dots, 2^n$ tatsächlich stetige Funktionen der Auszahlungsvektoren \mathbf{x} sind (vgl. Jianhua [56], S. 139). Um dies zu beweisen, legen wir die Koalitionsüberschüsse $\theta_i(\mathbf{x})$ für $i = 1, 2, \dots, 2^n$ in folgender Form dar. Darin steht $|\mathcal{A}|$ für die Anzahl der Koalitionen in \mathcal{A} :

$$\theta_i(\mathbf{x}) = \max\{\min\{e(S, \mathbf{x}) : S \in \mathcal{A}\} : \mathcal{A} \subseteq 2^N, |\mathcal{A}| = i\}. \quad (2.37)$$

Bevor wir auf die Stetigkeit der Koalitionsüberschüsse $\theta_i(\mathbf{x})$ eingehen, müssen wir zeigen, dass sie tatsächlich in der Form der Gleichungen (2.37) dargestellt werden können. Wir müssen also beweisen, dass die Gleichung (2.37) folgendes impliziert:

$$\theta_i(\mathbf{x}) \geq \theta_{i+1}(\mathbf{x}) \text{ für } i = 1, 2, \dots, 2^n. \quad (2.38)$$

Ersetzen wir in der Gleichung (2.37) i durch $i + 1$, erhalten wir

$$\theta_{i+1}(\mathbf{x}) = \max\{\min\{e(S, \mathbf{x}) : S \in \mathcal{A}\} : \mathcal{A} \subseteq 2^N, |\mathcal{A}| = i + 1\}. \quad (2.39)$$

⁴¹Dies folgt aus dem Satz von Weierstrass. Dieser Satz besagt, dass jede stetige Funktion über einer kompakten Menge ihr Maximum annimmt. Vgl. dazu z. B. Bronstein/Semendjajew [21], S. 262.

2.6 Beweis der Existenz und der Eindeutigkeit des Nukleolus

Da in der Gleichung (2.37) die Menge \mathcal{A} wegen $|\mathcal{A}| = i$ aus insgesamt i Koalitionen besteht, gilt

$$|\{e(S, \mathbf{x}) : S \in \mathcal{A}\}| = i. \quad (2.40)$$

Die Gleichung (2.40) besagt, dass für die Menge \mathcal{A} insgesamt i Koalitionsüberschüsse existieren, weil die Menge \mathcal{A} insgesamt i Koalitionen enthält. Folglich können wir in der Gleichung (2.39) für jede Menge $\{e(S, \mathbf{x}) : S \in \mathcal{A}\}$ solche Mengen $\{e(S, \mathbf{x}) : S \in \mathcal{A}\}$ mit $i + 1$ Elementen finden, dass die Menge mit i Elementen eine Teilmenge der Menge mit $i + 1$ Elementen ist. Die Wahl des Maximums in jeder Menge ergibt, dass $\theta_i(\mathbf{x}) \geq \theta_{i+1}(\mathbf{x})$ ist. Diese Ungleichung gilt für alle $i = 1, 2, \dots, 2^n - 1$. Somit haben wir bewiesen, dass die Ungleichungen (2.38) erfüllt sind.

Nun zeigen wir anhand der Gleichungen (2.37), dass die Überschüsse $\theta_i(\mathbf{x})$ für $i = 1, 2, \dots, 2^n$ stetige Funktionen von \mathbf{x} sind: Da $e(S, \mathbf{x}) = v(S) - x(S)$ eine stetige Funktion von \mathbf{x} ist und der Maximum- wie auch der Minimum-Operator stetige Funktionen sind, sind die Koalitionsüberschüsse θ_i für $i = 1, 2, \dots, 2^n$ ebenfalls stetige Funktionen von \mathbf{x} . \square

Wir zeigen nun, dass der Nukleolus eines kooperativen Spiels mit n Spielern nur ein Element enthält, wenn die Imputationsmenge des Spiels neben der Eigenschaft der Kompaktheit auch die Eigenschaft der Konvexität erfüllt:

Satz 2.6.2 (Eindeutigkeit des Nukleolus) *Wenn die Imputationsmenge kompakt, konvex und nicht-leer ist, besteht der Nukleolus aus höchstens einem Element.*

Beweis (Owen [97], S. 324–326):

Für den Beweis zeigen wir zunächst, dass der Vektor der geordneten Koalitionsüberschüsse Θ strikt quasi-konvex in lexikographischer Ordnung ist. Mit Hilfe dieser Eigenschaft und der Konvexität der Imputationsmenge werden wir anschliessend beweisen, dass der Nukleolus nur einen Auszahlungsvektor beinhaltet.

2 Kooperative Spiele mit zwei oder mehr Spielern

Der Vektor der geordneten Koalitionsüberschüsse Θ als Funktion von Imputationen heisst strikt quasi-konvex in lexikographischer Ordnung, wenn $\Theta(\mathbf{x}) = \Theta(\mathbf{y})$, für $\mathbf{x} \neq \mathbf{y}$ impliziert, dass

$$\Theta(\alpha\mathbf{x} + (1 - \alpha)\mathbf{y}) <_L \Theta(\mathbf{x}) = \Theta(\mathbf{y}) \text{ für } 0 < \alpha < 1. \quad (2.41)$$

Wir beweisen nun, dass Θ die Bedingung (2.41) erfüllt. Es sei $\mathbf{z} := \alpha\mathbf{x} + (1 - \alpha)\mathbf{y}$. Die zu \mathbf{x} und zu \mathbf{y} gehörenden Vektoren der geordneten Koalitionsüberschüsse sind

$$\begin{aligned} \Theta(\mathbf{x}) &= (e(S_1, \mathbf{x}), e(S_2, \mathbf{x}), \dots, e(S_{2^n}, \mathbf{x})), \\ \Theta(\mathbf{y}) &= (e(T_1, \mathbf{y}), e(T_2, \mathbf{y}), \dots, e(T_{2^n}, \mathbf{y})). \end{aligned}$$

Hier sind S_1, S_2, \dots, S_{2^n} und T_1, T_2, \dots, T_{2^n} zwei wie folgt geordnete Permutationen der Elemente von 2^N ($k = 1, 2, \dots, 2^n - 1$):

- (a) $e(S_k, \mathbf{x}) \geq e(S_{k+1}, \mathbf{x})$
- (b) Falls $e(S_k, \mathbf{x}) = e(S_{k+1}, \mathbf{x})$, dann gelte $e(S_k, \mathbf{y}) \geq e(S_{k+1}, \mathbf{y})$
- (c) $e(T_k, \mathbf{y}) \geq e(T_{k+1}, \mathbf{y})$.

Nun nehmen wir an, dass j der erste Index ist, für den $e(S_j, \mathbf{x}) \neq e(S_j, \mathbf{y})$ gilt. Es sei $q := e(S_j, \mathbf{x})$. Für eine beliebige Koalition $S \in 2^N$ gilt:

$$\begin{aligned} e(S, \mathbf{z}) &= v(S) - x(\mathbf{z}) \\ &= v(S) - \sum_{i \in S} [\alpha x_i + (1 - \alpha)y_i] \\ &= v(S) - \alpha \sum_{i \in S} x_i - (1 - \alpha) \sum_{i \in S} y_i \\ &= \alpha \left[v(S) - \sum_{i \in S} x_i \right] + (1 - \alpha) \left[v(S) - \sum_{i \in S} y_i \right] \\ &= \alpha e(S, \mathbf{x}) + (1 - \alpha) e(S, \mathbf{y}). \end{aligned}$$

Aus der Definition von j folgt, dass für alle $k < j$ die Gleichung $e(S_k, \mathbf{x}) = e(S_k, \mathbf{y})$ gilt und somit

$$e(S_k, \mathbf{z}) = e(S_k, \mathbf{x}). \quad (2.42)$$

2.6 Beweis der Existenz und der Eindeutigkeit des Nukleolus

Zudem muss für alle $k \geq j$ die Ungleichung

$$e(S_k, \mathbf{y}) \leq e(S_j, \mathbf{x}) \quad (2.43)$$

erfüllt sein. Würde die Ungleichung (2.43) nicht gelten, so müsste für mindestens einen Index $k \geq j$ die Ungleichung $e(S_k, \mathbf{y}) > e(S_j, \mathbf{x}) = q$ gelten. Dies würde aber bedeuten, dass die Ungleichung $e(S, \mathbf{y}) > q$ ebenfalls erfüllt sein muss, wenn für eine beliebige Koalition S die Ungleichung $e(S, \mathbf{x}) > q$ gilt. Darüber hinaus gäbe es mindestens eine Koalition, nämlich die Koalition S_k , für welche $e(S_k, \mathbf{y}) > q$ gelten würde, während aber $e(S_k, \mathbf{x}) \leq q$ wäre. Folglich wäre unsere Annahme $\Theta(\mathbf{x}) = \Theta(\mathbf{y})$ verletzt. Somit muss für alle $k \geq j$ die Ungleichung (2.43) gelten.

Ist nun $k = j$, so gilt $e(S_j, \mathbf{y}) \leq e(S_j, \mathbf{x}) = q$. Wegen der Wahl von j und der Bedingung (c) muss in diesem Fall $e(S_j, \mathbf{y}) < e(S_j, \mathbf{x}) = q$ sein. Ist aber $k > j$ und $e(S_k, \mathbf{x}) = e(S_j, \mathbf{x}) = q$, so gilt wegen der Eigenschaft (b), dass $e(S_k, \mathbf{y}) \leq e(S_j, \mathbf{y}) < q$. Für $k \geq j$ und $e(S_k, \mathbf{x}) = q$ haben wir also gezeigt, dass $e(S_k, \mathbf{y}) < q$ ist. Falls nun $k \geq j$ und $e(S_k, \mathbf{x}) < q$ ist, so muss wegen (2.43) die Ungleichung $e(S_k, \mathbf{y}) \leq q$ gelten. Für $k \geq j$ gilt also

$$\begin{aligned} &\text{entweder} \quad e(S_k, \mathbf{x}) < q, \quad e(S_k, \mathbf{y}) \leq q \\ &\text{oder} \quad e(S_k, \mathbf{x}) = q, \quad e(S_k, \mathbf{y}) < q. \end{aligned}$$

Somit liegt

$$e(S_k, \mathbf{z}) = \alpha e(S_k, \mathbf{x}) + (1 - \alpha) e(S_k, \mathbf{y}) < q \quad (2.44)$$

vor. Aus (2.42) und (2.44) folgt also für einen beliebigen Index k

$$\begin{aligned} e(S_k, \mathbf{z}) &= e(S_k, \mathbf{x}) \quad \text{für } k < j, \\ e(S_k, \mathbf{z}) &< q \quad \text{für } k \geq j. \end{aligned}$$

Die letzten beiden Eigenschaften implizieren, dass der Vektor $\Theta(\mathbf{z})$ lexikographisch kleiner ist als der Vektor $\Theta(\mathbf{x})$. Somit haben wir (2.41) bewiesen. Nun nehmen wir an, dass der Nukleolus nicht-eindeutig ist und betrachten zwei Vektoren $\mathbf{x} \in \mathcal{N}(\Gamma)$

2 Kooperative Spiele mit zwei oder mehr Spielern

und $\mathbf{y} \in \mathcal{N}(\Gamma)$, mit $\mathbf{x} \neq \mathbf{y}$. Unsere Annahme beinhaltet, dass $\Theta(\mathbf{x}) = \Theta(\mathbf{y})$ gilt. Aus der Konvexität der Imputationsmenge $\mathcal{I}(N, v)$ folgt, dass der Vektor

$$\mathbf{z} = \lambda \mathbf{x} + (1 - \lambda) \mathbf{y} \quad \text{für } 0 < \lambda < 1$$

auch ein Element von $\mathcal{I}(N, v)$ ist. Dies impliziert wegen (2.41), dass $\Theta(\mathbf{z}) <_L \Theta(\mathbf{x})$. Dieses Resultat bedeutet wiederum, dass $\mathbf{x} \notin \mathcal{N}(\Gamma)$. Somit liegt ein Widerspruch vor und \mathbf{x} und \mathbf{y} können nicht gleichzeitig ein Element des Nukleolus sein. \square

2.7 Zusammenfassung

Im ersten Abschnitt des Kapitels 2 haben wir zunächst die charakteristische Funktion eines kooperativen Spiels definiert. Diese ordnet jeder Koalition von Spielern eine nicht-negative reellwertige Zahl zu, welche Koalitionswert genannt wird. Der Koalitionswert repräsentiert den maximalen Betrag, den die Mitglieder der Koalition gemeinsam realisieren können, unabhängig vom Verhalten der Spieler ausserhalb der Koalition. Dieser Wert kann je nach Spielsituation unterschiedlich gross sein. Durch die Normierung der charakteristischen Funktion ist es jedoch möglich, ihn durch einen zwischen null und eins liegenden Betrag auszudrücken. In einem weiteren Schritt haben wir gezeigt, dass mehrere kooperative Spiele mit unterschiedlichen charakteristischen Funktionen dieselbe Spielsituation beschreiben können. Solche Spiele nennt man strategisch äquivalente kooperative Spiele. Jede Gruppe strategisch äquivalenter kooperativer Spiele kann durch ein entsprechendes $(0 - 1)$ -normiertes kooperatives Spiel dargestellt werden. Im Abschnitt 2.1 haben wir uns anschliessend mit den grundlegenden Eigenschaften der charakteristischen Funktion eines kooperativen Spiels befasst. Dabei wurden die Eigenschaften Monotonie, Superadditivität und Konvexität diskutiert. Eine charakteristische Funktion ist monoton, falls die Vergrösserung einer beliebigen Koalition zu keiner Verminderung des Koalitionswertes führt. Ist eine charakteristische Funktion superadditiv, so lohnt es sich für disjunkte Koalitionen zusammenzuarbeiten. Eine charakteristische Funktion heisst konvex, falls die Steigerung des Wertes einer Koalition durch den Eintritt eines Spielers in diese Koalition umso höher ausfällt, je grösser die Koalition ist.

In der Theorie der kooperativen Spiele geben verschiedene Lösungskonzepte Antwort darauf, anhand welcher Kriterien der von den Spielern durch Koalitionsbildung realisierte gemeinsame Gewinn unter den Mitgliedern der Koalition aufzuteilen ist. Im Abschnitt 2.2 haben wir die Forderungen nach Anreizkompatibilität und Stabilität eines Auszahlungsvektors zur Aufteilung des Wertes der grossen Koalition aufgestellt. Der Kern als ein Lösungskonzept erfüllt beide Kriterien in dem Sinne, dass er solche Auszahlungsvektoren beinhaltet, die die Eigenschaften der Effizienz,

2 Kooperative Spiele mit zwei oder mehr Spielern

individuellen Rationalität und Gruppenrationalität besitzen. Ein Auszahlungsvektor heisst effizient, falls er den gesamten Wert der grossen Koalition unter den Spielern vollständig aufteilt. Individuelle Rationalität bzw. Gruppenrationalität besagt, dass bei der Aufteilung des Wertes der grossen Koalition jedem Spieler bzw. jeder Koalition von Spielern mindestens der Betrag zugeteilt werden soll, der seinem bzw. ihrem Koalitionswert entspricht. Einigen sich die Spieler zur Aufteilung des Wertes der grossen Koalition auf einen Auszahlungsvektor, der ein Element des Kerns des Spiels ist, so haben kein Spieler und keine Koalition von Spielern ein Interesse, die grosse Koalition zu verlassen. Neben diesen Charakteristika hat der Kern einige andere wünschenswerte Eigenschaften nicht. Mit Hilfe der erwähnten Auswahlkriterien, nämlich Effizienz, individueller Rationalität und Gruppenrationalität, reduziert der Kern zwar die Menge der als Lösung in Frage kommenden Auszahlungsvektoren, er beinhaltet aber in der Regel mehrere Auszahlungsvektoren. Er liefert also nicht notwendigerweise eine eindeutige Spiellösung. Darüber hinaus ist die Existenz des Kerns nicht für jedes kooperative Spiel gesichert.

Um die Existenz des Kerns überprüfen zu können, haben wir im Abschnitt 2.3 mit Hilfe der balancierten Koalitionssysteme eine notwendige und hinreichende Bedingung für einen nicht-leeren Kern vorgestellt. Eine Menge von Koalitionen, also ein Koalitionssystem, heisst balanciert oder ausgewogen, falls jeder Spieler des Spiels seine Gesamtressourcen, z. B. seine Arbeitszeit, vollständig auf die Koalitionen des Koalitionssystems aufteilt. Liegt ein balanciertes Spiel vor, so kann man von der Existenz des Kerns ausgehen.

Im Abschnitt 2.4 haben wir uns mit der Fragestellung befasst, welches Element des Kerns als Lösung eines Spiels ausgewählt werden sollte. In diesem Zusammenhang wurde zuerst das Konzept des Nukleolus vorgestellt. Der Nukleolus ist ein Lösungskonzept, das unter den Kernallokationen denjenigen Allokationsvektor wählt, der die Unzufriedenheit der Koalitionen lexikographisch minimiert. In diesem Zusammenhang wird der Nukleolus als ein Lösungskonzept der Konsensfindung bezeichnet. Ist der Kern eines Spiels nicht-leer, verschärft der Nukleolus die Kernbedingungen solange, bis diese verschärften Bedingungen von einem einzigen Auszahlungsvektor erfüllt

sind. Umgekehrt lockert der Nukleolus die Kernbedingungen, wenn ein kooperatives Spiel einen leeren Kern besitzt. Somit ist der Nukleolus ein Konzept, dessen Lösungsmenge stets nicht-leer ist und nur noch einen einzigen Auszahlungsvektor enthält. Da der Nukleolus immer existiert, eindeutig ist und auf der Idee der Konsensfindung beruht, hat er in der Theorie der kooperativen Spiele weite Verbreitung gefunden. Anschliessend haben wir gezeigt, dass die Berechnung des Nukleolus mit steigender Anzahl der Spieler aufwändiger wird und daher die Verwendung mathematischer Verfahren zur Bestimmung des Nukleolus unumgänglich ist. Zuletzt wurde in diesem Abschnitt das Kohlberg-Kriterium erörtert, anhand dessen überprüft werden kann, ob ein gegebener Auszahlungsvektor dem Nukleolus des Spiels entspricht.

Im Abschnitt 2.5 haben wir weitere wichtige Eigenschaften des Nukleolus behandelt und den Nukleolus axiomatisch begründet. Wir haben gezeigt, dass der Nukleolus konsistent, kovariant und symmetrisch ist, aber die Eigenschaft der Monotonie nicht aufweist. Die Konsistenz besagt, dass sich die Auszahlungen der verbleibenden Spieler nicht ändern, wenn das Spiel dadurch reduziert wird, dass einzelne Spieler ihre Auszahlung gemäss dem Lösungskonzept erhalten. Ein Lösungskonzept heisst kovariant, wenn sowohl die Masseinheit, mit der die Koalitionswerte gemessen werden, wie auch die den Spielern unabhängig vom Spiel zur Verfügung stehenden Beträge die Lösung des Spiels nicht beeinflussen. Ist die Auszahlung eines Spielers gemäss einem Lösungskonzept unabhängig von der Nummerierung bzw. Reihenfolge der Spieler, so nennt man das Lösungskonzept symmetrisch. Die Monotonie eines Lösungskonzepts liegt dann vor, wenn sich die Auszahlung mindestens eines Mitglieds einer Koalition, deren Wert gestiegen ist, verbessert und sich die Auszahlungen aller anderen Mitglieder nicht verschlechtern.

Zum Schluss des Kapitels 2 haben wir im Abschnitt 2.6 für den an der Beweisführung interessierten Leser die Beweise der Eindeutigkeit und der Existenz des Nukleolus geführt.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

3.1 Methoden der ersten Generation

3.1.1 Das Verfahren von Kohlberg

Aus der Definition des Lösungskonzepts des Nukleolus im Abschnitt 2.4 geht hervor, dass die Berechnung des Nukleolus eine aufwändige Aufgabe ist, weil alle möglichen Koalitionen berücksichtigt werden. In der Literatur kennt man unterschiedliche Verfahren zur Berechnung des Nukleolus kooperativer Spiele. Diese Verfahren beruhen auf verschiedenen mathematischen Methoden, die für die praktische Anwendbarkeit des jeweiligen Verfahrens eine grosse Rolle spielen. Dennoch ziehen die meisten für die Berechnung des Nukleolus entwickelten Verfahren die Methoden der linearen Optimierung heran.¹ Dabei finden wir in der Literatur neben solchen Verfahren, die die Lösung mehrerer linearer Programme in Anspruch nehmen, auch Vorgehensweisen, die für die Berechnung des Nukleolus nur das Lösen eines linearen Programms erfordern. Die beiden Verfahren, die den Nukleolus mit Hilfe eines einzigen linearen Programms berechnen, zählen wir zu den Methoden der ersten Generation. Das erste derartige Verfahren stammt von Kohlberg [65]. In diesem Abschnitt stellen wir dieses Verfahren kurz vor.

Kohlberg [65] hat in seinem Artikel als erster gezeigt, dass der Nukleolus mit Hilfe eines einzigen linearen Programms berechnet werden kann. Wie wir noch erläutern werden, beinhaltet das lineare Programm aber eine grosse Zahl von Nebenbedingungen. Zudem sind die Koeffizienten der Variablen in den Nebenbedingungen des linearen Programms sehr gross. Aus diesen beiden Gründen stellt das Verfahren von Kohlberg eine in der Praxis kaum taugliche Methode zur Berechnung des Nukleolus

¹Eine Einführung in die lineare Optimierung befindet sich im Anhang A.1.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

kooperativer Spiele dar. Das Verfahren von Kohlberg zeigt also nur die theoretische Möglichkeit, dass der Nukleolus als Lösung einer einzigen linearen Optimierungsaufgabe bestimmt werden kann. Daher werden wir hier dieses Verfahren nicht ausführlich behandeln und verweisen für die Beweise der nachstehenden Aussagen auf Kohlberg [65].

Es seien $\Gamma = (N, v)$ ein kooperatives Spiel mit n Spielern und S_1, S_2, \dots, S_{2^n} Elemente der Potenzmenge von N . Kohlberg weist nach, dass eine positive reellwertige Zahl $t_0 = t_0(n, \mathcal{I}(N, v))^2$ derart existiert, dass der Nukleolus des Spiels $\Gamma = (N, v)$ für alle $t \geq t_0$ die eindeutige Lösung des folgenden linearen Optimierungsproblems ist:

$$\min z \tag{3.1}$$

unter den Nebenbedingungen

$$z \geq \sum_{j=1}^{2^n} t^j e(S_j, \mathbf{x}) \quad \text{für alle Permutationen von } \{S_1, S_2, \dots, S_{2^n}\} \tag{3.2}$$

$$x_1 + x_2 + \dots + x_n = v(N) \tag{3.3}$$

$$x_i \geq 0, \quad i = 1, 2, \dots, n. \tag{3.4}$$

Hierbei gewährleisten die Nebenbedingungen (3.3) und (3.4), dass die Lösung des linearen Programms ein Element der Imputationsmenge des Spiels ist. Kohlberg zeigt, dass für $(0-1)$ -normierte kooperative Spiele mit n Spielern $t \geq t_0 = n^{2^n}$ gelten muss.

Die Idee des Verfahrens von Kohlberg können wir wie folgt erläutern: Es seien $\Gamma = (N, v)$ ein kooperatives Spiel mit n Spielern und \mathbf{x} der Nukleolus des Spiels, wobei

$$\begin{aligned} \Theta(\mathbf{x}) &= (e(S_1, \mathbf{x}), e(S_2, \mathbf{x}), \dots, e(S_{2^n}, \mathbf{x})) \in \mathbb{R}^{2^n} \text{ mit} \\ e(S_i, \mathbf{x}) &\geq e(S_j, \mathbf{x}) \quad \text{für } 1 \leq i \leq j \leq 2^n \end{aligned}$$

den zu \mathbf{x} gehörenden Vektor der geordneten Überschüsse darstellt. Da in diesem Fall $\Theta(\mathbf{x})$ lexikographisch minimal ist, gilt für jede Imputation $\mathbf{y} \in \mathcal{I}(N, v)$, $\mathbf{x} \neq \mathbf{y}$,

²Die Notation $t_0 = t_0(n, \mathcal{I}(N, v))$ besagt, dass die Zahl t_0 nur von der Anzahl der Spieler und der Imputationsmenge direkt abhängt, nicht aber von der charakteristischen Funktion des Spiels. Das bedeutet, dass das von Kohlberg vorgeschlagene lineare Programm auf alle kooperativen Spiele angewendet werden kann.

3.1 Methoden der ersten Generation

dass der Vektor der geordneten Überschüsse $\Theta(\mathbf{y})$ lexikographisch grösser als $\Theta(\mathbf{x})$ ist. Kohlberg weist nach, dass für $\Theta(\mathbf{x})$ eine Folge positiver, fallender Gewichte t^j , $j = 1, 2, \dots, 2^n$, ($t > 1$) existiert, für die der Nukleolus die einzige Imputation ist, welche die Summe der so gewichteten Koalitionsüberschüsse, also

$$t^{2^n} e(S_1, \mathbf{x}) + t^{2^n-1} e(S_2, \mathbf{x}) + \dots + t e(S_{2^n}, \mathbf{x}), \quad (3.5)$$

minimiert. In dieser Gewichtung der Überschüsse ist die Schlichtungsidee des Nukleolus implizit wiedergegeben, weil der am wenigsten zufriedenen Koalition S_1 das höchste Gewicht zugemessen wird usw. Das heisst, dass der Schlichter der Koalition S_1 die höchste Priorität und der Koalition S_{2^n} die niedrigste Priorität zuordnet. Das Gewichtungsschema (3.5) garantiert, dass der Nukleolus nicht nur zur lexikographischen Minimierung der maximalen Koalitionsüberschüsse führt, sondern auch die Summe der gewichteten Koalitionsüberschüsse minimiert. Eine Schwierigkeit besteht nun aber darin, dass im Voraus nicht bekannt ist, welche Koalition bezüglich des Nukleolus des Spiels den höchsten Überschuss besitzt, welche den zweithöchsten usw. Dieses Problem kann jedoch durch die Einbeziehung aller möglichen Permutationen der Koalitionen S_1, S_2, \dots, S_{2^n} überwunden werden. Kohlberg beweist, dass der Nukleolus diejenige Imputation ist, die für alle möglichen Permutationen der Koalitionen S_1, S_2, \dots, S_{2^n} die maximale Summe der gewichteten Koalitionsüberschüsse minimiert.

Will man den Nukleolus eines kooperativen Spiels mit n Spielern mit Hilfe des obigen linearen Programms berechnen, ist man mit den beiden folgenden Schwierigkeiten konfrontiert:

1. Schwierigkeit: Da in der Menge $\{S_1, S_2, \dots, S_{2^n}\}$ die Anzahl der Koalitionen 2^n ist, gibt es insgesamt $2^n!$ Permutationen dieser Koalitionen. Somit beinhaltet das lineare Programm insgesamt $2^n! + 1$ Nebenbedingungen (ohne die Nichtnegativitätsbedingungen (3.4)).³ Folglich steigt die Anzahl der Nebenbedingungen mit steigender

³Da der Überschuss der grossen Koalition N und der leeren Menge \emptyset für alle Imputationen null ist, könnte man diese Koalitionen im linearen Programm (3.1) – (3.4) ausser Acht lassen. Die Anzahl der Nebenbedingungen würde in diesem Fall $(2^n - 2)! + 1$ betragen.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Anzahl der Spieler exponentiell.⁴ Beispielsweise beträgt die Anzahl der Nebenbedingungen für Spiele mit nur $n = 3$ Spielern bereits $2^3! + 1 = 40321$ und für kooperative Spiele mit $n = 4$ Spielern gar $\sim 2.1 \times 10^{13}$.

2. Schwierigkeit: Die zweite Schwierigkeit besteht darin, dass die t -Koeffizienten sehr gross sind. Da wir jedes kooperative Spiel $(0 - 1)$ -normieren können, geht Kohlberg von $(0 - 1)$ -normierten Spielen aus und zeigt, dass der Koeffizient t_0 mindestens den Wert n^{2^n} besitzen muss, damit der Nukleolus die eindeutige Lösung des obigen linearen Programms ist. Demnach haben im linearen Programm die t -Koeffizienten die Werte $t = n^{2^n}, \dots, (n^{2^n})^{2^n}$. Beispielsweise nehmen für $n = 4$ die t -Koeffizienten Werte zwischen $\sim 10^4$ und $\sim 10^{77}$ an. Dies führt bei numerischen Berechnungen mit Gleitkommaarithmetik zu Rundungsfehlern. Aus diesen beiden Gründen ist der Algorithmus von Kohlberg praktisch nicht geeignet, den Nukleolus kooperativer Spiele mit mehr als drei Spielern zu berechnen.

3.1.2 Das Verfahren von Owen

Die zweite Methode, die wir zu den Verfahren der ersten Generation zählen, ist die Vorgehensweise nach Owen [96]. Owen geht von dem von Kohlberg [65] präsentierten linearen Programm (3.1) – (3.4) zur Berechnung des Nukleolus kooperativer Spiele mit n Spielern aus und zeigt, dass die Anzahl der Nebenbedingungen des linearen Programms auf $4^n + 1$ (für $n = 4$ insgesamt $4^4 + 1 = 257$ Nebenbedingungen) reduziert werden kann. Zu diesem Zweck stellt er ein zum linearen Programm (3.1) – (3.4) äquivalentes lineares Programm auf, in welchem aber die Variablen die gleichen hohen t -Koeffizienten besitzen wie in dem von Kohlberg vorgeschlagenen linearen Programm. Die Reduktion der Anzahl der Nebenbedingungen in dem noch vorzustellenden linearen Programm von Owen geschieht zu Lasten der Anzahl der Variablen. Während nämlich die lineare Optimierungsaufgabe von Owen $2^{n+1} + n$ Variablen besitzt, beträgt die Anzahl der Variablen im linearen Programm (3.1) – (3.4)

⁴Unter dem „exponentiellen“ Anstieg der Anzahl der Nebenbedingungen verstehen wir, dass die Anzahl der Restriktionen bei einer steigenden Anzahl der Spieler schneller als polynomial wächst. Weitere Ausführungen dazu sind im Anhang F zu finden.

3.1 Methoden der ersten Generation

von Kohlberg insgesamt nur $n + 1$. Wir stellen nun das von Owen vorgeschlagene lineare Programm vor.

Wir betrachten ein kooperatives Spiel $\Gamma = (N, v)$ mit n Spielern und der Spielermenge $N = \{1, 2, \dots, n\}$. Wir nehmen wieder an, dass S_1, S_2, \dots, S_{2^n} alle möglichen Koalitionen der Spielermenge N sind. Wie wir im vorangehenden Abschnitt erläutert haben, beweist Kohlberg, dass der Nukleolus derjenigen Imputation entspricht, die für alle möglichen Permutationen der Koalitionen S_1, S_2, \dots, S_{2^n} die maximale Summe der gewichteten Koalitionsüberschüsse minimiert. Die Menge aller Permutationen über 2^N sei Π . In diesem Fall können wir das von Kohlberg vorgeschlagene lineare Programm (3.1) – (3.4) als ein sog. Sattelpunktproblem⁵ wie folgt darstellen:

$$\min_{\mathbf{x} \in \mathcal{I}(N, v)} \max_{\pi \in \Pi} \left\{ \sum_{j=1}^{2^n} t^j e(S_{\pi(j)}, \mathbf{x}) \right\}. \quad (3.6)$$

Repräsentieren wir nun eine Permutation $\pi \in \Pi$ mit Hilfe einer $2^n \times 2^n$ -Permutationsmatrix \mathbf{P} , indem wir den Elementen p_{jl} ($j, l = 1, 2, \dots, 2^n$) dieser Matrix die Werte

$$p_{jl} = \begin{cases} 1, & \text{falls } \pi(j) = l \\ 0, & \text{sonst} \end{cases}$$

zuweisen, so können wir in (3.6) jeden Koalitionsüberschuss $e(S_{\pi(j)}, \mathbf{x})$ durch

$$\sum_{l=1}^{2^n} e(S_l, \mathbf{x}) p_{jl}$$

ersetzen. Da nun jede Spalte der Matrix \mathbf{P} nur eine Eins enthält und alle anderen Einträge der Spalte Nullen sind, ist die Summe der Einträge in jeder Spalte eins. Das Gleiche gilt analog auch für jede Zeile. Somit nimmt das Sattelpunktproblem (3.6) die folgende äquivalente Gestalt an:

⁵Für die Definition von Sattelpunktproblemen und die Umwandlung eines Sattelpunktproblems in eine lineare Optimierungsaufgabe vgl. Anhang C.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

$$\min_{\mathbf{x}} \max_{\mathbf{P}} \sum_{j=1}^{2^n} \sum_{l=1}^{2^n} t^j e(S_l, \mathbf{x}) p_{jl} \quad (3.7)$$

unter den Nebenbedingungen

$$\sum_{j=1}^{2^n} p_{jl} = 1, \quad l = 1, 2, \dots, 2^n, \quad (3.8)$$

$$\sum_{l=1}^{2^n} p_{jl} = 1, \quad j = 1, 2, \dots, 2^n, \quad (3.9)$$

$$p_{jl} \geq 0, \quad j, l = 1, 2, \dots, 2^n, \quad (3.10)$$

$$x_1 + x_2 + \dots + x_n = v(N), \quad (3.11)$$

$$x_i \geq 0, \quad i = 1, 2, \dots, n. \quad (3.12)$$

Durch die Anwendung der Dualitätstheorie der mathematischen Programmierung auf den Maximierungsteil können wir nun das obige Sattelpunktproblem als eine lineare Optimierungsaufgabe ausdrücken (vgl. dazu Anhang C):

$$\min_{\mathbf{x}, \mathbf{u}, \mathbf{w}} \sum_{j=1}^{2^n} u_j + \sum_{l=1}^{2^n} w_l \quad (3.13)$$

unter den Nebenbedingungen

$$u_j + w_l \geq t^j e(S_l, \mathbf{x}), \quad j, l = 1, 2, \dots, 2^n \quad (3.14)$$

$$x_1 + x_2 + \dots + x_n = v(N) \quad (3.15)$$

$$x_i \geq 0, \quad i = 1, 2, \dots, n. \quad (3.16)$$

Somit haben wir gezeigt, dass das lineare Programm (3.13) – (3.16) aus der Optimierungsaufgabe (3.6) bzw. aus dem von Kohlberg vorgeschlagenen linearen Programm (3.1) – (3.4) hergeleitet werden kann. Das von Owen [96] in die Literatur eingeführte lineare Programm (3.13) – (3.16) zur Berechnung des Nukleolus und das lineare Programm (3.1) – (3.4) sind äquivalent.

Die Anzahl der Variablen in der linearen Optimierungsaufgabe (3.13) – (3.16) beträgt $2^n + 2^n + n = 2^{n+1} + n$ und diejenige der Nebenbedingungen $2^n 2^n + 1 = 4^n + 1$ (ohne die Nichtnegativitätsbedingungen). Beispielsweise müsste zur Berechnung des

3.2 Methoden der zweiten Generation

Nukleolus eines kooperativen Spiels mit $n = 4$ Spielern mit Hilfe des Verfahrens von Owen ein lineares Programm mit $2^{n+1} + n = 36$ Variablen und $4^n + 1 = 257$ Nebenbedingungen gelöst werden. Wir stellen fest, dass die vorgestellte äquivalente Darstellung des linearen Programms von Kohlberg zwar die Anzahl der Nebenbedingungen von $2^n! + 1$ auf $4^n + 1$ reduziert hat, aber gleichzeitig die Anzahl der Variablen von $n + 1$ auf $2^{n+1} + n$ gestiegen ist. Zudem bleibt die Problematik der hohen Werte der t -Koeffizienten bestehen. Daher stellt das Verfahren von Owen, wie das Verfahren von Kohlberg, keine praktikable Methode zur Bestimmung des Nukleolus dar. Diese Verfahren sind aber insofern von Bedeutung, als sie aufzeigen, dass zur Berechnung des Nukleolus Methoden der linearen Optimierung herangezogen werden können. Wie wir noch vorstellen werden, bilden die Verfahren der ersten Generation folglich die Grundlage von mehreren Verfahren, die in der Literatur zur Berechnung des Nukleolus kooperativer Spiele vorgeschlagen werden. Darüberhinaus legen die Verfahren der ersten Generation dar, dass der Nukleolus als eindeutige Lösung eines einzigen linearen Programms berechnet werden kann. Wegen der oben geschilderten Nachteile der Verfahren erster Generation wurden in der Literatur andere Verfahren zur Berechnung des Nukleolus entwickelt. In den nächsten zwei Abschnitten werden wir verschiedene Algorithmen vorstellen, die die Nachteile der Verfahren erster Generation beseitigen sollen.

3.2 Methoden der zweiten Generation

3.2.1 Das Verfahren von Bruyneel

Die Methoden der zweiten Generation sind dadurch gekennzeichnet, dass sie zur Berechnung des Nukleolus kooperativer Spiele minimal balancierte Koalitionssysteme heranziehen. Eines von insgesamt zwei Verfahren, das den Nukleolus mit Hilfe minimal balancierter Koalitionssysteme berechnet, ist das in diesem Abschnitt vorgestellte von Bruyneel [22]. Das andere, das wir ebenfalls zur zweiten Generation zählen, ist das Verfahren von Dragan [35], das wir im nächsten Abschnitt erörtern werden.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Bevor wir die einzelnen Schritte des Verfahrens von Bruyneel [22] erläutern, beschreiben wir die Grundidee des Algorithmus. Die Vorgehensweise von Bruyneel [22] zur Berechnung des Nukleolus eines kooperativen Spiels ist die bei der Einführung des Nukleolus im Abschnitt 2.4 dargelegte Schlichtermethode. Bei der Umsetzung der Schlichtermethode zur Berechnung des Nukleolus setzt das Verfahren von Bruyneel voraus, dass alle minimal balancierten Koalitionssysteme und ihre Gewichtsvektoren bekannt sind.⁶ Diese Vorbedingung beeinträchtigt die Praktikabilität des Verfahrens erheblich. Wie noch gezeigt wird, steigt nämlich die Anzahl der minimal balancierten Koalitionssysteme mit zunehmender Spielerzahl exponentiell an, sodass das Verfahren von Bruyneel [22] zur Berechnung des Nukleolus von Spielen mit lediglich kleiner Spielermenge N geeignet scheint.

Sind alle minimal balancierten Koalitionssysteme und ihre Gewichtsvektoren bekannt, so bestimmt Bruyneel [22] mit deren Hilfe in einer ersten Phase unter allen möglichen Koalitionen $S \in 2^N$ diejenigen, deren minimaler Koalitionsüberschuss am grössten ist. Anschliessend modifiziert er die charakteristische Funktion des Spiels. Die modifizierte charakteristische Funktion gewährleistet, dass der minimale Koalitionsüberschuss dieser Koalitionen bewahrt bleibt. Für die weiteren Schritte sind somit nur jene Imputationen zulässig, die den Überschuss der am wenigsten zufriedenen Koalitionen minimieren. In der zweiten Phase werden sodann die Koalitionen mit dem zweithöchsten Koalitionsüberschuss bestimmt. Anschliessend modifiziert man erneut die charakteristische Funktion usw. Ist der Koalitionsüberschuss jeder Koalition $S \in 2^N$ minimiert, so kann der Nukleolus des Spiels in der dritten Phase, wie wir noch zeigen werden, aus der zuletzt modifizierten charakteristischen Funktion abgelesen werden.

Es seien nun $\Gamma = (N, v)$ ein kooperatives Spiel mit der Spielermenge $N = \{1, 2, \dots, n\}$ und $\mathcal{B} = \{S_1, S_2, \dots, S_m\}$ ein minimal balanciertes Koalitionssystem mit den Gewichten $\alpha_1, \alpha_2, \dots, \alpha_m$.⁷ Wie wir auf S. 40 f. erläutert haben, stiftet in diesem

⁶Eine Methode zur Bestimmung aller minimal balancierten Koalitionssysteme einer Menge $N = \{1, 2, \dots, n\}$ finden wir in Peleg [101].

⁷Der Einfachheit halber werden wir, wenn dies nicht zu Verwechslungen führt, das Gewicht einer Koalition $S \in \mathcal{B}$ auch als α_S bezeichnen.

3.2 Methoden der zweiten Generation

Fall jede Koalition $S \in \mathcal{B}$ dem Koalitionssystem \mathcal{B} bezüglich der jeweiligen Imputation $\mathbf{x} \in \mathcal{I}(N, v)$ den Koalitionsüberschuss $\alpha_S e(S, \mathbf{x})$. Um diejenigen Koalitionen zu bestimmen, deren minimale Koalitionsüberschüsse am höchsten sind, betrachtet Bruyneel [22] zunächst die Summe aller dieser gewichteten Koalitionsüberschüsse, also

$$\sum_{S \in \mathcal{B}} \alpha_S e(S, \mathbf{x}). \quad (3.17)$$

Diese Summe kann als der Überschuss des balancierten Koalitionssystems \mathcal{B} bezüglich der Imputation \mathbf{x} interpretiert werden. Bruyneel [22] zeigt nun, dass der Überschuss jedes balancierten Koalitionssystems \mathcal{B} von der gewählten Imputation \mathbf{x} unabhängig ist. Somit hat der Überschuss stets denselben Wert. Die Gültigkeit dieser Aussage können wir mit Hilfe der Definitionen des Koalitionsüberschusses und der balancierten Koalitionssysteme folgendermassen beweisen. Für den Ausdruck (3.17) gilt

$$\sum_{S \in \mathcal{B}} \alpha_S e(S, \mathbf{x}) = \sum_{S \in \mathcal{B}} \alpha_S [v(S) - x(S)] = \sum_{S \in \mathcal{B}} \alpha_S v(S) - \sum_{S \in \mathcal{B}} \alpha_S x(S). \quad (3.18)$$

Es sei nun $\mathbf{1}_S := (1_S(1), 1_S(2), \dots, 1_S(n))$ der sog. Inzidenzvektor der Koalition $S \subset N$, wobei $1_S(i) = 1$ für $i \in S$ und $1_S(i) = 0$ für $i \notin S$ ist. Dann können wir die zweite Summe auf der rechten Seite der Gleichung (3.18) wie folgt umschreiben:

$$\begin{aligned} \sum_{S \in \mathcal{B}} \alpha_S x(S) &= \alpha_1 x(S_1) + \alpha_2 x(S_2) + \dots + \alpha_m x(S_m) \\ &= \alpha_1 \sum_{i \in S_1} x_i + \alpha_2 \sum_{i \in S_2} x_i + \dots + \alpha_m \sum_{i \in S_m} x_i \\ &= x_1 \underbrace{\sum_{j=1}^m \alpha_j 1_{S_j}(1)}_{=1} + x_2 \underbrace{\sum_{j=1}^m \alpha_j 1_{S_j}(2)}_{=1} + \dots + x_n \underbrace{\sum_{j=1}^m \alpha_j 1_{S_j}(n)}_{=1} \\ &= x_1 + x_2 + \dots + x_n \\ &= \sum_{i=1}^n x_i = v(N). \end{aligned} \quad (3.19)$$

Das zweite Gleichheitszeichen in (3.19) gilt, weil \mathbf{x} eine Imputation ist. Setzen wir (3.19) in die Gleichung (3.18) ein, dann erhalten wir

$$t(\mathcal{B}) := \sum_{S \in \mathcal{B}} \alpha_S e(S, \mathbf{x}) = \sum_{S \in \mathcal{B}} \alpha_S v(S) - v(N), \quad \forall \mathbf{x} \in \mathcal{I}(N, v). \quad (3.20)$$

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Da die rechte Seite der Gleichung (3.20) von \mathbf{x} unabhängig ist, ist der Überschuss des balancierten Koalitionssystems \mathcal{B} , also $t(\mathcal{B})$, bezüglich aller $\mathbf{x} \in \mathcal{I}(N, v)$ konstant. \square

Zur Veranschaulichung betrachten wir unser Einführungsbeispiel und rechnen für ein balanciertes Koalitionssystem \mathcal{B} den Wert von $t(\mathcal{B})$ aus. Zur Erinnerung geben wir die charakteristische Funktion des Spiels nochmals an:

$$\begin{aligned} v(\{1\}) &= 0, & v(\{2\}) &= 0, & v(\{3\}) &= 0, \\ v(\{1, 2\}) &= 6, & v(\{1, 3\}) &= 16, & v(\{2, 3\}) &= 26, \\ v(\{1, 2, 3\}) &= 42. \end{aligned}$$

Man kann zeigen,⁸ dass für die Spielermenge $N = \{1, 2, 3\}$ ausschliesslich die folgenden Koalitionssysteme für die jeweiligen (eindeutigen) Gewichtsvektoren α minimal balanciert sind:

$$\begin{array}{llll} \mathcal{B}_1 &= \{\{1\}, \{2, 3\}\} & \text{mit } \alpha &= (1, 1), \\ \mathcal{B}_2 &= \{\{2\}, \{1, 3\}\} & \text{mit } \alpha &= (1, 1), \\ \mathcal{B}_3 &= \{\{3\}, \{1, 2\}\} & \text{mit } \alpha &= (1, 1), \\ \mathcal{B}_4 &= \{\{1\}, \{2\}, \{3\}\} & \text{mit } \alpha &= (1, 1, 1), \\ \mathcal{B}_5 &= \{\{1, 2\}, \{1, 3\}, \{2, 3\}\} & \text{mit } \alpha &= (1/2, 1/2, 1/2). \end{array} \quad (3.21)$$

Mit Hilfe der Gleichung (3.20) bestimmen wir exemplarisch den Überschuss des minimal balancierten Koalitionssystems \mathcal{B}_5 :

$$t(\mathcal{B}_5) = \frac{1}{2}6 + \frac{1}{2}16 + \frac{1}{2}26 - 42 = -18, \quad \forall \mathbf{x} \in \mathcal{I}(N, v). \quad (3.22)$$

Das Koalitionssystem \mathcal{B}_5 besitzt also stets den Überschuss -18 .

Wir zeigen nun, dass dank dieser Eigenschaft balancierter Koalitionssysteme \mathcal{B} in der ersten Phase der Methode von Bruyneel diejenigen Koalitionen ermittelt werden können, deren minimale Koalitionsüberschüsse am höchsten sind. Bruyneel definiert zu diesem Zweck

⁸Vgl. dazu Beispiel 2.3.3, S. 39.

3.2 Methoden der zweiten Generation

$$\varphi(\mathcal{B}) := \begin{cases} \frac{t(\mathcal{B})}{\sum_{S \in \mathcal{B}} \alpha_S}, & \text{falls } t(\mathcal{B}) \leq 0, \\ \frac{t(\mathcal{B})}{\sum_{\substack{S \in \mathcal{B} \\ |S| \geq 2}} \alpha_S}, & \text{falls } t(\mathcal{B}) > 0. \end{cases} \quad (3.23)$$

Wir erläutern zunächst, was die Fallunterscheidung bei der Definition von $\varphi(\mathcal{B})$ beinhaltet und zeigen anschliessend, welche wichtige Eigenschaft $\varphi(\mathcal{B})$ besitzt. Für den Fall $t(\mathcal{B}) \leq 0$ kann der Wert $\varphi(\mathcal{B})$ als ein gewogenes arithmetisches Mittel⁹ der Koalitionsüberschüsse $e(S, \mathbf{x})$, $S \in \mathcal{B}$, interpretiert werden, denn es gilt

$$\frac{t(\mathcal{B})}{\sum_{S \in \mathcal{B}} \alpha_S} = \frac{\alpha_1 e(S_1, \mathbf{x}) + \alpha_2 e(S_2, \mathbf{x}) + \dots + \alpha_m e(S_m, \mathbf{x})}{\alpha_1 + \alpha_2 + \dots + \alpha_m}, \quad (3.24)$$

wobei sich die Koeffizienten $\frac{\alpha_j}{\sum_{k=1}^m \alpha_k}$, $j = 1, 2, \dots, m$, zu eins addieren. Ist hingegen $t(\mathcal{B}) = \sum_{S \in \mathcal{B}} \alpha_S v(S) - v(N) > 0$, so gilt $\sum_{S \in \mathcal{B}} \alpha_S v(S) > v(N)$. Gemäss dem Bondavera-Shapley-Satz (vgl. Satz 2.3.2, S. 43) ist in diesem Fall der Kern des Spiels leer. Folglich ist mindestens eine der Kernbedingungen verletzt. Mit anderen Worten gibt es mindestens eine Koalition $S \subset N$, für die die Kernbedingung $v(S) \leq x(S)$ für keinen Allokationsvektor \mathbf{x} erfüllt werden kann. Ist aber die Imputationsmenge des Spiels nicht-leer, so muss $v(\{i\}) \leq x(\{i\})$ für alle $i \in N$ gelten. Um nun bei der Berechnung von $\varphi(\mathcal{B})$ von effizienten Allokationsvektoren auszugehen, die auch individuell rational sind, unterscheidet Bruyneel in (3.23) zwischen Ein- und Mehr-Spieler-Koalitionen.

Wie wir sogleich zeigen werden, hat der oben definierte Wert $\varphi(\mathcal{B})$ die Eigenschaft, dass der Überschuss keiner Koalition $S \in \mathcal{B}$ und damit deren Unzufriedenheit kleiner sein kann als $\varphi(\mathcal{B})$. Dies gilt für alle Imputationen $\mathbf{x} \in \mathcal{I}(N, v)$. Der Wert $\varphi(\mathcal{B})$ ist also

⁹Allgemein ist ein *gewichtetes* oder *gewogenes arithmetisches Mittel* der Zahlen y_1, y_2, \dots, y_n für nicht-negative Gewichte g_1, g_2, \dots, g_n , $\sum_{i=1}^n g_i = 1$, durch $\bar{y}_g := \sum_{i=1}^n g_i y_i$ definiert. Man kann zeigen, dass der Wert \bar{y}_g nicht kleiner als $\min\{y_1, y_2, \dots, y_n\}$ und nicht grösser als $\max\{y_1, y_2, \dots, y_n\}$ sein kann. Es gilt also $\bar{y}_g \in [\min\{y_1, y_2, \dots, y_n\}, \max\{y_1, y_2, \dots, y_n\}]$. Vgl. dazu Bamberg/Baur [4], S. 17, Bösel [23], S. 51 f., und Bortz [15], S. 53 f.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

der kleinste Koalitionsüberschuss, der von jeder Koalition $S \in \mathcal{B}$ realisiert werden kann. Folglich ist $\varphi(\mathcal{B})$ eine untere Schranke für die Überschüsse der Koalitionen $S \in \mathcal{B}$. Das können wir wie folgt beweisen:

Gegeben sei $b := \max_{S \in \mathcal{B}} e(S, \mathbf{x})$, $\forall \mathbf{x} \in \mathcal{I}(N, v)$. Wir wollen zeigen, dass für alle Imputationen der Überschuss jeder Koalition $S \in \mathcal{B}$ nicht grösser ist als $\varphi(\mathcal{B})$:

$$\varphi(\mathcal{B}) \leq b, \quad \forall S \in \mathcal{B}, \quad \forall \mathbf{x} \in \mathcal{I}(N, v).$$

Wir treffen die folgende Fallunterscheidung:

- Falls $t(\mathcal{B}) \leq 0$ ist, müssen wir nachweisen, dass

$$\varphi(\mathcal{B}) := \frac{t(\mathcal{B})}{\sum_{S \in \mathcal{B}} \alpha_S} \leq b \Rightarrow t(\mathcal{B}) \leq \sum_{S \in \mathcal{B}} \alpha_S b. \quad (3.25)$$

In Anwendung der Definitionen von $t(\mathcal{B})$ und b können wir

$$t(\mathcal{B}) = \sum_{S \in \mathcal{B}} \alpha_S \underbrace{e(S, \mathbf{x})}_{\leq b} \leq \sum_{S \in \mathcal{B}} \alpha_S b \quad (3.26)$$

schreiben. Somit ist die Ungleichung (3.25) erfüllt.

- Falls $t(\mathcal{B}) > 0$ ist, müssen wir beweisen, dass

$$\varphi(\mathcal{B}) := \frac{t(\mathcal{B})}{\sum_{\substack{S \in \mathcal{B} \\ |S| \geq 2}} \alpha_S} \leq b \Rightarrow t(\mathcal{B}) \leq \sum_{\substack{S \in \mathcal{B} \\ |S| \geq 2}} \alpha_S b.$$

Für den Beweis ziehen wir die Definitionen von $t(\mathcal{B})$ und b heran. Zudem gilt auf Grund der individuellen Rationalität der Imputationen $\mathbf{x} \in \mathcal{I}(N, v)$ die Ungleichung $e(S, \mathbf{x}) \leq 0$ für alle $S \in \mathcal{B}$ mit $|S| = 1$:

$$t(\mathcal{B}) = \sum_{S \in \mathcal{B}} \alpha_S e(S, \mathbf{x}) = \sum_{\substack{S \in \mathcal{B} \\ |S|=1}} \alpha_S \underbrace{e(S, \mathbf{x})}_{\leq 0} + \sum_{\substack{S \in \mathcal{B} \\ |S| \geq 2}} \alpha_S \underbrace{e(S, \mathbf{x})}_{\leq b} \leq \sum_{\substack{S \in \mathcal{B} \\ |S| \geq 2}} \alpha_S b. \quad \square$$

3.2 Methoden der zweiten Generation

Wir betrachten beispielsweise das für den Gewichtsvektor $\alpha = (1/2, 1/2, 1/2)$ minimal balancierte Koalitionssystem $\mathcal{B}_5 = \{\{1, 2\}, \{1, 3\}, \{2, 3\}\}$ des Einführungsbeispiels. Der kleinste Koalitionsüberschuss, der von jeder Koalition $S \in \mathcal{B}_5$ realisiert werden kann, beträgt

$$\varphi(\mathcal{B}_5) = \frac{t(\mathcal{B}_5)}{\sum_{S \in \mathcal{B}_5} \alpha_S} = \frac{-18}{\frac{1}{2} + \frac{1}{2} + \frac{1}{2}} = -12.$$

Es sei $\mathcal{M} (\neq \emptyset)$ die Menge aller minimal balancierten Koalitionssysteme der Spielermenge N . Bruyneel [22] berechnet nun für alle minimal balancierten Koalitionssysteme $\mathcal{B} \in \mathcal{M}$ den jeweiligen Wert $\varphi(\mathcal{B})$. Wegen des Tatbestands, dass für jedes minimal balancierte Koalitionssystem \mathcal{B} der Wert $\varphi(\mathcal{B})$ über die Imputationsmenge konstant ist und die Menge \mathcal{M} alle Koalitionen $S \in 2^N$ enthält, ist das Maximum unter den gefundenen Werten $\varphi(\mathcal{B})$ derjenige Koalitionsüberschuss, der von jeder Koalition $S \in 2^N$ in der Imputationsmenge erzielt werden kann. Dieser maximale Wert von $\varphi(\mathcal{B})$ für alle $\mathcal{B} \in \mathcal{M}$ sei

$$\epsilon^1 := \max_{\mathcal{B} \in \mathcal{M}} \varphi(\mathcal{B}). \quad (3.27)$$

Der Wert ϵ^1 entspricht also dem kleinstmöglichen Koalitionsüberschuss, der von jeder Koalition $S \in 2^N$ über die Imputationsmenge $\mathcal{I}(N, v)$ erzielt werden kann. Da nun der Überschuss der den minimal balancierten Koalitionssystemen $\mathcal{B} \in \mathcal{M}$ mit $\varphi(\mathcal{B}) = \epsilon^1$ angehörenden Koalitionen nicht kleiner sein kann als $\varphi(\mathcal{B})$, ist der kleinste realisierbare Überschuss dieser Koalitionen ϵ^1 . Die Menge dieser Koalitionen sei

$$\mathcal{S}_1 := \{S \mid S \in \mathcal{B} \in \mathcal{T}_1\} \quad \text{mit} \quad \mathcal{T}_1 := \{\mathcal{B} \in \mathcal{M} \mid \varphi(\mathcal{B}) = \epsilon^1\}. \quad (3.28)$$

Somit beinhaltet die Menge \mathcal{S}_1 jene Koalitionen, deren minimaler Koalitionsüberschuss am höchsten ist. Zur Veranschaulichung bestimmen wir die Menge \mathcal{S}_1 unseres Einführungsbeispiels.

Alle minimal balancierten Koalitionssysteme der Spielermenge $N = \{1, 2, 3\}$ wurden auf Seite 84 aufgelistet: $\mathcal{M} = \{\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, \mathcal{B}_4, \mathcal{B}_5\}$. Für diese Menge berechnen wir die folgenden Werte

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

$$\begin{aligned}
t(\mathcal{B}_1) &= 1 \cdot 0 + 1 \cdot 26 - 42 = -16 & \Rightarrow & \varphi(\mathcal{B}_1) = -16/2 = -8, \\
t(\mathcal{B}_2) &= 1 \cdot 0 + 1 \cdot 16 - 42 = -26 & \Rightarrow & \varphi(\mathcal{B}_2) = -26/2 = -13, \\
t(\mathcal{B}_3) &= 1 \cdot 0 + 1 \cdot 6 - 42 = -36 & \Rightarrow & \varphi(\mathcal{B}_3) = -36/2 = -18, \\
t(\mathcal{B}_4) &= 1 \cdot 0 + 1 \cdot 0 + 1 \cdot 0 - 42 = -42 & \Rightarrow & \varphi(\mathcal{B}_4) = -42/3 = -14, \\
t(\mathcal{B}_5) &= \frac{1}{2}6 + \frac{1}{2}16 + \frac{1}{2}26 - 42 = -18 & \Rightarrow & \varphi(\mathcal{B}_5) = -18/1.5 = -12.
\end{aligned}$$

Der kleinstmögliche Koalitionsüberschuss, der von allen Koalitionen des Einführungsbeispiels erzielt werden kann, beträgt somit

$$\epsilon^1 := \max_{\mathcal{B} \in \mathcal{M}} \varphi(\mathcal{B}) = -8. \quad (3.29)$$

Unter allen minimal balancierten Koalitionssystemen besitzt nur \mathcal{B}_1 die Eigenschaft $\varphi(\mathcal{B}_1) = \epsilon^1 = -8$. Demzufolge gilt

$$\mathcal{T}_1 = \{\mathcal{B}_1\} \text{ und } \mathcal{S}_1 = \{\{1\}, \{2, 3\}\}.$$

Die Koalitionen $\{1\}, \{2, 3\}$ sind also diejenigen Koalitionen des Einführungsbeispiels, deren minimaler Überschuss mit -8 am höchsten ist.

Um im nächsten Schritt die Koalitionen zu ermitteln, deren minimaler Koalitionsüberschuss am zweithöchsten ist, modifiziert Bruyneel die charakteristische Funktion des Spiels, sodass der minimale Überschuss -8 der Koalitionen $S \in \mathcal{S}_1$ bewahrt bleibt und somit die minimal balancierten Koalitionssysteme $\mathcal{B} \in \mathcal{T}_1$ und die Koalitionen $S \in \mathcal{S}_1$ von der weiteren Betrachtung ausgeschlossen werden können. Die Modifizierung der charakteristischen Funktion v des Spiels $\Gamma = (N, v)$ erfolgt durch

$$w(S) := \begin{cases} v(S), & \text{falls } S \notin \mathcal{S}_1, \\ v(S), & \text{falls } S \in \mathcal{S}_1, \epsilon^1 > 0, |S| = 1, \\ v(S) - \epsilon^1, & \text{sonst.} \end{cases} \quad (3.30)$$

Die modifizierte charakteristische Funktion $w(S)$ lässt also die Werte der Koalitionen $S \notin \mathcal{S}_1$ unverändert. Um die individuelle Rationalität der Auszahlungsvektoren zu gewährleisten, werden die Koalitionswerte der Einerkoalitionen $S \in \mathcal{S}_1, |S| = 1$, ebenfalls nicht geändert, falls $\epsilon^1 > 0$ ist. Die Werte $v(S)$ aller übrigen Koalitionen

3.2 Methoden der zweiten Generation

$S \in \mathcal{S}_1$ werden durch $w(S) := v(S) - \epsilon^1$ ersetzt, weil auf Grund von (3.28) für $S \in \mathcal{S}_1$ und $\mathbf{x} \in \mathcal{I}(N, v)$ gilt

$$e(S, \mathbf{x}) = v(S) - x(S) = \epsilon^1 \Rightarrow x(S) = \sum_{i \in S} x_i = v(S) - \epsilon^1.$$

Für das Einführungsbeispiel nimmt die modifizierte charakteristische Funktion die folgende Gestalt an:

$$\begin{aligned} w(\{1\}) &:= 0 - (-8) = 8, & w(\{1, 2\}) &:= 6, \\ w(\{2\}) &:= 0, & w(\{1, 3\}) &:= 16, \\ w(\{3\}) &:= 0, & w(\{2, 3\}) &:= 26 - (-8) = 34. \end{aligned}$$

Auf Grund der individuellen Rationalität und der Gruppenrationalität von Auszahlungsvektoren impliziert diese modifizierte charakteristische Funktion, dass nunmehr diejenigen Imputationen zur Aufteilung des Wertes der grossen Koalition in Frage kommen, die der Einerkoalition $\{1\}$ den Wert 8 und der Koalition $\{2, 3\}$ den Wert 34 zuteilen. Während wir noch nicht sagen können, wie der Wert 34 der Koalition $\{2, 3\}$ auf die Spieler 2 und 3 aufgeteilt werden soll, können wir aber bereits feststellen, dass die Auszahlung des Spielers 1 im Nukleolus 8 betragen muss. Wegen der oben erläuterten Eigenschaft von ϵ^1 ist dieser Wert die höchstmögliche realisierbare Auszahlung für diesen Spieler.

Nach der Modifikation der charakteristischen Funktion des Spiels werden im nächsten Schritt der Wert ϵ^2 sowie die Mengen \mathcal{T}_2 und \mathcal{S}_2 bestimmt, falls die Menge \mathcal{S}_1 nicht alle Koalitionen $S \subset N$ enthält. In diesem Schritt werden diejenigen Koalitionen ermittelt, deren Koalitionsüberschuss am zweithöchsten ist. Analog zu oben wird anschliessend die charakteristische Funktion $w(S)$ modifiziert. Für die Bestimmung der Koalitionen mit dem dritthöchsten Koalitionsüberschuss können somit im nächsten Schritt die minimal balancierten Koalitionssysteme $\mathcal{B} \in \mathcal{T}_2$ und die Koalitionen $S \in \mathcal{S}_2$ ausser Acht gelassen werden. Falls noch Koalitionen existieren, die bisher nicht ausgeschlossen werden konnten, müssen weitere Schritte durchgeführt werden, bis die minimalen Koalitionsüberschüsse aller Koalitionen gefunden worden sind. Bruyneel [22], S. 46 ff., beweist, dass in diesem Fall die modifizierten Werte

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

der Einerkoalitionen dem Nukleolus des Spiels entsprechen. Ist also $w^*(\{j\})$, für alle $j \in N$, der Wert der Einerkoalition $\{j\}$ nach der Modifikation der charakteristischen Funktion im letzten Schritt m , so ist der Nukleolus des Spiels gegeben durch

$$\mathcal{N} = (w^*(\{1\}), w^*(\{2\}), \dots, w^*(\{n\})). \quad (3.31)$$

Bruyneel [22], S. 37 ff., weist ebenfalls nach, dass in jedem Schritt i die Mengen \mathcal{T}_i und \mathcal{S}_i nicht-leer sind, $\epsilon^1 > \epsilon^2 > \dots > \epsilon^m$ mit $\epsilon^m < 0$ gilt und ferner die Anzahl der Schritte m endlich ist.

Wir fassen die einzelnen Schritte des Algorithmus von Bruyneel [22] zusammen. Es seien \mathcal{T}^i die Menge aller minimal balancierten Koalitionssysteme, die ab dem i -ten Schritt ausser Acht gelassen werden, und Δ^i die Menge der Koalitionen, deren maximale Überschüsse bereits minimiert worden sind. Das Verfahren von Bruyneel führt zum Nukleolus eines kooperativen Spiels mit n Spielern, indem es die Mengen \mathcal{S}_i und Δ^i sowie die Mengen \mathcal{T}_i und \mathcal{T}^i für $i = 0, 1, \dots, m$ durch folgende Schritte rekursiv konstruiert:

Initialisierung

- $\mathcal{S}_0 := \{\emptyset, N\}$, $\Delta^0 := \{\emptyset, N\}$, $\mathcal{T}_0 := \emptyset$, $\mathcal{T}^0 := \emptyset$,
- $i := 1$, $w_1(S) := v(S)$ für alle $S \notin \mathcal{S}_0$.

Phase 1

Für alle $\mathcal{B} \in \mathcal{M} \setminus \mathcal{T}^0$ werden zunächst die nachstehenden Grössen bestimmt:

- $t_1(\mathcal{B}) := \sum_{S \in \mathcal{B}} \alpha_S w_1(S) - v(N)$,
- $\varphi_1(\mathcal{B}) := \begin{cases} \frac{t_1(\mathcal{B})}{\sum_{\substack{S \in \mathcal{B} \\ S \notin \Delta^0}} \alpha_S}, & \text{falls } t_1(\mathcal{B}) \leq 0 \\ \frac{t_1(\mathcal{B})}{\sum_{\substack{S \in \mathcal{B} \\ S \notin \Delta^0 \\ |S| \geq 2}} \alpha_S}, & \text{falls } t_1(\mathcal{B}) > 0, \end{cases}$

3.2 Methoden der zweiten Generation

- $\epsilon^1 := \max_{\mathcal{B} \in \mathcal{M} \setminus \mathcal{T}^0} \varphi_1(\mathcal{B}),$
- $\mathcal{T}_1 := \{\mathcal{B} \notin \mathcal{T}^0 \mid \varphi_1(\mathcal{B}) = \epsilon_1\}$ und $\mathcal{T}^1 := \mathcal{T}^0 \cup \mathcal{T}_1,$
- $\mathcal{S}_1 := \{S \notin \Delta^0 \mid S \in \mathcal{B} \in \mathcal{T}_1\}$ und $\Delta^1 := \Delta^0 \cup \mathcal{S}_1.$

Phase 2

In der zweiten Phase werden die folgenden Grössen rekursiv festgelegt:

- Für alle $S \notin \mathcal{S}_0$

$$w_{i+1}(S) := \begin{cases} w_i(S), & \text{falls } S \notin \mathcal{S}_i \\ w_i(S), & \text{falls } S \in \mathcal{S}_i, \epsilon^i > 0, |S| = 1 \\ w_i(S) - \epsilon^i, & \text{sonst,} \end{cases}$$

- $t_{i+1}(\mathcal{B}) := \sum_{S \in \mathcal{B}} \alpha_S w_{i+1}(S) - v(N),$ für alle $\mathcal{B} \notin \mathcal{T}^i,$
- Für alle $\mathcal{B} \notin \mathcal{T}^i$

$$\varphi_{i+1}(\mathcal{B}) := \begin{cases} \frac{t_{i+1}(\mathcal{B})}{\sum_{\substack{S \in \mathcal{B} \\ S \notin \Delta^i}} \alpha_S}, & \text{falls } t_{i+1}(\mathcal{B}) \leq 0 \\ \frac{t_{i+1}(\mathcal{B})}{\sum_{\substack{S \in \mathcal{B} \\ S \notin \Delta^i \\ |S| \geq 2}} \alpha_S}, & \text{falls } t_{i+1}(\mathcal{B}) > 0, \end{cases}$$

- $\epsilon^{i+1} := \max_{\mathcal{B} \notin \mathcal{T}^i} \varphi_{i+1}(\mathcal{B}),$
- $\mathcal{T}_{i+1} := \{\mathcal{B} \notin \mathcal{T}^i \mid \varphi_{i+1}(\mathcal{B}) = \epsilon^{i+1}\}$ und $\mathcal{T}^{i+1} := \mathcal{T}^i \cup \mathcal{T}_{i+1},$
- $\mathcal{S}_{i+1} := \{S \notin \Delta^i \mid S \in \mathcal{B} \in \mathcal{T}_{i+1}\}$ und $\Delta^{i+1} := \Delta^i \cup \mathcal{S}_{i+1}.$

Die Phase 2 wird solange durchgeführt, bis $\Delta^i = 2^N$ erreicht ist, wobei nach jeder Iteration der Zähler i um eins erhöht wird.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Phase 3

Sei m die letzte Iteration, für die $\Delta^m = 2^N$ erfüllt ist. In der dritten Phase wird die charakteristische Funktion für alle $S \notin \mathcal{S}_0$ folgendermassen modifiziert:

$$w_{m+1}(S) := \begin{cases} w_m(S), & \text{falls } S \notin \mathcal{S}_m \\ w_m(S) - \epsilon_m, & \text{falls } S \in \mathcal{S}_m. \end{cases}$$

Anschliessend wird $w^*(S) := w_{k+1}(S)$ definiert, für alle $S \in \mathcal{S}_k$, $k = 1, 2, \dots, m$. Der Nukleolus des Spiels ist dann gegeben durch (vgl. Bruyneel [22], S. 46)

$$\hat{\mathbf{w}} := (w^*(\{1\}), w^*(\{2\}), \dots, w^*(\{n\})).$$

Zur Veranschaulichung des vorgestellten Algorithmus von Bruyneel berechnen wir den Nukleolus des Einführungsbeispiels.

Beispiel 3.2.1

Wir haben oben bereits die erste Phase des Algorithmus durchgeführt. Dabei haben wir festgestellt, dass $\epsilon^1 = -8$, $\mathcal{T}_1 := \{\mathcal{B}_1\}$ und $\mathcal{S}_1 := \{\{1\}, \{2, 3\}\}$ ist. Demzufolge gilt

$$\mathcal{T}^1 := \mathcal{T}^0 \cup \mathcal{T}_1 = \{\mathcal{B}_1\} \text{ und } \Delta^1 := \Delta^0 \cup \mathcal{S}_1 = \{\{1\}, \{2, 3\}, \emptyset, N\}.$$

Zu Beginn der zweiten Phase erhöhen wir den Index i um eins, also auf $i = 2$. Auf S. 89 haben wir die charakteristische Funktion modifiziert. Wegen der Initialisierung $w_1(S) := v(S)$, $\forall S \notin \mathcal{S}_0$, entspricht diese modifizierte charakteristische Funktion im obigen Ablauf des Algorithmus der charakteristischen Funktion w_2 . Mit Hilfe von w_2 bestimmen wir $\varphi_2(\mathcal{B})$, $\forall \mathcal{B} \notin \mathcal{T}^1$:

$$\begin{aligned} t_2(\mathcal{B}_2) &= -26 &\Rightarrow \varphi_2(\mathcal{B}_2) &= -26/2 = -13, \\ t_2(\mathcal{B}_3) &= -36 &\Rightarrow \varphi_2(\mathcal{B}_3) &= -36/2 = -18, \\ t_2(\mathcal{B}_4) &= -34 &\Rightarrow \varphi_2(\mathcal{B}_4) &= -34/2 = -17, \\ t_2(\mathcal{B}_5) &= -14 &\Rightarrow \varphi_2(\mathcal{B}_5) &= -14/1 = -14. \end{aligned}$$

Somit erhalten wir $\epsilon^2 := -13$, $\mathcal{T}_2 := \{\mathcal{B}_2\}$ und $\mathcal{S}_2 := \{\{2\}, \{1, 3\}\}$ mit

$$\mathcal{T}^2 := \mathcal{T}^1 \cup \mathcal{T}_2 = \{\mathcal{B}_1, \mathcal{B}_2\} \text{ und } \Delta^2 := \Delta^1 \cup \mathcal{S}_2 = \{\{1\}, \{2, 3\}, \{2\}, \{1, 3\}, \emptyset, N\}.$$

3.2 Methoden der zweiten Generation

Wir haben also diejenigen Koalitionen ermittelt, deren minimaler Überschuss am zweithöchsten ist, also $\epsilon^2 = -13$ beträgt. Es handelt sich um die Koalitionen $\{2\}$ und $\{1, 3\}$. Der Überschuss der Einerkoalition $\{2\}$ ist demnach nur dann am geringsten, wenn seine Auszahlung 13 ist. Mit anderen Worten: Dem Spieler 2 muss gemäss dem Konzept des Nukleolus die Auszahlung 13 zugeteilt werden.¹⁰ Da $\Delta^2 \neq 2^N$ ist, führen wir den nächsten Schritt durch, wobei jetzt $i = 3$ ist. Wir modifizieren nun die charakteristische Funktion w_2 , sodass jede Koalition $S \in \Delta^2$ ihren geringstmöglichen Überschuss realisiert:

$$\begin{aligned} w_3(\{1\}) &:= 8, & w_3(\{1, 2\}) &:= 6, \\ w_3(\{2\}) &:= 0 - (-13) = 13, & w_3(\{1, 3\}) &:= 16 - (-13) = 29, \\ w_3(\{3\}) &:= 0, & w_3(\{2, 3\}) &:= 34. \end{aligned}$$

Wir berechnen nun für alle $\mathcal{B} \notin \mathcal{T}^2$ die Werte $\varphi_3(\mathcal{B})$:

$$\begin{aligned} t_3(\mathcal{B}_3) &= -36 \quad \Rightarrow \quad \varphi_3(\mathcal{B}_3) = -36/2 = -18, \\ t_3(\mathcal{B}_4) &= -21 \quad \Rightarrow \quad \varphi_3(\mathcal{B}_4) = -21/1 = -21, \\ t_3(\mathcal{B}_5) &= -7.5 \quad \Rightarrow \quad \varphi_3(\mathcal{B}_5) = -7.5/0.5 = -15. \end{aligned}$$

Folglich gelangen wir zu $\epsilon^3 := -15$, $\mathcal{T}_3 := \{\mathcal{B}_5\}$ und $\mathcal{S}_3 := \{\{1, 2\}\}$. Die Koalition $\{1, 2\}$ besitzt demnach den dritthöchsten Koalitionsüberschuss $\epsilon^3 = -15$. Darüber hinaus gilt $\mathcal{T}^3 := \mathcal{T}^2 \cup \mathcal{T}_3 = \{\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_5\}$ und $\Delta^3 := \Delta^2 \cup \mathcal{S}_3 = \{\{1\}, \{2, 3\}, \{2\}, \{1, 3\}, \{1, 2\}, \emptyset, N\} \neq 2^N$. Folglich führen wir eine weitere Iteration aus und setzen $i = 4$. Die neue modifizierte charakteristische Funktion lautet

$$\begin{aligned} w_4(\{1\}) &:= 8, & w_4(\{1, 2\}) &:= 6 - (-15) = 21, \\ w_4(\{2\}) &:= 13, & w_4(\{1, 3\}) &:= 29, \\ w_4(\{3\}) &:= 0, & w_4(\{2, 3\}) &:= 34. \end{aligned}$$

¹⁰Wir haben in der ersten Phase des Verfahrens festgestellt, dass die Auszahlung des Spielers 1 im Nukleolus 8 betragen muss. In diesem Schritt haben wir nun gezeigt, dass die Auszahlung des zweiten Spielers 13 beträgt. Somit verbleibt dem Spieler 3 die Auszahlung $v(N) - (8 + 13) = 21$. Da wir aber veranschaulichen wollen, wie der Algorithmus arbeitet, führen wir alle Schritte des Verfahrens aus.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Für alle $\mathcal{B} \notin \mathcal{T}^3$ lassen sich die Werte $\varphi_4(\mathcal{B})$ wie folgt festlegen:

$$\begin{aligned} t_4(\mathcal{B}_3) &= -21 &\Rightarrow \varphi_4(\mathcal{B}_3) &= -21/1 = -21, \\ t_4(\mathcal{B}_4) &= -21 &\Rightarrow \varphi_4(\mathcal{B}_4) &= -21/1 = -21. \end{aligned}$$

Der viertgrösste minimale Koalitionsüberschuss ist somit $\epsilon^4 := -21$. Hierbei erhalten wir $\mathcal{T}_4 := \{\mathcal{B}_3, \mathcal{B}_4\}$, $\mathcal{S}_4 := \{\{3\}\}$, $\mathcal{T}^4 := \mathcal{T}^3 \cup \mathcal{T}_4 = \{\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, \mathcal{B}_4, \mathcal{B}_5\}$ und $\Delta^4 := \Delta^3 \cup \mathcal{S}_4 = \{\{1\}, \{2, 3\}, \{2\}, \{1, 3\}, \{1, 2\}, \{3\}, \emptyset, N\}$. Daher ist $\Delta^4 = 2^N$ erreicht und es resultiert $m = 4$.

Nun folgt die Phase 3 des Algorithmus von Bruyneel:

$$\begin{aligned} w_5(\{1\}) &:= 8, & w_5(\{1, 2\}) &:= 21, \\ w_5(\{2\}) &:= 13, & w_5(\{1, 3\}) &:= 29, \\ w_5(\{3\}) &:= 0 - (-21) = 21, & w_5(\{2, 3\}) &:= 34. \end{aligned}$$

• $k = 1$:

$$\begin{aligned} \mathcal{S}_1 = \{\{1\}, \{2, 3\}\} &\Rightarrow w^*(\{1\}) &:= w_2(\{1\}) &= 8, \\ &w^*(\{2, 3\}) &:= w_2(\{2, 3\}) &= 34. \end{aligned}$$

• $k = 2$:

$$\begin{aligned} \mathcal{S}_2 = \{\{2\}, \{1, 3\}\} &\Rightarrow w^*(\{2\}) &:= w_3(\{2\}) &= 13, \\ &w^*(\{1, 3\}) &:= w_3(\{1, 3\}) &= 29. \end{aligned}$$

• $k = 3$:

$$\mathcal{S}_3 = \{\{1, 2\}\} \Rightarrow w^*(\{1, 2\}) := w_4(\{1, 2\}) = 21.$$

• $k = 4$:

$$\mathcal{S}_4 = \{\{3\}\} \Rightarrow w^*(\{3\}) := w_5(\{3\}) = 21.$$

Der Nukleolus des Spiels lautet somit:

$$\mathcal{N}(N, v) = \hat{\mathbf{w}} = (w^*(\{1\}), w^*(\{2\}), w^*(\{3\})) = (8, 13, 21).$$

Wie die obigen Erläuterungen und das anschliessend vorgeführte Beispiel zeigen, müssen für die Anwendung des Verfahrens von Bruyneel zur Berechnung des Nukleolus eines kooperativen Spiels $\Gamma = (N, v)$ mit n Spielern zunächst alle möglichen

3.2 Methoden der zweiten Generation

minimal balancierten Koalitionssysteme der Spielermenge N mit ihren jeweiligen Gewichtsvektoren bestimmt werden. Wie wir oben bestimmt haben, existieren für $n = 3$ insgesamt fünf minimal balancierte Koalitionssysteme. Shapley [123] zeigt, dass für $n = 4$ die Anzahl aller minimal balancierten Koalitionssysteme 41 beträgt und listet diese auf. Um die Anzahl minimal balancierter Koalitionssysteme für $n \geq 5$ zu bestimmen, definiert Shapley [123] sog. *irreduzible* minimal balancierte Koalitionssysteme. Ein minimal balanciertes Koalitionssystem nennt er *reduzibel*, falls es direkt von minimal balancierten Koalitionssystemen niedrigerer Grösse generiert werden kann, sonst heisst es *irreduzibel*.¹¹ Die Summe der Anzahl beider minimal balancierten Koalitionssysteme ergibt dann die Zahl aller minimal balancierten Koalitionssysteme. Shapley [123] stellt fest, dass für $n = 5$ bereits insgesamt 234 und für $n = 6$ gar 71805 irreduzible minimal balancierte Koalitionssysteme existieren. Dieses exponentielle Wachstum der Anzahl der minimal balancierten Koalitionssysteme bei steigender Spielerzahl macht die praktische Anwendung des Verfahrens nur auf Spiele mit einer geringen Anzahl von Spielern möglich. Die Bestimmung aller minimal balancierten Koalitionssysteme einer Spielermenge N , $|N| \geq 4$, ist also sehr aufwändig. Sind aber diese Koalitionssysteme z. B. mit Hilfe des von Peleg [101] vorgeschlagenen Algorithmus einmal gewonnen worden, so können sie zur Berechnung des Nukleolus jedes kooperativen Spiels mit derselben Spielermenge N herangezogen werden. Die Menge \mathcal{M} muss also in diesem Fall nicht jedes mal neu bestimmt werden. Die Bedeutung des Verfahrens von Bruyneel liegt besonders darin, dass es den Nukleolus eines kooperativen Spiels mit Hilfe minimal balancierter Koalitionssysteme berechnet.

3.2.2 Das Verfahren von Dragan

Neben dem Verfahren von Bruyneel zählt der Algorithmus von Dragan [35] zu den Verfahren der zweiten Generation. Dieses Verfahren stützt sich, dem Algorithmus von Bruyneel ähnlich, auf das Konzept der minimal balancierten Koalitionssysteme und berechnet den Prenukleolus kooperativer Spiele. Stimmen die Imputations-

¹¹Shapley [123] hebt hervor, dass für $n = 2, 3, 4$ kein irreduzibles minimal balanciertes Koalitionssystem existiert.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

menge $\mathcal{I}(N, v)$ und die Preimputationsmenge $\mathcal{PI}(N, v)$ eines kooperativen Spiels $\Gamma = (N, v)$ überein, ist also für jeden Allokationsvektor $\mathbf{x} \in \mathcal{PI}(N, v)$ das Kriterium der individuellen Rationalität erfüllt, so ermittelt das Verfahren von Dragan zugleich den Nukleolus des Spiels, da in diesem Fall der Nukleolus und der Prenukleolus zusammenfallen. Dragan [35] beschreibt in seinem Artikel zuerst die direkte Form des Algorithmus. Die direkte Form benötigt alle minimal balancierten Koalitionssysteme und deren (eindeutige) Gewichtsvektoren. Anschliessend modifiziert Dragan die direkte Form mit Hilfe einer Folge linearer Programme, sodass nicht mehr alle minimal balancierten Koalitionssysteme bestimmt werden, sondern nur diejenigen, die zur Berechnung des Prenukleolus nötig sind. Das Verfahren von Dragan findet eine Menge minimal balancierter Koalitionssysteme, mit deren Hilfe die Berechnung des Prenukleolus erfolgt.

Wir beschreiben zunächst die der direkten Form des Algorithmus von Dragan zu Grunde liegende Idee, welche sich nur geringfügig von der Idee des Verfahrens von Bruyneel [22] unterscheidet. Gegeben sei ein kooperatives Spiel $\Gamma = (N, v)$ mit n Spielern, dessen Imputationsmenge $\mathcal{I}(N, v)$ und Preimputationsmenge $\mathcal{PI}(N, v)$ übereinstimmen. Es sei $\mathcal{P}_0 := 2^N \setminus \{\emptyset, N\}$ mit $p := |\mathcal{P}_0| = 2^n - 2$. Die Menge \mathcal{P}_0 des Einführungsbeispiels beinhaltet z. B. die folgenden Koalitionen:

$$\mathcal{P}_0 := \{\{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}\} \quad \text{mit} \quad p := |\mathcal{P}_0| = 2^3 - 2 = 6.$$

Die Menge aller minimal balancierten Koalitionssysteme der Spielermenge N sei \mathcal{M} . Des Weiteren nehmen wir an, dass die Menge \mathcal{M} und der Gewichtsvektor jedes minimal balancierten Koalitionssystems $\mathcal{B} \in \mathcal{M}$ bekannt sind. In diesem Abschnitt bezeichnen wir einen Gewichtsvektor als einen p -dimensionalen Zeilenvektor:¹²

$$\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_p),$$

wobei für $S_i \in \mathcal{B}$ das Gewicht α_i positiv ist ($\alpha_i > 0$) und für jede Koalition $S_i \notin \mathcal{B}$ das Gewicht α_i verschwindet ($\alpha_i = 0$, $i = 1, 2, \dots, p$). Wie wir im vorangehenden

¹²Bisher haben wir einen Gewichtsvektor eines minimal balancierten Koalitionssystems $\mathcal{B} \in \mathcal{M}$ als einen $|\mathcal{B}|$ -dimensionalen Zeilenvektor definiert. In diesem Abschnitt definieren wir ihn als einen $(2^n - 2)$ -dimensionalen Zeilenvektor für alle $\mathcal{B} \in \mathcal{M}$ und folgen damit der Notation von Dragan [35].

3.2 Methoden der zweiten Generation

Abschnitt (vgl. S. 84) gezeigt haben, ist zum Beispiel $\mathcal{B} = \{\{1, 2\}, \{1, 3\}, \{2, 3\}\}$ ein minimal balanciertes Koalitionssystem der Spielermenge des Einführungsbeispiels, wobei wir den Gewichtsvektor von \mathcal{B} nunmehr darstellen als¹³

$$\boldsymbol{\alpha} = (\alpha_{\{1\}}, \alpha_{\{2\}}, \alpha_{\{3\}}, \alpha_{\{1,2\}}, \alpha_{\{1,3\}}, \alpha_{\{2,3\}}) = (0, 0, 0, 1/2, 1/2, 1/2).$$

Um den kleinsten Überschuss zu finden, der von jeder Koalition realisiert werden kann, verwendet Dragan [35], wie bereits Bruyneel [22], die auf S. 83 f. erläuterte Eigenschaft von balancierten Koalitionssystemen. Gemäss dieser Eigenschaft ist der Überschuss jedes balancierten Koalitionssystems \mathcal{B}

$$t(\mathcal{B}) := \sum_{S \in \mathcal{P}_0} \alpha_S e(S, \mathbf{x}) = \sum_{S \in \mathcal{P}_0} \alpha_S v(S) - v(N) \quad (3.32)$$

über die Imputationsmenge konstant und damit insbesondere für alle Imputationen $\mathbf{x} \in \mathcal{I}(N, v)$ gleich gross (vgl. dazu Gleichung (3.20), S. 83). Analog zur Vorgehensweise von Bruyneel berechnet Dragan anschliessend das gewichtete arithmetische Mittel der Koalitionsüberschüsse für jedes minimal balancierte Koalitionssystem $\mathcal{B} \in \mathcal{M}$, ohne jedoch die individuelle Rationalität der Auszahlungsvektoren zu verlangen. Demzufolge ersetzt Dragan für $\mathcal{B} \in \mathcal{M}$ die Definition von $\varphi(\mathcal{B})$ im Algorithmus von Bruyneel (vgl. die Definitionsgleichung (3.23) auf S. 85) durch

$$K(\mathcal{B}) := \frac{\sum_{S \in \mathcal{P}_0} \alpha_S e(S, \mathbf{x})}{\sum_{S \in \mathcal{P}_0} \alpha_S} = \frac{t(\mathcal{B})}{\sum_{S \in \mathcal{P}_0} \alpha_S}. \quad (3.33)$$

Dragan macht also keine Fallunterscheidung bei der Definition des gewichteten arithmetischen Mittels der Koalitionsüberschüsse, weil die individuelle Rationalität der Auszahlungsvektoren für die Berechnung des Prenukleolus nicht erfüllt werden muss. Ähnlich wie im Algorithmus von Bruyneel definiert Dragan danach

$$K_1 := \max_{\mathcal{B} \in \mathcal{M}} K(\mathcal{B}) \quad (3.34)$$

¹³Wir stellen dabei wiederum das Gewicht einer Koalition S als α_S dar. Vgl. dazu Fussnote 7 auf S. 82.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

und zeigt, dass K_1 der geringstmögliche Koalitionsüberschuss ist, der von jeder Koalition $S \in 2^N$ realisiert werden kann. Es sei nun $\mathcal{B}_1 = \{S_1, S_2, \dots, S_{m_1}\} \in \mathcal{M}$ ein minimal balanciertes Koalitionssystem, welches die Eigenschaft $K(\mathcal{B}) = K_1$ besitzt. Wie wir im vorangehenden Abschnitt gezeigt haben, weist in diesem Fall der minimale Überschuss der Koalitionen $S \in \mathcal{B}_1$ den Wert K_1 auf. Folglich sind bei der Berechnung des Prenukleolus des Spiels $\Gamma = (N, v)$ ausschliesslich jene Auszahlungsvektoren \mathbf{x} relevant, die die linearen Gleichungen und Ungleichungen

$$e(S, \mathbf{x}) = K_1, \quad \forall S \in \mathcal{B}^1 := \mathcal{B}_1, \quad (3.35)$$

$$e(S, \mathbf{x}) \leq K_1, \quad \forall S \in \mathcal{P}_1 := \mathcal{P}_0 \setminus \mathcal{B}^1, \quad (3.36)$$

$$x(N) = v(N) \quad (3.37)$$

erfüllen. Das heisst, dass der Prenukleolus des Spiels eine Lösung des Gleichungs- und Ungleichungssystems (3.35)–(3.37) ist.¹⁴ Die Gleichungen (3.35) besagen, dass der Überschuss jeder Koalition $S \in \mathcal{B}_1$ den Wert K_1 annehmen muss. Die Ungleichungen (3.36) gewährleisten, dass dabei der Überschuss jeder Koalition $S \notin \mathcal{B}_1$, $S \neq N, \emptyset$, höchstens K_1 ist. Die Gleichung (3.37) garantiert schliesslich die Effizienz der Auszahlungsvektoren \mathbf{x} .

Falls das minimal balancierte Koalitionssystem $\mathcal{B}_1 = \{S_1, S_2, \dots, S_{m_1}\}$ insgesamt n Koalitionen enthält, also $m_1 = n$ ist, besteht das Gleichungssystem (3.35) aus n linearen Gleichungen. Auf Grund des Tatbestands, dass \mathcal{B}_1 ein minimal balanciertes Koalitionssystem ist, sind die Koeffizienten des Gleichungssystems (3.35) eindeutig. Das Gleichungssystem (3.35) ist also wohldefiniert. Man kann nun zeigen, dass in diesem Fall die Gleichungen in (3.35) linear unabhängig sind und somit eine eindeutige Lösung besitzen.¹⁵ Da der Prenukleolus eine Lösung des linearen Gleichungs- und Ungleichungssystems (3.35)–(3.37) und somit ebenfalls eine Lösung des Gleichungssystems (3.35) ist, muss die eindeutige Lösung des Gleichungssystems (3.35) mit dem Prenukleolus des Spiels übereinstimmen.

¹⁴Mit Hilfe der starken Form des Bondavera-Shapley-Satzes (vgl. Satz 2.3.2, S. 43) beweist Dragan [35], S. 122, dass das Gleichungs- und Ungleichungssystem (3.35)–(3.37) immer einen nicht-leeren zulässigen Bereich besitzt.

¹⁵Für den Beweis vgl. Zumsteg [157], S. 70, und Charnes/Kortanek [25], S. 34.

3.2 Methoden der zweiten Generation

Für den Fall $m_1 \neq n$ definiert Dragan $\mathcal{M}_1 := \{\mathcal{B} \mid \mathcal{B} \in \mathcal{M}, \mathcal{B} \neq \mathcal{B}^1\}$. Anschließend bestimmt er ein weiteres minimal balanciertes Koalitionssystem $\mathcal{B}_2 \in \mathcal{M}_1$, für welches das maximale gewichtete arithmetische Mittel der Überschüsse der Koalitionen $S \in \mathcal{B}_2$ den Wert

$$K_2 := \max_{\mathcal{B} \in \mathcal{M}_1} K(\mathcal{B}) = \max_{\mathcal{B} \in \mathcal{M}_1} \left\{ \frac{\sum_{S \in \mathcal{P}_1} \alpha_S e(S, \mathbf{x})}{\sum_{S \in \mathcal{P}_1} \alpha_S} \right\} \quad (3.38)$$

besitzt, welcher von jeder Koalition $S \in \mathcal{P}_1$ realisiert werden kann. Da aber bereits gefunden worden ist, dass K_1 der geringstmögliche Überschuss der Koalitionen $S \in \mathcal{B}_1$ ist, muss bei der Bestimmung von K_2 die Erfüllung der Gleichungen (3.35) gewährleistet werden. Wir zeigen nun, dass dies bei der Berechnung von K_2 implizit garantiert ist. Da $\mathcal{P}_1 = \mathcal{P}_0 \setminus \mathcal{B}^1$ ist, gilt $\mathcal{P}_0 = \mathcal{P}_1 \cup \mathcal{B}^1$ mit $\mathcal{P}_1 \cap \mathcal{B}^1 = \emptyset$. Demzufolge können wir die folgenden Gleichungen aufstellen:

$$\sum_{S \in \mathcal{P}_0} \alpha_S e(S, \mathbf{x}) = \sum_{S \in \mathcal{P}_1} \alpha_S e(S, \mathbf{x}) + \sum_{S \in \mathcal{B}^1} \alpha_S e(S, \mathbf{x}) \quad (3.39)$$

$$\Rightarrow \sum_{S \in \mathcal{P}_1} \alpha_S e(S, \mathbf{x}) = \sum_{S \in \mathcal{P}_0} \alpha_S e(S, \mathbf{x}) - \sum_{S \in \mathcal{B}^1} \alpha_S e(S, \mathbf{x}). \quad (3.40)$$

Wenden wir die Gleichung (3.32) auf den ersten Ausdruck und die Gleichung (3.35) auf den zweiten Ausdruck der rechten Seite von (3.40) an, dann erhalten wir durch Verwendung von $\mathcal{P}_0 = \mathcal{P}_1 \cup \mathcal{B}^1$ mit $\mathcal{P}_1 \cap \mathcal{B}^1 = \emptyset$

$$\begin{aligned} \sum_{S \in \mathcal{P}_1} \alpha_S e(S, \mathbf{x}) &= \left[\sum_{S \in \mathcal{P}_0} \alpha_S v(S) - v(N) \right] - \sum_{S \in \mathcal{B}^1} \alpha_S K_1 \\ &= \left[\sum_{S \in \mathcal{P}_1} \alpha_S v(S) + \sum_{S \in \mathcal{B}^1} \alpha_S v(S) - v(N) \right] - \sum_{S \in \mathcal{B}^1} \alpha_S K_1 \\ &= \sum_{S \in \mathcal{B}^1} \alpha_S [v(S) - K_1] + \sum_{S \in \mathcal{P}_1} \alpha_S v(S) - v(N). \end{aligned} \quad (3.41)$$

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Setzen wir die Gleichung (3.41) in die Gleichung (3.38) ein, so erhalten wir

$$K_2 := \max_{\mathcal{B} \in \mathcal{M}_1} \left\{ \frac{\sum_{S \in \mathcal{B}^1} \alpha_S [v(S) - K_1] + \sum_{S \in \mathcal{P}_1} \alpha_S v(S) - v(N)}{\sum_{S \in \mathcal{P}_1} \alpha_S} \right\}. \quad (3.42)$$

Aus der Definitionsgleichung (3.42) ist ersichtlich, dass Dragan bei der Berechnung von K_2 zugleich die charakteristische Funktion des Spiels modifiziert (vgl. die erste Summe im Zähler von (3.42)). Während beim Verfahren von Bruyneel die Modifizierung der charakteristischen Funktion durch (3.30) erfolgt (vgl. S. 88), wird beim Vorgehen nach Dragan keine Fallunterscheidung getroffen, weil bei der Berechnung des Prenukleolus die individuelle Rationalität der Auszahlungsvektoren nicht verlangt wird. Folglich modifiziert Dragan die charakteristische Funktion des Spiels derart, dass er die Koalitionswerte $v(S)$, für alle $S \in \mathcal{B}^1$, um K_1 vermindert. Somit muss auf Grund der Gleichungen (3.35) nunmehr

$$e(S, \mathbf{x}) = K_1 \Rightarrow x(S) = v(S) - K_1, \forall S \in \mathcal{B}^1 \quad (3.43)$$

gelten. Die Definitionsgleichung (3.42) besagt zudem, dass K_2 für jeden effizienten Auszahlungsvektor \mathbf{x} denselben Wert besitzt, weil jeder Term in der geschweiften Klammer von (3.42) konstant ist.

Dragan beweist, dass das durch (3.38) bzw. (3.42) gefundene minimal balancierte Koalitionssystem \mathcal{B}_2 mit $K(\mathcal{B}_2) = K_2$ mindestens eine Koalition S beinhaltet, deren Überschuss $e(S, \mathbf{x}) = K_2$ von den Überschüssen der Koalitionen $S \in \mathcal{B}^1$ (die Gleichungen in (3.35)) linear unabhängig ist. Es sei \mathcal{B}^2 die Vereinigungsmenge derjenigen Koalitionen, deren minimale Überschüsse in der ersten und der zweiten Iteration des Verfahrens von Dragan bestimmt worden sind. Wir definieren also $\mathcal{B}^2 := \mathcal{B}^1 \cup \mathcal{B}_2$. Falls die Anzahl der Koalitionen in der Vereinigungsmenge \mathcal{B}^2 nicht geringer als die Anzahl der Spieler n ist und die Überschüsse von beliebigen n Koalitionen der Menge \mathcal{B}^2 linear unabhängig sind, ist die eindeutige Lösung des Gleichungssystems dieser linear unabhängigen Koalitionsüberschüsse der Prenukleolus des Spiels. Im umgekehrten Fall werden weitere Iterationen durchgeführt, bis insgesamt n linear unabhängige Koalitionsüberschüsse ermittelt worden sind.

3.2 Methoden der zweiten Generation

Wie das Verfahren von Bruyneel geht die oben beschriebene direkte Form des Verfahrens von Dragan ebenfalls davon aus, dass alle minimal balancierten Koalitionssysteme der Spielermenge N und deren Gewichtsvektoren im Voraus bekannt sind. Um diesen Nachteil zu umgehen, modifiziert Dragan die beschriebene direkte Form seines Algorithmus derart, dass die Zahl K_i und das minimal balancierte Koalitionssystem \mathcal{B}_i erst in der Iteration i bestimmt werden. Das heisst, dass die Menge \mathcal{M} aller minimal balancierten Koalitionssysteme und der Gewichtsvektor jedes minimal balancierten Koalitionssystems $\mathcal{B} \in \mathcal{M}$ nicht im Voraus ermittelt werden. Stattdessen werden lediglich diejenigen minimal balancierten Koalitionssysteme $\mathcal{B} \in \mathcal{M}$ bestimmt, die zur Berechnung des Prenukleolus benötigt werden. Dies geschieht im Wege der Lösung eines linearen Programms in der jeweiligen Iteration. Im Folgenden stellen wir die in Dragan [35] angegebene Modifikation der obigen direkten Form des Verfahrens vor.

Um das lineare Programm aufzustellen, mit dessen Hilfe Dragan in der Iteration i (≥ 1) das minimal balancierte Koalitionssystem \mathcal{B}_i und seinen Gewichtsvektor ermittelt sowie den Wert K_i berechnet, nehmen wir an, dass $\mathcal{B}^0 = \emptyset$, $\mathcal{M}_0 := \mathcal{M}$ ist und die folgenden Mengen, die wir weiter oben bereits eingeführt haben, für $h := 1, 2, \dots, i$ bekannt sind:

$$\bullet \quad \mathcal{B}^h := \mathcal{B}_1 \cup \mathcal{B}_2 \cup \dots \cup \mathcal{B}_h, \quad (3.44)$$

$$\bullet \quad \mathcal{P}_h := \mathcal{P}_0 \setminus \mathcal{B}^h, \quad (3.45)$$

$$\bullet \quad \mathcal{M}_h := \mathcal{M}_{h-1} \setminus \{\mathcal{B} \mid \mathcal{B} \in \mathcal{M}, \mathcal{B} \subset \mathcal{B}^h\}. \quad (3.46)$$

Wir definieren nun $K_S := K_h$ für $S \in \mathcal{B}_h$ und $h < i$. Aus der Definitionsgleichung (3.42) folgt analog, dass in der Iteration i der direkten Form des Verfahrens der Wert K_i durch

$$K_i := \max_{\mathcal{B} \in \mathcal{M}_{i-1}} \left\{ \frac{\sum_{S \in \mathcal{B}^{i-1}} \alpha_S [v(S) - K_S] + \sum_{S \in \mathcal{P}_{i-1}} \alpha_S v(S) - v(N)}{\sum_{S \in \mathcal{P}_{i-1}} \alpha_S} \right\}$$

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

gewonnen wird.¹⁶ Da hier $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_{2^n-2})$ ein Gewichtsvektor ist, müssen bei dieser Maximierungsaufgabe für jedes minimal balancierte Koalitionssystem $\mathcal{B} \in \mathcal{M}_{i-1}$ die Bedingungen

$$\bullet \sum_{\substack{S \in \mathcal{B} \in \mathcal{M}_{i-1} \\ j \in S}} \alpha_S = 1, \quad \forall j \in N, \quad (3.47)$$

$$\bullet \sum_{S \in \mathcal{P}_{i-1}} \alpha_S > 0, \quad (3.48)$$

$$\bullet \alpha_S \geq 0, \quad \forall S \in \mathcal{P}_0 \quad (3.49)$$

erfüllt werden.¹⁷ Definieren wir

$$\alpha_0 := \sum_{S \in \mathcal{P}_{i-1}} \alpha_S, \quad \mu_0 := \frac{1}{\alpha_0} \quad \text{und} \quad \mu_S := \frac{\alpha_S}{\alpha_0}, \quad \forall S \in \mathcal{P}_0, \quad (3.50)$$

dann können wir in der Iteration i das minimal balancierte Koalitionssystem \mathcal{B}_i , dessen Gewichtsvektor α und den Wert K_i als Lösung des des folgenden linearen Programms gewinnen:

$$(D_i): \quad K_i := \max_{\mu_S, \mu_0} \left\{ \sum_{S \in \mathcal{B}^{i-1}} \mu_S [v(S) - K_S] + \sum_{S \in \mathcal{P}_{i-1}} \mu_S v(S) - \mu_0 v(N) \right\}$$

unter den Nebenbedingungen

$$\bullet \sum_{\substack{S \in \mathcal{P}_0 \\ j \in S}} \mu_S - \mu_0 = 0, \quad j = 1, 2, \dots, n \quad (3.51)$$

$$\bullet \sum_{S \in \mathcal{P}_{i-1}} \mu_S = 1 \quad (3.52)$$

$$\bullet \mu_0 > 0, \quad \mu_S \geq 0, \quad \forall S \in \mathcal{P}_0. \quad (3.53)$$

Es sei $(\mu, \mu_0) = (\mu_1, \mu_2, \dots, \mu_{2^n-2}, \mu_0)$ eine Lösung des linearen Programms (D_i) .¹⁸ Jede Koalition, für die in der Lösung des linearen Programms $\mu_S > 0$ gilt, ist ein

¹⁶Die Beweisführung für diese Aussage ist analog zu derjenigen für (3.42), die wir auf S. 100 vorgestellt haben.

¹⁷Jede Summation in (3.47) erfolgt über alle Koalitionen $S \in \mathcal{P}_{i-1}$, die den Spieler $j \in N$ enthalten.

¹⁸Aus der Definition von K_i folgt, dass für $\mathcal{M}_{i-1} \neq \emptyset$ der Wert K_i immer existiert. Folglich besitzt das lineare Programm (D_i) immer eine Lösung, wenn \mathcal{M}_{i-1} nicht-leer ist.

3.2 Methoden der zweiten Generation

Element des minimal balancierten Koalitionssystems \mathcal{B}_i ,¹⁹ weil aus (3.50) folgt, dass jede solche Koalition S ein positives Gewicht α_S besitzt. Zur Veranschaulichung berechnen wir nun erneut den Nukleolus des Einführungsbeispiels mit Hilfe des modifizierten Verfahrens von Dragan.

Beispiel 3.2.2

Für das Einführungsbeispiel gilt $\mathcal{P}_0 := \{\{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}\}$. Es sei $\mathcal{B}_0 := \emptyset$. Wir stellen zunächst das lineare Programm (D₁) auf, um den Wert K_1 und anschliessend das minimal balancierte Koalitionssystem \mathcal{B}_1 zu bestimmen.²⁰

$$\begin{aligned}
 (\text{D}_1) : \quad & \max_{\mu_S, \mu_0} \{0\mu_{\{1\}} + 0\mu_{\{2\}} + 0\mu_{\{3\}} + 6\mu_{\{1,2\}} + 16\mu_{\{1,3\}} + 26\mu_{\{2,3\}} - 42\mu_0\} \\
 & \text{unter den Nebenbedingungen} \\
 & \bullet \ j = 1 \Rightarrow \mu_{\{1\}} + \mu_{\{1,2\}} + \mu_{\{1,3\}} - \mu_0 = 0 \\
 & \bullet \ j = 2 \Rightarrow \mu_{\{2\}} + \mu_{\{1,2\}} + \mu_{\{2,3\}} - \mu_0 = 0 \\
 & \bullet \ j = 3 \Rightarrow \mu_{\{3\}} + \mu_{\{1,3\}} + \mu_{\{2,3\}} - \mu_0 = 0 \\
 & \bullet \ \mu_{\{1\}} + \mu_{\{2\}} + \mu_{\{3\}} + \mu_{\{1,2\}} + \mu_{\{1,3\}} + \mu_{\{2,3\}} = 1 \\
 & \bullet \ \mu_0 \geq 0, \ \mu_S \geq 0, \ \forall S \in \mathcal{P}_0.
 \end{aligned}$$

Als Lösung erhalten wir²¹

$$(\boldsymbol{\mu}^*, \mu_0^*) = (\mu_{\{1\}}^*, \mu_{\{2\}}^*, \mu_{\{3\}}^*, \mu_{\{1,2\}}^*, \mu_{\{1,3\}}^*, \mu_{\{2,3\}}^*, \mu_0^*) = (\tfrac{1}{2}, 0, 0, 0, 0, \tfrac{1}{2}, \tfrac{1}{2})$$

mit dem optimalen Wert $K_1 = -8$ der Zielfunktion des linearen Programms. Das minimal balancierte Koalitionssystem \mathcal{B}_1 ist somit gegeben durch

$$\mathcal{B}_1 := \{S \mid \mu_S^* > 0\} = \{\{1\}, \{2, 3\}\}.$$

Der kleinstmögliche Überschuss, der von jeder Koalition des Spiels realisiert werden kann, ist also -8 . Die Menge \mathcal{B}_1 enthält folglich jene Koalitionen, deren minimaler

¹⁹Für den Beweis, dass jede Lösung des linearen Programms (D_i) dem Gewichtsvektor eines minimal balancierten Koalitionssystems entspricht, vgl. Owen [97], S. 228, und Peleg/Sudhölter [102], S. 40.

²⁰Für die charakteristische Funktion des Einführungsbeispiels vgl. z. B. S. 84.

²¹Zur Lösung des Optimierungsproblems (D₁) haben wir die Software LINDO verwendet. Diese Software ist erhältlich unter www.lindo.com (30.09.2004).

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Überschuss -8 beträgt. Um den Prenukleolus des Einführungsbeispiels zu berechnen, muss nun in einem weiteren Schritt im Wege der Lösung eines zweiten linearen Programms (D_2) das minimal balancierte Koalitionssystem \mathcal{B}_2 bestimmt werden, weil das Koalitionssystem \mathcal{B}_1 insgesamt lediglich $2 < n = 3$ Koalitionen beinhaltet. Zu diesem Zweck definieren wir

$$\begin{aligned}\mathcal{B}^1 &:= \mathcal{B}^0 \cup \mathcal{B}_1 = \{\{1\}, \{2, 3\}\}, \\ \mathcal{P}_1 &:= \mathcal{P}_0 \setminus \mathcal{B}_1 = \{\{2\}, \{3\}, \{1, 2\}, \{1, 3\}\}\end{aligned}$$

und stellen das lineare Programm (D_2) auf:

$$(D_2) : \quad \max_{\mu_S, \mu_0} \left\{ \underbrace{8\mu_{\{1\}} + 34\mu_{\{2,3\}}}_{S \in \mathcal{B}^1} + \underbrace{0\mu_{\{2\}} + 0\mu_{\{3\}} + 6\mu_{\{1,2\}} + 16\mu_{\{1,3\}}}_{S \in \mathcal{P}_1} - 42\mu_0 \right\}$$

unter den Nebenbedingungen

- $j = 1 \Rightarrow \mu_{\{1\}} + \mu_{\{1,2\}} + \mu_{\{1,3\}} - \mu_0 = 0$
- $j = 2 \Rightarrow \mu_{\{2\}} + \mu_{\{1,2\}} + \mu_{\{2,3\}} - \mu_0 = 0$
- $j = 3 \Rightarrow \mu_{\{3\}} + \mu_{\{1,3\}} + \mu_{\{2,3\}} - \mu_0 = 0$
- $\mu_{\{2\}} + \mu_{\{3\}} + \mu_{\{1,2\}} + \mu_{\{1,3\}} = 1$
- $\mu_0 \geq 0, \mu_S \geq 0, \forall S \in \mathcal{P}_0.$

Die Lösung des linearen Programms (D_2) ergibt

$$(\boldsymbol{\mu}^*, \mu_0^*) = (\mu_{\{1\}}^*, \mu_{\{2\}}^*, \mu_{\{3\}}^*, \mu_{\{1,2\}}^*, \mu_{\{1,3\}}^*, \mu_{\{2,3\}}^*, \mu_0^*) = (0, \frac{1}{2}, 0, 0, \frac{1}{2}, 0, \frac{1}{2}),$$

wobei der Zielfunktionswert des linearen Programms im Optimum -13 beträgt, also $K_2 = -13$ ist. Folglich gilt

$$\mathcal{B}_2 := \{S \mid \mu_S^* > 0\} = \{\{2\}, \{1, 3\}\}.$$

Da keine der Koalitionen $S \in \mathcal{B}_2$ gleichzeitig ein Element des Koalitionssystems \mathcal{B}^1 ist, wurden die minimalen Überschüsse dieser Koalitionen noch nicht gefunden. Der kleinste Überschuss jeder Koalition $S \in \mathcal{B}_2$ ist also $K_2 = -13$. Wir stellen fest, dass das Koalitionssystem

$$\mathcal{B}^2 := \mathcal{B}^1 \cup \mathcal{B}_2 = \{\{1\}, \{2\}, \{1, 3\}, \{2, 3\}\}$$

3.2 Methoden der zweiten Generation

mehr Koalitionen beinhaltet als die Anzahl der Spieler des Spiels, denn $|\mathcal{B}^2| = 4 \geq n = 3$. Wir überprüfen also, ob die Überschüsse von drei beliebigen Koalitionen im Koalitionssystem \mathcal{B}^2 linear unabhängig sind. Zu diesem Zweck geben wir die Überschüsse aller Koalitionen $S \in \mathcal{B}^2$ an:

$$\begin{aligned} S = \{1\} &\in \mathcal{B}_1 &\Rightarrow v(\{1\}) - x_1 &= -8, \\ S = \{2, 3\} &\in \mathcal{B}_1 &\Rightarrow v(\{2, 3\}) - x_2 - x_3 &= -8, \\ S = \{2\} &\in \mathcal{B}^2 \setminus \mathcal{B}^1 &\Rightarrow v(\{2\}) - x_2 &= -13, \\ S = \{1, 3\} &\in \mathcal{B}^2 \setminus \mathcal{B}^1 &\Rightarrow v(\{1, 3\}) - x_1 - x_3 &= -13. \end{aligned}$$

Setzen wir die Werte der Koalitionen ein, dann erhalten wir

$$\begin{aligned} x_1 &= 8 \\ x_2 + x_3 &= 34 \\ x_2 &= 13 \\ x_1 + x_3 &= 29. \end{aligned}$$

Wir erkennen, dass z. B. die ersten drei Gleichungen eine eindeutige Lösung

$$(x_1, x_2, x_3) = (8, 13, 21),$$

haben, also linear unabhängig sind. Folglich ist diese eindeutige Lösung der Prenukleolus des Spiels. Da der Kern des Einführungsbeispiels nicht-leer ist,²² erfüllt der Prenukleolus des Spiels das Kriterium der individuellen Rationalität. Folglich fällt der Prenukleolus mit dem Nukleolus des Spiels zusammen.

Die oben beschriebene Modifikation der direkten Form des Verfahrens führt dazu, dass zur Berechnung des Prenukleolus nicht mehr alle minimal balancierten Koalitionssysteme und deren Gewichtsvektoren benötigt werden. Bevor wir weitere Eigenschaften des Verfahrens von Dragan erläutern, geben wir zusammenfassend die einzelnen Schritte dieses Verfahrens an.

²²Dies erkennen wir daran, dass $K_1 = -8 < 0$ ist.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Initialisierung

$\mathcal{B}^0 := \emptyset$, $\mathcal{P}_0 := 2^N \setminus \{\emptyset, N\}$, $i := 1$.

Schritt 1

Wir lösen das lineare Programm (D_i) . Der Wert von K_i entspricht dem optimalen Wert der Zielfunktion von (D_i) . Ist $(\boldsymbol{\mu}^*, \mu_0^*)$ eine optimale Lösung von (D_i) , so beinhaltet das minimal balancierte Koalitionssystem \mathcal{B}_i Koalitionen S , für die $\mu_S^* > 0$ gilt.

Schritt 2

Wir definieren die folgenden Koalitionssysteme

$$\mathcal{B}^i := \mathcal{B}^{i-1} \cup \mathcal{B}_i \quad \text{und} \quad \mathcal{P}_i := \mathcal{P}_{i-1} \setminus \mathcal{B}_i.$$

Schritt 3

Ist $|\mathcal{B}^i| \geq n$, so überprüfen wir, ob die Überschüsse von n Koalitionen $S \in \mathcal{B}^i$ linear unabhängig sind. Falls dies zutrifft, gehen wir zu Schritt 4, sonst erhöhen wir i um eins auf $i := i + 1$ und gehen zu Schritt 1.

Schritt 4

Wir lösen das lineare Gleichungssystem

$$e(S, \mathbf{x}) = K_S, \quad \forall S \in \mathcal{B}^i. \quad (3.54)$$

Die Lösung von (3.54) ist der Prenukleolus des Spiels.

Nachdem wir nun oben den Algorithmus von Dragan vorgestellt haben, erläutern wir im Folgenden wichtige Eigenschaften des Verfahrens. Im Unterschied zu der direkten Form des Verfahrens von Dragan muss bei dem obigen modifizierten Algorithmus in jeder Iteration mit Hilfe des linearen Programms (D_i) ein minimal balanciertes Koalitionssystem bestimmt werden. Ohne Nicht-Negativitätsbedingungen (3.53) besteht die Koeffizientenmatrix der Nebenbedingungen (kurz: Koeffizienten- oder Nebenbedingungsmatrix) des linearen Programms (D_i) für ein kooperatives Spiel mit n Spielern aus $(n + 1)$ Zeilen (insgesamt n Nebenbedingungen für (3.51) sowie eine Nebenbedingung für (3.52)) und $2^n - 1$ Spalten. Der Algorithmus berechnet den

3.2 Methoden der zweiten Generation

Prenukleolus des Spiels in maximal $n - 1$ Iterationen. Folglich beträgt die Anzahl der zu lösenden linearen Programme der Form (D_i) im ungünstigsten Fall $n - 1$. Das sieht man wie folgt: Aus der Definition der balancierten Koalitionssysteme folgt, dass jeder Spieler ein Mitglied mindestens einer Koalition $S \in \mathcal{B}$ sein muss, falls \mathcal{B} ein balanciertes Koalitionssystem ist. Demzufolge müssen alle Spieler des Spiels in \mathcal{B} enthalten sein. Das ist für $|\mathcal{B}| = 1$ nur dann der Fall, wenn $\mathcal{B} = \{N\}$ ist. Da das Koalitionssystem \mathcal{P}_0 diesen Fall ausschliesst, wird jedes minimal balancierte Koalitionssystem mindestens zwei Koalitionen beinhalten. In der ersten Iteration wird also das minimal balancierte Koalitionssystem \mathcal{B}_1 aus nicht weniger als zwei Koalitionen bestehen, wobei die Überschüsse dieser Koalitionen linear unabhängig sind. Da der Algorithmus gewährleistet, dass in jeder Iteration mindestens eine Koalition $S \notin \mathcal{B}^1$ festgelegt wird, sodass der Koalitionsüberschuss dieser Koalition und die Überschüsse der Koalitionen in \mathcal{B}^1 linear unabhängig sind, müssen wir im ungünstigsten Fall insgesamt $n - 1$ lineare Programme lösen, um n Koalitionen zu bestimmen, deren Überschüsse linear unabhängig sind.

Vergleichen wir das Verfahren von Dragan mit dem von Bruyneel, so stellen wir Folgendes fest. Das Verfahren von Bruyneel ist ein Verfahren zur Berechnung des Nukleolus, während mit Hilfe des Algorithmus von Dragan der Prenukleolus berechnet wird. Das Verfahren von Dragan kann zur Berechnung des Nukleolus herangezogen werden, falls die Imputationsmenge und die Preimputationsmenge des Spiels übereinstimmen. Beide Verfahren stützen sich auf das Konzept der minimal balancierten Koalitionssysteme und benutzen insofern implizit das Kohlberg-Kriterium.²³ Das Verfahren von Bruyneel verlangt die Bestimmung aller minimal balancierten Koalitionssysteme und deren Gewichtsvektoren in der ersten Stufe. Dafür muss aber kein lineares Programm gelöst werden. Hingegen erfordert das modifizierte Verfahren von Dragan, dass in jeder Iteration ein lineares Programm der Form (D_i) gelöst wird. Auf Grund der hohen Anzahl der minimal balancierten Koalitionssysteme kann das Verfahren von Bruyneel nur auf Spiele mit kleiner Spielerzahl angewendet werden,

²³Für das Kohlberg-Kriterium vgl. S. 52.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

während mit Hilfe des Verfahrens von Dragan die Berechnung des Prenukleolus von Spielen mit einer grösseren Anzahl von Spielern möglich ist.

Zwar kann mit Hilfe des Algorithmus von Dragan im Vergleich zu demjenigen von Bruyneel der Prenukleolus kooperativer Spiele mit grosser Spielerzahl n berechnet werden, aber in jeder Iteration muss ein lineares Programm der Form (D_i) gelöst werden, dessen Koeffizientenmatrix die Dimension $(n + 1) \times (2^n - 1)$ besitzt. Beispielsweise besteht für $n = 5$ die Koeffizientenmatrix aus 5 Zeilen und 31 Spalten und für $n = 10$ aus 11 Zeilen und $2^n - 1 = 1023$ Spalten. Die Dimension der Koeffizientenmatrix steigt also mit zunehmender Spielerzahl n exponentiell an. Da in jeder Iteration des Verfahrens von Dragan die Dimension der Koeffizientenmatrix des linearen Programms (D_i) unverändert bleibt, wird die Berechnung des Prenukleolus mit steigender Spielerzahl n sehr aufwändig. In den nächsten Abschnitten stellen wir andere Verfahren vor, die zur Berechnung des Nukleolus kooperativer Spiele ebenfalls erfordern, dass lineare Programme gelöst werden, aber die Dimension der Koeffizientenmatrix im Laufe der Iterationen reduzieren.

3.3 Methoden der dritten Generation

3.3.1 Das Verfahren von Maschler et al.

In den vorangehenden Abschnitten haben wir verschiedene Verfahren vorgestellt, die zur Berechnung des Nukleolus bzw. des Prenukleolus kooperativer Spiele entwickelt worden sind. Der Ansatz der Verfahren der ersten Generation besteht darin, ein einziges lineares Programm zu lösen. Die Verfahren der zweiten Generation nutzen Eigenschaften minimal balancierter Koalitionssysteme. Wir stellen in diesem Abschnitt und den nächsten Abschnitten dieses Kapitels andere Verfahren dar, die durch die Ermittlung einer Lösung einer Folge linearer Programme den Nukleolus kooperativer Spiele berechnen. Diese Verfahren nennen wir Methoden der dritten Generation. Ihnen liegt das von Maschler et al. [82] vorgeschlagene Verfahren zu Grunde, das wir in diesem Abschnitt beschreiben.

3.3 Methoden der dritten Generation

Im Abschnitt 2.4, S. 47 f., haben wir erläutert, dass der Nukleolus auf der konstruktiven Idee der Schlichtung beruht. Maschler et al. [82] bilden die verschiedenen Phasen eines Schlichtungsverfahrens algorithmisch ab. Es sei $\Gamma = (N, v)$ ein kooperatives Spiel und $\mathcal{S}^0 := \{S \mid S \subset N, S \neq \emptyset\}$ die Menge der echten Koalitionen von Spielern dieses Spiels.²⁴ Um den Nukleolus des Spiels zu berechnen, bestimmen Maschler et al. [82] zunächst diejenigen Koalitionen $S \in \mathcal{S}^0$, deren minimaler Koalitionsüberschuss ϵ^1 über der Imputationsmenge $\mathcal{I}(N, v)$ (vgl. S. 47) gemäss

$$\epsilon^1 := \min_{\mathbf{x} \in \mathcal{I}(N, v)} \max_{S \in \mathcal{S}^0} e(S, \mathbf{x}) = \min_{\mathbf{x} \in \mathcal{I}(N, v)} \max_{S \in \mathcal{S}^0} [v(S) - x(S)] \quad (3.55)$$

am grössten ist. Der Definition zufolge ist ϵ^1 der kleinste Wert, für den der ϵ -Kern des Spiels nicht-leer ist (vgl. S. 35). Demnach bilden die zugehörigen Imputationen den Kleinst-Kern

$$X^1 := \{\mathbf{x} \in \mathcal{I}(N, v) : \max_{S \in \mathcal{S}^0} e(S, \mathbf{x}) = \epsilon^1\} \quad (3.56)$$

des Spiels. Da der Kleinst-Kern eines kooperativen Spiels mit einer nicht-leeren Imputationsmenge stets nicht-leer ist, ist $X^1 \neq \emptyset$. Die Menge der Koalitionen, deren kleinstmöglicher Überschuss ϵ^1 ist, sei

$$\mathcal{S}_1 := \{S \in \mathcal{S}^0 : e(S, \mathbf{x}) = \epsilon^1, \forall \mathbf{x} \in X^1\}. \quad (3.57)$$

Die Koalitionen $S \in \mathcal{S}_1$ sind also diejenigen Koalitionen, deren Unzufriedenheit minimal ist. Aus der Definition von X^1 folgt, dass für mindestens eine Koalition $S \in \mathcal{S}^0$ die Gleichung $e(S, \mathbf{x}) = \epsilon^1$ für alle $\mathbf{x} \in X^1$ erfüllt ist, dass also das Koalitionssystem \mathcal{S}_1 nicht-leer ist. Beinhaltet \mathcal{S}_1 alle Koalitionen $S \in \mathcal{S}^0$, so besitzt jede Koalition $S \neq \emptyset, N$ den gleichen kleinstmöglichen Überschuss ϵ^1 . Da in diesem Fall die Überschüsse aller Koalitionen und damit deren Unzufriedenheiten minimiert worden sind und der kleinste Überschuss jeder Koalition $S \neq \emptyset, N$ gleich gross ist, ist der Vektor der geordneten Überschüsse lexikographisch minimal. Folglich ist X^1 der Nukleolus des Spiels. Auf Grund der Eindeutigkeit des Nukleolus enthält X^1 nur eine Imputation. Die Abfolge der eben beschriebenen Schritte bilden eine Iteration. Ist $\mathcal{S}_1 \neq \mathcal{S}^0$,

²⁴Jede Teilmenge S von N heisst eine echte Koalition, falls $S \neq \emptyset, N$ gilt.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

so bestimmen Maschler et al. in einer weiteren Iteration bezüglich der Menge der Imputationen $X^1 \subset \mathcal{I}(N, v)$ diejenigen Koalitionen $S \in \mathcal{S}^1 := \mathcal{S}^0 \setminus \mathcal{S}_1$, deren minimale Überschüsse den zweithöchsten Wert ϵ^2 annehmen. Analog zu den obigen Ausführungen stellen die Autoren die Menge der zugehörigen Imputationen $X^2 \subset X^1$ und das jeweilige Koalitionssystem \mathcal{S}_2 fest. Falls nun $\mathcal{S}_1 \cup \mathcal{S}_2 = \mathcal{S}^0$ gilt, ist der Nukleolus des Spiels gefunden. Sonst müssen weitere Iterationen durchgeführt werden, bis der Nukleolus des Spiels erreicht wird. Maschler et al. [82], S. 334, beweisen, dass auf dem angegebenen Wege nach endlich vielen Iterationen der Nukleolus jedes kooperativen Spiels berechnet werden kann.

Wir stellen nun die einzelnen Iterationen des Verfahrens formal dar. Allgemein sei κ die Anzahl der Iterationen, die zur Berechnung des Nukleolus eines kooperativen Spiels $\Gamma = (N, v)$ durchgeführt werden. Zudem definieren wir in der Startsituation $X^0 := \mathcal{I}(N, v)$. Gemäss den obigen Ausführungen werden zur Berechnung des Nukleolus mit Hilfe des Verfahrens von Maschler et al. für $k = 1, 2, \dots, \kappa$ eine endliche Folge $X^0 \supset X^1 \supset \dots \supset X^\kappa$ von Teilmengen der Imputationsmenge $\mathcal{I}(N, v)$ und eine endliche Folge $\mathcal{S}^0 \supset \mathcal{S}^1 \supset \dots \supset \mathcal{S}^\kappa$ von Koalitionssystemen rekursiv durch

- $\epsilon^k := \min_{\mathbf{x} \in X^{k-1}} \max_{S \in \mathcal{S}^{k-1}} e(S, \mathbf{x}),$
- $X^k := \{\mathbf{x} \in X^{k-1} : \max_{S \in \mathcal{S}^{k-1}} e(S, \mathbf{x}) = \epsilon^k\},$
- $\mathcal{S}_k := \{S \in \mathcal{S}^{k-1} : e(S, \mathbf{x}) = \epsilon^k, \forall \mathbf{x} \in X^k\},$
- $\mathcal{S}^k := \mathcal{S}^{k-1} \setminus \mathcal{S}_k$

ermittelt, bis $\mathcal{S}^\kappa := \mathcal{S}^k = \emptyset$ erreicht wird. Maschler et al. [82] nennen die Menge X^κ das *lexikographische Zentrum* des kooperativen Spiels $\Gamma = (N, v)$ und weisen nach, dass es für jedes kooperative Spiel genau ein Element enthält, das zudem mit dem Nukleolus des Spiels zusammenfällt.²⁵

Die Bestimmung des lexikographischen Zentrums eines kooperativen Spiels $\Gamma = (N, v)$ kann im Wege der Lösung einer Folge linearer Programme erfolgen. Beispiels-

²⁵Für den Beweis vgl. Maschler et al. [82], S. 334–335.

3.3 Methoden der dritten Generation

weise kann in der ersten Iteration des Verfahrens der Wert ϵ^1 mit Hilfe des linearen Optimierungsproblems

$$\begin{aligned} \text{LP}_1 : \quad & \min_{\mathbf{x}, \epsilon} \epsilon \\ & \text{unter den Nebenbedingungen} \\ & x(S) + \epsilon \geq v(S), \quad \forall S \in \mathcal{S}^0 \\ & \mathbf{x} \in X^0 \end{aligned}$$

berechnet werden, wobei ϵ^1 der optimale Zielfunktionswert von LP_1 ist.²⁶ Um das Verfahren von Maschler et al. zu operationalisieren, gibt Kopelowitz [66] in einer nicht veröffentlichten Arbeit eine Methode an, die mit Hilfe der Lösung von linearen Programmen der Form LP_1 das lexikographische Zentrum kooperativer Spiele berechnet (zitiert nach Maschler et al. [82], S. 331). Gemäss dieser Methode wird zunächst an Hand des linearen Programms LP_1 das Minimum der maximalen Koalitionsüberschüsse für alle $\mathbf{x} \in X^0$, also der Wert ϵ^1 , berechnet. Anschliessend werden die oben beschriebene Menge X^1 sowie die Koalitionen, deren Überschüsse für alle $\mathbf{x} \in X^1$ dem Wert ϵ^1 entsprechen, bestimmt. Die Menge dieser Koalitionen haben wir als \mathcal{S}_1 bezeichnet.

Besitzt das lineare Programm LP_1 eine eindeutige Lösung, so ist sie das einzige Element von X^1 . Andernfalls besteht die Menge X^1 aus den Lösungen des Ungleichungssystems, das durch das Einsetzen von ϵ^1 in die Nebenbedingungen des linearen Programms LP_1 entsteht.²⁷ Die Elemente der Menge \mathcal{S}_1 können durch die Ermittlung der Lösung des folgenden linearen Programms für jede Koalition $S \in \mathcal{S}^0$ ermittelt werden:

$$\begin{aligned} \text{LP}_{e(S, \mathbf{x})} : \quad & \min_{\mathbf{x}} e(S, \mathbf{x}) \\ & \text{unter der Nebenbedingung} \\ & \mathbf{x} \in X^1. \end{aligned} \tag{3.58}$$

Es sei e^* der Wert der Zielfunktion im Optimum. Falls dieser Wert kleiner als ϵ^1

²⁶Zur Minimierung des Maximums linearer Funktionen mit Hilfe eines linearen Programms der Form LP_1 vgl. Anhang B.

²⁷An Hand des Einführungsbeispiels werden wir dies nachfolgend verdeutlichen.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

ist, gehört die Koalition S nicht zu \mathcal{S}_1 . Denn $e^* < \epsilon^1$ bedeutet, dass in der Menge X^1 noch die Möglichkeit besteht, den Überschuss der Koalition S und damit deren Unzufriedenheit zu reduzieren.²⁸

Falls $\mathcal{S}^1 := \mathcal{S}^0 \setminus \mathcal{S}_1$ nicht-leer ist, sind noch die Überschüsse der Koalitionen $S \in \mathcal{S}^1$ zu minimieren. Dafür kann ein zweites lineares Programm gelöst werden, wobei dieses den Koalitionen $S \in \mathcal{S}_1$ den in der ersten Iteration berechneten minimalen Überschuss ϵ^1 gewährleisten muss:

$$\begin{aligned} & \min_{\mathbf{x}, \epsilon} \epsilon \\ & \text{unter den Nebenbedingungen} \\ & x(S) + \epsilon \geq v(S), \quad \forall S \in \mathcal{S}^1 := \mathcal{S}^0 \setminus \mathcal{S}_1 \\ & x(S) + \epsilon^1 = v(S), \quad \forall S \in \mathcal{S}_1 \\ & \mathbf{x} \in X^0. \end{aligned} \tag{3.59}$$

Dieses lineare Programm kann durch Anwendung der Definition von X^1 (vgl. Definitionsgleichung (3.56)) wie folgt dargestellt werden:

$$\begin{aligned} \text{LP}_2 : \quad & \min_{\mathbf{x}, \epsilon} \epsilon \\ & \text{unter den Nebenbedingungen} \\ & x(S) + \epsilon \geq v(S), \quad \forall S \in \mathcal{S}^1 \\ & \mathbf{x} \in X^1. \end{aligned}$$

Die Lösung des LP_2 ergibt den zweithöchsten Koalitionsüberschuss ϵ^2 mit den entsprechenden Imputationen $\mathbf{x} \in X^2$ und Koalitionen \mathcal{S}_2 . Ist jede Koalition $S \in \mathcal{S}^0$ ein Element von $\mathcal{S}_1 \cup \mathcal{S}_2$, so ist der Überschuss jeder Koalition des Spiels minimiert worden. Im umgekehrten Fall wird ein weiteres lineares Programm gelöst. Dieser Prozess wird solange durchgeführt, bis $\mathcal{S}^k = \emptyset$ erreicht wird.²⁹

²⁸Maschler et al. [82] beschreiben in ihrer Arbeit nicht, wie man die Menge \mathcal{S}_k algorithmisch bestimmen kann. Eine explizite Antwort auf dieses Problem finden wir in Sankaran [113], S. 331. Er schlägt zur Ermittlung des Koalitionssystems \mathcal{S}_k die Lösung linearer Programme der Form $\text{LP}_{e(\mathcal{S}, \mathbf{x})}$ vor.

²⁹Die Beschreibung der Methode von Kopelowitz [66] wurde zitiert nach Bruyneel [22].

3.3 Methoden der dritten Generation

An Hand des Einführungsbeispiels illustrieren wir nun die oben beschriebenen Schritte. Wir definieren $\mathcal{S}^0 := \{\{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}\}$ und $X^0 := \mathcal{I}(N, v)$. Das lineare Programm LP_1 zur Berechnung von ϵ^1 ist gegeben durch

$$\begin{aligned}
 LP_1 : \quad & \min_{\mathbf{x}, \epsilon} \epsilon \\
 & \text{unter den Nebenbedingungen} \\
 & x_1 + \epsilon \geq 0 \\
 & x_2 + \epsilon \geq 0 \\
 & x_3 + \epsilon \geq 0 \\
 & x_1 + x_2 + \epsilon \geq 6 \\
 & x_1 + x_3 + \epsilon \geq 16 \\
 & x_2 + x_3 + \epsilon \geq 26 \\
 & x_1 + x_2 + x_3 = 42 \\
 & x_i \geq 0, \quad i = 1, 2, 3.
 \end{aligned}$$

Als Lösung erhalten wir $\epsilon^1 = -8$ und $\mathbf{x}^* = (8, 8, 26)$.³⁰ Um die Menge X^1 zu bestimmen, setzen wir für ϵ den Wert $\epsilon^1 = -8$ in die Nebenbedingungen von LP_1 ein. Somit besteht die Menge X^1 aus den Imputationen, die die folgenden Bedingungen erfüllen:

$$x_1 + x_2 \geq 14 \tag{3.60}$$

$$x_1 + x_3 \geq 24 \tag{3.61}$$

$$x_2 + x_3 \geq 34 \tag{3.62}$$

$$x_1 \geq 8 \tag{3.63}$$

$$x_2 \geq 8 \tag{3.64}$$

$$x_3 \geq 8 \tag{3.65}$$

$$x_1 + x_2 + x_3 = 42 \tag{3.66}$$

³⁰Hier ist zu bemerken, dass der Auszahlungsvektor $\mathbf{x}^* = (8, 8, 26)$ womöglich nicht die einzige Lösung des linearen Programms LP_1 ist. Wichtig ist hier und in den nachfolgenden Iterationen k , $k = 1, 2, \dots, \kappa$, des Verfahrens nur der Wert von ϵ^k , an Hand dessen die Menge X^1 und die Koalitionen $S \in \mathcal{S}_1$ ermittelt werden können.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

$$x_i \geq 0, \quad i = 1, 2, 3. \quad (3.67)$$

Lösen wir die Nebenbedingung (3.66) nach $x_2 + x_3$ auf und setzen diese sodann in (3.62) ein, so erhalten wir $42 - x_1 \geq 34 \Rightarrow x_1 \leq 8$. Die Kombination dieses Ergebnisses mit (3.63) ergibt, dass $x_1 = 8$ ist. Daraus folgt $x_2 + x_3 = 34$. Setzen wir $x_1 = 8$ in (3.60) bzw. in (3.61) ein, so erhalten wir $x_2 \geq 6$ bzw. $x_3 \geq 16$. Aus $x_2 \geq 6$ und (3.64) folgt $x_2 \geq 8$. Aus $x_3 \geq 16$ und (3.65) folgt $x_3 \geq 16$. Somit können wir die Menge X^1 , die der Kleinst-Kern des Spiels ist, wie folgt darstellen:

$$X^1 := \{(x_1, x_2, x_3) : x_1 = 8, x_2 + x_3 = 34, x_2 \geq 8, x_3 \geq 16\}.$$

Für die Bestimmung von \mathcal{S}^1 suchen wir diejenigen Koalitionen $S \in \mathcal{S}^0$, deren Überschüsse für alle $\mathbf{x} \in X^1$ den Wert $\epsilon^1 = -8$ annehmen. Im Wege der Lösung eines linearen Programms der Form $\text{LP}_{e(S, \mathbf{x})}$ für jede $S \in \mathcal{S}^0$ erhalten wir

$$\mathcal{S}_1 := \{S \in \mathcal{S}^0 : e(S, \mathbf{x}) = -8, \quad \forall \mathbf{x} \in X^1\} = \{\{1\}, \{2, 3\}\}.$$

Somit ist $\epsilon^1 = -8$ der realisierbare minimale Überschuss der Koalitionen $\{1\}$ und $\{2, 3\}$. Die Koalitionen, deren minimaler Überschüsse noch nicht ermittelt worden sind, bilden die Menge $\mathcal{S}^1 := \mathcal{S}^0 \setminus \mathcal{S}_1 = \{\{1, 2\}, \{1, 3\}, \{2\}, \{3\}\}$.³¹ Da $\mathcal{S}^1 \neq \emptyset$ ist, es also noch Koalitionen gibt, deren minimaler Überschuss berechnet werden soll, muss für \mathcal{S}^1 das lineare Programm

$$\begin{aligned} \text{LP}_2 : \quad & \min_{\mathbf{x}, \epsilon} \epsilon \\ & \text{unter den Nebenbedingungen} \\ & x_2 + \epsilon \geq 0 \\ & x_3 + \epsilon \geq 0 \\ & x_1 + x_2 + \epsilon \geq 6 \\ & x_1 + x_3 + \epsilon \geq 16 \\ & \mathbf{x} \in X^1 \end{aligned}$$

³¹Hierbei gehen wir in dieser Arbeit immer implizit davon aus, dass für die Berechnung der minimalen Überschüsse der Koalitionen $S \in \mathcal{S}^k$ ($k = 1, 2, \dots, \kappa$) nur jene Imputationen herangezogen werden, die den Koalitionen $S \in \mathcal{S}^{k-1}$ ihren bereits ermittelten minimalen Überschuss gewährleisten.

3.3 Methoden der dritten Generation

gelöst werden. Betrachten wir die Eigenschaften der Auszahlungsvektoren $\mathbf{x} \in X^1$ im linearen Programm LP_2 explizit, so können wir LP_2 durch

$$\begin{aligned} LP'_2 : \quad & \min_{\mathbf{x}, \epsilon} \epsilon \\ & \text{unter den Nebenbedingungen} \\ & x_2 + \epsilon \geq 0 \\ & x_3 + \epsilon \geq 8 \\ & x_2 \geq 8 \\ & x_3 \geq 16 \\ & x_2 + x_3 = 34 \end{aligned}$$

ersetzen, dessen Lösung $\epsilon^2 = -13$ mit $(x_2, x_3) = (13, 21)$ ist. Um die Menge X^2 zu bestimmen, setzen wir $\epsilon^2 = -13$ in die Nebenbedingungen von LP'_2 ein:

$$x_2 \geq 13 \tag{3.68}$$

$$x_3 \geq 21 \tag{3.69}$$

$$x_2 \geq 8 \tag{3.70}$$

$$x_3 \geq 16 \tag{3.71}$$

$$x_2 + x_3 = 34. \tag{3.72}$$

Aus (3.68) und (3.70) folgt, dass $x_2 \geq 13$ ist. Ferner gilt wegen (3.69) und (3.71) die Bedingung $x_3 \geq 21$. Somit ist

$$\begin{aligned} X^2 &:= \{(x_1, x_2, x_3) : x_1 = 8, x_2 \geq 13, x_3 \geq 21, x_2 + x_3 = 34\} \\ &= \{(8, 13, 21)\}. \end{aligned}$$

Da X^2 nur einen Auszahlungsvektor enthält, entspricht dieser dem lexikographischen Zentrum des Spiels und stimmt somit mit dem Nukleolus des Spiels überein. Das Verfahren von Maschler et al. kann also derart modifiziert werden, dass keine weiteren Berechnungen mehr durchgeführt werden müssen, falls in einer k -ten Iteration die Menge X^k lediglich aus einem Element besteht. Wir führen jedoch zur Illustration

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

weitere Iterationen durch, bis das von Maschler et al. geforderte Abbruchkriterium $\mathcal{S}^k = \emptyset$ erreicht wird.

Die Koalitionen $S \in \mathcal{S}^1$, deren realisierbarer minimaler Überschuss $\epsilon^2 = -13$ ist, können wiederum an Hand des linearen Programms $\text{LP}_{e(S, \mathbf{x})}$ bestimmt werden:

$$\mathcal{S}_2 := \{S \in \mathcal{S}^1 : e(S, \mathbf{x}) = -13, \forall \mathbf{x} \in X^2\} = \{\{1, 3\}, \{2\}\}.$$

Die Menge $\mathcal{S}^2 := \mathcal{S}^1 \setminus \mathcal{S}_2 = \{\{1, 2\}, \{3\}\}$ enthält Koalitionen, deren kleinstmögliche Überschüsse noch minimiert werden müssen. Analog zu der ersten und zweiten Iteration erhalten wir in der dritten Iteration

$$\epsilon^3 = -15, \quad X^3 = X^2, \quad \mathcal{S}_3 := \{\{1, 2\}\}, \quad \mathcal{S}^3 := \mathcal{S}^2 \setminus \mathcal{S}_3 = \{\{3\}\}$$

und in der vierten Iteration

$$\epsilon^4 = -21, \quad X^4 = X^3, \quad \mathcal{S}_4 := \{\{3\}\}, \quad \mathcal{S}^4 := \mathcal{S}^3 \setminus \mathcal{S}_4 = \emptyset.$$

Somit ist $\mathcal{N}(N, v) = X^4 = (8, 13, 21)$ der Nukleolus des Einführungsbeispiels.

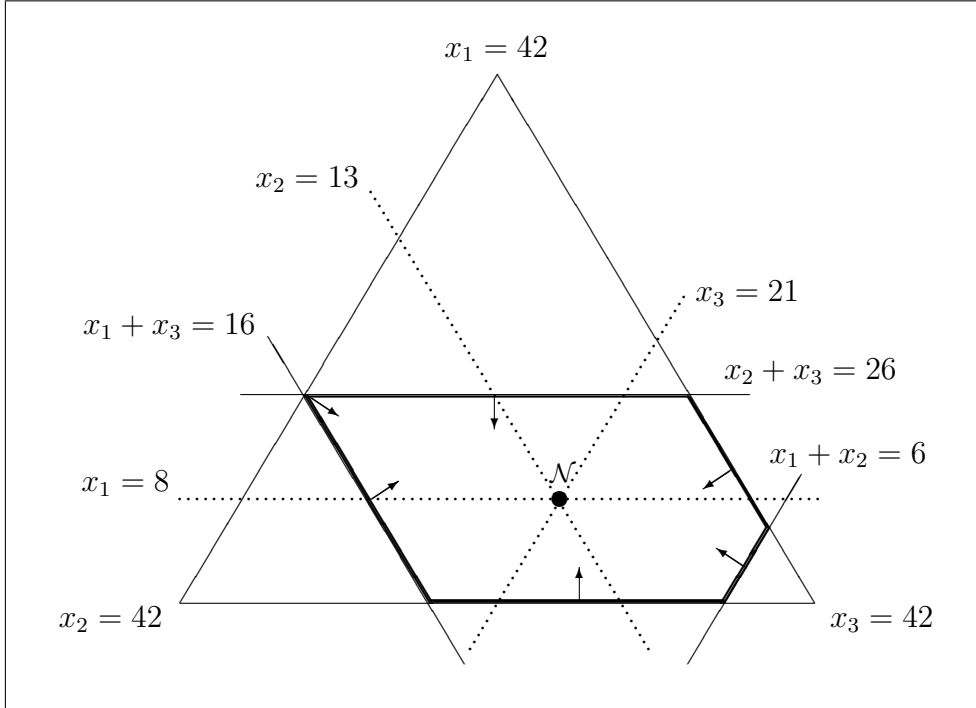
Die Lösung linearer Programme LP_k und die Bestimmung der Mengen X^k , $k = 1, 2, \dots, \kappa$, bedeutet, dass in jeder Iteration die Restriktionen

$$x(S) + \epsilon \geq v(S), \quad \forall S \in \mathcal{S}^k = \mathcal{S}^{k-1} \setminus \mathcal{S}_k \quad (3.73)$$

um den gleichen Wert ϵ^k tendenziell verschärft werden. Geometrisch heisst das z. B. für das Einführungsbeispiel, dass die Geraden, die diese Restriktionen repräsentieren, um den gleichen Wert ϵ^k verschoben werden. Die Abbildung 3.1 illustriert die Verschiebungsrichtung der Geraden, die den Kern des Einführungsbeispiels eingrenzen³² und den durch die Verschiebung erreichten geometrischen Ort des Nukleolus im Kern des Spiels.

³²Der Kern des Einführungsbeispiels wurde in der Abbildung 2.1 auf S. 30 dargestellt.

Abbildung 3.1: Nukleolus des Einführungsbeispiels



Das oben beschriebene Verfahren von Maschler et al. [82] stellt die Grundidee der anderen Verfahren der dritten Generation dar, welche wir in den nächsten Abschnitten vorstellen werden. Während sich die in den vorangegangenen Abschnitten vorgestellten Verfahren der zweiten Generation zur Berechnung des Nukleolus bzw. des Prenukleolus kooperativer Spiele auf besondere Eigenschaften minimal balancierter Koalitionssysteme stützen, verwenden also die in den nächsten Abschnitten vorgestellten weiteren Verfahren der dritten Generation die Idee von Maschler et al. [82] zur Bestimmung des lexikographischen Zentrums kooperativer Spiele. Trotz dieser unterschiedlichen Ansätze besteht jedoch zwischen den Verfahren beider Generationen ein interessanter Zusammenhang. Man kann nämlich zeigen, dass der in der Iteration k des Verfahrens von Maschler et al. [82] berechnete Wert ϵ^k mit dem

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

gemäss dem Algorithmus von Bruyneel [22] bzw. gemäss dem Algorithmus von Dragan [35] in der Iteration i berechneten Wert ϵ^i bzw. K_i übereinstimmt.³³

Die Anzahl der Iterationen, die durchgeführt werden, um den Nukleolus eines kooperativen Spiels mit Hilfe des Verfahrens von Maschler et al. [82] zu berechnen, hängt von der Anzahl der Koalitionen ab, deren Überschüsse in der jeweiligen Iteration k ($k = 1, 2, \dots, \kappa$) minimal werden. Wird in jeder Iteration der minimale Überschuss lediglich einer Koalition berechnet, so sind in diesem ungünstigsten Fall für ein Spiel mit n Spielern insgesamt $2^n - 2$ lineare Programme der Form LP_k zu lösen. Da x_1, x_2, \dots, x_n und ϵ die Variablen der linearen Programme sind, besitzt jedes zu lösende lineare Programm $n + 1$ Variablen. Die Anzahl der Nebenbedingungen jedes linearen Programms beträgt $2^n - 2 + 1 = 2^n - 1$, denn man hat eine Nebenbedingung für jede echte Koalition und eine Nebenbedingung für die Effizienzeigenschaft $x(N) = v(N)$. Somit sind in jedem linearen Optimierungsproblem insgesamt $2^n - 1$ Nebenbedingungen zu berücksichtigen. Während beispielsweise für ein kooperatives Spiel mit $n = 3$ Spielern in jeder Iteration ein lineares Programm mit $2^3 - 1 = 7$ Nebenbedingungen gelöst werden muss, beträgt für $n = 10$ die Anzahl der Beschränkungen bereits 1023. Also besteht ein Nachteil dieses Verfahrens darin, dass die Anzahl der Nebenbedingungen jedes linearen Programms mit steigender Spielerzahl exponentiell zunimmt. Zudem werden im ungünstigsten Fall insgesamt $\mathcal{O}(4^n)$ lineare Programme³⁴ der Form $LP_{e(S,x)}$ gelöst,³⁵ um in jeder Iteration k , $k = 1, 2, \dots, \kappa$, jene

³³Die Berechnung des Nukleolus des Einführungsbeispiels mit Hilfe der drei genannten Verfahren bestätigt diese Aussage. Für einen allgemeinen Beweis der Aussage vgl. Bruyneel [22], S. 47 f., und Dragan [35], S. 129 f.

³⁴Mit der Notation $\mathcal{O}(\cdot)$ wird die Komplexität eines Algorithmus ausgedrückt. Eine Beschreibung dieser Notation ist im Anhang F zu finden.

³⁵Der ungünstigste Fall tritt ein, wenn die Menge \mathcal{S}_k in jeder Iteration nur ein Element enthält. Somit entspricht die maximale Anzahl der zu lösenden linearen Programme der Form $LP_{e(S,x)}$ dem Wert

$$\begin{aligned}
 (2^n - 2) + (2^n - 3) + \dots + 1 &= (2^n - 2) + (2^n - 3) + \dots + 2^n - (2^n - 1) \\
 &= 2^n(2^n - 2) - (1 + 2 + \dots + 2^n - 2) \\
 &= 4^n - 2^n 2 - \frac{[2^n(2^n + 1)]}{2} + 2 = 4^n - 2^n 2 - \frac{1}{2}4^n - \frac{1}{2}2^n + 2 \\
 &= \frac{1}{2}4^n - \frac{5}{2}2^n + 2.
 \end{aligned}$$

Koalitionen zu bestimmen, deren realisierbare minimale Überschüsse ϵ^k betragen. Sankaran [113] zeigt aber, dass das Verfahren von Maschler et al. insofern verbessert werden kann, als die Anzahl der zu lösenden linearen Programme der Form $\text{LP}_{e(S, \mathbf{x})}$ reduziert wird. Das Verfahren von Sankaran [113] stellen wir im nächsten Abschnitt vor.

3.3.2 Das Verfahren von Sankaran

Sankaran [113] entwickelt auf der Basis des Verfahrens von Maschler et al. [82] einen verbesserten Algorithmus zur Berechnung des Nukleolus. Es gelingt ihm, die Anzahl der linearen Optimierungsaufgaben, die zur Berechnung des Nukleolus kooperativer Spiele gelöst werden, gegenüber dem Verfahren von Maschler et al. zu reduzieren. Zur Beschreibung des Algorithmus von Sankaran gehen wir in den weiteren Erläuterungen von denselben Koalitionssystemen \mathcal{S}_k und \mathcal{S}^k , demselben Wert ϵ^k und derselben Teilmenge X^k der Imputationsmenge des Spiels aus, die wir bei der Beschreibung des Verfahrens von Maschler et al. definiert haben (vgl. dazu S. 110). Die Reduzierung der Anzahl linearer Programme im Algorithmus von Sankaran gelingt einerseits durch die Kombination der Berechnung von ϵ^{k+1} mit der Bestimmung des Koalitionssystems \mathcal{S}_k . Andererseits wird nicht mehr für jede Koalition $S \in \mathcal{S}^{k-1}$ ein lineares Programm der Art $\text{LP}_{e(S, \mathbf{x})}$ (vgl. S. 111) gelöst, um jene Koalitionen festzustellen, deren minimale Unzufriedenheit berechnet worden ist. Die folgende Überlegung bildet dazu den Hintergrund.

Aus der Definition des lexikographischen Zentrums eines kooperativen Spiels $\Gamma = (N, v)$ auf S. 110 geht hervor, dass für jede Koalition $S \in \mathcal{S}_k$ die Beziehung

$$e(S, \mathbf{x}) = \epsilon^k \quad \text{für alle } \mathbf{x} \in X^k$$

gilt. Das Koalitionssystem \mathcal{S}_k enthält jene Koalitionen aus \mathcal{S}^{k-1} , deren minimale Koalitionsüberschüsse unter allen Koalitionen $S \in \mathcal{S}^{k-1}$ am höchsten sind und insbesondere denselben Wert ϵ^k annehmen. Dies impliziert, dass jede Koalition $T \in \mathcal{S}^k := \mathcal{S}^{k-1} \setminus \mathcal{S}_k$ einen geringeren minimalen Überschuss als ϵ^k besitzt. Es sei s die Anzahl der Koalitionen in der Menge \mathcal{S}_k , also $s := |\mathcal{S}_k|$. Falls nun \mathcal{S}_k eine echte

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Teilmenge von \mathcal{S}^{k-1} ist, existiert eine Imputation $\tilde{\mathbf{x}} \in X^k$, für die die Summe der grössten $s+1$ Koalitionsüberschüsse der Koalitionen in \mathcal{S}^{k-1} kleiner als $(s+1)\epsilon^k$ ist. Sind eine solche Imputation $\tilde{\mathbf{x}} \in X^k$ und s bekannt, kann die Menge \mathcal{S}_k bestimmt werden. Die Menge \mathcal{S}_k enthält nämlich jene Koalitionen aus der Menge \mathcal{S}^{k-1} , deren minimale Koalitionsüberschüsse für jede solche Imputation $\tilde{\mathbf{x}} \in X^k$ mit ϵ^k übereinstimmen. Ist hingegen $\mathcal{S}_k = \mathcal{S}^{k-1}$, so gilt, dass $e(S, \mathbf{x}) = \epsilon^k$ für alle $\mathbf{x} \in X^k$ und für alle $S \in \mathcal{S}^{k-1}$. Folglich ist der Überschuss jeder Koalition minimal, weil in diesem Fall $\mathcal{S}^k := \mathcal{S}^{k-1} \setminus \mathcal{S}_k = \emptyset$ ist.

Zur Illustration greifen wir auf die Bestimmung des Nukleolus des Einführungsbeispiels an Hand des Verfahrens von Maschler et al. zurück und nehmen an, dass $k = 1$ ist. Für diesen Fall haben wir festgestellt (vgl. S. 113 f.), dass $\mathcal{S}^0 := \{\{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}\}$, $\epsilon^1 = -8$ und $\mathcal{S}_1 := \{\{1\}, \{2, 3\}\}$ gilt. Folglich ist $s := |\mathcal{S}_1| = 2$. Unter allen Koalitionen $S \in \mathcal{S}^0$ besitzen die Koalitionen $\{1\}$ und $\{2, 3\}$ den höchsten minimalen Überschuss $\epsilon^1 = -8$, während alle anderen Koalitionen $T \in \mathcal{S}^1 := \mathcal{S}^0 \setminus \mathcal{S}_1 = \{\{1, 2\}, \{1, 3\}, \{2\}, \{3\}\}$ einen niedrigeren minimalen Überschuss als $\epsilon^1 = -8$ aufweisen. Somit gilt für alle $\mathbf{x} \in X^1$

$$\underbrace{e(\{2, 3\}, \mathbf{x})}_{=\epsilon^1} + \underbrace{e(\{1\}, \mathbf{x})}_{=\epsilon^1} + \underbrace{\max_{T \in \mathcal{S}^1} e(T, \mathbf{x})}_{< \epsilon^1} < (s+1)\epsilon^1 = 3\epsilon^1 = -24. \quad (3.74)$$

Die Ungleichung (3.74) besagt, dass für alle Imputationen $\mathbf{x} \in X^k = X^1$ die Summe der grössten drei ($s+1 = 3$) Überschüsse der Koalitionen $S \in \mathcal{S}^0$ kleiner als $(s+1)\epsilon^1$ ist.

Wir nehmen nun an, dass ϵ^k bekannt ist und die Menge \mathcal{S}_k bestimmt werden soll. Zu diesem Zweck stellt Sankaran zunächst die Anzahl der Elemente des Koalitionssystems \mathcal{S}_k , also die Zahl s , fest. Dafür macht er von der oben erläuterten Eigenschaft von s Gebrauch, dass die Summe der grössten $s+1$ Koalitionsüberschüsse der Koalitionen in \mathcal{S}^{k-1} kleiner als $(s+1)\epsilon^k$ ist. Er setzt $t := 1$ und ermittelt mit Hilfe von

$$\min_{\mathbf{x} \in X^k} \max_{\substack{\hat{\mathcal{S}} \subseteq \mathcal{S}^{k-1} \\ |\hat{\mathcal{S}}|=t+1}} \sum_{S \in \hat{\mathcal{S}}} e(S, \mathbf{x}) \quad (3.75)$$

3.3 Methoden der dritten Generation

zuerst die Summe der grössten $t + 1 = 2$ Überschüsse der Koalitionen in \mathcal{S}^{k-1} über alle Imputationen $\mathbf{x} \in X^k$. Die Optimierungsaufgabe (3.75) kann durch ein lineares Programm in der Form

$$\begin{aligned} \text{LP}(k, t) : \quad & \min_{\mathbf{x}, z} z \\ & \text{unter den Nebenbedingungen} \\ & z \geq \sum_{S \in \hat{\mathcal{S}}} e(S, \mathbf{x}), \quad \forall \hat{\mathcal{S}} \subseteq \mathcal{S}^{k-1} \text{ mit } |\hat{\mathcal{S}}| = t + 1 \quad (3.76) \\ & \mathbf{x} \in X^k \quad (3.77) \end{aligned}$$

repräsentiert werden (vgl. dazu Anhang B). Es seien z^* der optimale Zielfunktionswert und \mathbf{x}^k eine optimale Lösung von $\text{LP}(k, t)$.³⁶ Ist der berechnete optimale Wert z^* kleiner als $(t + 1)\epsilon^k$, so entspricht t der Anzahl der Koalitionen in der Menge \mathcal{S}_k . Folglich gilt $|\mathcal{S}_k| = s = t + 1$. Im umgekehrten Fall erhöht Sankaran den Wert von t um eins, $t := t + 1$, und berechnet mit Hilfe von $\text{LP}(k, t)$ jetzt die Summe der grössten $t + 1 = 3$ Koalitionsüberschüsse usw., bis schliesslich $t + 1 = |\mathcal{S}^{k-1}|$ oder $z^* < (t + 1)\epsilon^k$ erreicht wird. Ist $t + 1 = |\mathcal{S}^{k-1}|$, so ist der Überschuss jeder Koalition minimiert worden und somit stimmt \mathbf{x}^k mit dem Nukleolus des Spiels überein. Ist hingegen $z^* < (t + 1)\epsilon^k$, so gilt $\mathcal{S}_k := \{S \in \mathcal{S}^{k-1} \mid e(S, \mathbf{x}^k) = \epsilon^k\}$.

Auf der Basis dieser Überlegungen schlägt Sankaran zusammenfassend folgende Methode zur Bestimmung von \mathcal{S}_k vor:

- Im Wege der Lösung einer Folge linearer Programme $\text{LP}(k, t)$ bestimmt man den kleinsten existierenden Wert von t , für den es eine Imputation $\tilde{\mathbf{x}} \in X^k$ gibt, so dass für $\tilde{\mathbf{x}}$ die Summe der $t + 1$ grössten Koalitionsüberschüsse der Koalitionen in \mathcal{S}^{k-1} kleiner als $(t + 1)\epsilon^k$ ist.
- Existiert ein solcher Wert für t , so besteht die Menge \mathcal{S}_k genau aus denjenigen Koalitionen $S \in \mathcal{S}^{k-1}$, deren Koalitionsüberschüsse für die Imputation $\tilde{\mathbf{x}}$ mit ϵ^k übereinstimmen. In diesem Fall enthält die Menge \mathcal{S}_k insgesamt $t + 1$ Koalitionen.

³⁶Aus der Problemstellung (3.75) folgt, dass das lineare Programm $\text{LP}(k, t)$ immer eine Lösung besitzt, falls $X^k \neq \emptyset$ und $\mathcal{S}^{k-1} \neq \emptyset$ ist.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

- Existiert kein solches t , so ist $|\mathcal{S}_k| = |\mathcal{S}^{k-1}|$ und somit $\mathcal{S}^k = \emptyset$. In diesem Fall ist X^k der Nukleolus des Spiels.

Wie oben erwähnt worden ist, unterscheidet sich das Verfahren von Sankaran von demjenigen von Maschler et al. nicht nur in der Bestimmung der Menge \mathcal{S}_k . Liegt nämlich der Fall $z^* < (t+1)\epsilon^k$ vor, kann der nächsthöhere Überschuss ϵ^{k+1} ($\epsilon^{k+1} < \epsilon^k$) durch $\epsilon^{k+1} := z^* - t\epsilon^k$ berechnet werden, ohne dass man in jeder Iteration ein lineares Programm der Form (3.59) (vgl. S. 112) lösen muss. Diese Aussage folgt direkt aus den Erläuterungen auf S. 119 und dem linearen Programm $\text{LP}(k, t)$. Die Ungleichung (3.74) illustriert diesen Tatbestand. Damit man aber ϵ^{k+1} in der angegebenen Weise berechnen kann, muss ϵ^k bekannt sein. Deswegen berechnet Sankaran zu Beginn des Algorithmus den Wert ϵ^1 an Hand des bei der Beschreibung des Verfahrens von Maschler et al. vorgestellten linearen Programms LP_1 (vgl. S. 111). Die Berechnung von ϵ^2 erfolgt dann mit Hilfe von ϵ^1 , diejenige von ϵ^3 mit Hilfe von ϵ^2 usw.

Bevor wir für das Verfahren von Sankaran untersuchen, wie viele lineare Programme höchstens gelöst werden müssen und wie hoch die Anzahl der Nebenbedingungen dieser linearen Programme ist, geben wir den aus einer Initialisierung und den nachfolgenden Schritten bestehenden Algorithmus von Sankaran [113] an.

Initialisierung

$$k := 1, \quad X^0 := \mathcal{I}(N, v), \quad \mathcal{S}^0 := \{S \mid S \subset N, \ S \neq \emptyset\}.$$

Schritt 1

Wir lösen das lineare Programm

$$\begin{aligned} & \min_{\mathbf{x}, \epsilon} \epsilon \\ & \text{unter den Nebenbedingungen} \\ & \epsilon \geq e(S, \mathbf{x}), \quad \forall S \in \mathcal{S}^0 \\ & \mathbf{x} \in X^0. \end{aligned}$$

Es seien ϵ^1 der optimale Wert der Zielfunktion und \mathbf{x}^0 eine optimale Lösung des linearen Programms sowie $X^1 := \{\mathbf{x} \in X^0 \mid e(S, \mathbf{x}) \leq \epsilon^1, \ \forall S \in \mathcal{S}^0\}$.

Schritt 2

Falls $|\mathcal{S}^{k-1}| = 1$ ist, definieren wir $s(k) := 1$, $\mathbf{x}^k := \mathbf{x}^{k-1}$ und gehen zu Schritt 7.

Schritt 3

Es sei $t := 1$.

Schritt 4

Wir lösen das lineare Programm $\text{LP}(k, t)$. Es seien z^* der optimale Wert der Zielfunktion und \mathbf{x}^k ein optimaler Lösungsvektor von $\text{LP}(k, t)$.

Schritt 5

Falls $z^* < (t+1)\epsilon^k$ oder $t+1 = |\mathcal{S}^{k-1}|$ ist, dann gehen wir zu Schritt 6, sonst setzen wir $t := t+1$ und gehen zu Schritt 4.

Schritt 6

Falls $z^* < (t+1)\epsilon^k$ ist, enthält die Menge \mathcal{S}_k insgesamt t Koalitionen. Wir definieren $s(k) := t$ und berechnen $\epsilon^{k+1} := z^* - t\epsilon^k$. Andernfalls setzen wir $s(k) := |\mathcal{S}^{k-1}|$.

Schritt 7

Wir bestimmen die Menge $\mathcal{S}_k := \{S \in \mathcal{S}^{k-1} \mid e(S, \mathbf{x}^k) = \epsilon^k\}$.

Schritt 8

Falls $s(k) = |\mathcal{S}^{k-1}|$ ist, ist der Nukleolus des Spiels gefunden und entspricht der Imputation \mathbf{x}^k . Wir beenden die Berechnung.

Schritt 9

Wir definieren

- $\mathcal{S}^k := \mathcal{S}^{k-1} \setminus \mathcal{S}_k$,
- $X^{k+1} := \{\mathbf{x} \in X^k \mid e(S, \mathbf{x}) \leq \epsilon^{k+1}, \forall S \in \mathcal{S}^k\}$,
- $k := k+1$

und gehen zu Schritt 2.

Sankaran [113], S. 333 f., beweist, dass der oben dargestellte Algorithmus nach endlich vielen Iterationen gegen das lexikographische Zentrum des Spiels und folglich gegen den Nukleolus des Spiels konvergiert.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Um den Algorithmus aus der Sicht der Komplexitätstheorie zu analysieren, nehmen wir an, dass $\kappa \in \mathbb{N}$ der letzte Iterationsschritt im obigen Algorithmus ist. Um den Nukleolus eines kooperativen Spiels mit Hilfe des Algorithmus von Sankaran zu berechnen, müssen in jeder Iteration k , $1 \leq k < \kappa$, maximal $|\mathcal{S}_k|$ und in der letzten Iteration κ maximal $(|\mathcal{S}_\kappa| - 1)$ lineare Programme der Form $\text{LP}(k, t)$ gelöst werden. Somit ist die Anzahl der zu lösenden linearen Programme für ein Spiel mit n Spielern maximal $\sum_{k=1}^{\kappa} |\mathcal{S}_k| = 2^n$, also $\mathcal{O}(2^n)$. Während das lineare Programm im Schritt 1 des Algorithmus $2^n - 1$ Nebenbedingungen beinhaltet, besitzt jedes lineare Programm $\text{LP}(k, t)$ insgesamt

$$\binom{|\mathcal{S}^{k-1}|}{t+1} + 2^n = \frac{|\mathcal{S}^{k-1}|!}{(t+1)!(|\mathcal{S}^{k-1}| - t - 1)!} + 2^n$$

Nebenbedingungen, wobei der kombinatorische Teil des Ausdrucks die Anzahl der Nebenbedingungen in der Form (3.76) und 2^n diejenige der Nebenbedingungen (3.77) ist. Jedes lineare Programm $\text{LP}(k, t)$ hat insgesamt die $n + 1$ Variablen x_1, x_2, \dots, x_n und z . Wie wir auf S. 120 dargelegt haben, gilt in unserem Einführungsbeispiel $|\mathcal{S}^0| = 6$ für $k = 1$. Das lineare Programm $\text{LP}(k, t)$ beinhaltet in diesem Fall beispielsweise für $t = 2$

$$\binom{6}{3} + 2^3 = \frac{6!}{3!(6-3)!} + 2^3 = 28$$

Nebenbedingungen. Wir zeigen nun, dass die Anzahl der Nebenbedingungen der zu lösenden linearen Programme der Gestalt $\text{LP}(k, t)$ gesenkt werden kann. Um die Anzahl der Nebenbedingungen in $\text{LP}(k, t)$ zu reduzieren, formuliert Sankaran das lineare Programm $\text{LP}(k, t)$ wie folgt um. Es sei Ω_t^k die Menge aller Kombinationen von $(t+1)$ Koalitionen in \mathcal{S}^{k-1} . Jede Kombination bilden wir durch eine Menge $\mu := \{\mu_S\}_{S \in \mathcal{S}^{k-1}}$ ab, wobei wir $\mu_S = 1$ setzen, falls die Koalition S in der Kombination enthalten ist, und sonst $\mu_S = 0$ gelten soll. Für $S^0 := \{\{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}\}$ und $t = 3$ ist z. B. $\{\{1\}, \{1, 2\}, \{1, 3\}, \{2, 3\}\}$ eine Kombination von $t+1 = 4$ Koalitionen in S^0 , wobei wir diese Kombination durch $\mu := \{1, 0, 0, 1, 1, 1\} \in \Omega_3^1$ repräsentieren.

3.3 Methoden der dritten Generation

Mit dieser Notation können wir nun das lineare Programm $LP(k, t)$ in der Fassung der Optimierungsaufgabe (3.75) wie folgt umschreiben:

$$LP_\mu : \quad \min_{\mathbf{x} \in X^k} \max_{\mu \in \Omega_t^k} \sum_{S \in \mathcal{S}^{k-1}} \mu_S e(S, \mathbf{x})$$

unter den Nebenbedingungen

$$\sum_{S \in \mathcal{S}^{k-1}} \mu_S = t + 1 \quad (3.78)$$

$$\mu_S \in \{0, 1\}, \quad \forall S \in \mathcal{S}^{k-1}. \quad (3.79)$$

Ohne Beschränkung der Allgemeinheit können wir die Nebenbedingungen (3.79) ersetzen durch:

$$0 \leq \mu_S \leq 1, \quad \forall S \in \mathcal{S}^{k-1}.^{37}$$

Wir können dieses Sattelpunktproblem LP_μ in ein lineares Programm umwandeln.³⁸ Zu diesem Zweck betrachten wir den Maximierungsteil des obigen Optimierungsproblems, also den Ausdruck

$$\max_{\mu \in \Omega_t^k} \sum_{S \in \mathcal{S}^{k-1}} \mu_S e(S, \mathbf{x}). \quad (3.80)$$

Die Zielfunktion in (3.80) ist eine lineare Funktion von μ mit den (von μ unabhängigen) Koeffizienten $e(S, \mathbf{x})$. Gemäss dem starken Dualitätssatz (vgl. Anhang A.1, S. 287) stimmen für die beiden folgenden, zueinander dualen linearen Programme

$$\begin{aligned} & \max_{\mu \in \Omega_t^k} \sum_{S \in \mathcal{S}^{k-1}} \mu_S e(S, \mathbf{x}) \\ & \text{unter den Nebenbedingungen} \\ & \sum_{S \in \mathcal{S}^{k-1}} \mu_S = t + 1 \quad \text{Dualvariable} =: \alpha \\ & 0 \leq \mu_S \leq 1, \quad \forall S \in \mathcal{S}^{k-1} \quad \text{Dualvariable} =: \beta_S, \quad \forall S \in \mathcal{S}^{k-1} \end{aligned}$$

³⁷Die Richtigkeit dieser Aussage können wir wie folgt begründen. Im linearen Programm LP_μ wird die Summe $\sum_{S \in \mathcal{S}^{k-1}} \mu_S e(S, \mathbf{x})$ über $\mu \in \Omega_t^k$ maximiert. Da nun zudem jede Variable μ_S maximal den Wert 1 annehmen darf und die Bedingung (3.78) eingehalten werden muss, werden im Optimum insgesamt $t + 1$ Variablen, deren zugehörige Koeffizienten $e(S, \mathbf{x})$ die $t + 1$ grössten Überschüsse bilden, den Wert 1 besitzen. Stimmen die Überschüsse aller Koalitionen überein, so existiert stets eine durch (3.79) geforderte ganzzahlige Lösung, die die Bedingung (3.78) erfüllt.

³⁸Vgl. dazu Owen [97], S. 59 f., und Anhang C dieser Arbeit.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

und

$$\begin{aligned} \min_{\alpha, \beta_S} \quad & (t+1)\alpha + \sum_{S \in \mathcal{S}^{k-1}} \beta_S \\ \text{unter den Nebenbedingungen} \\ & \alpha + \beta_S \geq e(S, \mathbf{x}), \quad \forall S \in \mathcal{S}^{k-1} \end{aligned} \quad (3.81)$$

$$\beta_S \geq 0, \quad \forall S \in \mathcal{S}^{k-1} \quad (3.82)$$

α frei

die Zielfunktionswerte im Optimum überein. Somit erhalten wir durch Anwendung der Dualitätstheorie auf den Maximierungsteil des Optimierungsproblems LP_μ das lineare Programm:

$$\begin{aligned} \text{DLP}_\mu(k, t) : \quad & \min_{\mathbf{x}, \alpha, \beta_S} \quad (t+1)\alpha + \sum_{S \in \mathcal{S}^{k-1}} \beta_S \\ & \text{unter den Nebenbedingungen} \end{aligned}$$

$$e(S, \mathbf{x}) \leq \alpha + \beta_S, \quad \forall S \in \mathcal{S}^{k-1} \quad (3.83)$$

$$\beta_S \geq 0, \quad \forall S \in \mathcal{S}^{k-1} \quad (3.84)$$

$$e(S, \mathbf{x}) \leq \epsilon^j, \quad \forall j = 1, \dots, k-1, \quad \forall S \in \mathcal{S}_j \quad (3.85)$$

$$e(S, \mathbf{x}) \leq \epsilon^k, \quad \forall S \in \mathcal{S}^{k-1} \quad (3.86)$$

$$\mathbf{x} \in \mathcal{I}(N, v). \quad (3.87)$$

Die Nebenbedingungen (3.83) und (3.84) stimmen mit den Nebenbedingungen (3.81) bzw. (3.82) überein. Die Nebenbedingungen (3.85), (3.86) und (3.87) gewährleisten, dass in LP_μ die Bedingung $\mathbf{x} \in X^k$ erfüllt ist.

Ohne die Nicht-Negativitätsbedingungen (3.84) und die Bedingungen (3.87) besitzt das Optimierungsproblem $\text{DLP}_\mu(k, t)$ insgesamt $|\mathcal{S}^{k-1}| + 2^n$ Nebenbedingungen³⁹ und $|\mathcal{S}^{k-1}| + n + 1$ Variablen ($|\mathcal{S}^{k-1}|$ Variablen für β_S , n Variablen für \mathbf{x} und eine Variable für α). Da das Koalitionssystem \mathcal{S}^{k-1} , inklusive der Koalitionen \emptyset und N ,

³⁹Die Anzahl der Nebenbedingungen von (3.83) beträgt $|\mathcal{S}^{k-1}|$ und diejenige von (3.85) und (3.86) gemeinsam 2^n . Dies werden wir noch an Hand des Einführungsbeispiels am linearen Programm $\text{DLP}_\mu(1, 1)$, S. 128, verdeutlichen.

3.3 Methoden der dritten Generation

maximal 2^n Koalitionen enthalten kann, ist bei dem Algorithmus von Sankaran sowohl die Anzahl der Nebenbedingungen als auch die Anzahl der Variablen von der Ordnung $\mathcal{O}(2^n)$.

Beispielsweise beinhaltet das lineare Programm $\text{DLP}_\mu(k, t) := \text{DLP}_\mu(1, 2)$ für $|\mathcal{S}^0| = 6$ insgesamt $6 + 2^3 = 14$ Nebenbedingungen und $6 + 3 + 1 = 10$ Variablen, während die lineare Optimierungsaufgabe $\text{LP}(k, t) := \text{LP}(1, 2)$ für $|\mathcal{S}^0| = 6$ insgesamt 28 Nebenbedingungen und 4 Variablen besitzt (vgl. S. 124). Wir stellen also fest, dass die Umformulierung von $\text{LP}(k, t)$ zwar die Anzahl der Nebenbedingungen verringert, aber gleichzeitig die Variablenzahl von $n + 1$ auf $|\mathcal{S}^{k-1}| + n + 1$ erhöht.

Vergleichen wir nun das Verfahren von Sankaran mit dem Verfahren von Maschler et al. im Hinblick auf die zu lösenden linearen Programme, so stellen wir Folgendes fest: Die Durchführung des Algorithmus von Sankaran erfordert nur zur Berechnung von ϵ^1 die Lösung eines linearen Programms, während bei der Methode von Maschler et al. zur Bestimmung jedes Wertes ϵ^k , $k = 1, 2, \dots, \kappa$, ein lineares Programm gelöst wird. Zudem ist die Anzahl der linearen Programme zur Festlegung der Koalitionssysteme \mathcal{S}_k , $k = 1, 2, \dots, \kappa$, beim Verfahren von Maschler et al. von der Ordnung $\mathcal{O}(4^n)$, während sie beim Algorithmus von Sankaran von der Ordnung $\mathcal{O}(2^n)$ ist. Hingegen ist die Anzahl der Variablen der gemäss dem Verfahren von Sankaran zu lösenden linearen Programme von der Ordnung $\mathcal{O}(2^n)$, während die im Algorithmus von Maschler et al. zu lösenden linearen Programme nur noch $n + 1$ Variablen enthalten.

Zur Illustration des Algorithmus von Sankaran lösen wir mit Hilfe des Verfahrens von Sankaran das Einführungsbeispiel. Wir geben zunächst nochmals die charakteristische Funktion des Spiels an:

$$\begin{aligned} v(\{1\}) &= 0, & v(\{2\}) &= 0, & v(\{3\}) &= 0, \\ v(\{1, 2\}) &= 6, & v(\{1, 3\}) &= 16, & v(\{2, 3\}) &= 26, \\ v(\{1, 2, 3\}) &= 42. \end{aligned}$$

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Zur Berechnung des Nukleolus führen wir die einzelnen Schritte des Algorithmus durch:

Initialisierung

- $k := 1$,
- $X^0 := \mathcal{I}(N, v) = \{\mathbf{x} \in \mathbb{R}_{\geq 0}^3 \mid x_1 + x_2 + x_3 = 42\}$,
- $S^0 := \{\{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}\}$.

Das lineare Programm zur Berechnung von ϵ^1 haben wir bereits auf S. 113 gelöst: $\epsilon^1 = -8$.

Iteration $k = 1$

Wir setzen $t := 1$. Anstelle von $\text{LP}(k, t) = \text{LP}(1, 1)$ lösen wir $\text{DLP}_\mu(1, 1)$.⁴⁰ Durch Heranziehung von $e(S, \mathbf{x}) := v(S) - \sum_{i \in S} x_i$ stellen wir das lineare Programm $\text{DLP}_\mu(1, 1)$ auf:

$$\begin{aligned} \text{DLP}_\mu(1, 1) : \quad & \min 2\alpha + \beta_1 + \beta_2 + \beta_3 + \beta_4 + \beta_5 + \beta_6 \\ & \text{unter den Nebenbedingungen} \\ & x_i + \alpha + \beta_i \geq 0, \quad i = 1, 2, 3 \\ & x_1 + x_2 + \alpha + \beta_4 \geq 6 \\ & x_1 + x_3 + \alpha + \beta_5 \geq 16 \\ & x_2 + x_3 + \alpha + \beta_6 \geq 26 \\ & x_i \geq 8, \quad i = 1, 2, 3 \\ & x_1 + x_2 \geq 14 \\ & x_1 + x_3 \geq 24 \\ & x_2 + x_3 \geq 34 \\ & x_1 + x_2 + x_3 = 42 \\ & \beta_i \geq 0, \quad i = 1, \dots, 6 \end{aligned}$$

⁴⁰Zur Lösung der Optimierungsaufgaben haben wir wiederum die Software LINDO verwendet. Diese Software ist erhältlich unter www.lindo.com (30.09.2004).

3.3 Methoden der dritten Generation

$$\begin{aligned} x_i &\geq 0, \quad i = 1, 2, 3 \\ \alpha &\quad \text{frei.} \end{aligned}$$

Als Lösung erhalten wir im Optimum $z^* = -16$. Da $-16 \not\leq (t+1)\epsilon^1 = 2 \cdot (-8) = -16$ ist, erhöhen wir den Wert von t um eins, $t := t + 1 = 2$, und stellen das nächste lineare Programm $\text{DLP}_\mu(1, 2)$ auf. Das nun zu lösende lineare Programm $\text{DLP}_\mu(1, 2)$ unterscheidet sich von $\text{DLP}_\mu(1, 1)$ nur in der Zielfunktion, genauer gesagt nur im Koeffizienten t . Als Lösung von $\text{DLP}_\mu(1, 2)$ ergeben unsere Berechnungen $z^* = -29$ mit $\mathbf{x}^1 = (8, 13, 21)$. Es gilt

$$z^* = -29 < (t + 1) \cdot \epsilon^1 = 3 \cdot (-8) = -24.$$

Die Menge jener Koalitionen, deren minimale Überschüsse $\epsilon^1 = -8$ betragen, umfasst $t = 2$ Koalitionen und ist gegeben durch

$$\mathcal{S}_1 := \{S \in \mathcal{S}^0 \mid e(S, \mathbf{x}^1) = \epsilon^1 = -8\} = \{\{1\}, \{2, 3\}\}.$$

Folglich beinhaltet die Menge

$$\mathcal{S}^1 := \mathcal{S}^0 \setminus \mathcal{S}_1 = \{\{2\}, \{3\}, \{1, 2\}, \{1, 3\}\}$$

jene Koalitionen, deren minimale Überschüsse noch ermittelt werden müssen. Zu diesem Zweck berechnen wir den Wert von ϵ^2

$$\epsilon^2 := z^* - t \cdot \epsilon^1 = -29 - 2 \cdot (-8) = -13$$

und erhöhen k um eins, $k := k + 1 = 2$.

Iteration $k = 2$

Zu Beginn setzen wir wiederum $t := 1$. Als den optimalen Wert der Zielfunktion von $\text{DLP}_\mu(k, t) = \text{DLP}_\mu(2, 1)$ erhalten wir

$$z^* = -26 \not\leq (t + 1) \cdot \epsilon^2 = 2 \cdot (-13) = -26.$$

Wir setzen $t := t + 1 = 2$ und lösen das entsprechende lineare Programm $\text{DLP}_\mu(2, 2)$. Die Lösung ergibt

$$z^* = -41 < (t + 1) \cdot \epsilon^2 = 3 \cdot (-13) = -39$$

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

mit $\mathbf{x}^2 = (8, 13, 21)$. Daraus folgt, dass

$$\begin{aligned}\mathcal{S}_2 &:= \{S \in \mathcal{S}^1 \mid e(S, \mathbf{x}^2) = \epsilon^2 = -13\} = \{\{2\}, \{1, 3\}\}, \\ \mathcal{S}^2 &:= \mathcal{S}^1 \setminus \mathcal{S}_2 = \{\{3\}, \{1, 2\}\}, \\ \epsilon^3 &:= z^* - t \cdot \epsilon^2 = -41 - 2 \cdot (-13) = -15\end{aligned}$$

gilt. Wir erhöhen k wiederum um eins, $k := k + 1 = 3$.

Iteration $k = 3$

Wir definieren erneut $t := 1$ und stellen das lineare Programm $\text{DLP}_\mu(k, t) = \text{DLP}_\mu(3, 1)$ auf. Die Lösung ergibt

$$z^* = -36 < (t + 1) \cdot \epsilon^3 = 2 \cdot (-15) = -30$$

mit $\mathbf{x}^3 = (8, 13, 21)$. Folglich gilt

$$\begin{aligned}\mathcal{S}_3 &:= \{S \in \mathcal{S}^2 \mid e(S, \mathbf{x}^3) = \epsilon^3 = -15\} = \{\{1, 2\}\}, \\ \mathcal{S}^3 &:= \mathcal{S}^2 \setminus \mathcal{S}_3 = \{\{3\}\}, \\ \epsilon^4 &:= z^* - t \cdot \epsilon^3 = -36 - 2 \cdot (-15) = -21.\end{aligned}$$

Wir setzen $k := k + 1 = 4$.

Iteration $k = 4$

Wir prüfen die Bedingung $|\mathcal{S}^{k-1}| = 1$ (Schritt 2) und stellen fest, dass $|\mathcal{S}^3| = 1$ ist. Demzufolge setzen wir $s(k) = s(4) := 1$ und $\mathbf{x}^k := \mathbf{x}^{k-1}$. Also ist $\mathbf{x}^4 := \mathbf{x}^3 = (8, 13, 21)$. Somit erhalten wir

$$\mathcal{S}_4 := \{S \in \mathcal{S}^3 \mid e(S, \mathbf{x}^4) = -21\} = \{\{3\}\}.$$

Da $|\mathcal{S}_4| = 1$ ist, stellt $\mathbf{x}^4 = (8, 13, 21)$ den Nukleolus des Spiels dar.

3.3.3 Das Verfahren von Potters et al.

Zur Berechnung des Nukleolus eines kooperativen Spiels schlagen Potters et al. [105] einen weiteren Algorithmus vor. Dieser beruht auf der Konzeption von Maschler et al. [82] und führt somit nach der Lösung von endlich vielen linearen Programmen

3.3 Methoden der dritten Generation

zum Nukleolus des Spiels. Ein wesentlicher Unterschied des Verfahrens von Potters et al. im Vergleich zu den Verfahren von Maschler et al. und von Sankaran besteht darin, dass im Algorithmus von Potters et al. zur Bestimmung der Koalitionen, deren realisierbare minimale Überschüsse berechnet worden sind, keine linearen Programme gelöst werden. Wie wir noch erläutern werden, ermöglicht die von den Autoren vorgeschlagene spezielle Formulierung der zu lösenden linearen Programme die Bestimmung dieser Koalitionen direkt aus dem optimalen Simplex-Tableau heraus. Zur Lösung dieser linearen Optimierungsaufgaben schlagen Potters et al. [105] eine der Simplex-Methode⁴¹ ähnliche Methode vor, welche die Bestimmung der Koalitionen, deren minimale Überschüsse ermittelt worden sind, mit der Aufstellung des nächsten linearen Programms verbindet. Damit gelingt es, einerseits die gewonnenen Informationen weiterzugeben und andererseits die Anzahl der Nebenbedingungen des nächsten zu lösenden linearen Programms zu reduzieren. Die Autoren zeigen, dass es ausreicht, zur Berechnung des Nukleolus eines kooperativen Spiels mit n Spielern höchstens $n - 1$ lineare Programme zu lösen, wobei die maximale Anzahl der Nebenbedingungen und der Variablen dieser linearen Programme $2^n - 1$ bzw. $2^n + n - 1$ beträgt.

Wir geben in diesem Abschnitt zunächst die allgemeine Form des Verfahrens von Potters et al. [105] an. Anschliessend zeigen wir, wie die oben erwähnten speziellen linearen Programme aufgestellt werden können. Danach erläutern wir die einzelnen Schritte zur Lösung dieser linearen Programme.

Gegeben sei ein kooperatives Null-normiertes⁴² Spiel $\Gamma = (N, v)$ mit der Imputationsmenge $\mathcal{I}(N, v) := \{\mathbf{x} \in \mathbb{R}_{\geq 0}^n \mid x(N) = v(N)\}$. Die allgemeine Form des von Potters et al. vorgeschlagenen Verfahrens besteht aus der Initialisierung und aus den nachfolgenden Schritten:

Initialisierung $\Pi := \mathcal{I}(N, v)$, $\mathcal{S} := 2^N \setminus \{N, \emptyset\}$.

Schritt 1 Man ersetze Π durch $\Pi' := \{\mathbf{x} \in \Pi \mid \max_{S \in \mathcal{S}} e(S, \mathbf{x}) \text{ ist minimal}\}$.

⁴¹Für die Simplex-Methode zur Lösung linearer Programme vgl. Anhang A.2.

⁴²Ein kooperatives Spiel $\Gamma = (N, v)$ nennt man Null-normiert, wenn $v(\{i\}) = 0$, $\forall i \in N$, gilt.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Schritt 2 Man entferne aus \mathcal{S} mindestens eine Koalition S , deren Überschuss $e(S, \mathbf{x})$ für alle $\mathbf{x} \in \Pi$ konstant ist.

Schritt 3 Falls $\mathcal{S} = \emptyset$ ist, ist der Nukleolus des Spiels gefunden. Sonst gehe man zu Schritt 1.

Hierbei bilden die Schritte 1, 2 und 3 eine Iteration des Algorithmus. Im Schritt 1 werden jene Imputationen bestimmt, die den maximalen Überschuss der Koalitionen minimieren. Im zweiten Schritt wird mindestens eine Koalition ermittelt, deren realisierbarer maximaler Überschuss mit dem im Schritt 1 berechneten kleinstmöglichen Überschuss übereinstimmt. Diese Koalition wird aus \mathcal{S} entfernt. Wir werden nachher zeigen, dass in jeder Iteration mindestens eine solche Koalition existiert. Der Algorithmus erfordert, dass die Schritte 1 und 2 solange durchgeführt werden, bis die Überschüsse aller Koalitionen minimiert worden sind. Der Unterschied zwischen dieser allgemeinen Form des Verfahrens von Potters et al. [105] und dem Verfahren von Maschler et al. [82] besteht darin, dass hier nicht alle Koalitionen festgestellt werden müssen, die in der Menge der Imputationen Π einen konstanten Koalitionsüberschuss haben, sondern nur mindestens eine solche Koalition. Hierfür ist ausschlaggebend, dass die Bestimmung mindestens einer solchen Koalition einfacher ist als die Bestimmung aller dieser Koalitionen.

Um den oben angegebenen Algorithmus operationalisieren zu können, definieren Potters et al. [105]

$$c := \max_{\mathbf{x} \in \Pi} \max_{S \in \mathcal{S}} e(S, \mathbf{x}), \quad (3.88)$$

$$y_S := c - [v(S) - x(S)], \quad \forall S \in \mathcal{S}. \quad (3.89)$$

Der Wert c entspricht dem maximalen Überschuss, der über die aktuelle Imputationsmenge Π und das Koalitionssystem \mathcal{S} realisiert werden kann. Die Variable y_S , $S \in \mathcal{S}$, ist als die Differenz zwischen diesem maximalen Koalitionsüberschuss c und dem Koalitionsüberschuss der Koalition S bezüglich einer Imputation $\mathbf{x} \in \Pi$ definiert. Da c

3.3 Methoden der dritten Generation

der maximale Überschuss ist, ist y_S stets nicht-negativ. Während der Initialisierung zu Beginn des Algorithmus ist der maximale Koalitionsüberschuss gegeben durch

$$c := \max_{\mathbf{x} \in \mathcal{I}(N, v)} \max_{S \in \mathcal{S}} e(S, \mathbf{x}) = \max_{S \in \mathcal{S}} v(S).$$

Der Grund dafür ist, dass eine Koalition $S \neq N$ mit dem höchsten Koalitionswert $v(S)$ den höchstmöglichen Koalitionsüberschuss realisieren wird, falls ihre Auszahlung $x(S)$ null ist. Zum Beispiel besitzt in unserem Einführungsbeispiel die Koalition $S = \{2, 3\}$ den höchsten Koalitionswert mit $v(\{2, 3\}) = 26$. Folglich beträgt der maximale Überschuss 26, wenn $x_2 = x_3 = 0$ ist. Also kann zu Beginn des Algorithmus der höchstmögliche Überschuss im Einführungsbeispiel nicht grösser als $c = 26$ sein. Durch Heranziehung dieses Tatbestands definieren Potters et al. [105] für einen beliebigen Spieler $i' \in N$ das folgende Gleichungssystem und zeigen, dass die Lösungsmenge des Gleichungssystems mit der Imputationsmenge Π übereinstimmt:

$$x(N) = v(N) \tag{3.90}$$

$$y_S - x(S) = c - v(S), \quad \text{falls } i' \notin S, \ S \in \mathcal{S} \tag{3.91}$$

$$y_S + x(N \setminus S) = c + v(N) - v(S), \quad \text{falls } i' \in S, \ S \in \mathcal{S} \tag{3.92}$$

$$\mathbf{x} \in \mathbb{R}_{\geq 0}^n, \mathbf{y} \in \mathbb{R}_{\geq 0}^{|\mathcal{S}|}. \tag{3.93}$$

Die Gleichung (3.90) und die Anforderung $\mathbf{x} \in \mathbb{R}_{\geq 0}^n$ gewährleisten die Effizienz und die individuelle Rationalität der Vektoren \mathbf{x} im Optimum. Die beiden Gleichungen (3.91) und (3.92) können analog zu (3.89) interpretiert werden, weil sie direkt aus (3.89) folgen: Die Gleichung (3.91) ist genau dieselbe wie die Gleichung (3.89), und die Gleichung (3.92) kann als

$$\begin{aligned} y_S &= c + v(N) - v(S) - x(N \setminus S) = c + v(N) - v(S) - [x(N) - x(S)] \\ &= c + \underbrace{v(N) - x(N)}_{=0} - [v(S) - x(S)] = c - [v(S) - x(S)] \end{aligned}$$

umgeschrieben werden und stimmt somit mit (3.89) überein. Bevor wir weitere Eigenschaften des Gleichungssystems (3.90)-(3.93) erläutern, stellen wir dieses für das Einführungsbeispiel auf. Wählen wir beispielsweise $i' = 3 \in N$, so stellt sich das

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Gleichungssystem (3.90)-(3.93) in der Initialisierung des Verfahrens von Potters et al. folgendermassen dar:

$$\begin{array}{llllll}
S = N = \{1, 2, 3\} & \Rightarrow & x_1 + x_2 + x_3 & = & 42 \\
S = \{1\} & \Rightarrow & i' \notin S & \Rightarrow & y_{\{1\}} - x_1 & = 26 - 0 = 26 \\
S = \{1, 2\} & \Rightarrow & i' \notin S & \Rightarrow & y_{\{1,2\}} - x_1 - x_2 & = 26 - 6 = 20 \\
S = \{1, 3\} & \Rightarrow & i' \in S & \Rightarrow & y_{\{1,3\}} + x_2 & = 26 + 42 - 16 = 52 \\
S = \{2\} & \Rightarrow & i' \notin S & \Rightarrow & y_{\{2\}} - x_2 & = 26 - 0 = 26 \\
S = \{2, 3\} & \Rightarrow & i' \in S & \Rightarrow & y_{\{2,3\}} + x_1 & = 26 + 42 - 26 = 42 \\
S = \{3\} & \Rightarrow & i' \in S & \Rightarrow & y_{\{3\}} + x_1 + x_2 & = 26 + 42 - 0 = 68.
\end{array}$$

Zur Illustration wird in der Tabelle 3.1 die Koeffizientenmatrix dieses Gleichungssystems angegeben. In der ersten Zeile der Tabelle 3.1 sind die Variablen des Gleichungssystems aufgeführt. Jede andere Zeile repräsentiert eine Gleichung des Gleichungssystems und beinhaltet die Koeffizienten der Variablen und den Wert der rechten Seite der Gleichung. Ein Vorteil der Repräsentation der Imputationsmenge durch

Tabelle 3.1: Koeffizientenmatrix

x_1	x_2	x_3	$y_{\{1\}}$	$y_{\{1,2\}}$	$y_{\{1,3\}}$	$y_{\{2\}}$	$y_{\{2,3\}}$	$y_{\{3\}}$	d
1	1	1	0	0	0	0	0	0	42
-1	0	0	1	0	0	0	0	0	26
-1	-1	0	0	1	0	0	0	0	20
0	1	0	0	0	1	0	0	0	52
0	-1	0	0	0	0	1	0	0	26
1	0	0	0	0	0	0	1	0	42
1	1	0	0	0	0	0	0	1	68

das Gleichungssystem (3.90)-(3.93) besteht darin, dass das Gleichungssystem bereits eine Basis besitzt. In der Tabelle 3.1 erscheinen nämlich exakt so viele Variablen (die Variablen $\{y_S\}_{S \in \mathcal{S}}$ und die Variable $x_{i'}$) in genau einer Gleichung, wie es Zeilen gibt. Potters et al. zeigen nun, dass nach jeder Iteration des Algorithmus ein Gleichungssystem der Gestalt (3.90)-(3.93) für das aktuelle Π und für das aktuelle Koalitionssystem $\mathcal{S} \subseteq 2^N \setminus \{\emptyset, N\}$ aufgestellt werden kann. Mit der folgenden Defini-

tion stellen Potters et al. das oben vorgestellte Gleichungssystem allgemein für alle Iterationen des Algorithmus dar:

Definition 3.3.1 *Gegeben sei ein System von Gleichungen und Ungleichungen*

$$\begin{aligned} \mathbf{Ax} + \mathbf{By} &= \mathbf{d} \geq \mathbf{0} \\ \mathbf{x} \in \mathbb{R}_{\geq 0}^n, \mathbf{y} \in \mathbb{R}_{\geq 0}^{|\mathcal{S}|}, \end{aligned} \tag{3.94}$$

wobei \mathbf{A} eine $(q \times n)$ -Matrix, \mathbf{B} eine $(q \times |\mathcal{S}|)$ -Matrix und \mathbf{d} ein reellwertiger $(q \times 1)$ -Spaltenvektor sind. Das System (3.94) heisst eine Beschreibung des Paares (Π, \mathcal{S}) , wenn folgende Bedingungen erfüllt sind:

1. Ist $\mathbf{x} \in \Pi$, so hat das System (3.94) eine Lösung (\mathbf{x}, \mathbf{y}) .
2. Das System (3.94) hat q Basisvariablen. Das heisst, in diesem System gibt es q Variablen, die genau in einer Gleichung erscheinen, und zwar derart, dass jede davon in einer anderen Gleichung steht.
3. Für jede Imputation $\mathbf{x} \in \Pi$ ist \mathbf{y} eindeutig, sodass (\mathbf{x}, \mathbf{y}) eine Lösung des Systems (3.94) ist.

Wie wir bereits erläutert haben, besagt die Definition 3.3.1, dass in jeder Iteration des Verfahrens die aktuelle Menge der Imputationen $\Pi \subseteq \mathcal{I}(N, v)$, die im Schritt 1 des Algorithmus definiert wird, durch (3.94) repräsentiert werden kann.

Wir zeigen nun, dass das für die Initialisierung des Verfahrens aufgestellte Gleichungssystem (3.90)-(3.93) alle drei Bedingungen der Definition 3.3.1 erfüllt: Das Gleichungssystem besitzt für $\mathbf{x} \in \Pi$ eine Lösung (\mathbf{x}, \mathbf{y}) , denn es beschreibt eine Spielsituation. Wie in der Tabelle 3.1 für das Einführungsbeispiel exemplarisch illustriert wird, trifft die Bedingung 2 für das Gleichungssystem ebenfalls zu. Da wegen (3.93) die Variablen $\{y_S\}_{S \in \mathcal{S}}$ nicht-negativ sind und für jedes gegebene $\mathbf{x} \in \Pi$ die Variablen $\{y_S\}_{S \in \mathcal{S}}$ eindeutig bestimmt werden können, ist auch die Bedingung 3 erfüllt.⁴³ Somit kann die Imputationsmenge Π für das Koalitionssystem \mathcal{S} durch das Gleichungssystem (3.90)-(3.93) repräsentiert werden.

⁴³Da y_S auf Grund der Definitionsgleichung (3.89) als eine affine Funktion von \mathbf{x} betrachtet werden kann, ist die Bedingung 3 der Definition 3.3.1 immer erfüllt.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Der Algorithmus von Potters et al. erfordert im Schritt 1 (vgl. S. 131), dass wir die aktuelle Imputationsmenge Π durch eine andere Π' ersetzen, sodass

$$\Pi' := \{ \mathbf{x} \in \Pi \mid \max_{S \in \mathcal{S}} e(S, \mathbf{x}) \text{ ist minimal} \} \quad (3.95)$$

gilt. Die Menge Π' beinhaltet Imputationen $\mathbf{x} \in \Pi$, die den maximalen Überschuss der Koalitionen $S \in \mathcal{S}$ minimieren. Aus den Definitionen für y_S und c folgt, dass diese Imputationen gleichzeitig diejenigen Imputationen sind, die zur höchstmöglichen Reduktion des maximalen Überschusses der Koalitionen $S \in \mathcal{S}$ führen. Demnach kann die durch (3.95) definierte Imputationsmenge Π' ebenfalls durch

$$\Pi' := \{ \mathbf{x} \in \Pi \mid \min_{S \in \mathcal{S}} y_S(\mathbf{x}) \text{ ist maximal} \} \quad (3.96)$$

repräsentiert werden. Um die Imputationsmenge Π' zu finden, deren Elemente $\mathbf{x} \in \Pi$ den Wert $\min_{S \in \mathcal{S}} y_S(\mathbf{x})$ maximieren, definieren wir zunächst eine neue Variable $t := \min_{S \in \mathcal{S}} y_S(\mathbf{x})$. Diese Definition impliziert, dass $t \leq y_S$ für alle $S \in \mathcal{S}$ ist. Folglich können wir mit Hilfe des folgenden linearen Programms die neue Imputationsmenge Π' bestimmen:⁴⁴

$$\begin{aligned} & \max_{\mathbf{x}, y_S, t} t \\ & \text{unter den Nebenbedingungen} \\ & \mathbf{x} \in \Pi \end{aligned} \quad (3.97)$$

$$t \leq y_S, \quad \text{für alle } S \in \mathcal{S} \quad (3.98)$$

$$t \geq 0, \quad x_i \geq 0, \quad i = 1, 2, \dots, n. \quad (3.99)$$

In diesem Programm kann die Nebenbedingung (3.97) durch $\mathbf{Ax} + \mathbf{By} = \mathbf{d}$ ersetzt werden, sobald ein System (3.94) bzw. (3.90)-(3.93) zur Beschreibung des Paares (Π, \mathcal{S}) vorliegt.⁴⁵

⁴⁴Vgl. dazu Anhang B.

⁴⁵An dieser Stelle merken wir an, dass die Variable t zwar eine freie Variable ist, aber in der Problemlösung nie negativ sein wird, weil $y_S > 0$ ist. Im Gegensatz dazu ist die Variable ϵ der linearen Programme, die z. B. nach dem Verfahren von Maschler et al. [82] gelöst werden (vgl. S. 111), auch eine freie Variable, die tatsächlich sowohl positive als auch negative Werte annehmen wird.

$$\begin{aligned} \text{LP:} \quad & \max_{\mathbf{x}, \mathbf{y}, t} t \\ & \text{unter den Nebenbedingungen} \\ & \mathbf{Ax} + \mathbf{By} = \mathbf{d} \end{aligned} \tag{3.100}$$

$$t \leq y_S, \quad \text{für alle } S \in \mathcal{S} \tag{3.101}$$

$$t \geq 0, \quad x_i \geq 0, \quad i = 1, 2, \dots, n. \tag{3.102}$$

Um die Ungleichungen (3.101) als Gleichungen darzustellen, definieren wir

$$z_S + t := y_S. \tag{3.103}$$

Es gilt, dass

$$t \leq y_S \Rightarrow y_S - t \geq 0 \Rightarrow z_S \geq 0, \quad \forall S \in \mathcal{S}. \tag{3.104}$$

Demzufolge können wir das lineare Programm LP wie folgt umschreiben:

$$\begin{aligned} \text{LP'} : \quad & \max_{\mathbf{x}, \mathbf{z}, t} t \\ & \text{unter den Nebenbedingungen} \\ & \mathbf{Ax} + \mathbf{Bz} + (\mathbf{Be}_S)t = \mathbf{d}, \\ & t \geq 0, \quad \mathbf{x} \geq \mathbf{0}, \quad \mathbf{z} \geq \mathbf{0}, \end{aligned}$$

wobei $\mathbf{e}_S \in \mathbb{R}^{|\mathcal{S}|}$ der Indikatorvektor von $S \in \mathcal{S}$ ist.⁴⁶ Zur Veranschaulichung bilden wir nun in der Tabelle 3.2 die Nebenbedingungsmatrix des linearen Programms LP' des Einführungsbeispiels für die erste Iteration.

Potters et al. [105] verwenden in ihrem Algorithmus lineare Programme der Form LP' zur Berechnung des Nukleolus kooperativer Spiele. Sie schlagen ein von ihnen als „erweiterte Simplex-Methode“ bezeichnetes Verfahren zur Berechnung des Nukleolus vor. Es handelt sich dabei um eine Modifikation der Simplex-Methode. Im Folgenden beschreiben wir die einzelnen Schritte der „erweiterten Simplex-Methode“.

Um das lineare Programm LP' zu lösen, bringen die Autoren zunächst die Variable t durch eine Pivotoperation in die Basis.⁴⁷ Wenn t eine Basisvariable ist, erscheint sie

⁴⁶Für $\mathcal{S} := \{\{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}\}$ ist beispielsweise $\mathbf{e}_{\{1,3\}} = (0 \ 0 \ 0 \ 0 \ 1 \ 0)^T$.

⁴⁷Das ist ein in der Literatur üblicher Schritt zur Lösung linearer Programme der Form LP'. Vgl. dazu z. B. Vanderbei [141], S. 167-171.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Tabelle 3.2: Nebenbedingungsmatrix des LP' für das Einführungsbeispiel

t	x_1	x_2	x_3	$z_{\{1\}}$	$z_{\{1,2\}}$	$z_{\{1,3\}}$	$z_{\{2\}}$	$z_{\{2,3\}}$	$z_{\{3\}}$	d
0	1	1	1	0	0	0	0	0	0	42
1	-1	0	0	1	0	0	0	0	0	26
1	-1	-1	0	0	1	0	0	0	0	20
1	0	1	0	0	0	1	0	0	0	52
1	0	-1	0	0	0	0	1	0	0	26
1	1	0	0	0	0	0	0	1	0	42
1	1	1	0	0	0	0	0	0	1	68

nur in einer Gleichung. Es sei j diejenige Zeile, die nach der Pivotoperation die Variable t enthält. Damit t die Basis nicht verlässt, wird bei den weiteren Pivotoperationen kein Pivotelement aus der Zeile j gewählt. Nach dieser ersten Pivotoperation addieren sich die Koeffizienten der Zielfunktionszeile und diejenigen der Zeile j auf null. Das heisst, die Zeile j besteht aus den Koeffizienten, die sich von den Koeffizienten der Zielfunktionszeile nur dadurch unterscheiden, dass sie jeweils die umgekehrten Vorzeichen aufweisen.⁴⁸ Somit können wir die Zeile j als die Zielfunktion des linearen Programms betrachten und weitere Pivotoperationen durchführen, bis wir ein Optimum gefunden haben. Ein Optimum des linearen Programms LP' ist erreicht, wenn alle Koeffizienten der Zielfunktionszeile, also der Zeile j , durch Pivotoperationen nicht-negativ werden. Dies impliziert, dass die Pivotspalte immer eine der Spalten ist, die einen negativen Eintrag in der Zeile j hat.

Wir nehmen an, dass nach der Maximierung von t die Zielfunktionszeile j die folgende Gestalt annimmt:

$$\sum_{i \in N} a_{ji} x_i + \sum_{S \in \mathcal{S}} b_{jS} z_S + t = d_j. \quad (3.105)$$

Die Zeile j ist die einzige Zeile, die die Variable t beinhaltet. Im Optimum haben wir in dieser Zeile nur noch nicht-negative Koeffizienten, d. h. $a_{ji} \geq 0$, $\forall i$ und $b_{jS} \geq 0$,

⁴⁸Für den Beweis dieser Eigenschaft und Beispiele dazu vgl. z. B. Vanderbei [141], S. 169 f., Owen [97], S. 56 f., und Owen [98], S. 209 f.

3.3 Methoden der dritten Generation

$\forall S$. Des Weiteren ist jede Variable x_i bzw. jede Variable z_S eine Nichtbasisvariable, wenn $a_{ji} > 0$ bzw. $b_{jS} > 0$ gilt.⁴⁹ Das heisst formal

$$a_{ji} > 0 \quad \Rightarrow \quad x_i = 0, \quad (3.106)$$

$$b_{jS} > 0 \quad \Rightarrow \quad z_S = 0. \quad (3.107)$$

Dies kann man wie folgt begründen. Im Optimum impliziert die Zielfunktionszeile (3.105) den maximalen Wert von t . Da alle Koeffizienten a_{ji} und b_{jS} nicht-negativ sind, müssen die Variablen x_i und z_S den Wert Null annehmen, damit t maximal wird.⁵⁰

Somit erhalten wir den maximalen Wert von t wegen (3.106) und (3.107) als

$$t = d_j =: \hat{t}. \quad (3.108)$$

Demnach können wir die Zeile j durch die folgenden Gleichungen ersetzen

$$\begin{aligned} x_i &= 0 && \text{für alle } i \text{ mit } a_{ji} > 0, \\ z_S &= 0 && \text{für alle } S \text{ mit } b_{jS} > 0, \\ t &= d_j =: \hat{t}. \end{aligned} \quad (3.109)$$

Auf Grund der Definitionen für y_S und z_S gilt⁵¹

$$z_S = y_S - \hat{t} \quad (3.110)$$

$$= c - [v(S) - x(S)] - \hat{t} \quad (3.111)$$

$$= (c - \hat{t}) - e(S, \mathbf{x}) \quad \text{für alle } S \in \mathcal{S}. \quad (3.112)$$

Daher ist \hat{t} derjenige Wert, um den der maximale Koalitionsüberschuss c für das nächste lineare Programm reduziert wird. Der maximale Koalitionsüberschuss in der

⁴⁹Es kann aber auch der Fall vorliegen, dass im Optimum in der Zielfunktionszeile der Koeffizient einer Nichtbasisvariablen zwar nicht-negativ ist, aber den Wert Null hat. In diesem Fall kann man diese Variable mit einer Pivotoperation in die Basis nehmen, ohne dabei den optimalen Wert der Zielfunktion zu ändern. Der Wert der Zielfunktion wird sich nicht ändern, denn die Variable wird im Optimum immer noch den Wert Null besitzen, obwohl sie zu einer Basisvariablen wird. Aus diesem Grund betrachten wir hier nur diejenigen Nichtbasisvariablen, die einen positiven Koeffizienten in der Zeile j aufweisen.

⁵⁰Diese Aussage ist in der Literatur als Komplementärschlupfbedingung bekannt. Eine allgemeine Beschreibung dieser Bedingung ist im Anhang A.1, S. 287, zu finden.

⁵¹Vgl. dazu die Definitionsgleichungen (3.89) und (3.103).

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

nächsten Iteration ist also $c - \hat{t}$. Für die nächste Iteration entfernen nun Potters et al. die Gleichung $t = \hat{t}$ (vgl. (3.109)) aus dem linearen Programm, benennen die Variable z_S als y_S , $\forall S \in \mathcal{S}$ und erhalten somit die Beschreibung des neuen Paares (Π', \mathcal{S}) für den ausgewählten Spieler i' . Hierbei beinhaltet Π' jene Imputationen, die eine Lösung des linearen Programms LP' sind. Eine Variable y_S ist jetzt die Differenz zwischen dem (neuen) aktuellen maximalen Koalitionsüberschuss $c - \hat{t}$ und $e(S, \mathbf{x})$ (vgl. Gleichung 3.112).

Gemäss dem Algorithmus von Potters et al. [105] ermitteln wir im zweiten Schritt jeder Iteration mindestens eine Koalition, die für alle Imputationen in Π' einen konstanten Überschuss besitzt. Mindestens eine solche Koalition $S \in \mathcal{S}$ wird für die nächste Iteration aus \mathcal{S} entfernt, weil deren Überschuss, also ihre Unzufriedenheit, minimiert ist. Zu diesem Zweck definieren Potters et al. [105] die sog. *elementare Gleichung*:

Definition 3.3.2 (elementare Gleichung) Eine Gleichung k der Koeffizientenmatrix der Nebenbedingungen im Optimum des linearen Programms LP' wird als *elementar* bezeichnet, wenn es in dieser Zeile genau eine Koalition $S \in \mathcal{S}$ gibt, so dass

$$b_{kS} \neq 0, \quad (3.113)$$

$$b_{kS'} = 0 \text{ für alle } S' \in \mathcal{S}, S' \neq S \text{ und} \quad (3.114)$$

$$a_{ki} = 0 \text{ für alle } i \in N \quad (3.115)$$

gelten. Die Koalition S heisst *Index* der Zeile k .

Somit ist eine Gleichung k elementar, falls diese im Optimum die Form $z_S = d_k$ besitzt, also die Variable z_S über die aktuelle Imputationsmenge den konstanten Wert d_k hat. Der Überschuss jener Koalition $S \in \mathcal{S}$, für die diese Eigenschaft gilt, ist folglich ebenfalls konstant (vgl. dazu die Gleichung 3.112).

Lemma 3.3.1 Nach dem Schritt 1 jeder Iteration des Algorithmus besitzt das optimale Simplex-Tableau mindestens eine elementare Zeile.

Beweis (Potters et al. [105], S. 761):

Das lineare Programm LP' (vgl. S. 137) haben wir aufgestellt, indem wir im linearen Programm LP jede Variable y_S durch $z_S + t$ ersetzt haben. Demzufolge haben wir als Nebenbedingungen von LP' das Gleichungssystem

$$\mathbf{Ax} + \mathbf{Bz} + (\mathbf{Be}_S)t = \mathbf{d} \quad (3.116)$$

erhalten. Im Gleichungssystem (3.116) sind die Variablen z_S und die Variable $x_{i'}$ Basisvariablen. Die Koeffizienten dieser Basisvariablen bilden die sog. Koeffizientenmatrix der Basisvariablen oder kurz Basismatrix.⁵² Jede Spalte der Basismatrix ist eine Einheitsspalte und jede Zeile der Basismatrix ist eine Einheitszeile, sodass die Basismatrix eine Einheitsmatrix ist. Demzufolge beträgt die Summe der Einträge jeder Zeile dieser Matrix eins. Des Weiteren ist im Gleichungssystem (3.116) der Koeffizient der Variablen t in der Zeile, die die Basisvariable $x_{i'}$ enthält, null und in jeder Zeile, die eine Variable z_S beinhaltet, eins (vgl. dazu die Tabelle 3.2). Somit stimmt zu Beginn der Optimierung in jeder Gleichung des Gleichungssystems (3.116) die Summe der Koeffizienten der Variablen z_S mit dem Koeffizienten der Variablen t überein.⁵³ Diese Eigenschaft des Gleichungssystems (3.116) gilt auch im optimalen Simplex-Tableau, weil sie nach Pivotoperationen erhalten bleibt. Also gilt sie im Optimum auch für die Zeile j , die die Variable t beinhaltet. Da t im Optimum eine Basisvariable ist, ist die Spalte t eine Einheitsspalte mit einer Eins in der Zeile j . Somit gilt

$$\sum_{S \in \mathcal{S}} b_{jS} = 1. \quad (3.117)$$

Es existiert also mindestens eine Koalition $S \in \mathcal{S}$ mit $b_{jS} > 0$. Wenn wir die Zeile j durch die Gleichungen (3.109) (vgl. S. 139) ersetzen, ist für jede Variable z_S mit $b_{jS} > 0$ die Gleichung $z_S = 0$ eine elementare Gleichung. \square

Der Tatbestand, dass im Optimum des linearen Programms jede Variable z_S mit dem Koeffizienten $b_{jS} > 0$ in der Zielfunktionszeile den konstanten Wert Null haben

⁵²Vgl. dazu die Koeffizientenmatrix der Nebenbedingungen von LP' in der Tabelle 3.2 auf S. 138. In der Tabelle bilden die Spalten von x_3 bis einschliesslich $z_{\{3\}}$ die Basismatrix.

⁵³Vgl. dazu die Tabelle 3.2 auf S. 138.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

muss, damit die Variable t ihren maximalen Wert annimmt, impliziert, dass jede solche Koalition S in der aktuellen Imputationsmenge einen konstanten Überschuss, nämlich $c - \hat{t}$, besitzt.⁵⁴

Ist eine Zeile k elementar und S der Index der Zeile k , so entfernen wir die Koalition S aus der Menge \mathcal{S} . Zudem entfernen wir die Zeile k und die zu z_S gehörende Spalte aus dem Simplex-Tableau. Nach diesen Operationen besitzt die Koeffizientenmatrix der Nebenbedingungen immer noch eine Basis, weil wir jedes Mal die Spalte einer Basisvariablen und eine Zeile, die diese Variable enthält, aus dem Simplex-Tableau entfernen. Falls $\mathcal{S} \neq \emptyset$ ist, bilden wir die neue Spalte t , indem wir in jeder Zeile die Koeffizienten der Variablen z_S addieren. Anschliessend gehen wir zu Schritt 1 des Algorithmus von Potters et al. und führen eine neue Iteration mit dem modifizierten Simplex-Tableau durch.

Wir geben nun die einzelnen Schritte des Verfahrens von Potters et al. [105] detailliert an:

Initialisierung

- $\Pi := \mathcal{I}(N, v)$ und $\mathcal{S} := 2^N \setminus \{\emptyset, N\}$,
- $c := \max_{S \in \mathcal{S}} v(S)$,
- Man wähle einen beliebigen Spieler $i' \in N$ aus,
- Die Beschreibung des Paares (Π, \mathcal{S}) lautet

$$\begin{aligned} x(N) &= v(N) \\ y_S - x(S) &= c - v(S), & \text{falls } i' \notin S, \ S \in \mathcal{S} \\ y_S + x(N \setminus S) &= c + v(N) - v(S), & \text{falls } i' \in S, \ S \in \mathcal{S} \\ \mathbf{x} &\in \mathbb{R}_{\geq 0}^n, \mathbf{y} \in \mathbb{R}_{\geq 0}^{|\mathcal{S}|}. \end{aligned}$$

Es sei $\mathbf{Ax} + \mathbf{By} = \mathbf{d}$ die obige Beschreibung des Paares (Π, \mathcal{S}) für den ausgewählten Spieler i' in Matrixschreibweise.

⁵⁴Vgl. dazu die Bedingung (3.107) und ihre Begründung auf S. 139 sowie die Gleichung (3.112) und Reijnierse/Potters [103], S. 93.

Schritt 1

Beginnend mit einer Pivotoperation in der Spalte t löse man das folgende lineare Programm⁵⁵

$$\begin{aligned} \text{LP}' : \quad & \max_{\mathbf{x}, \mathbf{y}, t} t \\ & \text{unter den Nebenbedingungen} \\ & \mathbf{Ax} + \mathbf{By} + (\mathbf{Be}_S)t = \mathbf{d} \\ & t \geq 0, \mathbf{x} \geq \mathbf{0}, \mathbf{y} \geq \mathbf{0}. \end{aligned}$$

Schritt 2

Die Zeile j , die im optimalen Simplex-Tableau die Variable t enthält, sei

$$\sum_i a_{ji}x_i + \sum_{S \in \mathcal{S}} b_{jS}y_S + t = d_j.$$

- (i) Für alle $S \in \mathcal{S}$ mit $b_{jS} > 0$ entferne man die Koalitionen S aus \mathcal{S} , d. h.

$$\mathcal{S} := \mathcal{S} \setminus \{S \mid b_{jS} > 0, S \in \mathcal{S}\}.$$

- (ii) Für alle $i \in N$ mit $a_{ji} > 0$ ersetze man die Spalte x_i durch die Null-Spalte und füge die Gleichung $x_i = 0$ hinzu.

- (iii) Man entferne die Zeile j , welche nun die Form $t = d_j$ bekommen hat.

- (iv) Für jede elementare Gleichung k entferne man die Zeile k und die zu deren Index S gehörende Spalte S aus dem optimalen Simplex-Tableau sowie die Koalition S aus \mathcal{S} .

Schritt 3

Das resultierende Tableau ist wiederum von der Form $\mathbf{Ax} + \mathbf{By} = \mathbf{d}$. Falls $\mathcal{S} \neq \emptyset$ ist, bilde man die neue Spalte t und gehe zu Schritt 1. Die neue Spalte t in jeder

⁵⁵Das hier aufgestellte lineare Programm ist dasselbe wie das lineare Programm LP' auf S. 137. Wir haben hier nur die Variablen z_S umbenannt. Ein Vergleich der Tabellen 3.1 und 3.2 verdeutlicht, dass die Wahl dieser Notation zulässig ist. Wir folgen damit aus Kompatibilitätsgründen der Notation der Autoren.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Zeile besteht aus der Summe der Koeffizienten der Variablen y_S . Falls $\mathcal{S} = \emptyset$ ist, ist der Nukleolus des Spiels gefunden. Er stimmt mit dem Vektor \mathbf{x}^* in der optimalen Lösung $(\mathbf{x}^*, \mathbf{y}^*, \hat{t})$ des linearen Programms LP' überein.

Bezüglich des Aufwands des oben beschriebenen Algorithmus gelten folgende Aussagen. Für ein kooperatives Spiel mit n Spielern hat das Simplex-Tableau des ersten zu lösenden linearen Programms LP' insgesamt $2^n - 1$ Zeilen und $2^n + n$ Spalten. Diese werden aber im Laufe der Iterationen reduziert, da in jeder Iteration die elementaren Zeilen und die dazugehörenden Spalten aus dem Simplex-Tableau entfernt werden. Potters et al. [105], S. 762 f., zeigen, dass zur Berechnung des Nukleolus eines kooperativen Spiels mit n Spielern höchstens $n - 1$ lineare Programme gelöst werden. Weiterhin hat die von den Autoren vorgeschlagene Beschreibung eines Paares (Π, \mathcal{S}) , die die Koeffizientenmatrix der Nebenbedingungen des linearen Programms LP' bildet, bereits die Eigenschaft, dass sie eine Basis enthält. Folglich wird bei der Lösung linearer Programme der Form LP' zu Beginn keine Basislösung berechnet.

Ein Beispiel zur Illustration des Algorithmus werden wir zweckmässigerweise erst im Kapitel 5 vorstellen, wo wir den Algorithmus von Potters et al. [105] programmieren.

3.3.4 Das Verfahren von Derks/Kuipers

Die allgemeine Vorgehensweise bei den Methoden der dritten Generation zur Berechnung des Nukleolus kooperativer Spiele ist dadurch gekennzeichnet, dass man eine Folge linearer Programme löst. Während diese linearen Programme sowohl beim Verfahren von Maschler et al. [82] als auch beim Verfahren von Sankaran [113] mit Hilfe einer allgemein gebräuchlichen Methode wie z. B. der Simplex-Methode gelöst werden, zeigen Potters et al. [105], wie diese linearen Programme mit Hilfe des von ihnen als „erweiterte Simplex-Methode“ bezeichneten Verfahrens gelöst werden können. In diesem Abschnitt stellen wir das von Derks/Kuipers [32] vorgeschlagene Verfahren zur Berechnung des Prenukleolus kooperativer Spiele mit n Spielern vor. Derks/Kuipers [32] gehen in ihrem Algorithmus ebenfalls von einer Folge linearer Programme aus und zeigen, wie diese linearen Programme gelöst werden können. Wie wir noch

3.3 Methoden der dritten Generation

erläutern werden, verwenden Derks/Kuipers für die Ermittlung einer Lösung dieser linearen Programme zwar die Simplex-Methode, aber sie nutzen bestimmte Eigenschaften der Koeffizientenmatrix der Nebenbedingungen aus, um den Aufwand der Pivotoperationen zu reduzieren. Dabei bestehen die Koeffizientenmatrizen der Nebenbedingungen der linearen Programme nur aus n Zeilen und n Spalten. Zudem sind für ein kooperatives Spiel mit n Spielern höchstens $n - 1$ lineare Programme zu lösen, um den Prenukleolus des Spiels mit Hilfe des Verfahrens von Derks/Kuipers [32] zu berechnen.

Im Folgenden beschreiben wir zunächst im Grundsatz, wie der Prenukleolus gemäss dem Algorithmus von Derks/Kuipers [32] im Wege der Lösung einer Folge linearer Programme berechnet werden kann. Danach erläutern wir die von den Autoren zur Bestimmung der Lösung dieser linearen Programme vorgeschlagene spezielle „Simplex-Methode“. Anschliessend zeigen wir, wie die Anzahl der Rechenschritte für die Durchführung einer Pivotoperation reduziert werden kann.

Gegeben sei ein kooperatives Spiel $\Gamma = (N, w)$ mit n Spielern wobei N die Spielermenge und w die charakteristische Funktion des Spiels ist. Es sei $\mathcal{F} \subseteq 2^N$ ein Koalitionssystem, dessen Eigenschaften wir nachher erläutern werden. Die Berechnung des Prenukleolus des Spiels mit Hilfe des Verfahrens von Derks/Kuipers erfolgt dadurch, dass lineare Programme folgender Form gelöst werden:

$$\begin{aligned} \mathcal{P}(\mathcal{F}, w) : \quad & \min_{t, \mathbf{x}} \quad t \\ & \text{unter den Nebenbedingungen} \\ & x(S) = w(S), \quad \forall S \in \mathcal{F} \\ & x(S) + t \geq w(S), \quad \forall S \notin \mathcal{F}. \end{aligned}$$

Für die weiteren Ausführungen definieren wir für jede Koalition $S \subseteq N$ den n -dimensionalen Zeilenvektor $\mathbf{1}_S := (1_{S,1} \ 1_{S,2} \ \dots \ 1_{S,n})$, wobei $1_{S,i} := 1$, falls $i \in S$ ist, und $1_{S,i} := 0$, falls $i \notin S$ ist. Diesen Vektor nennen wir den *Inzidenzvektor* der Koalition $S \subseteq N$. Betrachten wir beispielsweise ein kooperatives Spiel mit $n = 3$ Spielern und die Koalition $S = \{2\}$, so ergibt sich der Inzidenzvektor dieser Koalition als $\mathbf{1}_{\{2\}} := (0 \ 1 \ 0)$. Ein weiterer Begriff, den Derks/Kuipers [32] für die

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Darstellung ihres Verfahrens verwenden, ist das sog. *saturierte Mengensystem*, auf dessen Bedeutung für die Berechnung des Nukleolus wir später auf S. 147 eingehen werden. Ein Mengensystem $\mathcal{F} \subseteq 2^N$ nennt man saturiert, falls jede Koalition $S \subseteq N$, deren Inzidenzvektor als Linearkombination⁵⁶ der Inzidenzvektoren der Koalitionen $T \in \mathcal{F}$, $T \neq S$, dargestellt werden kann, ein Element von \mathcal{F} ist. Zum Beispiel ist für $N = \{1, 2, 3\}$ das Mengensystem $\mathcal{F} = \{\{1\}, \{2, 3\}\}$ nicht-saturiert, weil die Linearkombination

$$1 \cdot \mathbf{1}_{\{1\}} + 1 \cdot \mathbf{1}_{\{2,3\}} = 1(1 \ 0 \ 0) + 1(0 \ 1 \ 1) = (1 \ 1 \ 1) \quad (3.118)$$

dem Inzidenzvektor der grossen Koalition N entspricht, welche aber kein Element von \mathcal{F} ist. Hingegen kann man analog zu oben zeigen, dass das Mengensystem $\mathcal{F} = \{\{1\}, \{2, 3\}, \{1, 2, 3\}\}$ saturiert ist.⁵⁷

Aus der Definition von saturierten Mengensystemen folgt, dass für jedes saturierte Mengensystem \mathcal{F} stets eine reellwertige $(p \times n)$ -Matrix \mathbf{F} mit dem Rang $p \leq n$ existiert, so dass die Inzidenzvektoren der Koalitionen $S \in \mathcal{F}$ als Linearkombination der Zeilen von \mathbf{F} dargestellt werden können. Das heisst, dass jedes saturierte Mengensystem \mathcal{F} durch eine Matrix \mathbf{F} vollständig repräsentiert werden kann. Für das saturierte Mengensystem $\mathcal{F} = \{\{1\}, \{2, 3\}, \{1, 2, 3\}\}$ erfüllt z. B. die Matrix

$$\mathbf{F} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix},$$

deren Rang $p = 2$ ist, diese Eigenschaft.⁵⁸

Wie wir noch ausführen werden, verwenden Derks/Kuipers [32] diese Eigenschaft saturierter Mengensysteme, um die jeweils zu lösenden linearen Programme zu vereinfachen. Ist ein Mengensystem \mathcal{F} nicht-saturiert, so nennen Derks/Kuipers [32] die

⁵⁶Für die Definition einer Linearkombination von Vektoren vgl. Anhang D.

⁵⁷Damit das Mengensystem \mathcal{F} saturiert ist, darf keine Linearkombination der Inzidenzvektoren der Koalitionen $S \in \mathcal{F}$ mit dem Inzidenzvektor einer Koalition $S \notin \mathcal{F}$, $S \subseteq N$, übereinstimmen. Wir zeigen dies exemplarisch für die Koalition $S = \{2\} \notin \mathcal{F}$. Dafür bilden wir die Linearkombination $\alpha_1(1 \ 0 \ 0) + \alpha_2(0 \ 1 \ 1) + \alpha_3(1 \ 1 \ 1) = \mathbf{1}_{\{2\}} = (0 \ 1 \ 0)$ und leiten daraus das Gleichungssystem $\alpha_1 + \alpha_3 = 0$, $\alpha_2 + \alpha_3 = 1$, $\alpha_2 + \alpha_3 = 0$ her. Aus den beiden letzten Gleichungen folgt, dass das Gleichungssystem keine Lösung besitzt.

⁵⁸Der Rang der Matrix \mathbf{F} ist $p = 2$, weil die Zeilen der Matrix linear unabhängig sind, also aus $\alpha_1(1 \ 0 \ 0) + \alpha_2(0 \ 1 \ 1) = (0 \ 0 \ 0)$ für $\alpha_1, \alpha_2 \in \mathbb{R}$, stets $\alpha_1 = \alpha_2 = 0$ folgt. Vgl. dazu Anhang D.

Schnittmenge aller saturierten Mengensysteme, die das Mengensystem \mathcal{F} beinhalten, die *Saturation* $\bar{\mathcal{F}}$ des Mengensystems \mathcal{F} . Die Saturation eines saturierten Mengensystems ist das Mengensystem selbst. Aus dieser Definition der Saturation folgt, dass $\bar{\mathcal{F}}$ das kleinste saturierte Mengensystem ist, welches das Mengensystem \mathcal{F} beinhaltet. Beispielsweise ist das Koalitionssystem $\bar{\mathcal{F}} = \{\{1\}, \{2, 3\}, \{1, 2, 3\}\}$ die Saturation des Koalitionssystems $\mathcal{F} = \{\{1\}, \{2, 3\}\}$.⁵⁹

Im Hinblick auf die Berechnung des Nukleolus kooperativer Spiele können wir saturierte Koalitionssysteme bzw. Saturationen wie folgt interpretieren: Wir nehmen an, dass wir den Nukleolus eines kooperativen Spiels $\Gamma = (N, v)$ mit n Spielern berechnen wollen. Es sei $\mathcal{F} \subseteq 2^N$ die Menge jener Koalitionen, deren Überschüsse bereits minimiert worden sind. Wie wir noch an Hand eines Beispiels verdeutlichen werden, kann in diesem Fall der minimale Überschuss jeder Koalition $T \notin \mathcal{F}$, deren Inzidenzvektor als Linearkombination der Koalitionen $S \in \mathcal{F}$ dargestellt werden kann, mit Hilfe der Überschüsse der Koalitionen $S \in \mathcal{F}$ bestimmt werden. Das heisst, der minimale Überschuss jeder solchen Koalition T ist implizit durch Linearkombination der minimalen Überschüsse der Koalitionen $S \in \mathcal{F}$ bestimmt. Folglich kann jede solche Koalition T im weiteren Verlauf der Berechnungen als eine Koalition betrachtet werden, deren Überschuss minimiert worden ist. Die Menge aller solcher Koalitionen bildet die Saturation $\bar{\mathcal{F}}$ des Koalitionssystems \mathcal{F} . Zur Illustration betrachten wir das Einführungsbeispiel. Wir haben bei der Berechnung des Nukleolus des Einführungsbeispiels z. B. mit Hilfe des Algorithmus von Maschler et al. [82] nach der Lösung des ersten linearen Programms gefunden (vgl. S. 113 f.), dass die Koalitionen $\{1\}$ und $\{2, 3\}$ denselben realisierbaren minimalen Überschuss $\epsilon^1 = -8$ besitzen und folglich

$$\begin{array}{rcl} x_1 & = & 8 \\ x_2 + x_3 & = & 34 \end{array} \Rightarrow \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 8 \\ 34 \end{pmatrix} \quad (3.119)$$

gelten muss. Jede Lösung des Gleichungssystems (3.119) ist also für die Koalitionen $\{1\}$ und $\{2, 3\}$ mit dem minimalen Überschuss von -8 verbunden. Die einzige Ko-

⁵⁹Dieser Tatbestand ist nicht ohne weiteres ersichtlich. Die Richtigkeit der Aussage kann aber mit Hilfe der oben angewendeten Techniken gezeigt werden.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

alition, deren Inzidenzvektor als Linearkombination der Inzidenzvektoren der Koalitionen $\{1\}$ und $\{2, 3\}$ dargestellt werden kann, ist die grosse Koalition $N = \{1, 2, 3\}$ (vgl. die Gleichung (3.118)). Den minimalen Überschuss der Koalition N können wir also durch die Linearkombination der beiden Gleichungen $x_1 = 8$ und $x_2 + x_3 = 34$ berechnen: Addieren wir nämlich die beiden Gleichungen, erhalten wir die Gleichung $x_1 + x_2 + x_3 = 42$, welche zum minimalen Überschuss null der grossen Koalition N führt. Wie wir noch erläutern werden, bildet die erörterte Eigenschaft saturierter Koalitionssysteme eine wichtige Grundlage für den Algorithmus von Derks/Kuipers [32].

Bevor wir das Verfahren von Derks/Kuipers ausführlich diskutieren, beschreiben wir zunächst allgemein die Vorgehensweise des Verfahrens. Die Berechnung des Prenukleolus eines kooperativen Spiels $\Gamma = (N, v)$ nach dem Verfahren von Derks/Kuipers erfolgt über mehrere Iterationen, die jeweils die unten beschriebenen Schritte beinhalten. Zu Beginn des Verfahrens definieren wir $\mathcal{F}^{(1)} := \{N\}$, $w^{(1)} := v$ und stellen das erste lineare Programm $\mathcal{P}(\mathcal{F}^{(1)}, w^{(1)})$ auf, das eine optimale Lösung besitzt.⁶⁰ Es seien \mathbf{x} und \mathbf{y} zwei beliebige optimale Lösungen des linearen Programms $\mathcal{P}(\mathcal{F}^{(1)}, w^{(1)})$ und $t^{(1)}$ der optimale Zielfunktionswert. Des Weiteren sei $\mathcal{A}^{(1)}$ ein nicht-leeres Koalitionssystem mit der Eigenschaft

$$\mathcal{A}^{(1)} \subseteq \{S \notin \mathcal{F}^{(1)} \mid x(S) = y(S) \ \forall \text{ optimalen Lösungen } (\mathbf{x}, t^{(1)}) \text{ und } (\mathbf{y}, t^{(1)})\}.$$

Die Menge $\mathcal{A}^{(1)}$ beinhaltet also diejenigen Koalitionen, die für alle optimalen Lösungen des linearen Programms denselben minimalen Koalitionsüberschuss besitzen. Später werden wir zeigen, dass die Menge $\mathcal{A}^{(1)}$ aus einer optimalen Lösung des zu $\mathcal{P}(\mathcal{F}^{(1)}, w^{(1)})$ dualen Problems bestimmt werden kann. Nachdem die Menge $\mathcal{A}^{(1)}$ festgelegt ist, werden durch

$$\begin{aligned} \bullet \quad \mathcal{F}^{(2)} &:= \overline{\mathcal{F}^{(1)} \cup \mathcal{A}^{(1)}}, \\ \bullet \quad w^{(2)}(S) &:= \begin{cases} x(S) & : \text{ falls } S \in \mathcal{F}^{(1)} \\ w^{(1)}(S) - t^{(1)} & : \text{ sonst,} \end{cases} \end{aligned}$$

⁶⁰Das lineare Programm $\mathcal{P}(\mathcal{F}^{(1)}, w^{(1)})$ entspricht dem linearen Programm LP_1 im Algorithmus von Maschler et al. [82] (vgl. S. 111 ff.). Maschler et al. [82], S. 334, zeigen, dass das lineare Programm LP_1 stets eine optimale Lösung besitzt.

3.3 Methoden der dritten Generation

das Koalitionssystem $\mathcal{F}^{(1)}$ und die charakteristische Funktion v des Spiels aktualisiert.⁶¹ In der nächsten Iteration lösen wir das lineare Optimierungsproblem $\mathcal{P}(\mathcal{F}^{(2)}, w^{(2)})$. Da in jeder Iteration k das Koalitionssystem $\mathcal{F}^{(k)}$ mindestens um eine neue Koalition erweitert wird, werden wir in der letzten Iteration $\mathcal{F}^{(k)} = 2^N$ erreichen. Derks/Kuipers zeigen, dass in diesem Fall die eindeutige Lösung des letzten linearen Programms mit dem Prenukleolus des Spiels zusammenfällt. Zudem weisen Derks/Kuipers [32], S. 4, nach, dass es ausreicht, zur Berechnung des Prenukleolus eines kooperativen Spiels mit n Spielern höchstens $n - 1$ lineare Programme zu lösen.

Nehmen wir an, dass $\mathcal{P}(\mathcal{F}^{(k-1)}, w^{(k-1)})$ in der Iteration $k - 1$ die Lösung $(\mathbf{x}^{(k-1)}, t^{(k-1)})$ besitzt. Da das Koalitionssystem $\mathcal{F}^{(k)}$ durch die Matrix $\mathbf{F}^{(k)}$ vollständig repräsentiert ist, können wir das lineare Programm $\mathcal{P}(\mathcal{F}^{(k)}, w^{(k)})$ folgendermassen umschreiben:

$$\begin{aligned} \mathcal{Q}(\mathbf{F}^{(k)}, \mathbf{x}^{(k-1)}, w^{(k)}) : \quad & \min_{t, \mathbf{x}} t \\ & \text{unter den Nebenbedingungen} \\ & \mathbf{F}^{(k)} \mathbf{x} = \mathbf{F}^{(k)} \mathbf{x}^{(k-1)} \end{aligned} \tag{3.120}$$

$$\mathbf{A}^{(k)} \mathbf{x} + \mathbf{1}t \geq \mathbf{w}^{(k)}. \tag{3.121}$$

Wie wir sogleich an Hand eines Beispiels illustrieren werden, stellen die Nebenbedingungen (3.120) analog zum linearen Programm $\mathcal{P}(\mathcal{F}^{(k)}, w^{(k)})$ bzw. $\mathcal{P}(\mathcal{F}, w)$ (vgl. S. 145) jene Koalitionen dar, deren minimaler Überschuss bereits ermittelt worden ist. Der Inzidenzvektor jeder solchen Koalition stimmt somit mit einer Linearkombination der Zeilen von $\mathbf{F}^{(k)}$ überein. In (3.121) ist $\mathbf{A}^{(k)}$ eine $(0, 1)$ -Matrix, deren Zeilen $(0, 1)$ -Zeilenvektoren mit n Komponenten sind. Die Zeilen dieser Matrix haben die Eigenschaft, dass sie nicht als Linearkombination der Zeilen der Matrix $\mathbf{F}^{(k)}$ dargestellt werden können. Die Zeilen der Matrix $\mathbf{A}^{(k)}$ sind somit die Inzidenzvektoren jener Koalitionen, die keine Elemente der saturierten Menge $\mathcal{F}^{(k)}$ sind. Der Vektor $\mathbf{w}^{(k)}$ ist der Spaltenvektor der Koalitionswerte derjenigen Koalitionen, deren Inzidenzvektor der jeweils zugehörigen Zeile der Matrix $\mathbf{A}^{(k)}$ entspricht. Der Vektor

⁶¹Während die Bestimmung von $w^{(2)}$ direkt gemäss Definition erfolgen kann, ist die Ermittlung des Mengensystems $\mathcal{F}^{(2)}$ aufwändiger. Wir werden die Vorgehensweise hierfür auf S. 151 f. vorstellen.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

$\mathbf{1}$ ist ein Spaltenvektor, dessen Komponenten alle Einsen sind. Nun können wir das letzte lineare Programm noch vereinfachen. In der Iteration k gilt für die Koalitionen $S \notin \mathcal{F}^{(k)}$, dass

$$w^{(k)}(S) = v(S) - \sum_{l=1}^{k-1} t^{(l)}.$$

Folglich ist (\mathbf{x}, t) dann und nur dann eine optimale Lösung des linearen Programms $\mathcal{Q}(\mathbf{F}^{(k)}, \mathbf{x}^{(k-1)}, w^{(k)})$, wenn $(\mathbf{x}, t + \sum_{l=1}^{k-1} t^{(l)})$ eine optimale Lösung des folgenden linearen Programms ist:

$$\begin{aligned} \mathcal{Q}(\mathbf{F}^{(k)}, \mathbf{x}^{(k-1)}, v) : \quad & \min_{t, \mathbf{x}} \quad t \\ & \text{unter den Nebenbedingungen} \\ & \mathbf{F}^{(k)} \mathbf{x} = \mathbf{F}^{(k)} \mathbf{x}^{(k-1)} \\ & \mathbf{A}^{(k)} \mathbf{x} + \mathbf{1}t \geq \mathbf{v}. \end{aligned} \tag{3.122}$$

Hier ist der Vektor \mathbf{v} ein Spaltenvektor der ursprünglichen, nicht aktualisierten Koalitionswerte derjenigen Koalitionen, deren Inzidenzvektor einer Zeile der Matrix $\mathbf{A}^{(k)}$ entspricht. Somit sind die beiden linearen Programme

$$\mathcal{Q}(\mathbf{F}^{(k)}, \mathbf{x}^{(k-1)}, w^{(k)}) \quad \text{und} \quad \mathcal{Q}(\mathbf{F}^{(k)}, \mathbf{x}^{(k-1)}, v)$$

äquivalent. Dies impliziert, dass wir im Weiteren die charakteristische Funktion des Spiels nicht aktualisieren müssen.

Zur Illustration geben wir die Nebenbedingungen des linearen Programms $\mathcal{Q}(\mathbf{F}^{(k)}, \mathbf{x}^{(k-1)}, v)$ des Einführungsbeispiels für $k = 1$ an. Gemäss der Annahme $\mathcal{F}^{(1)} := \{N\}$ ist $\mathbf{F}^{(1)} = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}$. Folglich gilt

$$\mathbf{F}^{(1)} \mathbf{x}^{(0)} = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = x_1 + x_2 + x_3.$$

Wegen der Effizienzeigenschaft der Imputationen $\mathbf{x} = (x_1, x_2, x_3) \in \mathbb{R}_+^n$ muss $\mathbf{F}^{(1)} \mathbf{x}^{(0)} = x_1 + x_2 + x_3 = v(N) = 42$ erfüllt sein. Die Inzidenzvektoren jener Koali-

3.3 Methoden der dritten Generation

tionen $S \subseteq N$, die kein Element von $\mathcal{F}^{(1)}$ sind, bilden dann die Matrix $\mathbf{A}^{(1)}$, sodass die Nebenbedingungen, die Ungleichungen sind, durch

$$\mathbf{A}^{(1)}\mathbf{x} + \mathbf{1}t = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} t \geq \begin{pmatrix} v(\{1,2\}) \\ v(\{1,3\}) \\ v(\{2,3\}) \\ v(\{1\}) \\ v(\{2\}) \\ v(\{3\}) \end{pmatrix} = \begin{pmatrix} 6 \\ 16 \\ 26 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

dargestellt werden können.

Nun zeigen wir, wie das Koalitionssystem $\mathcal{A}^{(k)}$ bestimmt und das Koalitionssystem $\mathcal{F}^{(k)}$ aktualisiert werden können. Wie bereits erwähnt, folgt aus der Definition der saturierten Mengensysteme, dass sich das Koalitionssystem $\mathcal{F}^{(k)}$ durch die Matrix $\mathbf{F}^{(k)}$ vollständig bestimmen lässt. Daher ist es hinreichend, wenn wir zeigen, wie wir die Matrix $\mathbf{F}^{(k)}$ aktualisieren können. Hierzu betrachten wir das folgende zu $\mathcal{Q}(\mathbf{F}^{(k)}, \mathbf{x}^{(k-1)}, v)$ duale Problem:⁶²

$$(\mathcal{D}) : \quad \max_{\mathbf{z}, \mathbf{u}} \quad \mathbf{z}^T \mathbf{F}^{(k)} \mathbf{x}^{(k-1)} + \mathbf{u}^T \mathbf{v}$$

unter den Nebenbedingungen

$$\mathbf{z}^T \mathbf{F}^{(k)} + \mathbf{u}^T \mathbf{A}^{(k)} = \mathbf{0}^T \tag{3.123}$$

$$\mathbf{u}^T \mathbf{1} = 1 \tag{3.124}$$

$$\mathbf{u} \geq \mathbf{0}. \tag{3.125}$$

Da der Zulässigkeitsbereich jedes Problems $\mathcal{Q}(\mathbf{F}^{(k)}, \mathbf{x}^{(k-1)}, v)$ nicht-leer und beschränkt ist, wird uns dessen Lösung z. B. mit der Simplex-Methode eine primale optimale Lösung $(\bar{\mathbf{x}}, \bar{t})$ und eine duale optimale Lösung $(\bar{\mathbf{z}}, \bar{\mathbf{u}})$ liefern. Aufgrund der Nebenbedingungen (3.124) und (3.125) existiert mindestens eine Variable u_i mit $u_i > 0$. Die entsprechende Zeile der Matrix $\mathbf{A}^{(k)}$ sei $\mathbf{a}_i^{(k)}$. Gemäss der Komplementärschlupfbedingung⁶³ muss für die Zeile $\mathbf{a}_i^{(k)}$ die Gleichung

$$\mathbf{a}_i^{(k)} \bar{\mathbf{x}} + \bar{t} = v_i \tag{3.126}$$

⁶²In diesem Abschnitt sind die Vektoren als Spaltenvektoren definiert, solange nichts anderes angenommen wird.

⁶³Für die Beschreibung der Komplementärschlupfbedingungen vgl. Anhang A.1, S. 287.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

erfüllt sein. Da v_i konstant ist und sich der Zielfunktionswert \bar{t} für alle optimalen Lösungen des linearen Programms in der Iteration k nicht ändert, ist $\mathbf{a}_i^{(k)} \bar{\mathbf{x}}$ ebenfalls in allen optimalen Lösungen (\mathbf{x}, \bar{t}) des linearen Programms $\mathcal{Q}(\mathbf{F}^{(k)}, \mathbf{x}^{(k-1)}, v)$ konstant. Folglich ist eine Koalition ein Element der Menge $\mathcal{A}^{(k)}$, falls deren Inzidenzvektor keine Zeile der Matrix $\mathbf{F}^{(k)}$ ist und mit der Zeile $\mathbf{a}_i^{(k)}$ übereinstimmt. Die Aktualisierung der Matrix $\mathbf{F}^{(k)}$ erfolgt, indem die Zeile $\mathbf{a}_i^{(k)}$ in $\mathbf{F}^{(k)}$ an beliebiger Stelle hinzugefügt wird. Falls der Vektor \mathbf{u} mehr als eine positive Komponente besitzt, beinhaltet die Menge $\mathcal{A}^{(k)}$ diejenigen Koalitionen $S \notin \mathcal{F}^{(k)}$, deren Inzidenzvektoren mit diesen Zeilen übereinstimmen und gleichzeitig keine Zeile der Matrix $\mathbf{F}^{(k)}$ sind. In diesem Fall fügen wir mehrere Zeilen zur Matrix $\mathbf{F}^{(k)}$ hinzu, solange dadurch keine lineare Abhängigkeit der Zeilen der aktualisierten Matrix $\mathbf{F}^{(k+1)}$ entsteht. Das Mengensystem \mathcal{F}^{k+1} beinhaltet nun jene Koalitionen, deren Inzidenzvektoren als Linearkombination der Zeilen von $\mathbf{F}^{(k+1)}$ dargestellt werden können. Enthält $\mathcal{F}^{(k+1)}$ alle möglichen Koalitionen $S \subseteq N$, so stimmt $\bar{\mathbf{x}}$ mit dem Prenukleolus des Spiels überein. Sonst lösen wir das nächste lineare Programm $\mathcal{Q}(\mathbf{F}^{(k+1)}, \mathbf{x}^{(k)}, v)$ (vgl. (3.122)) usw., bis der Prenukleolus des Spiels erreicht ist.

Wie wir zu Beginn dieses Abschnitts erwähnt haben, schlagen Derks/Kuipers zur Lösung linearer Programme der Gestalt $\mathcal{Q}(\mathbf{F}^{(k)}, \mathbf{x}^{(k-1)}, v)$ eine spezielle Variante der Simplex-Methode vor. Diese Variante hat die Besonderheit, dass die Matrix \mathbf{A} nicht explizit aufgestellt werden muss, sondern implizit durch die Matrix \mathbf{F} abgebildet wird. Wir beschreiben im Folgenden zunächst diese spezielle Variante der Simplex-Methode und erläutern, wie wir sie bei der Berechnung des Prenukleolus eines kooperativen Spiels anwenden können. Anschliessend geben wir zur Illustration dieser Methode ein Beispiel an.

Derks/Kuipers [32] gehen von einer für die Berechnung des Nukleolus eines kooperativen Spiels herangezogenen linearen Optimierungsaufgabe der Form (3.122) aus:

$$\mathcal{P} := \min_{t, \mathbf{x}} \{t \mid \mathbf{F}\mathbf{x} = \mathbf{c}, \mathbf{A}\mathbf{x} + \mathbf{1}t \geq \mathbf{b}\}.$$

Hier ist \mathbf{A} eine $(m \times n)$ -Matrix, \mathbf{F} eine $(p \times n)$ -Matrix mit dem Rang $p \leq n$, $\mathbf{1}$ ein $(m \times 1)$ -Spaltenvektor, dessen Komponenten alle Einsen sind, und \mathbf{b} ein $(m \times 1)$ -Spaltenvektor. Gemäss unserer Annahme, dass die Matrix \mathbf{F} den Rang p besitzt, sind die Zeilen von \mathbf{F} linear unabhängig. Die lineare Optimierungsaufgabe $\mathcal{Q}(\mathbf{F}^{(k)}, \mathbf{x}^{(k-1)}, v)$, die wir zur Berechnung des Prenukleolus eines kooperativen Spiels mit Hilfe dieses Verfahrens lösen, besitzt die Form von \mathcal{P} . Wir stellen nun die von Derks/Kuipers [32] vorgeschlagenen Schritte zur Lösung eines linearen Optimierungsproblems vom Typ \mathcal{P} vor:

Schritt 1

Wir nehmen an, dass das Gleichungssystem $\mathbf{F}\mathbf{x} = \mathbf{c}$ eine zulässige Lösung besitzt. Eine solche Lösung sei \mathbf{x}^0 . Ferner sei $t^0 := \max \{b_i - \mathbf{a}_i \mathbf{x}^0 \mid i = 1, \dots, m\}$, wobei \mathbf{a}_i die Zeile i der Matrix \mathbf{A} ist. In diesem Fall ist (\mathbf{x}^0, t^0) eine zulässige Lösung des linearen Optimierungsproblems \mathcal{P} .

Schritt 2

Es sei $\mathbf{A}^0 \mathbf{x} + \mathbf{1}t \geq \mathbf{b}^0$ ein Teilsystem von $\mathbf{A}\mathbf{x} + \mathbf{1}t \geq \mathbf{b}$, wobei die Matrix \mathbf{A}^0 und der Vektor \mathbf{b}^0 die Eigenschaften haben, dass $\mathbf{A}^0 \mathbf{x}^0 + \mathbf{1}t^0 = \mathbf{b}^0$ gilt und die Zeilen der Matrix

$$\begin{pmatrix} \mathbf{F} & \mathbf{0} \\ \mathbf{A}^0 & \mathbf{1} \end{pmatrix} \quad (3.127)$$

linear unabhängig sind. Die Matrix \mathbf{A}^0 hat mindestens eine Zeile. Denn aus der Definition von t^0 existiert eine Zeile \mathbf{a}_i der Matrix \mathbf{A} , so dass $\mathbf{a}_i \mathbf{x}^0 + t^0 = b_i$ erfüllt ist. Die Anzahl der Zeilen der Matrix \mathbf{A}^0 sei s . Der Rang der Matrix (3.127) ist somit $(p + s)$.

Schritt 3

Das zu \mathcal{P} duale lineare Programm ist:

$$\mathcal{D} := \max_{\mathbf{z}, \mathbf{u}} \{\mathbf{z}^T \mathbf{c} + \mathbf{u}^T \mathbf{b} \mid \mathbf{z}^T \mathbf{F} + \mathbf{u}^T \mathbf{A} = \mathbf{0}^T, \mathbf{u}^T \mathbf{1} = 1, \mathbf{u} \geq \mathbf{0}\},$$

wobei \mathbf{z} ein $(p \times 1)$ -Spaltenvektor und \mathbf{u} ein $(m \times 1)$ -Spaltenvektor ist. Man betrachte das Gleichungssystem

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

$$\begin{aligned} \mathbf{z}^T \mathbf{F} + \mathbf{u}_0^T \mathbf{A}^0 &= \mathbf{0}, \\ \mathbf{u}_0^T \mathbf{1} &= 1. \end{aligned} \tag{3.128}$$

Hierin ist \mathbf{A}^0 eine $(s \times n)$ -Matrix und \mathbf{u}_0 ein $(s \times 1)$ -Spaltenvektor. Die Anzahl der Unbekannten ist $(p + s)$. Die Koeffizientenmatrix des Gleichungssystems (3.128) besitzt $p + s$ linear unabhängige Spalten (vgl. (3.127)). Folglich ist $p + s$ gleichzeitig der Rang der Koeffizientenmatrix des Gleichungssystems (3.128), welcher daher mit der Anzahl der Unbekannten des Gleichungssystems (3.128) übereinstimmt. Deswegen besitzt dieses Gleichungssystem höchstens eine Lösung.⁶⁴ Diesbezüglich sind die folgenden drei Fälle denkbar:

1. Fall Das Gleichungssystem (3.128) hat eine eindeutige Lösung $(\mathbf{z}^*, \mathbf{u}_0^*)$ mit $\mathbf{u}_0^* \geq \mathbf{0}$.

In diesem Fall ist (\mathbf{x}^0, t^0) eine optimale Lösung des linearen Programms \mathcal{P} , denn es gilt

$$\begin{aligned} t^0 &= \underbrace{(\mathbf{z}^{*T} \mathbf{F} + \mathbf{u}_0^{*T} \mathbf{A}^0)}_{=\mathbf{0}} \mathbf{x}^0 + \underbrace{\mathbf{u}_0^{*T} \mathbf{1}}_{=1} t^0 \\ &= \mathbf{z}^{*T} \mathbf{F} \mathbf{x}^0 + \mathbf{u}_0^{*T} (\mathbf{A}^0 \mathbf{x}^0 + \mathbf{1} t^0) \\ &= \mathbf{z}^{*T} \mathbf{c} + \mathbf{u}_0^{*T} \mathbf{b} \\ &\leq \max_{\mathbf{z}, \mathbf{u}} \{ \mathbf{z}^T \mathbf{c} + \mathbf{u}^T \mathbf{b} \mid \mathbf{z}^T \mathbf{F} + \mathbf{u}^T \mathbf{A} = \mathbf{0}, \mathbf{u}^T \mathbf{1} = 1, \mathbf{u} \geq \mathbf{0} \} \\ &= \min_{t, \mathbf{x}} \{ t \mid \mathbf{F} \mathbf{x} = \mathbf{c}, \mathbf{A} \mathbf{x} + \mathbf{1} t \geq \mathbf{b} \}. \end{aligned}$$

2. Fall Das Gleichungssystem (3.128) hat zwar genau eine Lösung $(\mathbf{z}^*, \mathbf{u}_0^*)$, aber mindestens ein Element von \mathbf{u}_0^* ist negativ. Es sei \hat{i} der kleinste Index, für den $u_{0_i}^* < 0$ ist. Man definiere \mathbf{B}^0 als diejenige Matrix, die durch streichen der Zeile \hat{i} aus der Matrix \mathbf{A}^0 entsteht. Entsprechend sei $\bar{\mathbf{u}}_0^*$ derjenige Vektor, der entsteht, wenn die Komponente \hat{i} aus \mathbf{u}_0^* gestrichen wird. Da das Gleichungssystem (3.128) eine eindeutige Lösung hat, besitzt es keine Lösung mit $u_{0_i}^* = 0$. Folglich besitzt auch das Gleichungssystem

⁶⁴Zur Lösbarkeit linearer Gleichungssysteme vgl. Anhang D dieser Arbeit.

$$\begin{aligned}\mathbf{z}^T \mathbf{F} + \bar{\mathbf{u}}^T \mathbf{B}^0 &= \mathbf{0} \\ \bar{\mathbf{u}}^T \mathbf{1} &= 1\end{aligned}\tag{3.129}$$

keine Lösung. Gemäss dem Lemma von Farkas⁶⁵ hat aber das System

$$\begin{aligned}\mathbf{F}\mathbf{y} &= \mathbf{0} \\ \mathbf{B}^0 \mathbf{y} &= \mathbf{1}\end{aligned}\tag{3.130}$$

mit dem $(n \times 1)$ -Spaltenvektor \mathbf{y} eine Lösung \mathbf{y}^0 . Ist \mathbf{a}_i die aus der Matrix \mathbf{A}^0 gestrichene Zeile, dann gilt für diese Lösung $\mathbf{a}_i \mathbf{y}^0 > 1$. Diese Aussage können wir wie folgt beweisen. Da $(\mathbf{z}^*, \mathbf{u}_0^*)$ die Lösung des Gleichungssystems (3.128) ist, gilt

$$\begin{aligned}\mathbf{z}^{*T} \mathbf{F} + \mathbf{u}_0^{*T} \mathbf{A}^0 &= \mathbf{0} \\ \Rightarrow \mathbf{z}^{*T} \mathbf{F} + \bar{\mathbf{u}}_0^{*T} \mathbf{B}^0 + u_{0_i}^* \mathbf{a}_i &= \mathbf{0} \\ \Rightarrow u_{0_i}^* \mathbf{a}_i &= -(\mathbf{z}^{*T} \mathbf{F} + \bar{\mathbf{u}}_0^{*T} \mathbf{B}^0).\end{aligned}$$

Multiplizieren wir die letzte Gleichung von rechts mit \mathbf{y}^0 , so erhalten wir

$$\begin{aligned}u_{0_i}^* \mathbf{a}_i \mathbf{y}^0 &= -(\mathbf{z}^{*T} \mathbf{F} + \bar{\mathbf{u}}_0^{*T} \mathbf{B}^0) \mathbf{y}^0 \\ &= -(\mathbf{z}^{*T} \underbrace{\mathbf{F} \mathbf{y}^0}_{=0} + \bar{\mathbf{u}}_0^{*T} \underbrace{\mathbf{B}^0 \mathbf{y}^0}_{=1}) \\ &= -\bar{\mathbf{u}}_0^{*T} \mathbf{1}.\end{aligned}\tag{3.131}$$

Da $\mathbf{u}_0^{*T} \mathbf{1} = \bar{\mathbf{u}}_0^{*T} \mathbf{1} + u_{0_i}^* = 1$ ist, gilt $\bar{\mathbf{u}}_0^{*T} \mathbf{1} = 1 - u_{0_i}^*$. Setzen wir diese Gleichung in (3.131) ein, so gelangen wir zu

$$\begin{aligned}u_{0_i}^* \mathbf{a}_i \mathbf{y}^0 &= u_{0_i}^* - 1 \\ \Rightarrow \underbrace{u_{0_i}^*}_{< 0} (1 - \mathbf{a}_i \mathbf{y}^0) &= 1 \\ \Rightarrow \mathbf{a}_i \mathbf{y}^0 &> 1.\end{aligned}$$

3. Fall Das Gleichungssystem (3.128) hat keine Lösung. Gemäss dem Lemma von Farkas besitzt dann das Gleichungssystem

⁶⁵Zum Lemma von Farkas siehe Anhang D, S. 308 f.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

$$\begin{aligned}\mathbf{F}\mathbf{y} &= \mathbf{0} \\ \mathbf{A}^0\mathbf{y} &= \mathbf{1}\end{aligned}\tag{3.132}$$

eine Lösung. In diesem Fall definiere man $\mathbf{B}^0 := \mathbf{A}^0$. Die Lösung des Gleichungssystems (3.132) sei \mathbf{y}^0 .

Im ersten Fall ist eine optimale Lösung des Optimierungsproblems gefunden. In den anderen Fällen haben wir einen Vektor \mathbf{y}^0 ermittelt, so dass $\mathbf{F}\mathbf{y}^0 = \mathbf{0}$ und $\mathbf{A}^0\mathbf{y}^0 \geq \mathbf{1}$ gelten. Falls $\mathbf{A}\mathbf{y}^0 \geq \mathbf{1}$ vorliegt, dann ist der Vektor $(\mathbf{x}^0 + \lambda\mathbf{y}^0, t^0 - \lambda)$ für alle $\lambda \geq 0$ zulässig. Dies bedeutet, dass die Zielfunktion des linearen Programms unbeschränkt ist (vgl. dazu Derks/Kuipers [32], S. 7). Falls für einige Zeilen \mathbf{a} der Matrix \mathbf{A} gilt, dass $\mathbf{a}\mathbf{y}^0 < 1$, berechnen Derks/Kuipers [32] die maximale Schrittweite λ^0 , für die $(\mathbf{x}^0 + \lambda^0\mathbf{y}^0, t^0 - \lambda^0)$ noch zulässig ist, durch

$$\lambda^0 := \min \left\{ \frac{\mathbf{a}_j\mathbf{x}^0 + t^0 - b_j}{1 - \mathbf{a}_j\mathbf{y}^0} \mid \text{Für alle Zeilen } \mathbf{a}_j \text{ von } \mathbf{A} \text{ mit } \mathbf{a}_j\mathbf{y}^0 < 1 \right\}.\tag{3.133}$$

Darin sei \hat{j} der kleinste zu λ^0 gehörende Index j . Man berechne die neuen Werte:

$$(\mathbf{x}^1, t^1) = (\mathbf{x}^0 + \lambda^0\mathbf{y}^0, t^0 - \lambda^0).$$

Man bilde die neue Matrix \mathbf{A}^1 , indem man die Zeile \mathbf{a}_j als zusätzliche letzte Zeile der Matrix \mathbf{B}^0 hinzufügt.⁶⁶ Hierbei ist zu beachten, dass die Zeilen der neuen Matrix

$$\begin{pmatrix} \mathbf{F} & \mathbf{0} \\ \mathbf{A}^1 & \mathbf{1} \end{pmatrix}$$

linear unabhängig sind. Man wiederhole nun Schritt 3 mit $(\mathbf{A}^1, \mathbf{x}^1)$ usw.

Wir geben nun an, mit welchem Aufwand die oben vorgestellte Methode zur Berechnung einer Lösung von linearen Programmen vom Typ \mathcal{P} (vgl. S. 152) verbunden ist. Um ein lineares Programm \mathcal{P} mit der oben beschriebenen Simplex-Methode zu lösen, berechnen wir in jeder Iteration

- eine Lösung des Gleichungssystems $\mathbf{zF} + \mathbf{u}^i\mathbf{A}^i = \mathbf{0}$, $\mathbf{u}^i\mathbf{1} = \mathbf{1}$,

⁶⁶Diese Methode der Pivotoperation ist in der Literatur als Bland-Regel bekannt.

- gegebenfalls eine Lösung des Gleichungssystems $\mathbf{F}\mathbf{y} = \mathbf{0}$, $\mathbf{B}^i\mathbf{y} = \mathbf{1}$,
- gegebenfalls die optimale Schrittweite λ^i .

Da in beiden linearen Gleichungssystemen die Anzahl der Unbekannten n ist, kann jedes der Gleichungssysteme mit $\mathcal{O}(n^3)$ Operationen gelöst werden.⁶⁷ Die Berechnung von λ^i gemäss (3.133) kann mit $\mathcal{O}(nm)$ Operationen erfolgen. Für die zur Berechnung des Prenukleolus eines kooperativen Spiels mit n Spielern zu lösenden linearen Programme gilt $m = \mathcal{O}(2^n)$, weil die Matrix \mathbf{A} höchstens $2^n - 1$ Zeilen besitzt. Folglich ist $\mathcal{O}(n2^n)$ die Komplexität einer Pivotoperation der obigen Simplex-Methode. Wie wir auf S. 160 f. erläutern werden, zeigen Derks/Kuipers, dass durch eine spezielle Implementation dieser Simplex-Methode die Komplexität einer Pivotoperation um den Faktor n von $\mathcal{O}(n2^n)$ auf $\mathcal{O}(2^n)$ reduziert werden kann.

Zur Veranschaulichung der oben vorgestellten Simplex-Methode führen wir nun anhand des folgenden Beispiels eine Iteration durch:

Beispiel 3.3.1

Das zu lösende lineare Programm besitze die Form

$$\mathcal{P} := \min_{t, \mathbf{x}} \{t \mid \mathbf{F}\mathbf{x} = \mathbf{c}, \mathbf{A}\mathbf{x} + \mathbf{1}t \geq \mathbf{b}\}.$$

Gegeben seien

$$\mathbf{F} = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}, \quad \mathbf{c} = (9), \quad \mathbf{x} = \begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix}^T$$

und

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 5 \\ 5 \end{pmatrix}.$$

⁶⁷Für den Beweis vgl. z. B. Schwarz [118], S. 19., oder Engeln-Müllges/Reutter [37], S. 74 ff.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Schritt 1

- $\mathbf{x}^0 = (3 \ 3 \ 3)^T$ ist eine zulässige Lösung von $\mathbf{F}\mathbf{x} = \mathbf{c}$,
- $t^0 := \max \{b_i - \mathbf{a}_i\mathbf{x}^0 \mid i = 1, \dots, 6\} = -1$,
- Somit ist $(\mathbf{x}^0, t^0) = \left((3 \ 3 \ 3)^T, -1 \right)$ eine zulässige Lösung des linearen Programms.

Schritt 2

Für $i = 5$ und $i = 6$ ist die Gleichung $\mathbf{a}_i\mathbf{x}^0 + t^0 = b_i$ erfüllt. Deswegen besteht die Matrix \mathbf{A}^0 aus der fünften und der sechsten Zeile der Matrix \mathbf{A} :

$$\mathbf{A}^0 = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}.$$

Schritt 3

Wir stellen das Gleichungssystem (3.128) (vgl. S. 154) auf:

$$\begin{aligned} z \begin{pmatrix} 1 & 1 & 1 \end{pmatrix} + (u_1 \ u_2) \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} &= \begin{pmatrix} 0 & 0 & 0 \end{pmatrix} \\ u_1 + u_2 &= 1. \end{aligned}$$

Daraus ergibt sich:

$$z + u_1 + u_2 = 0$$

$$z + u_2 = 0$$

$$z + u_1 = 0$$

$$u_1 + u_2 = 1.$$

Setzen wir die letzte Gleichung in die erste ein, dann erhalten wir $z = -1$. Daraus folgt, dass $u_1 = -1$ und $u_2 = -1$ gelten muss. Diese beiden Ergebnisse widersprechen aber der letzten Gleichung. Also besitzt das obige Gleichungssystem keine Lösung. Da der 3. Fall vorliegt, definieren wir $\mathbf{B}^0 := \mathbf{A}^0$. Gemäss dem Lemma von Farkas besitzt das Gleichungssystem

$$\begin{aligned}
\mathbf{F}\mathbf{y} = \mathbf{0} &\Rightarrow \begin{pmatrix} 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = 0 \\
\mathbf{A}^0\mathbf{y} = \mathbf{1} &\Rightarrow \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}
\end{aligned} \tag{3.134}$$

eine Lösung. Führen wir die Matrix- und Vektormultiplikationen durch, so erhalten wir

$$\begin{aligned}
y_1 + y_2 + y_3 &= 0 \\
y_1 + y_3 &= 1 \\
y_1 + y_2 &= 1.
\end{aligned}$$

Als Lösung gewinnen wir

$$\mathbf{y}^0 := \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} 2 \\ -1 \\ -1 \end{pmatrix}.$$

Um zu überprüfen, ob die Zielfunktion des linearen Programms beschränkt ist, berechnen wir $\mathbf{A}\mathbf{y}^0$:

$$\mathbf{A}\mathbf{y}^0 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 2 \\ -1 \\ -1 \end{pmatrix} = \begin{pmatrix} -1 \\ -1 \\ -2 \\ 2 \\ 1 \\ 1 \end{pmatrix}.$$

Da $\mathbf{A}\mathbf{y}^0 \not\leq \mathbf{1}$ gilt, ist die Zielfunktion beschränkt. Zugleich sehen wir, dass für die Berechnung der optimalen Schrittweite λ die Zeilen $j = 1, 2, 3$ in Frage kommen. Somit ergibt sich (vgl. Gleichung (3.133), S. 156):

$$\lambda^0 := \min \left\{ 1, 1, \frac{4}{3} \right\} = 1.$$

Da der kleinste zu $\lambda^0 = 1$ gehörende Index j eins ist, gilt $\hat{j} = 1$. Die neue Ecke des zulässigen Bereichs mit einem kleineren Zielfunktionswert ist

$$(\mathbf{x}^1, t^1) = (\mathbf{x}^0 + \lambda^0 \mathbf{y}^0, t^0 - \lambda^0) = \left(\begin{pmatrix} 5 & 2 & 2 \end{pmatrix}^T, -2 \right).$$

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Wir bilden die Matrix \mathbf{A}^1 , indem wir die Zeile $\hat{j} = 1$ der Matrix \mathbf{A} der Matrix \mathbf{B}^0 hinzufügen, und gehen zu Schritt 3. Wir führen die nächste Iteration mit (\mathbf{x}^1, t^1) durch. Diesen Ablauf wiederholen wir, bis wir die Problemlösung gefunden oder andernfalls festgestellt haben, dass die Zielfunktion des Problems unbeschränkt ist.

Im Folgenden beschreiben wir nach Derks/Kuipers, wie die Komplexität einer Pivotoperation reduziert werden kann. Es sei ein lineares Programm $\mathcal{Q}(\mathbf{F}, \mathbf{d}, \mathbf{b})$ der Form (3.122) zur Berechnung des Prenukleolus eines kooperativen Spiels gegeben, wobei hier \mathbf{F} eine $(p \times n)$ -Matrix mit dem Rang p , \mathbf{d} ein $(n \times 1)$ -Spaltenvektor und \mathbf{b} ein dimensionsverträglicher Spaltenvektor ist. Da das lineare Programm $\mathcal{Q}(\mathbf{F}, \mathbf{d}, \mathbf{b})$ von der Form \mathcal{P} ist, können wir die oben beschriebene Simplex-Methode auf $\mathcal{Q}(\mathbf{F}, \mathbf{d}, \mathbf{b})$ anwenden. Wie wir bereits auf S. 157 dargelegt haben, erfordert jede Pivotisierung $\mathcal{O}(n2^n)$ elementare Operationen.⁶⁸ Um die Komplexität einer Pivotoperation zu reduzieren, nutzen Derks/Kuipers aus, dass die Matrix \mathbf{A} mit Hilfe der Matrix \mathbf{F} bestimmt werden kann. Die Matrix \mathbf{A} enthält nämlich alle möglichen $(0, 1)$ -Zeilen⁶⁹ der Dimension n , die nicht als Linearkombinationen der Zeilen von \mathbf{F} dargestellt werden können. Wir ordnen zunächst alle $(0, 1)$ -Vektoren, die n Komponenten besitzen, in lexikographisch steigender Reihenfolge, wobei der Nullvektor ausser Acht gelassen wird. Es gibt insgesamt $2^n - 1$ solche Vektoren. Der Vektor i sei \mathbf{a}_i . Ferner sei \mathbf{f} ein $1 \times (2^n - 1)$ -Zeilenvektor mit folgender Eigenschaft:

$$f_i := \begin{cases} 1 & : \text{ falls } \mathbf{a}_i \text{ eine Zeile von } \mathbf{A} \text{ ist,} \\ 0 & : \text{ sonst.} \end{cases}$$

Falls der Vektor \mathbf{f} bekannt ist,⁷⁰ können wir λ durch

$$\lambda := \min \left\{ \frac{\mathbf{a}_i \mathbf{x} + t - b_i}{1 - \mathbf{a}_i \mathbf{y}} \mid f_i = 1 \text{ und } \mathbf{a}_i \mathbf{y}^0 < 1 \right\}$$

berechnen, wobei (\mathbf{x}, t) die aktuell zulässige Lösung ist und \mathbf{y} eine Lösung des Gleichungssystems (3.130) bzw. (3.132). Wir zeigen nun, wie der Wert λ mit Hilfe des Vektors \mathbf{f} bestimmt werden kann, ohne dass hierfür die Matrix \mathbf{A} bekannt sein muss.

⁶⁸Vgl. Derks/Kuipers [32], S. 8.

⁶⁹Die Schreibweise „ $(0, 1)$ -Zeile“ bzw. „ $(0, 1)$ -Vektor“ besagt, dass die erwähnte Zeile bzw. der erwähnte Vektor aus Nullen und Einsen besteht.

⁷⁰Später werden wir zeigen, wie der Vektor \mathbf{f} bestimmt werden kann.

3.3 Methoden der dritten Generation

Es seien \mathbf{a} und \mathbf{g} zwei unmittelbar aufeinander folgende $(0, 1)$ -Zeilen der Dimension n , die in lexikographisch steigender Reihenfolge geordnet sind. Die Position der Null, die in \mathbf{a} am weitesten rechts steht, sei j . Des Weiteren sei \mathbf{u}^j ein $(1 \times n)$ -Zeilenvektor mit der Eigenschaft, dass die ersten $j - 1$ Komponenten Nullen, die Komponente j Eins und die restlichen Komponenten -1 sind. In diesem Fall gilt $\mathbf{g} = \mathbf{a} + \mathbf{u}^j$. Wir können mit Hilfe dieser Eigenschaft der Vektoren \mathbf{g} , \mathbf{a} und \mathbf{u}^j , beginnend mit dem lexikographisch kleinsten $(0, 1)$ -Vektor der Dimension n

$$\mathbf{e}^n := (0 \ 0 \ \dots \ 0 \ 1),$$

alle anderen $(0, 1)$ -Vektoren der Dimension n bestimmen. Gegeben sei z. B. $\mathbf{a} = (1 \ 0 \ 1)$. In diesem Fall sind $j = 2$ und $\mathbf{u}^j := (0 \ 1 \ -1)$. Somit ist $\mathbf{g} = \mathbf{a} + \mathbf{u}^j = (1 \ 1 \ 0)$. Der Vektor \mathbf{g} ist nach dem Vektor $\mathbf{a} = (1 \ 0 \ 1)$ der nächste lexikographisch grössere Vektor. Durch Verwendung dieser Eigenschaft der Vektoren \mathbf{g} , \mathbf{a} und \mathbf{u}^j schlagen Derks/Kuipers die folgenden Schritte zur Berechnung von λ vor:

Initialisierung

- $\mathbf{a} := \mathbf{e}^n$,
- $\lambda := \begin{cases} \frac{\mathbf{ax} + t - b_1}{1 - \mathbf{ay}} & : \text{ falls } f_1 = 1 \text{ und } \mathbf{ay} < 1 \\ \infty & : \text{ sonst,} \end{cases}$
- $i := 2$.

Schritt 1

Man bestimme die Position der Null, die in \mathbf{a} am weitesten rechts steht. Diese Position sei j .

Schritt 2

Man bestimme den nächsten, lexikographisch grösseren Vektor durch

$$\mathbf{a} := \mathbf{a} + \mathbf{u}^j$$

und aktualisiere gemäss

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

$$\mathbf{ax} := \mathbf{ax} + \mathbf{u}^j \mathbf{x},$$

$$\mathbf{ay} := \mathbf{ay} + \mathbf{u}^j \mathbf{y}.$$

Falls $\mathbf{ay} < 1$ und $f_i = 1$ ist, setze man $\lambda := \min \left\{ \lambda, \frac{\mathbf{ax} + t - b_i}{1 - \mathbf{ay}} \right\}$.

Schritt 3

Man erhöhe i um eins, $i := i + 1$. Falls $i \leq 2^n$ gilt, gehe man zu Schritt 1, andernfalls hat man den Wert von λ gefunden.

Die Bestimmung von λ mit Hilfe des obigen Algorithmus erfordert die Durchführung von $2^n - 1$ Iterationen. Innerhalb jeder Iteration beträgt die Anzahl der elementaren Operationen $\mathcal{O}(n)$, weil sowohl im Schritt 1 als auch im Schritt 2 die Anzahl der elementaren Operationen auf $\mathcal{O}(n)$ beschränkt ist. Somit hat eine Pivotoperation die Komplexität $\mathcal{O}(n2^n)$. Diese Komplexität können wir jedoch reduzieren. Die Bestimmung von j im Schritt 1 erfolgt in $n - (j - 1)$ elementaren Operationen.⁷¹ Da die ersten $j - 1$ Einträge des Vektors \mathbf{u}^j per Definition Nullen sind, lautet die Anzahl elementarer Operationen für die Berechnung von \mathbf{a} , \mathbf{ax} , und \mathbf{ay} im zweiten Schritt in jeder Iteration $n - (j - 1)$. Zudem entspricht 2^{j-1} der Anzahl der $(0, 1)$ -Vektoren, deren Komponente j die am weitesten rechts stehende Null ist. Die Anzahl der möglichen Positionen der am weitesten rechts stehenden Null ist n , d. h. $j = 1, 2, \dots, n$. Folglich ist die Komplexität der $2^n - 1$ Iterationen

$$\begin{aligned} \mathcal{O} \left(\sum_{j=1}^n (n - j + 1) 2^{j-1} \right) &= \mathcal{O} \left(\frac{(1+n)}{2} \sum_{j=1}^n 2^j - \frac{1}{2} \sum_{j=1}^n j 2^j \right) \\ &= \mathcal{O} \left(\frac{(1+n)}{2} 2(2^n - 1) - \frac{1}{2} [2^{n+1}(n-1) + 2] \right) \\ &= \mathcal{O}(2^{n+1} - n - 2) \\ &= \mathcal{O}(2^n). \end{aligned}$$

Die Komplexität einer Pivotoperation der oben beschriebenen Simplex-Methode ist also $\mathcal{O}(2^n)$.

⁷¹Beispielsweise sei $\mathbf{a} = (1 \ 0 \ 1 \ 1 \ 0 \ 1 \ 1 \ 1)$. Wir beginnen mit der letzten Komponente von \mathbf{a} und überprüfen rückwärts die Komponenten, bis wir 0 erreicht haben. Wir stellen in vier Schritten fest, dass $j = 5$ ist.

3.3 Methoden der dritten Generation

Nun wollen wir zeigen, wie wir den Vektor \mathbf{f} bestimmen können. Zunächst transformieren wir die Matrix \mathbf{F} , so dass der erste nicht-negative Eintrag in jeder Zeile 1 ist. Hierzu sei \mathbf{H} eine $(n \times n)$ -Matrix und \mathbf{h} ein $(n \times 1)$ -Spaltenvektor mit folgenden Eigenschaften. Diejenige Zeile der Matrix \mathbf{F} , deren erster nicht-negativer Eintrag an der Position i erscheint, wird zur Zeile i der Matrix \mathbf{H} . Zudem setzen wir $h_i := 0$. Jede Zeile i von \mathbf{H} , die nach der Anwendung dieser Regel nicht besetzt ist, wird zu einer Einheitszeile, deren Komponente i Eins ist. Ferner gilt $h_i := 1$. Falls zum Beispiel

$$\mathbf{F} = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}$$

ist, dann sind die Matrix \mathbf{H} und der Vektor \mathbf{h} wie folgt definiert:

$$\mathbf{H} := \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \text{ und } \mathbf{h} := \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}.$$

Somit ist die Matrix \mathbf{H} eine obere Dreiecksmatrix, deren Diagonalelemente Einsen sind. Folglich hat das Gleichungssystem

$$\boldsymbol{\alpha} \mathbf{H} = \mathbf{a}$$

für jeden Zeilenvektor \mathbf{a} eine eindeutige Lösung, wobei hier $\boldsymbol{\alpha}$ ein $(1 \times n)$ -Zeilenvektor ist. Definieren wir die Matrix \mathbf{H} und den Vektor \mathbf{h} wie oben angegeben, dann ist ein $(0, 1)$ -Zeilenvektor \mathbf{a} dann und nur dann keine Zeile der Matrix \mathbf{A} , wenn

$$\alpha_i h_i = 0 \quad \text{für alle } i = 1, 2, \dots, n$$

gilt. Um diese Aussage zu beweisen, halten wir zunächst fest, dass \mathbf{a} als Linearkombination von Zeilen der Matrix \mathbf{H} abgebildet werden kann, da das Gleichungssystem $\boldsymbol{\alpha} \mathbf{H} = \mathbf{a}$ eine Lösung besitzt.⁷² Nun nehmen wir an, dass $\alpha_i h_i = 0$ für alle $i = 1, 2, \dots, n$. Aus den Definitionen für \mathbf{H} und \mathbf{h} folgt, dass keine Zeile i von \mathbf{H} einer Zeile von \mathbf{F} entspricht, falls die Komponente i des Vektors \mathbf{h} Eins ist. Für jede solche

⁷²Für eine ausführliche Beschreibung dieser Eigenschaft von linearen Gleichungssystemen vgl. Padberg/Kütting [99], S. 55 f.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Zeile i bedeutet folglich die Gleichung $\alpha_i h_i = 0$, dass $\alpha_i = 0$ gilt. Das heisst wiederum, dass der Zeilenvektor \mathbf{a} auch als Linearkombination der Zeilen von \mathbf{F} dargestellt werden kann. Demzufolge kann gemäss der Definition der Matrix \mathbf{A} der Zeilenvektor \mathbf{a} keine Zeile der Matrix \mathbf{A} sein. Nehmen wir jetzt umgekehrt an, dass \mathbf{a} keine Zeile der Matrix \mathbf{A} ist. In diesem Fall kann \mathbf{a} durch Linearkombination der Zeilen von \mathbf{F} repräsentiert werden. Da für den Vektor $\boldsymbol{\alpha}$ der Zeilenvektor \mathbf{a} gleichzeitig als Linearkombination der Zeilen von \mathbf{H} dargestellt werden kann, muss $\alpha_i h_i = 0$ für alle $i = 1, 2, \dots, n$ gelten.

Zur Illustration überprüfen wir für $\mathbf{F} = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}$, ob ein $n = 3$ dimensionaler $(0, 1)$ -Zeilenvektor $\mathbf{a} = \begin{pmatrix} a_1 & a_2 & a_3 \end{pmatrix}$ eine Zeile der Matrix \mathbf{A} ist oder nicht. Soeben haben wir für diesen Fall die Matrix \mathbf{H} und den Vektor \mathbf{h} bestimmt. Wir lösen nun das Gleichungssystem $\boldsymbol{\alpha}\mathbf{H} = \mathbf{a}$

$$\begin{pmatrix} \alpha_1 & \alpha_2 & \alpha_3 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 \end{pmatrix}.$$

Durch Multiplikation auf der linken Seite erhalten wir

$$\begin{pmatrix} \alpha_1 & \alpha_1 + \alpha_2 & \alpha_1 + \alpha_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 \end{pmatrix}. \quad (3.135)$$

Um zu überprüfen, ob z. B. der Vektor $\mathbf{a} = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix}$ eine Zeile der Matrix \mathbf{A} ist, setzen wir ihn in (3.135) ein und lösen das Gleichungssystem

$$\begin{aligned} \alpha_1 &= 0 \\ \alpha_1 + \alpha_2 &= 1 \\ \alpha_1 + \alpha_3 &= 0. \end{aligned}$$

Als Lösung erhalten wir $\alpha_1 = 0$, $\alpha_2 = 1$ und $\alpha_3 = 0$. Da $\alpha_2 h_2 = 1 \neq 0$ ist, ist der Vektor $\mathbf{a} = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix}$ eine Zeile der Matrix \mathbf{A} und folglich ist die dem Vektor \mathbf{a} entsprechende Komponente des Vektors \mathbf{f} eine Eins. Auf diese Weise können nun alle n -dimensionalen $(0, 1)$ -Vektoren auf ihre Zugehörigkeit zur Matrix \mathbf{A} überprüft werden, um den Vektor \mathbf{f} zu bestimmen. Derks/Kuipers schlagen für diese Überprüfung die gleiche Vorgehensweise vor, die wir auf S. 161 bei der Berechnung von λ

erörtert haben. Sie beginnen nämlich wiederum mit dem lexikographisch kleinsten $(0, 1)$ -Vektor $\mathbf{e}^n := (0 \ 0 \ \dots \ 0 \ 1)$ und legen den jeweils lexikographisch grösseren Vektor mit Hilfe des auf S. 161 definierten Vektors \mathbf{u}^j fest. Anschliessend prüfen sie, ob dieser Vektor eine Zeile der Matrix \mathbf{A} ist, ob also die entsprechende Komponente des Vektors \mathbf{f} eine Eins ist.

Zum Abschluss fassen wir die Eigenschaften des Algorithmus von Derks/Kuipers kurz zusammen. Das Verfahren von Derks/Kuipers erfordert zur Berechnung des Prenukleolus eines kooperativen Spiels $\Gamma = (N, v)$ mit n Spielern die Lösung von höchstens $n - 1$ linearen Programmen. Für die Berechnung der Lösungen dieser Programme schlagen die Autoren eine spezielle Implementation der Simplex-Methode vor, um den Rechenaufwand einer Pivotoperation von $\mathcal{O}(n2^n)$ auf $\mathcal{O}(2^n)$ zu reduzieren. Ferner erfordert das Verfahren von Derks/Kuipers zur Berechnung des Prenukleolus nur die $(n \times n)$ -Matrix \mathbf{F} , während die jeweiligen linearen Programme, die gemäss den in den vorangegangenen Abschnitten vorgestellten Verfahren der dritten Generation gelöst werden, jede mögliche Koalition $S \subseteq N$ für die Problemlösung heranziehen. Folglich nimmt die Matrix \mathbf{F} bei den praktischen Berechnungen des Prenukleolus am Computer relativ weniger Speicherplatz in Anspruch. Dieser Tatbestand erlaubt es in der Praxis, den Prenukleolus kooperativer Spiele mit einer relativ grossen Anzahl an Spielern zu ermitteln. Beispielsweise haben Derks/Kuipers [32] ihren Algorithmus in der Programmiersprache C programmiert. Mit Hilfe ihrer Implementation kann der Prenukleolus der Spiele mit bis zu 32 Spielern berechnet werden.⁷³

Die im Kapitel 3 vorgestellten Verfahren zeigen auf, dass die Berechnung des Nukleolus im Allgemeinen eine sehr aufwändige Aufgabe ist. Deshalb wird in der Literatur untersucht, ob es möglich ist, den Nukleolus in besonderen Fällen mit relativ geringem Rechenaufwand zu berechnen. Beispielsweise beweist Wolff [149], dass der Prenukleolus eines Null-normierten kooperativen Spiels $\Gamma = (N, v)$ mit n Spielern mit der Gleichverteilung des Wertes $v(N)$ der grossen Koalition zwischen den Spie-

⁷³Eine 32-Bit-Programmier-Umgebung erlaubt es, maximal 2^{32} Einträge zu speichern. Somit kann in einer solchen Umgebung der Prenukleolus von Spielen mit höchstens 32 Spielern berechnet werden, weil für 32 Spieler insgesamt 2^{32} Koalitionswerte gespeichert werden müssen.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

lern $i \in N$ übereinstimmt,⁷⁴ falls die Werte aller Mehrspielerkoalitionen im Vergleich zu $v(N)$ im Sinne der folgenden Ungleichung „genügend“ klein sind:

$$\frac{v(S)}{|S| - 1} \leq \frac{v(N)}{n} \quad \text{für alle } S \subset N, |S| \geq 2. \quad (3.136)$$

Wolff [149] zeigt, dass die Ungleichung (3.136) mit geringem Aufwand abgeprüft werden kann. Beispielsweise legt er dar, dass die Prüfung der Ungleichung (3.136) für Spiele bis und mit 25 Spielern auf einem Rechner (Pentium 4, 2.4 GHz) nicht mehr als eine Sekunde erfordert. Folglich kann für die Bestimmung des Prenukleolus kooperativer Spiele zunächst untersucht werden, ob die Ungleichung (3.136) erfüllt ist. Gilt die Ungleichung (3.136), erfolgt die Berechnung des Prenukleolus ohne grossen Rechenaufwand. In diesem Fall ordnet der Prenukleolus jedem Spieler des Spiels den Wert $v(N)/n$ zu.

⁷⁴Wie wir auf S. 20 erläutert haben, kann die charakteristische Funktion kooperativer Spiele $(0-1)$ -normiert werden. Deswegen stellt die Voraussetzung, dass das jeweilige Spiel Null-normiert sein muss, keine Beschränkung der Allgemeinheit dar.

3.4 Zusammenfassung

In diesem Kapitel haben wir uns mit den unterschiedlichen in der Literatur entwickelten Verfahren zur Berechnung des Nukleolus kooperativer Spiele mit n Spielern befasst. Wir haben diese Verfahren in drei Gruppen unterteilt.

Zur ersten Gruppe zählen die Verfahren von Kohlberg [65] und von Owen [96], die den Nukleolus kooperativer Spiele mit Hilfe eines einzigen linearen Programms berechnen. Diese Verfahren haben wir Methoden der ersten Generation genannt. Das Verfahren von Kohlberg [65] ist das erste Verfahren, das darlegt, dass der Nukleolus im Wege der Lösung lediglich eines linearen Programms berechnet werden kann. Es besitzt aber einerseits eine hohe Anzahl an Nebenbedingungen und andererseits sind in den Nebenbedingungen die Koeffizienten der jeweiligen Variablen sehr gross. Das Verfahren von Owen [96] geht von dem von Kohlberg [65] angegebenen linearen Programm aus und zeigt, dass durch eine äquivalente Umformung dieses Programms die Anzahl der Nebenbedingungen verringert werden kann. Demgegenüber beinhaltet aber das von Owen [96] zur Berechnung des Nukleolus vorgeschlagene lineare Programm eine im Vergleich zu dem von Kohlberg vorgeschlagenen linearen Programm grössere Anzahl an Variablen. Zudem bleibt im von Owen vorgeschlagenen linearen Programm das Problem der hohen Koeffizienten der Variablen bestehen. Wegen dieser Nachteile stellen die Verfahren von Kohlberg [65] und von Owen [96] keine in der Praxis geeigneten Verfahren zur Berechnung des Nukleolus kooperativer Spiele dar. Sie sind aber insofern von Bedeutung, als sie theoretisch aufzeigen, wie der Nukleolus im Wege der Lösung lediglich eines einzigen linearen Programms berechnet werden kann. Die Tabelle 3.3 am Schluss dieses Abschnittes ermöglicht einen direkten Vergleich der Komplexität der beiden genannten Verfahren.

Zur zweiten Gruppe gehören die Verfahren von Bruyneel [22] und von Dragan [35], die den Nukleolus kooperativer Spiele mit Hilfe minimal balancierter Koalitionssysteme berechnen. Diese Verfahren haben wir Methoden der zweiten Generation genannt. Wir haben zunächst das Verfahren von Bruyneel [22] vorgeführt. Dieses Verfahren zieht zur Berechnung des Nukleolus alle möglichen minimal balancierten Koaliti-

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

onssysteme des Spiels und deren Gewichtsvektoren heran. Da aber die Anzahl der minimal balancierten Koalitionssysteme kooperativer Spiele mit wachsender Spielerzahl exponentiell steigt, eignet sich das Verfahren von Bruyneel [22] in der Praxis nur zur Berechnung des Nukleolus kooperativer Spiele mit bis zu vier Spielern. Die Bedeutung des Verfahrens liegt jedoch darin, dass es keine Lösung linearer Programme erfordert und somit eine neue Möglichkeit zur Berechnung des Nukleolus eröffnet. Dies gestattet Dragan [35], ein Verfahren zur Berechnung des Prenukleolus kooperativer Spiele zu entwickeln, das auf dem Konzept der minimal balancierten Koalitionssysteme beruht. Wir haben zunächst die direkte Form des Verfahrens von Dragan [35] vorgestellt, welches ebenfalls alle minimal balancierten Koalitionssysteme sowie deren Gewichtsvektoren heranzieht. Anschliessend haben wir die von Dragan vorgenommene Modifikation des Verfahrens beschrieben. Das modifizierte Verfahren erfordert nicht mehr die Berücksichtigung aller minimal balancierter Koalitionssysteme, sondern ermittelt im Verlaufe der Berechnungen im Wege der Lösung eines linearen Programms in jeder Iteration jeweils ein minimal balanciertes Koalitionssystem, das für die Ermittlung des Prenukleolus des Spiels benötigt wird. Somit kann das Verfahren von Dragan [35] im Gegensatz zu dem Verfahren von Bruyneel [22] in der Praxis auf Spiele mit hoher Spielerzahl angewendet werden.

Eine dritte Gruppe von Verfahren, die dadurch gekennzeichnet sind, dass sie zur Berechnung des Nukleolus die Lösung einer Folge linearer Programme erfordern, haben wir als Methoden der dritten Generation bezeichnet. Den Verfahren dieser Generation liegt der Algorithmus von Maschler et al. [82] zu Grunde. Wir haben zunächst das von Maschler et al. [82] in die Literatur eingeführte Konzept des lexikographischen Zentrums vorgestellt und gezeigt, dass dieses Konzept mit dem Nukleolus übereinstimmt. Das lexikographische Zentrum jedes kooperativen Spiels kann durch die Ermittlung einer Lösung einer Folge linearer Programme bestimmt werden. Die Berechnung des lexikographischen Zentrums jedes kooperativen Spiels mit Hilfe des Verfahrens von Maschler et al. erfordert die Lösung von höchstens $2^n - 2$ linearen Programmen, wobei jedes lineare Programm $n + 1$ Variablen und $2^n - 1$ Nebenbedingungen enthält. Um zu überprüfen, ob der minimale Überschuss jeder

Koalition erreicht ist, werden zudem insgesamt maximal $\mathcal{O}(4^n)$ lineare Programme gelöst (vgl. Tabelle 3.3). Ein weiteres Verfahren, das zu den Methoden der dritten Generation zählt, ist das Verfahren von Sankaran [113]. Im Abschnitt 3.3.2 haben wir dieses Verfahren vorgestellt. Das Verfahren von Sankaran hat gegenüber dem Verfahren von Maschler et al. den Vorteil, dass es im ungünstigsten Fall die Lösung von insgesamt $\mathcal{O}(2^n)$ linearen Programmen erfordert, um zu überprüfen, ob der minimale Überschuss jeder Koalition ermittelt ist. Zudem kombiniert das Verfahren von Sankaran die Bestimmung dieser Koalitionen mit der Berechnung des jeweiligen minimalen Koalitionsüberschusses, sodass im Unterschied zum Verfahren von Maschler et al. [82] nicht in jeder Iteration eine lineare Optimierungsaufgabe gelöst wird, sondern insgesamt nur eine einzige. Im Anschluss daran haben wir das Verfahren von Potters et al. [105] vorgestellt, das ebenfalls auf dem Verfahren von Maschler et al. [82] beruht. Im Verfahren von Potters et al. wird die Imputationsmenge des Spiels mittels eines spezifischen Gleichungssystems repräsentiert, welches die Nebenbedingungen des ersten zu lösenden linearen Programms bildet. Zudem wird nach jeder Iteration die Anzahl der Nebenbedingungen in dem jeweiligen linearen Programm reduziert, indem die Nebenbedingungen gestrichen werden, die zu jenen Koalitionen gehören, deren minimale Überschüsse ermittelt worden sind. Darüber hinaus erfolgt die Bestimmung der Koalitionen, deren minimale Überschüsse erreicht sind, direkt aus dem optimalen Simplex-Tableau der entsprechenden linearen Programme. Das Verfahren von Potters et al. [105] erfordert zur Berechnung des Nukleolus kooperativer Spiele die Lösung von maximal $n - 1$ linearen Programmen. Zwar besitzt das erste zu lösende lineare Programm $2^n - 1$ Nebenbedingungen und $2^n + n - 1$ Variablen, aber sowohl die Anzahl der Nebenbedingungen als auch die Anzahl der Variablen werden im Laufe der Berechnungen nach jeder Iteration reduziert (vgl. Tabelle 3.3). Ein weiterer Algorithmus, der zu den Methoden der dritten Generation zählt, ist das Verfahren von Derks/Kuipers [32]. Zur Berechnung des Prenukleolus kooperativer Spiele mit Hilfe des Verfahrens von Derks/Kuipers werden maximal $n - 1$ lineare Programme gelöst. Das Verfahren konzentriert sich jedoch darauf, eine geeignete Methode zur Lösung dieser linearen Optimierungsaufgaben zu entwickeln.

3 Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele

Derks/Kuipers nutzen zu diesem Zweck bestimmte Eigenschaften der Koeffizientenmatrix der Nebenbedingungen der linearen Programme aus. Bei der Aufstellung der linearen Programme von Derks/Kuipers werden nicht alle Nebenbedingungen explizit herangezogen. Stattdessen können sie durch eine $(n \times n)$ -Matrix \mathbf{F} bestimmt werden. Dadurch bietet sich die Möglichkeit, in der Praxis den Prenukleolus kooperativer Spiele mit hoher Spielerzahl zu berechnen.

Tabelle 3.3: Tabellarischer Überblick über die Komplexität der Algorithmen

	Verfahren	lin. Programme	Nebenbed.	Variablen
1. Generation	Kohlberg	1	$\mathcal{O}(2^n!)$	$\mathcal{O}(n)$
	Owen	1	$\mathcal{O}(4^n)$	$\mathcal{O}(2^n)$
2. Generation	Bruyneel	-	-	-
	Dragan	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(2^n)$
3. Generation	Maschler et al.	$\mathcal{O}(2^n), \mathcal{O}(4^n)$	$\mathcal{O}(2^n)$	$\mathcal{O}(2^n)$
	Sankaran	$\mathcal{O}(2^n)$	$\mathcal{O}(2^n)$	$\mathcal{O}(2^n)$
	Potters et al.	$\mathcal{O}(n)$	$\mathcal{O}(2^n)$	$\mathcal{O}(2^n)$
	Derks/Kuipers	$\mathcal{O}(n)$	$\mathcal{O}(n^2)$	$\mathcal{O}(n)$

Um die Komplexität der vorgestellten Verfahren zur Berechnung des Nukleolus kooperativer Spiele zusammenzufassen, geben wir in der Tabelle 3.3 einen Überblick über die beschriebenen Algorithmen. In der Tabelle sind die Anzahl der zu lösenden linearen Programme, der Nebenbedingungen und der Variablen für den ungünstigsten Fall angegeben, wobei n die Anzahl der Spieler ist. Hierbei merken wir an, dass bei der Interpretation der Angaben in der Tabelle 3.3 aus folgendem Grund Vorsicht geboten ist. Gemäss der Definition der Komplexität von Algorithmen (vgl. dazu Anhang F) wird nämlich der multiplikative Faktor des Ausdrucks in der Klammer von $\mathcal{O}(\cdot)$ vernachlässigt. Um verschiedene Algorithmen miteinander zu vergleichen, sollte man deshalb diese Algorithmen auch anhand praktischer Probleme erproben. Zu diesem Zweck haben wir in dieser Arbeit zwei Verfahren zur Berechnung kooperativer Spiele programmiert. Im übernächsten Kapitel stellen wir unsere Implementation

3.4 Zusammenfassung

der Verfahren vor und anschliessend vergleichen wir diese Verfahren an Hand von Berechnungen des Nukleolus verschiedener kooperativer Spiele miteinander.

4 Ausgewählte spezielle kooperative Spiele

4.1 Abstimmungsspiele: Das Verfahren von Wolsey

In der Literatur werden zahlreiche spezifische Typen kooperativer Spiele sowie zugehörige spezialisierte Verfahren zur Berechnung des jeweiligen Nukleolus diskutiert. Wegen der potentiellen Einsparung an Rechenzeit kann die Kenntnis des besonderen Spieltyps nützlich sein. In diesem Abschnitt stellen wir exemplarisch den Algorithmus von Wolsey [150] vor. Dieser Algorithmus berechnet den Nukleolus einer speziellen Form kooperativer Spiele, nämlich den Nukleolus von sog. einfachen kooperativen Spielen, die wir auf S. 32 kennen gelernt haben. In den nächsten Abschnitten des Kapitels geben wir einen Literaturhinweis über die wichtigsten speziellen kooperativen Spiele und die jeweiligen Algorithmen. Der Überblick soll dem Leser vor allem die eigene Vertiefung in die Literatur erleichtern. Daher verzichten wir insbesondere auf eine Behandlung der jeweiligen Algorithmen im Detail.

Wolsey [150] schlägt ein Verfahren zur Berechnung des Nukleolus *einfacher* kooperativer Spiele vor. Das Verfahren kann also ausschliesslich zur Berechnung des Nukleolus kooperativer Spiele angewendet werden, bei denen jeder Koalitionswert entweder Null oder Eins ist. Hierbei heisst jede Koalition mit dem Wert Null eine Verlustkoalition und entsprechend jede Koalition mit dem Wert Eins eine Gewinnkoalition. Die einfachen kooperativen Spiele finden beispielsweise Anwendung in Gremien, bei denen durch Abstimmung entschieden wird. Deshalb werden diese Spiele auch *Abstimmungsspiele* genannt. Der Algorithmus von Wolsey beruht auf der konzeptionellen Idee von Maschler et al. [82] bzw. von Kopelowitz [66]. Das bedeutet, dass zur Berechnung des Nukleolus abermals eine Folge linearer Programme gelöst wird. Das Verfahren hat zwei wesentliche Aspekte. Erstens bezieht es die Verlustkoalitionen nicht ein. Die Koalitionen, deren Wert Null ist, spielen also für die Berechnung des

4 Ausgewählte spezielle kooperative Spiele

Nukleolus keine Rolle. Zweitens geht das Verfahren von einer beliebig ausgewählten Teilmenge der Gewinnkoalitionen aus und zieht in jeder Iteration nötigenfalls weitere Gewinnkoalitionen heran. Diese zusätzlichen Gewinnkoalitionen werden mit Hilfe der binären linearen Optimierung bestimmt.

Bevor wir den Algorithmus von Wolsey vorstellen, geben wir einige grundlegende Definitionen an, die wir bei der Beschreibung des Verfahrens benötigen werden. Es sei $\Gamma = (N, v)$ ein einfaches kooperatives Spiel mit n Spielern. Die Menge der Gewinnkoalitionen bezeichnen wir als

$$\mathcal{W} := \{S \mid v(S) = 1, \ S \subseteq N\} \quad (4.1)$$

und die Menge der Verlustkoalitionen als

$$\mathcal{L} := \{S \mid v(S) = 0, \ S \subset N\}. \quad (4.2)$$

Es sei $\mathbf{y} = (y_1 \ y_2 \ \dots \ y_n) \in \mathbb{R}^n$ ein $(0, 1)$ -Vektor. Die Menge

$$S(\mathbf{y}) := \{i \mid y_i = 1\} \quad (4.3)$$

heisst die *Indexmenge* des Vektors \mathbf{y} . Sie repräsentiert eine Koalition $S \subseteq N$, für die $S = S(\mathbf{y})$ gilt. Beispielsweise stellt der Vektor $\mathbf{y} = (0 \ 1 \ 0 \ 1)$ die Koalition $S = S(\mathbf{y}) = \{2, 4\}$ dar. Umgekehrt bezeichnen wir in diesem Abschnitt \mathbf{y} als den Inzidenzvektor einer Koalition $S \subseteq N$, falls $y_i = 1$ für $i \in S$ und $y_i = 0$ für $i \notin S$ gelten. Zum Beispiel ist $\mathbf{y} = (0 \ 0 \ 1 \ 1)$ für ein kooperatives Spiel mit $n = 4$ Spielern der Inzidenzvektor der Koalition $S = \{3, 4\}$. Es sei $Y(\Gamma)$ die Menge aller $(0, 1)$ -Vektoren \mathbf{y} , die Inzidenzvektoren der Gewinnkoalitionen des Spiels $\Gamma = (N, v)$ sind. Wolsey [150], S. 228, zeigt, dass es reellwertige Zahlen a_{ij} und b_i ($i = 1, 2, \dots, m$, $j = 1, 2, \dots, n$) gibt, so dass die Menge $Y(\Gamma)$ jedes einfachen kooperativen Spiels $\Gamma = (N, v)$ durch

$$Y(\Gamma) := \left\{ \mathbf{y} \mid \sum_{j=1}^n a_{ij} y_j \geq b_i, \ y_j \in \{0, 1\} \right\} \quad (4.4)$$

bestimmt werden kann. Ist also ein $(0, 1)$ -Vektor \mathbf{y} ein Element von $Y(\Gamma)$, so ist dessen Indexmenge $S(\mathbf{y})$ eine Gewinnkoalition. Wolsey bezeichnet die Menge $Y(\Gamma)$ als eine

4.1 Abstimmungsspiele: Das Verfahren von Wolsey

Repräsentation des Spiels $\Gamma = (N, v)$, weil mit Hilfe dieser Menge der Wert jeder Koalition des Spiels bestimmt werden kann: Jede Koalition, deren Inzidenzvektor ein Element der Menge $Y(\Gamma)$ ist, ist eine Gewinnkoalition, und folglich beträgt der Wert dieser Koalition Eins. Alle anderen Koalitionen sind dann Verlustkoalitionen, deren Wert Null ist.

Ist ein einfaches kooperatives Spiel $\Gamma = (N, v)$ monoton,¹ so weist Wolsey [150], S. 228, nach, dass dieses Spiel durch

$$Y(\Gamma) := \left\{ \mathbf{y} \mid \sum_{j \in N \setminus L} y_j \geq 1, \quad \forall L \in \mathcal{L}, \quad y_j \in \{0, 1\}, \quad j = 1, 2, \dots, n \right\} \quad (4.5)$$

repräsentiert werden kann. Zur Illustration betrachten wir ein monotones einfaches kooperatives Spiel $\Gamma = (N, v)$ mit der Spielermenge $N = \{1, 2, 3\}$ und der charakteristischen Funktion

$$\begin{aligned} v(\{3\}) &= v(\{1, 3\}) = v(\{2, 3\}) = v(\{1, 2\}) = v(\{1, 2, 3\}) = 1 \\ v(\{1\}) &= v(\{2\}) = 0. \end{aligned}$$

Gemäss der Definitionsgleichung (4.5) beinhaltet die Menge $Y(\Gamma)$ jeden Vektor $\mathbf{y} \in \mathbb{R}^n$, der eine Lösung des folgenden Ungleichungssystems ist:

$$\begin{aligned} Y(\Gamma) : \quad y_2 + y_3 &\geq 1 \\ y_1 + y_3 &\geq 1 \\ y_1 + y_2 + y_3 &\geq 1 \\ y_i &\in \{0, 1\}, \quad i = 1, 2, 3. \end{aligned}$$

Durch Einsetzen der Inzidenzvektoren der Koalitionen des Spiels in das obige Ungleichungssystem können wir bestätigen, dass der Inzidenzvektor jeder Gewinnkoalition des Spiels eine Lösung des Ungleichungssystems ist und der Inzidenzvektor keiner Verlustkoalition die Bedingungen des Ungleichungssystems erfüllt. Beispielsweise verletzt der Inzidenzvektor $\mathbf{y} = (1 \ 0 \ 0)$ der Verlustkoalition $S = \{1\}$ die erste Ungleichung.

¹Für die Definition monotoner Spiele vgl. S. 17.

4 Ausgewählte spezielle kooperative Spiele

Ein monotones einfaches kooperatives Spiel $\Gamma = (N, v)$ nennt man ein *gewichtetes Abstimmungsspiel*,² wenn in diesem Spiel durch einen sog. *Gewichtsvektor* $\mathbf{w} = (w_1, w_2, \dots, w_n) \in \mathbb{R}_{>0}^n$ jedem Spieler $i \in N$ ein Gewicht w_i zugeordnet wird und für eine gegebene positive reellwertige Zahl q die charakteristische Funktion für alle Koalitionen $S \subseteq N$ folgende Form besitzt:

$$v(S) := \begin{cases} 1 & : \text{ falls } w(S) := \sum_{j \in S} w_j \geq q, \\ 0 & : \text{ sonst.} \end{cases} \quad (4.6)$$

Die Zahl q nennt man *Entscheidungsregel* oder auch *Abstimmungsregel* des Spiels. Die Menge der Gewinnkoalitionen der gewichteten Abstimmungsspiele lautet

$$\mathcal{W} := \left\{ S \mid w(S) := \sum_{j \in S} w_j \geq q, \quad S \subseteq N \right\}. \quad (4.7)$$

Bei den Abstimmungsspielen ist also eine Koalition S dann eine Gewinnkoalition, wenn die Summe der Gewichte der Spieler dieser Koalition nicht kleiner als die Entscheidungsregel q ist. In diesem Fall ist der Koalitionswert Eins. Im umgekehrten Fall heisst die Koalition S Verlustkoalition und hat den Koalitionswert Null. Ein gewichtetes Abstimmungsspiel mit dem Gewichtsvektor $\mathbf{w} = (w_1, w_2, \dots, w_n)$ und der Entscheidungsregel q bezeichnen wir mit $\Gamma = [q : w_1, w_2, \dots, w_n]$. Wolsey zeigt, dass jedes gewichtete Abstimmungsspiel Γ durch

$$Y(\Gamma) := \left\{ \mathbf{y} \mid \sum_{j=1}^n w_j y_j \geq q, \quad y_j \in \{0, 1\}, \quad j = 1, 2, \dots, n \right\} \quad (4.8)$$

repräsentiert werden kann. Zur Veranschaulichung der Repräsentation gewichteter Abstimmungsspiele betrachten wir das gewichtete Abstimmungsspiel $\Gamma = [5 : 3, 2, 2, 1]$. Für dieses Spiel besteht die Menge $Y(\Gamma)$ aus allen $(0, 1)$ -Vektoren $\mathbf{y} \in \mathbb{R}^4$, die die Ungleichung

$$Y(\Gamma) : 3y_1 + 2y_2 + 2y_3 + y_4 \geq 5, \quad y_j \in \{0, 1\}, \quad j = 1, 2, 3, 4 \quad (4.9)$$

²Die gewichteten Abstimmungsspiele diskutieren wir im Kapitel 6 ausführlich.

4.1 Abstimmungsspiele: Das Verfahren von Wolsey

erfüllen.³ Die Indexmenge jedes Vektors $\mathbf{y} \in Y(\Gamma)$ ist eine Gewinnkoalition. Zum Beispiel ist der Vektor $\mathbf{y} = (0 \ 1 \ 1 \ 1)$ eine Lösung der Ungleichung (4.9), und seine Indexmenge $S(\mathbf{y}) = \{2, 3, 4\}$ ist eine Gewinnkoalition.

Die folgende Definition führt einen wichtigen Begriff ein, den wir für die Beschreibung des Algorithmus von Wolsey benötigen.

Definition 4.1.1 (gültige Ungleichung) *Gegeben sei ein System linearer Ungleichungen in $(0, 1)$ -Variablen $\mathbf{y} \in \mathbb{R}^n$ mit der Lösungsmenge Y . Die Ungleichung*

$$\sum_{j=1}^n \pi_j y_j \geq \pi_0, \quad \pi_0, \pi_j \in \mathbb{R}, \quad y_j \in \{0, 1\}, \quad j = 1, \dots, n \quad (4.10)$$

heißt eine gültige Ungleichung (valid inequality) für die Lösungsmenge Y , wenn sie für alle $\mathbf{y} \in Y$ erfüllt ist.

Für ein einfaches kooperatives Spiel $\Gamma = (N, v)$ impliziert die Definition 4.1.1, dass eine Ungleichung $\sum_{j=1}^n \pi_j y_j \geq \pi_0$ dann und nur dann eine gültige Ungleichung für die Spielrepräsentation $Y(\Gamma)$ ist, wenn jeder Vektor $\mathbf{y} \in Y(\Gamma)$, dessen Indexmenge S eine Gewinnkoalition ist, diese Ungleichung erfüllt (vgl. (4.4), S. 174). Wie wir anschließend sehen werden, ist dieses Merkmal gültiger Ungleichungen bei der Berechnung des Nukleolus einfacher kooperativer Spiele nützlich.

Wie erwähnt, verlangt das Verfahren von Wolsey [150] zur Berechnung des Nukleolus einfacher kooperativer Spiele die Lösung einer Folge linearer Programme. Die Besonderheit dieses Verfahrens liegt einerseits darin, dass es als Nebenbedingungen des zu lösenden linearen Programms am Anfang nur eine Teilmenge der Gewinnkoalitionen berücksichtigt. Andererseits erfordert das Verfahren, dass in jeder Iteration zusätzlich ein binäres lineares Programm gelöst wird, dessen Nebenbedingungen der Repräsentation des Spiels entsprechen. Wie wir noch erläutern werden, können aus der berechneten Lösung dieses binären Programms die folgenden Informationen gewonnen werden. Erstens kann überprüft werden, ob der Nukleolus bereits gefunden worden ist. Zweitens können weitere Gewinnkoalitionen ausgewählt werden, die als

³Eine Ungleichung der Gestalt (4.9) nennt man in der Literatur *Knapsack-* oder auch *Rucksackungleichung*. Vgl. dazu Anhang A.4.

4 Ausgewählte spezielle kooperative Spiele

Nebenbedingungen des nächsten zu lösenden linearen Programms zusätzlich berücksichtigt werden, falls der Nukleolus noch nicht berechnet ist.

Bevor wir den Algorithmus von Wolsey [150] detailliert beschreiben, weisen wir noch auf die folgende wichtige Beziehung zwischen dem Nukleolus eines monotonen einfachen kooperativen Spiels und der gültigen Ungleichung der Spielrepräsentation hin:

Satz 4.1.1 *Es sei $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)$ der Nukleolus eines monotonen einfachen kooperativen Spiels $\Gamma = (N, v)$ mit $\alpha(N) := \sum_{i \in N} \alpha_i = 1$ und*

$$\alpha_0 := \min_{S \in \mathcal{W}} \alpha(S) = \min_{S \in \mathcal{W}} \sum_{j \in S} \alpha_j = \min_{S \in \mathcal{W}} \sum_{j=1}^n \alpha_j y_j^S.$$

Dann ist

$$\sum_{j=1}^n \alpha_j y_j \geq \alpha_0, \quad y_j \in \{0, 1\} \quad (4.11)$$

eine gültige Ungleichung für $Y(\Gamma)$. Ferner gibt es neben (4.11) keine andere gültige Ungleichung mit einem höheren Wert als α_0 .

Wir fügen den Beweis dieser Aussage an, da er für das Verständnis des Algorithmus von Wolsey entscheidend ist (vgl. Wolsey [150], S. 230):

Beweis:

Um den Nukleolus eines einfachen kooperativen Spiels im Wege der Lösung einer Folge linearer Programme zu berechnen, wird in der Ausgangssituation das folgende lineare Programm gelöst (vgl. Abschnitt 3.3.1, S. 111)

$$\begin{aligned} \text{LP1 :} \quad & \min_{\mathbf{x}, t} t \\ & t \geq v(S) - x(S), \quad \forall S \subset N, S \neq \emptyset \\ & x(N) = 1 \\ & x_i \geq 0, \quad i = 1, 2, \dots, n. \end{aligned}$$

Demnach ist der Nukleolus α des monotonen einfachen kooperativen Spiels $\Gamma = (N, v)$ notwendigerweise eine Lösung von LP1. Da in einem einfachen kooperativen

4.1 Abstimmungsspiele: Das Verfahren von Wolsey

Spiel alle Verlustkoalitionen den Wert Null haben, also $v(S) = 0$ für alle $S \in \mathcal{L}$ ist, sind diese Koalitionen für die Berechnung des Nukleolus redundant. Lassen wir nun in LP1 die Verlustkoalitionen ausser Acht und setzen zudem $v(S) = 1$ für alle $S \in \mathcal{W}$, so gelangen wir zu dem folgenden zu LP1 äquivalenten linearen Programm:

$$\begin{aligned} \text{LP2 : } \quad & \min_{\mathbf{x}, t} t \\ & t \geq 1 - x(S), \quad \forall S \in \mathcal{W}, S \neq N \end{aligned} \quad (4.12)$$

$$x(N) = 1 \quad (4.13)$$

$$x_i \geq 0, \quad i = 1, 2, \dots, n. \quad (4.14)$$

Da das hier betrachtete kooperative Spiel ein monotones einfaches Spiel ist, gilt einerseits $x(S) \leq x(N) = 1$ für alle $S \in \mathcal{W}$. Andererseits ist $x(S) \geq 0$ für alle $S \in \mathcal{W}$, weil $x_i \geq 0$ für alle $i \in N$ ist. Die Kombination der beiden Ungleichungen für $x(S)$ ergibt

$$0 \leq x(S) \leq 1 \quad \text{für alle } S \in \mathcal{W}. \quad (4.15)$$

Auch für die Variable t trifft $0 \leq t \leq 1$ zu, weil die Zielfunktion in LP2 minimiert wird und die Nebenbedingungen (4.12) des linearen Programms LP2 sowie die Bedingung (4.15) gelten. Definieren wir $x_0 := 1 - t$, dann gilt $0 \leq x_0 \leq 1$. Insbesondere sind nun die Minimierung der Variablen t und die Maximierung von x_0 zueinander äquivalent. Demnach können wir LP2 durch das folgende Maximierungsproblem ersetzen:

$$\begin{aligned} \text{LP3 : } \quad & \max_{\mathbf{x}, x_0} x_0 \\ & x(S) \geq x_0, \quad \forall S \in \mathcal{W}, S \neq N \\ & x(N) = 1 \\ & x_i \geq 0, \quad i = 1, 2, \dots, n. \end{aligned}$$

Nehmen wir an, dass $(\mathbf{x}^*; x_0^*)$ eine optimale Lösung dieses Programms ist, wobei $\alpha_0 := x_0^*$ dem maximalen Wert der Zielfunktion von LP3 entspricht. Ferner sei $\alpha := \mathbf{x}^*$. Dann gibt $(\alpha ; \alpha_0)$ für $Y(\Gamma)$ eine gültige Ungleichung an (vgl. die Ungleichung (4.11), S. 178), wobei α_0 maximal ist. \square

4 Ausgewählte spezielle kooperative Spiele

Als ein Resultat des Satzes 4.1.1 können wir die folgende Aussage treffen (vgl. Wolsey [150], S. 230): Es seien $\Gamma = (N, v)$ ein monotones einfaches kooperatives Spiel mit n Spielern und

$$\sum_{j=1}^n \pi_j y_j \geq \pi_0 \quad (4.16)$$

eine gültige Ungleichung für die Repräsentation $Y(\Gamma)$ dieses Spiels. Wenn die gültige Ungleichung (4.16) für den maximalen Wert von π_0 eindeutig ist, dann ist $\boldsymbol{\pi} = (\pi_1, \pi_2, \dots, \pi_n)$ der Nukleolus des Spiels $\Gamma = (N, v)$.

Vor diesem Hintergrund ermittelt das Verfahren von Wolsey [150], das wir unten beschreiben werden, zur Berechnung des Nukleolus eines monotonen einfachen kooperativen Spiels $\Gamma = (N, v)$ eine eindeutige gültige Ungleichung für die Repräsentation des Spiels.

Die Berechnung des Nukleolus eines einfachen kooperativen Spiels $\Gamma = (N, v)$ mit Hilfe des Verfahrens von Wolsey [150] erfolgt in der Regel über mehrere Iterationen, wobei in jeder Iteration ein lineares Programm und ein binäres lineares Programm gelöst werden. Hierbei bildet die Durchführung der nachstehenden Schritte 1 und 2 des Algorithmus jeweils eine Iteration des Verfahrens. Wir geben nun die Gestalt der einzelnen Schritte des Verfahrens und die Begründung dieser Schritte an. Es sei \mathcal{W} die Menge aller Gewinnkoalitionen des Spiels und \mathcal{V}_q die Menge der in der nachfolgend zu erläuternden Iteration (q, r) des Verfahrens ermittelten Koalitionen $S \subset N$, deren Überschüsse minimal sind. Ist also x_0^q die optimale Lösung des linearen Programms LP3 (vgl. S. 179), so gilt $x(S) = x_0^q$ für alle $S \in \mathcal{V}_q$.

Initialisierung

Es seien $k := 0$, $r := 0$ und $\mathcal{V}_0 := \emptyset$. Ferner sei $\mathcal{W}_{k,r}$ eine beliebige nicht-leere Teilmenge von \mathcal{W} .

Schritt 1

Wir lösen das lineare Programm

4.1 Abstimmungsspiele: Das Verfahren von Wolsey

$$\begin{aligned}
\text{LP : } \quad & \max_{\mathbf{x}, u} u \\
& \text{unter den Nebenbedingungen} \\
& x(S) \geq u, \quad \forall S \in \mathcal{W}_{k,r} \subseteq \mathcal{W} \setminus \bigcup_{q=0}^k \mathcal{V}_q \\
& x(S) = x_0^q, \quad \forall S \in \mathcal{V}_q, \quad q = 0, 1, \dots, k \\
& x(N) = 1 \\
& x_i \geq 0, \quad i = 1, 2, \dots, n.
\end{aligned}$$

Die Lösung dieses linearen Programms sei $(\mathbf{x}^{k,r}; x_0^{k,r})$, wobei $x_0^{k,r}$ zugleich der optimale Wert der Zielfunktion ist. Es sei

$$n^k := \left| \bigcup_{q=0}^k \mathcal{V}_q \right| \quad (4.17)$$

die Anzahl der Gleichungen in den Nebenbedingungen von LP für die Koalitionen $S \in \mathcal{V}_q$, $q = 0, 1, \dots, k$. Das bedeutet, dass n^k die Anzahl der Koalitionen ist, deren Situation nach der Iteration (k, r) nicht mehr verbessert werden kann. Der Unterschied zwischen LP und LP3 (vgl. S. 179) besteht darin, dass das lineare Programm LP nur von einer Teilmenge der Gewinnkoalitionen, also von $\mathcal{W}_{k,r} \subseteq \mathcal{W}$, ausgeht, während LP3 alle Gewinnkoalitionen berücksichtigt.

Schritt 2

Der Nukleolus des kooperativen Spiels $\Gamma = (N, v)$ liegt dann vor, wenn das lineare Programm LP eine eindeutige Lösung hat und sie zudem alle Nebenbedingungen von LP3 erfüllt. Da die Lösung von LP ermittelt wird, ohne alle möglichen Gewinnkoalitionen zu berücksichtigen, überprüfen wir, ob diese Lösung eine nicht in LP enthaltene Nebenbedingung von LP3 verletzt. In diesem Fall stellen wir ein neues

4 Ausgewählte spezielle kooperative Spiele

lineares Programm auf, indem wir die verletzte Nebenbedingung dem Programm LP hinzufügen. Zu diesem Zweck lösen wir das binäre lineare Programm⁴

$$\begin{aligned} \text{IP :} \quad & (n^k + 1) - \text{bestes Minimum von } \mathbf{x}^{k,r} \mathbf{y}^T \\ & \text{unter der Nebenbedingung} \\ & \mathbf{y} \in Y(\Gamma). \end{aligned}$$

Im Optimierungsproblem IP ist $\mathbf{x}^{k,r}$ die im Schritt 1 ermittelte optimale Lösung des linearen Programms LP. Der n -dimensionale $(0, 1)$ -Vektor \mathbf{y} besteht aus den Variablen von IP. Ferner ist $Y(\Gamma)$ eine Repräsentation des kooperativen Spiels $\Gamma = (N, v)$. Wir nehmen an, dass $(\mathbf{y}^{k,r}; z^{k,r})$ eine optimale Lösung von IP ist, wobei $z^{k,r}$ zugleich der optimale Wert der Zielfunktion ist.⁵ Wegen der Nebenbedingung $\mathbf{y} \in Y(\Gamma)$ ist die Indexmenge von $\mathbf{y}^{k,r}$ eine Gewinnkoalition.

Im Folgenden erläutern wir, warum durch das binäre lineare Programm IP ermittelt werden kann, ob die optimale Lösung von LP gleichzeitig eine optimale Lösung von LP3 ist oder welche Nebenbedingung von LP3 allenfalls verletzt wird. Zu diesem Zweck betrachten wir den Fall $k = 0, r = 0$. Die binäre lineare Optimierungsaufgabe IP hat in diesem Fall die Form

$$\begin{aligned} \min_{\mathbf{y}} \quad & \mathbf{x}^{0,0} \mathbf{y}^T \\ & \text{unter der Nebenbedingung} \\ & \mathbf{y} \in Y(\Gamma). \end{aligned}$$

Es sei $(\mathbf{y}^{0,0}; z^{0,0})$ eine optimale Lösung, wobei $z^{0,0}$ der optimale Zielfunktionswert ist. Diese Lösung kann wie folgt interpretiert werden. Die Indexmenge von $\mathbf{y}^{0,0}$ ist diejenige Koalition $\hat{S} \in \mathcal{W}$, deren Auszahlung $x^{0,0}(\hat{S})$ bezüglich der im Schritt 1 gefundenen optimalen Lösung $\mathbf{x}^{0,0}$ des linearen Programms LP am niedrigsten ist. Demnach erfüllt $\mathbf{x}^{0,0}$ alle Nebenbedingungen von LP3, falls $z^{0,0} \geq x_0^{0,0}$ ist, wobei $x_0^{0,0}$ der optimale Zielfunktionswert des im Schritt 1 gelösten linearen Programms

⁴Zur Lösung von binären linearen Programmen der Gestalt IP vgl. Anhang A.4.

⁵Eine Lösung von IP existiert immer, weil es für jedes einfache Spiel $\Gamma = (N, v)$, wie auf S. 174 erläutert, eine Repräsentation $Y(\Gamma)$ gibt.

4.1 Abstimmungsspiele: Das Verfahren von Wolsey

LP ist. Umgekehrt verletzt $\mathbf{x}^{0,0}$ die zur Koalition \hat{S} gehörende Nebenbedingung in LP3, falls $z^{0,0} = x^{0,0}(\hat{S}) < x_0^{0,0}$ ist. Denn in LP3 muss für alle $S \in \mathcal{W}$ die Bedingung $x^{0,0}(S) \geq x_0^{0,0}$ erfüllt sein. Liegt nun der Fall $z^{0,0} \geq x_0^{0,0}$ vor und hat das lineare Programm LP eine eindeutige Lösung, so ist $\mathbf{x}^{0,0}$ der Nukleolus des Spiels. In der Regel hat aber LP in der ersten Iteration noch keine eindeutige Lösung, sodass man die Menge \mathcal{V}_1 jener Koalitionen bestimmt, deren minimale Überschüsse bereits ermittelt worden sind. Anschliessend wird in der nächsten Iteration ein neues lineares Programm gelöst, in dem nun insgesamt $n^k + 1$ Nebenbedingungen in der Form von Gleichungen vorliegen. Aus diesem Grund berechnen wir in der nächsten Iteration im Programm IP das $(n^k + 1)$ -beste Minimum von $\mathbf{x}^{k,r} \mathbf{y}^T$. Um zu überprüfen, ob die gefundene Lösung von LP der Nukleolus des Spiels ist, vergleichen wir $z^{k,r}$ und $x_0^{k,r}$. Wir unterscheiden zwischen zwei Fällen:

- 1. Fall** Falls $z^{k,r} < x_0^{k,r}$ ist, ist die zu der Koalition $\hat{S} := S(\mathbf{y}^{k,r})$ gehörende Nebenbedingung im linearen Programm LP3 wegen

$$z^{k,r} < x_0^{k,r} \Rightarrow x^{k,r}(\hat{S}) < x_0^{k,r} \quad (4.18)$$

verletzt, weil in LP3 für jede Gewinnkoalition $S \in \mathcal{W}$ die Bedingung $x(S) \geq x_0$ erfüllt werden muss. Folglich fügen wir die Koalition \hat{S} der Menge $\mathcal{W}_{k,r}$ hinzu, d. h. wir setzen

$$\mathcal{W}_{k,r+1} := \mathcal{W}_{k,r} \cup \hat{S}.$$

Entsprechend erweitern wir LP um die zu der Koalition \hat{S} gehörende Nebenbedingung $x(\hat{S}) \geq u$. Da wir noch keine Lösung von LP3 gefunden haben, setzen wir $r := r + 1$ und gehen zurück zu Schritt 1.

- 2. Fall** Falls $z^{k,r} \geq x_0^{k,r}$ ist, erfüllt die gewonnene optimale Lösung von LP alle Nebenbedingungen von LP3. In diesem Fall überprüfen wir, ob diese Lösung eindeutig ist. Falls Eindeutigkeit vorliegt, ist der Nukleolus α gefunden, d. h. $\alpha := \mathbf{x}^{k,r}$. Falls die optimale Lösung von LP nicht-eindeutig ist, suchen wir in $\mathcal{W}_{k,r}$ diejenigen Koalitionen, deren zugehörige Nebenbedingungen für alle

4 Ausgewählte spezielle kooperative Spiele

optimalen Lösungen von LP als Gleichungen erfüllt sind, d. h. deren minimale Überschüsse ermittelt worden sind. Die Menge

$$\mathcal{V}_{k+1} := \left\{ S \mid x(S) = x_0^{k,r} \ \forall \text{ optimalen Lösungen von LP, } \forall S \in \mathcal{W}_{k,r} \right\}$$

enthält diese Koalitionen. Anschliessend setzen wir

$$x_0^k := x_0^{k,r}, \quad r := 0, \quad k := k + 1$$

und gehen zurück zu Schritt 1.

Vergleichen wir das Verfahren von Wolsey [150] mit den im dritten Kapitel vorgestellten Verfahren der Methoden der dritten Generation zur Berechnung des Nukleolus allgemeiner kooperativer Spiele, so stellen wir Folgendes fest. Das Verfahren von Wolsey [150] beruht zwar auf der konzeptionellen Idee von Maschler et al. [82], berücksichtigt aber zur Berechnung des Nukleolus einfacher kooperativer Spiele nur die Gewinnkoalitionen des Spiels. Demgegenüber können die Verlustkoalitionen ausser Acht gelassen werden. Dies reduziert die Anzahl der Nebenbedingungen der zu lösenden linearen Programme erheblich. Zudem zieht der Algorithmus zu Beginn lediglich eine beliebig ausgewählte Teilmenge der Gewinnkoalitionen des Spiels heran. Demnach ist die Anzahl der Nebenbedingungen der linearen Programme in der Regel niedriger als die Anzahl der Gewinnkoalitionen. Neben diesen Vorteilen hat das Verfahren von Wolsey einen Nachteil. Gemäss dem Verfahren wird nämlich in jeder Iteration zusätzlich ein binäres lineares Programm gelöst, um festzustellen, ob die Lösung des mit Hilfe nur einer Teilmenge der Gewinnkoalitionen aufgestellten linearen Programms gleichzeitig eine Lösung des linearen Programms mit allen Gewinnkoalitionen ist. Ist dies nicht der Fall, so kann ebenfalls mit Hilfe der Lösung dieses binären linearen Programms eine Koalition bestimmt werden, die in der nächsten Iteration berücksichtigt wird.

Wir haben in dieser Arbeit den oben beschriebenen Algorithmus von Wolsey zur Berechnung des Nukleolus gewichteter Abstimmungsspiele programmiert. Im Kapitel 5 werden wir die Implementation dieses Verfahrens beschreiben und zur Illustration

4.2 Zuordnungsspiele und verwandte kooperative Spiele

sowohl des Verfahrens als auch unserer Implementation ein einfaches kooperatives Spiel mit $n = 4$ Spielern als Beispiel angeben.

4.2 Zuordnungsspiele und verwandte kooperative Spiele

In diesem Abschnitt stellen wir zunächst das sog. *Zuordnungsspiel* vor, das von Shapley/Shubik [125] in die Literatur eingeführt wurde. Anschliessend gehen wir auf einige hiermit verwandte kooperative Spiele ein. Zunächst betrachten wir einen Markt, auf dem ein unteilbares Gut (Haus, Automobil usw.) gehandelt wird. Auf der Angebotsseite bietet jeder Anbieter eine Einheit des Gutes an. Auf der Nachfrageseite fragt jeder Nachfrager ebenfalls nur eine Einheit des Gutes nach. Es seien $\bar{M} = \{1, 2, \dots, m\}$ die Menge der Anbieter und $\bar{P} = \{1, 2, \dots, p\}$ die Menge der Nachfrager, wobei $\bar{M} \cap \bar{P} = \emptyset$ gelte. Ferner sei $m \leq p$.⁶ Jeder Anbieter (Verkäufer) i , $i = 1, 2, \dots, m$, ist bereit, sein Gut für mindestens c_i Geldeinheiten zu verkaufen. Jeder Nachfrager (Käufer) j , $j = 1, 2, \dots, p$, hat eine Zahlungsbereitschaft z_{ij} für das Gut des Anbieters i . Wir nehmen an, dass eine Transaktion zwischen einem Anbieter i und einem Nachfrager j zu einem Gesamtgewinn für diese beiden Akteure in der Höhe von $a_{ij} \geq 0$ Geldeinheiten führt. Diese Situation können wir in einem kooperativen Spiel abbilden. Zu diesem Zweck interpretieren wir die Nachfrager und die Anbieter als Spieler. Für jede Koalition $S \subset \bar{M} \cup \bar{P}$ entspricht der Koalitionswert $v(S)$ der maximalen Summe der Gewinne aus den Transaktionen zwischen den Spielern dieser Koalition. Falls $S \subset \bar{M}$ oder $S \subset \bar{P}$ ist, kommen keine Transaktionen zustande. Daher gilt in diesem Fall $v(S) = 0$. Ansonsten erfordert die Bestimmung des Wertes jeder Koalition die Lösung eines Optimierungsproblems. Im Allgemeinen können wir für jede Koalition $S \subseteq \bar{M} \cup \bar{P}$ den Wert $v(S)$ mit Hilfe des nachstehenden binären linearen Programms ermitteln,⁷ wobei x_{ij} , $i \in S \cap \bar{M}$, $j \in S \cap \bar{P}$ die Variablen sind:

⁶Mit Hilfe der Einbeziehung von sog. *Dummy-Spielern* werden wir sogleich erläutern, dass die Annahme $m \leq p$ keine Beschränkung der Allgemeinheit darstellt.

⁷Vgl. dazu Owen [97], S. 221, und Chin [27], S. 56.

4 Ausgewählte spezielle kooperative Spiele

$$\begin{aligned}
& \max \sum_{i \in S \cap \bar{M}} \sum_{j \in S \cap \bar{P}} a_{ij} x_{ij} \\
& \text{unter den Nebenbedingungen} \\
& \sum_{j \in S \cap \bar{P}} x_{ij} \leq 1 \quad \text{für alle } i \in S \cap \bar{M} \\
& \sum_{i \in S \cap \bar{M}} x_{ij} \leq 1 \quad \text{für alle } j \in S \cap \bar{P} \\
& x_{ij} \in \{0, 1\} \quad \text{für alle } i \in S \cap \bar{M}, j \in S \cap \bar{P}.
\end{aligned} \tag{4.19}$$

Der optimale Zielfunktionswert des linearen Programms entspricht nämlich dem Koalitionswert $v(S)$. Ist in der optimalen Lösung des obigen binären linearen Programms $x_{ij} = 1$, kommt zwischen dem Anbieter i und dem Nachfrager j eine Transaktion zustande. In diesem Fall drücken wir durch das Indexpaar (i, j) aus, dass der Anbieter i sein Gut dem Nachfrager j verkauft. Wenn $x_{ij} = 0$ ist, kommt es zu keiner Transaktion zwischen dem Anbieter i und dem Nachfrager j . Falls im Optimum des linearen Programms (4.19) für einen Anbieter $s \in S \cap \bar{M}$ gilt, dass $x_{sj} = 0$ für alle Nachfrager $j \in S \cap \bar{P}$, stellen wir diese Situation durch $(s, 0)$ dar. Hierbei nennt man den fiktiven Spieler 0 in der Literatur Dummy-Spieler. Die Notation $(s, 0)$ besagt, dass der Anbieter s sein Gut nicht verkaufen kann, weil er auf keinen Nachfrager trifft. Falls für einen Nachfrager $b \in S \cap \bar{P}$ gilt, dass $x_{ib} = 0$ für alle Anbieter $i \in S \cap \bar{M}$, dann verwenden wir unter Einbeziehung des Dummy-Spielers die Notation $(0, b)$. Das Indexpaar $(0, b)$ besagt, dass der Nachfrager b das Gut nicht kaufen kann, weil er keinen Anbieter findet. Die Indexpaare $(s, 0)$ und $(0, b)$ interpretieren wir als die Einerkoalitionen $\{s\}$ bzw. $\{b\}$. Es seien $M := \bar{M} \cup \{0\} = \{0, 1, 2, \dots, m\}$ und $P := \bar{P} \cup \{0\} = \{0, 1, 2, \dots, p\}$. Da $(i, 0)$ und $(0, j)$ besagen, dass für den Anbieter i bzw. für den Nachfrager j keine Transaktion zu Stande kommt, setzen wir $a_{00} = a_{i0} = a_{0j} = 0$ für alle $i \in \bar{M}$, $j \in \bar{P}$. Die Matrix $\mathbf{A} = (a_{ij})_{\substack{i=0, \dots, m \\ j=0, \dots, p}}$ sei für alle $i \in M$, $j \in P$ die $(m+1) \times (p+1)$ -Gewinnmatrix. Wenn es nun gelingt, unter Einbeziehung des Dummy-Spielers jedem Anbieter genau einen Nachfrager zuzuordnen und umgekehrt, so nennen wir das Ergebnis eine *Zuordnung* σ über der Menge

4.2 Zuordnungsspiele und verwandte kooperative Spiele

$M \cup P$. Ein derartiges kooperatives Spiel heisst *Zuordnungsspiel* (engl. *assignment game*).

Jedes Zuordnungsspiel kann auch graphentheoretisch definiert werden. Eine derartige Definition finden wir z. B. in Bilbao [7], S. 10.⁸ Es sei $G = (\bar{M} \cup \bar{P}, E)$ ein bipartiter Graph, wobei die grosse Koalition $N := \bar{M} \cup \bar{P}$ die Knotenmenge und E die Kantenmenge sind. Es sei $a : E \rightarrow \mathbb{R}_{\geq 0}$ eine (Kanten-)Bewertungsfunktion. Für jede Koalition $S \subseteq N$ sei $G' = (S, E_S)$ der zugehörige Teilgraph von G , wobei $E_S \subseteq E$. Das heisst, dass jede Koalition $S \subseteq N$ in zwei disjunkte Spielermengen (Anbieter und Nachfrager) aufgeteilt wird, so dass jeder Anbieter nur einem Nachfrager zugeordnet werden kann und umgekehrt. Die charakteristische Funktion eines Zuordnungsspiels ist dann gegeben durch:

$$v(S) := \max \left\{ \sum_{(i,j) \in Y} a_{ij} : Y \text{ ist ein vollständiges Matching in } G' \right\}. \quad (4.20)$$

Da das gehandelte Gut unteilbar ist und jeder Anbieter bzw. Nachfrager nur eine Einheit des Gutes anbietet bzw. nachfragt, wird der Wert $v(S)$ jeder Koalition S durch die Zweierkoalitionen $\{i, j\}$, $i \in S \cap M$, $j \in S \cap P$ bestimmt. Nach Voraussetzung ist a_{ij} der Gewinn der Koalition $S = \{i, j\}$, $i \in M$, $j \in P$. Somit können wir den Wert a_{ij} bei Zuordnungsspielen als den Wert der Koalition S interpretieren: $v(\{i, j\}) = a_{ij}$. Zur Illustration geben wir das folgende Beispiel an:

Beispiel 4.2.1 *Vier Häuser von vier verschiedenen Anbietern sind zu verkaufen. Die Anzahl der Nachfrager ist Fünf. Jeder der potentiellen Nachfrager möchte genau ein Haus erwerben. In der Tabelle 4.1 werden die Zahlungsbereitschaften z_{ij} der Nachfrager (Käufer) und die von den Anbietern (Verkäufern) verlangten Mindestpreise c_i (z. B. in CHF 10 000) wiedergegeben.*

Der Verkäufer 1 verlangt beispielsweise für sein Haus einen Preis in Höhe von mindestens 20. Der Käufer 1 ist bereit, für dieses Haus höchstens 26 zu bezahlen. Falls sich die beiden Marktteilnehmer über einen Preis von $20 \leq q \leq 26$ einigen, ist

⁸Für die in diesem Abschnitt verwendeten graphentheoretischen Begriffe siehe Anhang E.

4 Ausgewählte spezielle kooperative Spiele

Tabelle 4.1: Ein Zuordnungsspiel

Mindestpreise c_i	Zahlungsbereitschaften z_{ij} der Käufer				
	j = 1	j = 2	j = 3	j = 4	j = 5
$c_1 \geq 20$	26	27	24	25	29
$c_2 \geq 24$	28	27	31	32	27
$c_3 \geq 30$	29	31	33	36	34
$c_4 \geq 32$	34	34	37	39	40

Quelle: Vgl. Raghavan [108], S. 215.

der Vorteil des Verkäufers aus dieser Transaktion $q - c_1 = q - 20$ und der Vorteil des Käufers $z_{11} - q = 26 - q$. Der Gesamtvorteil der Koalition $\{1, 1\}$ ist somit $a_{11} = (q - 20) + (26 - q) = 6$. Folglich gilt im Allgemeinen

$$v(\{i, j\}) = a_{ij} = \begin{cases} z_{ij} - c_i, & \text{falls } z_{ij} \geq c_i \\ 0, & \text{falls } z_{ij} < c_i. \end{cases} \quad (4.21)$$

Somit stellt sich die Gewinnmatrix \mathbf{A} folgendermassen dar:

$$\mathbf{A} = (a_{ij})_{\substack{i=0,\dots,4 \\ j=0,\dots,5}} = \left(\begin{array}{c|ccccc} 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 6 & 7 & 4 & 5 & 9 \\ 0 & 4 & 3 & 7 & 8 & 3 \\ 0 & 0 & 1 & 3 & 6 & 4 \\ 0 & 2 & 2 & 5 & 7 & 8 \end{array} \right).$$

Die erste Zeile bzw. Spalte der Matrix \mathbf{A} repräsentieren den Dummy-Spieler. Im Beispiel 4.2.1 erhalten wir den Wert der grossen Koalition $N := M \cup P$ gemäss (4.19) als den optimalen Wert der Zielfunktion der folgenden Maximierungsaufgabe:

$$\begin{aligned} & \max \sum_{i=1}^4 \sum_{j=1}^5 a_{ij} x_{ij} \\ & \text{unter den Nebenbedingungen} \\ & \sum_{j=1}^5 x_{ij} \leq 1, \quad i = 1, 2, 3, 4 \\ & \sum_{i=1}^4 x_{ij} \leq 1, \quad j = 1, 2, 3, 4, 5 \\ & x_{ij} \in \{0, 1\}, \quad i = 1, 2, 3, 4, \quad j = 1, 2, 3, 4, 5. \end{aligned}$$

4.2 Zuordnungsspiele und verwandte kooperative Spiele

Die Lösung dieses binären linearen Programms⁹ ergibt $v(N) = 28$. Die zugehörige optimale Zuordnung lautet:

$$\sigma^* := \{(0, 0), (0, 1), (1, 2), (2, 3), (3, 4), (4, 5)\}.$$

In der Literatur finden wir spezifische Verfahren zur Berechnung des Nukleolus des Zuordnungsspiels. Hierfür schlagen Solymosi [130] bzw. Solymosi/Raghavan [132] einen Algorithmus vor, der auf der Konzeption von Maschler et al. [82] beruht. Der Algorithmus hat den Vorteil, dass er nur jene Zweierkoalitionen berücksichtigt, die aus genau einem Anbieter und lediglich einem Nachfrager ($|S| = 2$) bestehen.¹⁰ Hierbei kann der Anbieter bzw. der Nachfrager in S ein Dummy-Spieler sein.

Das Matchingspiel

Das sog. *Matchingspiel* ist ein spezielles kooperatives Spiel, das mit dem Zuordnungsspiel in einem engen Zusammenhang steht. Das Matchingspiel bildet eine Spielsituation ab, die beispielsweise darin besteht, an einem Ausbildungszentrum Dozenten und Kurse einander so zuzuordnen, dass möglichst viele Kurse gehalten werden können. Hierbei soll jeder Dozent nur einen Kurs halten, und jeder Kurs soll nur von einem Dozenten gehalten werden. Um das Matchingspiel formal darzulegen, nehmen wir an, dass $N := \bar{M} \cup \bar{P}$ die Spielermenge ist, wobei $\bar{M} \cap \bar{P} = \emptyset$. Es sei $G = (\bar{M} \cup \bar{P}, E)$ ein Graph mit der Kantenmenge E und der (Kanten-)Bewertungsfunktion $a : E \rightarrow \mathbb{R}_{\geq 0}$. Gegeben sei für jede Koalition $S \subset N$ ein Teilgraph $G' = (S, E_S)$ von G , wobei $E_S \subseteq E$. Die charakteristische Funktion des Matchingspiels lautet dann (vgl. Bilbao [7], S. 10, und Kern/Paulusma [59], S. 4):

$$v(S) := \max \left\{ \sum_{(i,j) \in Y} a_{ij} : Y \subseteq E_S \text{ ist ein Matching in } G' \right\}. \quad (4.22)$$

⁹Wir haben das Optimierungsproblem mit der Software LINDO gelöst. Diese Software ist erhältlich unter www.lindo.com (26.02.2004).

¹⁰Huberman [54] beweist, dass für die Berechnung des Nukleolus eines allgemeinen kooperativen Spiels mit nicht-leerem Kern nur die sog. *essentiellen Koalitionen* relevant sind. In einem kooperativen Spiel $\Gamma = (N, v)$ ist eine Koalition $S \neq \emptyset$ essentiell, falls für $S = S_1 \cup S_2$, $S_1 \cap S_2 = \emptyset$ und $S_1 \neq \emptyset \neq S_2$ gilt, dass $v(S) > v(S_1) + v(S_2)$. Solymosi/Raghavan [132] zeigen, dass für die Berechnung des Nukleolus jedes Zuordnungsspiels lediglich die erwähnten Zweierkoalitionen erforderlich sind.

4 Ausgewählte spezielle kooperative Spiele

Kern/Paulusma [59] betrachten eine spezielle Klasse der Matchingspiele, bei denen das Gewicht jeder Kante eins ist. Das oben erwähnte Beispiel der Situation der Kursbelegung ist ein solches Matchingspiel, da es dabei darum geht, möglichst viele Kurse zu belegen. Graphentheoretisch können wir jede zulässige Zuordnung eines Dozenten zu einem Kurs als eine Kante betrachten. In diesem Fall ist die charakteristische Funktion solcher Matchingspiele wie folgt definiert:

$$v(S) := \max \{|Y| : Y \subseteq E_S \text{ ist ein Matching in } G'\} .^{11} \quad (4.23)$$

Der Wert einer Koalition S entspricht also der maximalen Kantenanzahl des Matchings $Y \subseteq E_S$. Das bedeutet, dass Y ein maximales Matching in E_S ist. Kern/Paulusma [59] schlagen einen spezifischen Algorithmus zur Berechnung des Nukleolus dieser Matchingspiele vor. Der Algorithmus beruht auf dem von Maschler et al. [82] entwickelten Verfahren. Kern/Paulusma zeigen, dass sich die Berechnung des Kleinstkerns vereinfachen lässt, wenn alle Kantengewichte eins sind. Auf diesem Wege gelingt es ihnen, die Anzahl der Nebenbedingungen der linearen Programme zur Berechnung des Kleinstkerns zu reduzieren.

Das Permutationsspiel

Das sog. *Permutationsspiel* ist eine weitere Klasse spezieller kooperativer Spiele, die mit den Zuordnungsspielen in Verbindung gebracht werden. Dieses Spiel wurde von Tijs et al. [140] in die Literatur eingeführt. Wir gehen im Weiteren von einer Spielsituation aus, in der jeder Spieler einen Auftrag hat und hierzu mit einer Ressource in der Form einer Maschine j ausgestattet ist. Wir setzen voraus, dass jede Maschine j nur einen Auftrag ausführen kann. Die Ausführung des Auftrags mit einer Maschine j generiere die Auszahlung k_{ij} an den Spieler i . Es sei $\mathbf{K} := (k_{ij})_{\substack{i \in N \\ j \in N}}$. Wir nehmen an, dass es sich für einige Spieler lohnen kann, ihren Auftrag mit der Maschine eines anderen Spielers auszuführen. Es sei $\pi : N \rightarrow N$ eine bijektive Abbildung der Spielermenge N auf sich selbst. Die Funktion π ist also eine Permutation von N . Deshalb bezeichnen wir die Abbildung π als N -Permutation. Des Weiteren sei Π_N

¹¹Im Bereich des Operations Research bezeichnet man dieses Problem als Matching maximaler Kardinalität.

4.2 Zuordnungsspiele und verwandte kooperative Spiele

die Menge aller möglichen Permutationen der Menge N . Ferner setzen wir voraus, dass jede Koalition $S \subset N$ lediglich sich selbst organisieren kann, so dass $\pi(i) = i$ für alle $i \in N \setminus S$ gilt. Das bedeutet im obigen Beispiel, dass die Spieler einer Koalition nur untereinander Absprachen vereinbaren können, wer mit welcher Maschine seinen Auftrag ausführt. Hierbei besteht das Ziel darin, die maximale Summe der Werte $k_{i\pi(i)}$ für alle $i \in S$ über alle zulässigen Permutationen von S zu erreichen. Die in diesem Wege gewonnene maximale Summe bezeichnet man als den Wert der Koalition S . Definieren wir nun Π_S als die Menge aller Permutationen $\pi \in \Pi_N$, wobei $\pi(i) = i$ für alle $i \in N \setminus S$ ist, so ist der Wert jeder Koalition S eines Permutationsspiels definiert durch (vgl. Curiel [28], S. 56, und Bilbao [7], S. 12)

$$v(S) := \max_{\pi \in \Pi_S} \sum_{i \in S} k_{i\pi(i)} \quad \text{für alle } S \subseteq N. \quad (4.24)$$

Eine Möglichkeit, die charakteristische Funktion eines Permutationsspiels zu bestimmen, besteht darin, dass wir für jede Koalition $S \subseteq N$ das folgende binäre lineare Programm lösen. Der Wert der Koalition S entspricht dem optimalen Zielfunktionswert dieses linearen Programms (vgl. dazu Curiel [28], S. 56):

$$\begin{aligned} & \max \sum_{i \in S} \sum_{j \in S} k_{ij} x_{ij} \\ & \text{unter den Nebenbedingungen} \\ & \sum_{j \in S} x_{ij} = 1 \quad \text{für alle } i \in S \\ & \sum_{i \in S} x_{ij} = 1 \quad \text{für alle } j \in S \\ & x_{ij} \in \{0, 1\} \quad \text{für alle } i, j \in S. \end{aligned} \quad (4.25)$$

Curiel [28], S. 56, zeigt einen interessanten Zusammenhang zwischen den Zuordnungsspielen und den Permutationsspielen auf. Sie weist nach, dass jedes Zuordnungsspiel mit der Spielermenge $N := \bar{M} \cup \bar{P}$ und der Gewinnmatrix $\mathbf{A} = (a_{ij})_{\substack{i \in \bar{M} \\ j \in \bar{P}}}$ zu einem Permutationsspiel mit der selben Spielermenge und der Matrix \mathbf{K} mit den Elementen

$$k_{ij} := \begin{cases} a_{ij}, & \text{falls } i \in \bar{M}, j \in \bar{P} \\ 0, & \text{sonst} \end{cases} \quad (4.26)$$

4 Ausgewählte spezielle kooperative Spiele

äquivalent ist. Solymosi et al. [133] nutzen diesen Tatbestand aus, um einen spezifischen Algorithmus zur Berechnung des Nukleolus der sog. *zyklischen Permutationsspiele* zu entwickeln.¹² Die Autoren nennen ein Permutationsspiel *zyklisch*, falls eine sog. zyklische N -Permutation

$$\pi(i) = \begin{cases} i + 1, & \text{falls } i \neq n \\ 1, & \text{falls } i = n \end{cases} \quad (4.27)$$

existiert, für welche

$$v(N) = \sum_{i \in N} k_{i\pi(i)} \quad (4.28)$$

der maximal realisierbare Wert ist. Eine zyklische N -Permutation bedeutet, dass kein Spieler $i \in N$ seinen Auftrag mit seiner eigenen Maschine ausführt. Bei den zyklischen Permutationsspielen existiert also eine optimale Zuweisung aller Aufträge zu allen Maschinen, welche zu einer zyklischen Permutation aller Spieler führt. Solymosi et al. [133] zeigen, dass man den Nukleolus zyklischer Permutationsspiele mit Hilfe eines modifizierten Algorithmus von Solymosi/Raghavan [132] in $\mathcal{O}(n^4)$ elementaren Schritten berechnen kann.

Das Nachbarspiel

Das sog. *Nachbarspiel* (engl. *neighbour games*) ist ein weiterer interessanter Spezialfall eines Zuordnungsspiels, der in der Literatur diskutiert wird. Dieser Spieltyp wurde von Klijn et al. [62] in die Literatur eingeführt. Er ist dadurch gekennzeichnet, dass nur bestimmte Paare von Spielern eine Koalition bilden können. Beispielhaft (vgl. Hammers et al. [50], S. 2) nehmen wir an, dass ein Fluss durch n Regionen fließt, wobei die Regionen i und $i + 1$, $i = 1, 2, \dots, n - 1$, aneinander grenzen. Damit der Fluss zum Transport der Waren zwischen den Regionen genutzt werden kann, sollen Häfen gebaut werden. Wir setzen voraus, dass sich jede Region aus finanziellen Gründen höchstens einen Hafen leisten kann. Damit bleiben jeder Region zwei Optionen: Sie kann entweder ihren eigenen Hafen errichten oder sie kann mit einer ihrer Nachbarregionen an der Grenze einen gemeinsamen Hafen bauen, um Kosten

¹²Wir finden in der Literatur kein spezifisches Verfahren zur Berechnung des Nukleolus von allgemeinen Permutationsspielen.

4.2 Zuordnungsspiele und verwandte kooperative Spiele

zu sparen. Somit können in Nachbarspielen nur diejenigen Spieler miteinander kooperieren, die (z. B. räumlich) benachbart sind. Wir nehmen an, dass die Regionen daran interessiert sind, maximale Kosteneinsparungen zu realisieren.

Um das Nachbarspiel formal zu definieren, bringen wir die Spieler des Spiels $N = \{1, 2, \dots, n\}$ in eine Reihenfolge. Es sei $\pi : N \rightarrow N$ eine Permutation von N . Für $i, r \in N$ besagt $\pi(i) = r$, dass der Spieler i in der Reihenfolge die Position r besitzt. Zwei Spieler i und j nennen wir Nachbarn, falls $|\pi(i) - \pi(j)| = 1$ ist. Für Nachbarspiele verwenden wir die Notation (i, j) , falls die Spieler i und j Nachbarn sind, so dass $\pi(j) = \pi(i) + 1$ ist. Für alle Nachbarspieler (i, j) sei die Kosteneinsparung $a_{ij} \geq 0$ gegeben. Für eine Koalition $S \subseteq N$ definieren wir μ als die (potentiell leere) Menge der disjunkten Indexpaare (i, j) von Nachbarn $i, j \in S$. Da auf diese Weise jedem Spieler i nur einer seiner Nachbarn zugeordnet werden kann, ist die Menge μ ein Matching über der Menge S . Es sei nun $M(S)$ die Menge aller möglichen Matchings μ für eine Koalition $S \subseteq N$. Damit können wir die charakteristische Funktion eines Nachbarspiels wie folgt definieren (vgl. Klijn et al. [62], S. 4):

$$v(S) := \max \left\{ \sum_{(i,j) \in \mu} a_{ij} \mid \mu \in M(S) \right\} \quad \text{für alle } S \subseteq N. \quad (4.29)$$

Zur Illustration betrachten wir das oben erläuterte Beispiel der Nachbarregionen. Für dieses Beispiel nehmen wir an, dass $N = \{1, 2, 3, 4\}$ und $\pi(i) = i$ für alle $i \in N$ sind. Demnach lauten die Nachbarpaare $(1, 2)$, $(2, 3)$ und $(3, 4)$. Folglich sind die Werte aller anderen Paare Null. Nehmen wir an, dass $a_{12} = 10$, $a_{23} = 20$ und $a_{34} = 30$ (in CHF Mio.) gilt. Für diesen Fall haben wir die charakteristische Funktion des dazugehörigen Nachbarspiels in der Tabelle 4.2 wiedergegeben.

Hamers et al. [50] schlagen einen speziellen Algorithmus vor, der zur Berechnung des Nukleolus eines Nachbarspiels mit n Spielern $\mathcal{O}(n^2)$ elementare Operationen erfordert. Da die Nachbarspiele eine besondere Form der Zuordnungsspiele sind und die Zuordnungsspiele stets einen nicht-leeren Kern besitzen, ist auch der Kern jedes Nachbarspiels nicht-leer. Somit ziehen Hamers et al. [50] zur Berechnung des Nukleolus eines Nachbarspiels nur diejenigen Zweierkoalitionen heran, die Nachbarpaare beinhalten, da nur diese essentiell sind.

4 Ausgewählte spezielle kooperative Spiele

Tabelle 4.2: Charakteristische Funktion eines Nachbarspiels

S	$v(S)$
$\{1, 2\}$	10
$\{2, 3\}$	20
$\{3, 4\}$	30
$\{1, 2, 3\}$	20
$\{1, 2, 4\}$	10
$\{1, 3, 4\}$	30
$\{2, 3, 4\}$	30
$\{1, 2, 3, 4\}$	40

Quelle: Klijn et al. [62], S. 5.

4.3 Zusammenhängende balancierte kooperative Spiele

Es sei $\Gamma = (N, v)$ ein kooperatives Spiel mit der Spielermenge $N = \{1, 2, \dots, n\}$, wobei die Spieler in der Reihenfolge $1, 2, \dots, n$ geordnet seien. Beinhaltet eine Koalition S jeden Spieler k , der zwischen zwei Mitgliedern i und j der Koalition S liegt, so nennen wir diese Koalition eine *Intervallkoalition*. Eine Koalition S heisst also eine Intervallkoalition, falls für alle $i, j \in S$ mit $i \leq j$ jeder Spieler k , $i \leq k \leq j$, ebenfalls ein Mitglied dieser Koalition ist. Ein kooperatives Spiel bezeichnen wir als *zusammenhängend*, falls in diesem Spiel jede essentielle Koalition eine Intervallkoalition ist. Zur Illustration betrachten wir das folgende Beispiel (Raghavan [108], S. 224):

Ein Reparatuer soll insgesamt n Kunden besuchen und anschliessend wieder zu seinem Standort zurückkehren. Die Reisekosten des Reparateurs werden von den Kunden gemeinsam getragen. Die Menge der Kunden sei $N = \{1, 2, \dots, n\}$. Die Reihenfolge, in der der Reparatuer seine Kunden besuchen wird, ist wie folgt vorgegeben:

$$0 \rightarrow 1 \rightarrow 2 \rightarrow \dots \rightarrow n \rightarrow 0.$$

Darin ist 0 der Standort des Reparateurs. Wir nehmen an, dass die Reisekosten des Reparateurs von i nach j bekannt sind und $c_{ij} \geq 0$ Geldeinheiten betragen. Die gesamten Reisekosten des Reparateurs sind somit $C(N) := \sum_{i=0}^n c_{i(i+1)}$, wobei wir $c_{n(n+1)} := c_{n0}$ setzen. Die oben vorgegebene Reihenfolge könnte z. B. derart festgelegt

4.3 Zusammenhängende balancierte kooperative Spiele

worden sein, dass die gesamten Reisekosten $C(N)$ minimal werden. Das zu lösende Problem besteht in diesem Fall in der Zuteilung der gesamten Kosten $C(N)$ auf die Kunden. Um diese Situation als ein kooperatives Spiel modellieren zu können, betrachten wir eine Menge von Kunden $S = \{i_1, i_2, \dots, i_k\}$ mit $i_1 < i_2 < \dots < i_k$. Es sei $C(S)$ die Gesamtsumme der Reisekosten des Reparateurs, der die Kunden in der Menge S in der folgenden vorgegebenen Reihenfolge besucht:

$$0 \rightarrow i_1 \rightarrow i_2 \rightarrow \dots \rightarrow i_k \rightarrow 0.$$

Der Wert einer Koalition $S \subseteq N$ wird in der Literatur durch

$$v^*(S) := C(N) - C(N \setminus S) \quad (4.30)$$

definiert. Diesen Wert können wir als den Anteil der Koalition $S \subseteq N$ an den gesamten entstandenen Kosten interpretieren. Ein möglicher Finanzierungsvorschlag $\mathbf{x} = (x_1, x_2, \dots, x_n)$ sollte zunächst kostendeckend sein, so dass $x(N) := \sum_{i \in N} x_i = C(N)$ erfüllt ist. Damit jede Koalition S mit diesem Vorschlag einverstanden ist, wird man zudem verlangen, dass für alle Koalitionen $S \subseteq N$ die Ungleichung $x(S) := \sum_{i \in S} x_i \leq C(S)$ gilt, das Spiel also einen nicht-leeren Kern besitzt. Derks/Kuipers [31] zeigen, dass mindestens ein solcher Zuteilungsvektor \mathbf{x} genau dann existiert, wenn die Ungleichung

$$C(N) \leq C(S) + C(N \setminus S) \quad (4.31)$$

für alle Koalitionen S erfüllt ist. Die Ungleichung (4.31) besagt, dass der Kern des Spiels nicht-leer ist. Somit ist das Spiel insbesondere balanciert. Falls für ein zusammenhängendes kooperatives Spiel die Ungleichung (4.31) gilt, nennt man es ein *zusammenhängendes balanciertes* Spiel.

In Solymosi et al. [131] und Kuipers et al. [70] finden wir Algorithmen zur Berechnung des Nukleolus eines zusammenhängenden balancierten kooperativen Spiels. Die Komplexität beider Algorithmen ist $\mathcal{O}(n^4)$, wobei n die Anzahl der Spieler ist. Der Algorithmus von Solymosi et al. [131] beruht auf dem Algorithmus von Maschler et al. [82] und berechnet ebenfalls im Wege der Lösung einer Folge linearer Programme

4 Ausgewählte spezielle kooperative Spiele

den Nukleolus. Hierzu ziehen Solymosi et al. [131] für die Berechnung des Nukleolus zusammenhängender kooperativer Spiele nur die Intervallkoalitionen heran. Der Algorithmus startet mit einer Kernallokation, die mit Hilfe der in Derks/Kuipers [31] vorgeschlagenen Methode bestimmt wird. Der Algorithmus von Kuipers et al. [70] beruht ebenfalls auf dem Verfahren von Maschler et al. [82], wobei im Algorithmus von Kuipers et al. [70] keine linearen Programme gelöst werden und nur die Intervallkoalitionen für die Berechnung des Nukleolus relevant sind.

4.4 Bilaterale Marktspiele mit komplementären Gütern

Die Spielermenge N eines kooperativen Spiels sei die Vereinigungsmenge zweier disjunkten Mengen P und Q . Wir setzen voraus, dass P eine Menge von insgesamt p Produzenten ist, die jeweils mit einer Maschine ausgestattet sind und dieselbe Technologie verwenden. Es sei Q die Menge der insgesamt q Arbeitnehmer. Wir nehmen an, dass die Maschinen und die Arbeitskräfte komplementäre Inputs in einem Produktionsprozess sind. In diesem Fall definieren wir den Wert einer Koalition S von Produzenten und Arbeitnehmern durch die von dieser Koalition erbrachte Outputmenge. Diese Produktionsmenge ist ein Resultat der Koalition der $|S \cap P|$ Produzenten und der $|S \cap Q|$ Arbeitnehmer. Es sei λ der Inputkoeffizient für die Arbeitnehmer. Wir nehmen an, dass der Inputkoeffizient für die Produzenten eins beträgt. Folglich drückt λ die Faktorintensität aus. In diesem Fall lautet die Produktionsfunktion wie folgt (vgl. Legros [74], S. 1):

$$v(S) := \min \{|S \cap P|, \lambda \cdot |S \cap Q|\} \text{ für alle } S \subset P \cup Q, \lambda \in \mathbb{R}_{\geq 0}. \quad (4.32)$$

Jedes kooperative Spiel, dessen charakteristische Funktion durch (4.32) definiert ist, nennt man *bilaterales Marktspiel mit komplementären Gütern*. Legros [74], S. 2, zeigt, dass der Kern dieser Spiele nicht-leer ist. Zur Berechnung des Nukleolus eines kooperativen bilateralen Marktspiels mit komplementären Gütern gibt Legros [74] geschlossene Formelausdrücke an, an Hand derer wir den Nukleolus berechnen können.

4.5 Peer-Group- und Veto-Rich-Spiele

Wir betrachten eine Organisation, in der jedes Mitglied i durch eine vorgegebene Hierarchie eine direkte oder indirekte Verbindung zum Leiter der Organisation hat. Hierbei erfolgt die indirekte Verbindung mit Hilfe mindestens eines anderen Mitglieds, das in der Hierarchie zwischen dem Leiter und dem Mitglied i steht. Jedes Mitglied i besitzt in der Hierarchie ökonomische Optionen, die zu potentielltem Gewinn führen können. Der Gewinn eines Mitglieds i wird nur dann realisiert, wenn der Leiter die Option genehmigt und alle anderen Mitglieder, die in der Hierarchie zwischen i und dem Leiter der Organisation stehen, mit dem Mitglied i kooperieren. Wir bezeichnen die Menge aller Mitglieder der Organisation einschliesslich des Leiters der Organisation als die Spielermenge. In dieser Situation ist für ein Mitglied i diejenige Koalition wichtig, die neben diesem Mitglied den Leiter der Organisation und alle in der Hierarchie dazwischen stehenden Organisationsmitglieder einschliesst. Denn nur diese Koalition ermöglicht dem Mitglied i , seinen Gewinn zu realisieren. Eine solche Koalition heisst *Peer-Group*. Brânzei et al. [18] beschreiben die Hierarchie der Spieler durch einen *Baum*, dessen Wurzel den Leiter der Organisation repräsentiert. Der Rest der Spieler wird durch die anderen Knoten des Baumes illustriert. Das kooperative Spiel, das aus einer Peer-Group-Situation entsteht, nennen wir *Peer-Group-Spiel*. Dieser Spieltyp wurde von Brânzei et al. [18] in die Literatur eingeführt. Die Spielermenge eines Peer-Group-Spiels entspricht der Knotenmenge des Baumes.

Gegeben seien die Spielermenge $N = \{1, 2, \dots, n\}$ und der knotenbewertete Baum T mit der Knotenmenge N und dem (Knoten-)Gewichtsvektor $\mathbf{w} = (w_1, w_2, \dots, w_n) \in \mathbb{R}_{\geq 0}^n$. Im obigen Beispiel der Peer-Group-Situation entspricht w_i dem vom Mitglied i potentiell ausgelösten Gewinn. Wir nehmen an, dass der Knoten 1 die Wurzel des Baumes ist. Wir schreiben $j \preceq i$, falls der Knoten j auf dem Pfad von der Wurzel 1 zum Knoten i liegt. Um alle Peer-Group-Situationen definieren zu können, verwenden wir die folgende Notation:

$$P(i) := \{j \in N : j \preceq i\} \text{ für alle } i \in N.$$

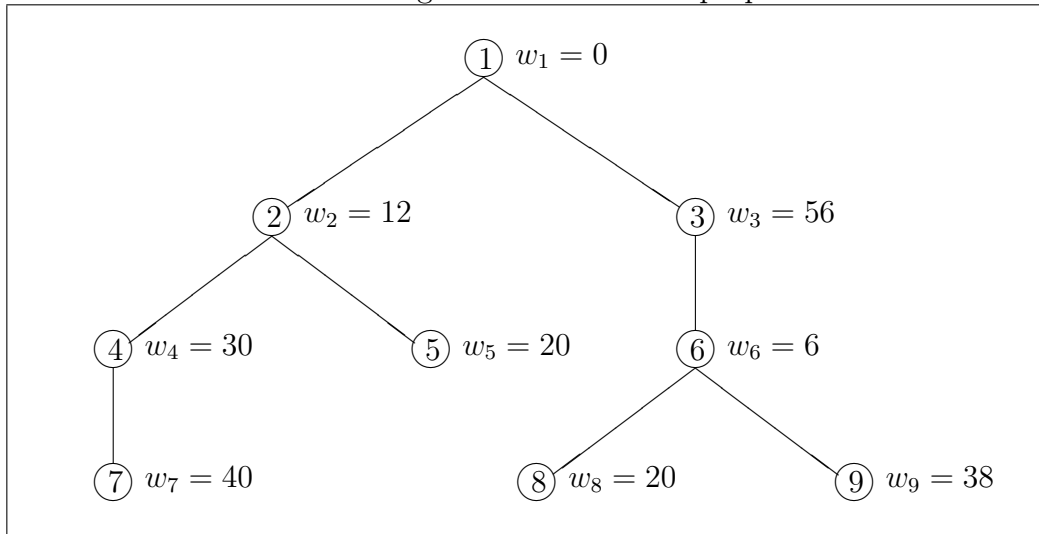
4 Ausgewählte spezielle kooperative Spiele

Für einen Spieler i beinhaltet also die Menge $P(i)$ alle vorgesetzten Spieler, mit denen i eine Peer-Group bilden kann. Kommt nun für einen Spieler i eine Peer-Group zustande, so können neben w_i alle anderen Gewinne w_j , $j \preceq i$, realisiert werden. Deshalb zieht man in der Literatur bei der Definition des Wertes jeder Koalition S jede Menge $P(i) \subseteq S$ heran (vgl. Brânzei et al. [18]):

$$v(S) := \sum_{i: P(i) \subseteq S} w_i \quad \text{für alle } S \subseteq N. \quad (4.33)$$

Wir betrachten das in Abbildung 4.1 durch den Baum dargestellte Peer-Group-Spiel mit der Spielermenge $N = \{1, 2, \dots, 9\}$ und dem Gewichtsvektor $\mathbf{w} = (w_1, w_2, \dots, w_9) = (0, 12, 56, 30, 20, 6, 40, 20, 38)$.

Abbildung 4.1: Ein Peer-Group-Spiel



Quelle: Brânzei et al. [20], S. 6.

In diesem Fall sehen die Mengen $P(i)$, $i = 1, 2, \dots, 9$, wie folgt aus:

$$\begin{array}{lll} P(1) = \{1\}, & P(4) = \{1, 2, 4\}, & P(7) = \{1, 2, 4, 7\}, \\ P(2) = \{1, 2\}, & P(5) = \{1, 2, 5\}, & P(8) = \{1, 3, 6, 8\}, \\ P(3) = \{1, 3\}, & P(6) = \{1, 3, 6\}, & P(9) = \{1, 3, 6, 9\}. \end{array}$$

Die Werte der einzelnen Koalitionen können wir mit Hilfe der Definitionsgleichung (4.33) bestimmen. Wir geben exemplarisch die Werte der Koalitionen $\{1, 2, 7\}$,

$\{1, 3, 6\}$, $\{1, 6, 9\}$ und $\{2, 5\}$ an:

$$\begin{aligned} v(\{1, 2, 7\}) &= w_1 + w_2 = 12, \\ v(\{1, 3, 6\}) &= w_1 + w_3 + w_6 = 62, \\ v(\{1, 6, 9\}) &= w_1 = 0, \\ v(\{2, 5\}) &= 0. \end{aligned} \tag{4.34}$$

Brânzei et al. [18] zeigen verschiedene Anwendungen von Peer-Group-Spielen im Operations Research und in den Wirtschaftswissenschaften auf. Zur Illustration stellen wir im Weiteren den Fall der Vickrey-Auktion vor. Bei dieser Auktionsform hat der Verkäufer eines Objektes einen Reservationspreis r , welcher der Mindestpreis ist, den er von den potentiellen Käufern (Bieter) mindestens abverlangt. Wir nehmen an, dass es n Bieter (Spieler) gibt. Jeder Spieler $i \in N = \{1, 2, \dots, n\}$ unterbreitet in einem versiegelten Umschlag ein Gebot b_i . Nachdem die Umschläge geöffnet worden sind, erhält der Bieter mit dem höchsten Gebot den Zuschlag, und zwar zur Höhe des zweithöchsten Gebots. Es sei w_i der Wert des Objektes für Spieler i . Man kann zeigen, dass für jeden Spieler $i \in N$ das Gebot $b_i = w_i$ eine dominante Strategie ist (vgl. z. B. Varian [142], S. 299).

Um diese Situation als ein Peer-Group-Spiel abzubilden, nehmen wir an, dass Kooperation zwischen den Spielern möglich ist. Zudem unterstellen wir $w_1 > w_2 > \dots > w_n \geq r$.¹³ Wir halten zunächst fest, dass bei keiner Kooperation von Spielern der Spieler 1 den Zuschlag erhalten wird, weil er das Objekt am höchsten bewertet und seine dominante Strategie $b_1 = w_1$ wählt. In diesem Fall ist $v(\{1\}) = w_1 - w_2$, weil der Spieler 1 die Auktion zum Preis w_2 gewinnt. Für alle übrigen Spieler $i > 1$ gilt $v(\{i\}) = 0$. Wenn aber alle Spieler miteinander kooperieren, dann besteht die dominante Strategie der grossen Koalition darin, dass der Spieler 1 das Gebot $b_1 = w_1$ abgibt, während alle anderen Spieler $i > 1$ mit dem Reservationspreis $b_i = r$ bieten. Kommt nun lediglich eine Koalition $S \subset N$ zustande, so nehmen wir an, dass der Spieler $h \in S$ das Objekt zum Preis w_h am höchsten bewertet. Dieser Spieler gibt in

¹³Diese Annahme ist der Übersichtlichkeit halber getroffen worden und bildet somit keine Beschränkung der Allgemeinheit. Die nachfolgenden Aussagen können auch für den Fall $w_1 \geq w_2 \geq \dots > w_n \geq r$ gemacht werden.

4 Ausgewählte spezielle kooperative Spiele

diesem Fall das Gebot w_h ab. Die anderen Spieler der Koalition S unterbreiten das Gebot r . Das ist die dominante Strategie dieser Koalition. Wenn wir annehmen, dass die übrigen Spieler ebenfalls ihre dominante Strategie spielen, können wir zwischen zwei Fällen unterscheiden:

- Falls der Spieler 1 der Koalition S nicht angehört, gilt $h \neq 1$. In diesem Fall erhält der Spieler 1 (oder die Koalition, zu der Spieler 1 gehört) das Objekt. Demnach ist der Wert $v(S)$ der Koalition S Null.
- Falls der Spieler 1 ein Mitglied der Koalition S ist, entspricht w_1 dem höchsten Gebot dieser Koalition. In diesem Fall gilt $b_i = r$ für alle $i \in S \setminus \{1\}$. Nehmen wir an, dass das zweithöchste Gebot w_k ist. Der Wert der Koalition S lautet dann $v(S) = w_1 - w_k$.

Für die Berechnung des Nukleolus solcher Spiele schlagen Brânzei et al. [20] einen spezifischen Algorithmus vor. Die Autoren weisen nach, dass die Berechnung des Nukleolus eines Peer-Group-Spiels durch Heranziehung von sog. *stark essentiellen* Koalitionen erfolgen kann. Sie zeigen, dass die Menge aller stark essentiellen Koalitionen aus allen Koalitionen besteht, mit Hilfe derer man den Kern eines Spiels gerade noch bestimmen kann. Folglich determinieren diese Koalitionen bei einem Spiel mit nicht-leerem Kern gleichzeitig den Nukleolus des Spiels. Brânzei et al. [20] legen dar, dass jedes Peer-Group-Spiel mit n Spielern insgesamt $2n - 1$ stark essentielle Koalitionen besitzt und der Nukleolus des Spiels in $O(n^2)$ elementaren Operationen berechnet werden kann.

Für einfache Peer-Group-Spiele, bei denen der Baum T lediglich aus einer einzigen Kantenfolge besteht, so dass $P(1) = \{1\}, P(2) = \{1, 2\}, \dots, P(n) = \{1, 2, \dots, n\}$ gilt, schlagen Brânzei et al. [18] einen alternativen Algorithmus vor. Dieser Algorithmus gewinnt den Nukleolus als Lösung eines Systems mit n linearen Gleichungen mit n Unbekannten.

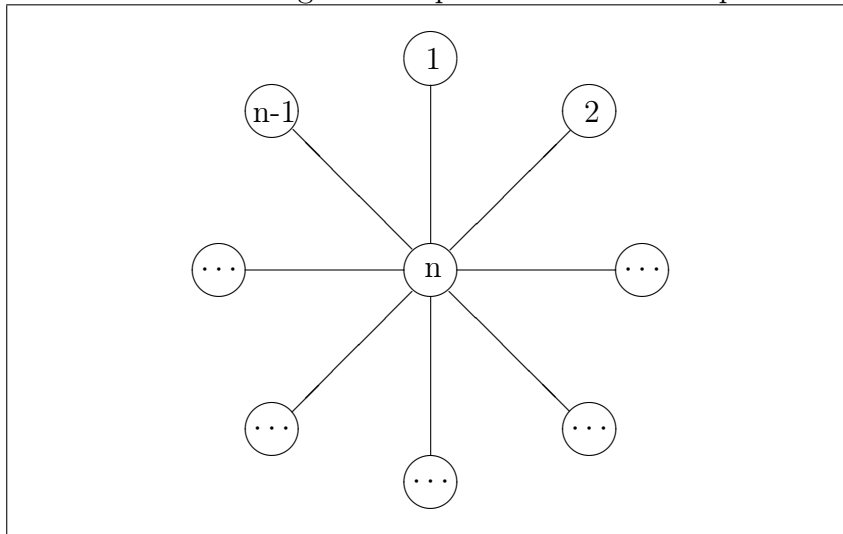
Das Veto-Rich-Spiel

Das sog. *Veto-Rich-Spiel* (*Veto-kontrolliertes Spiel*) ist ein spezielles kooperatives Spiel, das in einem engen Zusammenhang zum Peer-Group-Spiel steht. In einem

4.5 Peer-Group- und Veto-Rich-Spiele

kooperativen Spiel $\Gamma = (N, v)$ heisst ein Spieler $i \in N$ *Vetospieler*, falls $v(S) = 0$ für alle $S \subset N$ mit $i \notin S$ gilt. Solange also eine Koalition den Vetospieler i nicht enthält, ist der Wert dieser Koalition Null. Ein kooperatives Spiel nennt man *Veto-Rich-Spiel*, wenn das Spiel mindestens einen Vetospieler besitzt. Die Abbildung 4.2 zeigt die Graphenstruktur eines Veto-Rich-Spiels mit dem alleinigen Veto-Spieler n .

Abbildung 4.2: Graph eines Veto-Rich-Spiels



Quelle: Kuipers et al. [70], S. 557.

Zum Beispiel können wir einen Auktionsmarkt als Veto-Rich-Spiel interpretieren. In einem solchen Markt ist nämlich der Wert einer Koalition Null, solange sie den Verkäufer nicht enthält (vgl. Arin/Feltkamp [2], S. 62 und S. 65). Zur Berechnung des Nukleolus des Veto-Rich-Spiels entwickeln Arin/Feltkamp [2] einen Algorithmus. Während die meisten Algorithmen zur Berechnung des Nukleolus die Lösung linearer Programme erfordern, gewinnt der Algorithmus von Arin/Feltkamp den Nukleolus, ohne dass man dafür ein lineares Programm lösen muss. Zudem ist der Algorithmus auch auf Veto-Rich-Spiele mit einem leeren Kern anwendbar. Demgegenüber schlagen Kuipers et al. [70] einen anderen Algorithmus vor, der den Nukleolus eines Veto-Rich-Spiels mit nicht-leerem Kern berechnet. Hierfür nutzen die Autoren den Tatbestand aus, dass im Veto-Rich-Spiel nur die Einerkoalitionen sowie alle Zweierkoalitionen,

4 Ausgewählte spezielle kooperative Spiele

denen der Veto-Spieler angehört, essentiell sind. Für die Berechnung des Nukleolus ziehen die Autoren nur diese essentiellen Koalitionen heran.

4.6 Kooperative Spiele über kostenminimale spannende Bäume

Wir beginnen mit einem anschaulichen Beispiel (vgl. Domschke/Drexel [34], S. 71). Hierzu betrachten wir eine Anzahl von Ortschaften (z. B. Städte), für die ein Versorgungsnetz (z. B. ein Netz von Wasser-, Gas- oder Telefonleitungen) gebaut werden soll. Dabei sollen alle Ortschaften direkt oder indirekt durch das Versorgungsnetz miteinander verbunden werden. Verzweigungspunkte des Netzes befinden sich nur in den Ortschaften. Die Kosten jeder möglichen Verbindung zweier Ortschaften seien bekannt. Wir suchen ein Versorgungsnetz, dessen Gesamtkosten minimal sind. Diese Aufgabe nennt man in der Literatur das Problem des (kosten-)minimalen spannenden Baumes.

Um dieses Problem graphentheoretisch zu modellieren, gehen wir von einem kantenbewerteten ungerichteten Graphen G aus, dessen Knoten die Ortschaften darstellen. Jede mögliche direkte Verbindung zwischen zwei Ortschaften wird durch eine Kante des Graphen G repräsentiert, wobei das Gewicht der Kante den Baukosten dieser Verbindung entspricht. Das hier gesuchte Versorgungsnetz ist ein (kosten-)minimaler spannender Baum (engl. minimum cost spanning tree)¹⁴ des Graphen G . Dieses Problem können wir als ein kooperatives Spiel interpretieren. Zu diesem Zweck nehmen wir an, dass die Finanzierung des Versorgungsnetzes durch die Ortschaften kostendeckend sein soll. In diesem Fall stellt sich die Frage nach der Aufteilung der minimalen Gesamtkosten auf die Ortschaften. Claus/Kleitman [26] haben diese Aufgabe in die Literatur eingeführt. Diese Aufgabe zählt zur Klasse der sog. Kostenallokationsprobleme. Eine erste spieltheoretische Behandlung dieser Aufgabe stammt von Bird [9]. Das Problem der Kostenallokation über minimale spannende Bäume wurde von Granot/Huberman [47] als ein kooperatives Spiel in charakteristischer

¹⁴Für die Bestimmung eines minimalen spannenden Baumes gibt es in der Literatur verschiedene effiziente Algorithmen, auf die wir hier nicht näher eingehen werden. Vgl. dazu z. B. Kruskal [69], Prim [107] und Dijkstra [33].

4.6 Kooperative Spiele über kostenminimale spannende Bäume

Form ausgedrückt. Gemäss dieser Formulierung interpretieren wir die Ortschaften, also die Knoten des Baumes, als Spieler, wobei $N = \{1, 2, \dots, n\}$ die Spielermenge ist. Wir nehmen an, dass das Versorgungsnetz eine Quelle 0 hat und fügen diese der Knotenmenge N hinzu. Es sei nun $G = (N \cup \{0\}, E)$ ein zusammenhängender ungerichteter Graph, der das Versorgungsnetz darstellt. Hierbei sei E die Menge der Kanten $e_{ij} := (i, j)$, also der Verbindungen zwischen jeweils zwei Ortschaften i und j . Es sei $c : E \rightarrow \mathbb{R}_{\geq 0}$ eine gegebene Gewichtungsfunktion. Die Kosten der Verbindung zwischen zwei Knoten i und j bezeichnen wir als c_{ij} , wobei $c_{ij} = c_{ji}$ gilt. Die zum Problem der Kostenallokation bezüglich minimaler spannender Bäume gehörende charakteristische Funktion ordnet jeder Koalition $S \subseteq N$ die Kosten eines Teilgraphen zu, der alle Elemente von S zur Quelle 0 mit minimalen Kosten verbindet. Wir unterstellen nun, dass $G' := (S \cup \{0\}, E_S)$ ein Teilgraph von G ist. Wir erhalten die charakteristische Funktion eines kostenminimalen spannenden Baumes wie folgt:

$$c(S) := \min \left\{ \sum_{e_{ij} \in T_S} c_{ij} : T_S \text{ ist ein spannender Baum von } G' \right\}. \quad (4.35)$$

Zur Illustration geben wir ein Beispiel an. Es sei $N = \{1, 2, 3\}$. Der Graph wird in Abbildung 4.3 wiedergegeben.

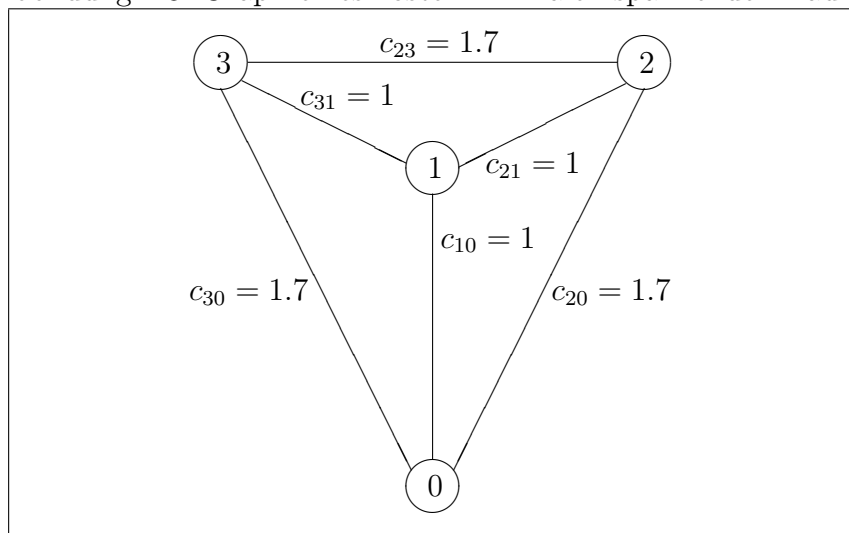
Durch Anwendung der Definitionsgleichung (4.35) erhalten wir die folgenden Koalitionswerte:

$$\begin{array}{lll} c(\{1\}) = 1, & c(\{1, 2\}) = 2, & c(\{1, 2, 3\}) = 3. \\ c(\{2\}) = 1.7, & c(\{1, 3\}) = 2, & \\ c(\{3\}) = 1.7, & c(\{2, 3\}) = 3.4, & \end{array}$$

Granot/Huberman [47] zeigen, dass jedes kooperative Spiel bezüglich kostenminimaler spannender Bäume einen nicht-leeren Kern besitzt. Da nun bei der Bestimmung des Nukleolus kooperativer Spiele mit nicht-leerem Kern nur die essentiellen Koalitionen eine zentrale Rolle spielen, legen die Autoren dar, wie diese Koalitionen in einem Spiel bezüglich kostenminimaler spannender Bäume bestimmt werden können. Die Berechnung des Nukleolus erfolgt dann z. B. mit Hilfe des Verfahrens von

4 Ausgewählte spezielle kooperative Spiele

Abbildung 4.3: Graph eines kostenminimalen spannenden Baumes



Quelle: Granot/Huberman [47], S. 3.

Maschler et al. [82], wobei nur essentielle Koalitionen in Betracht gezogen werden.

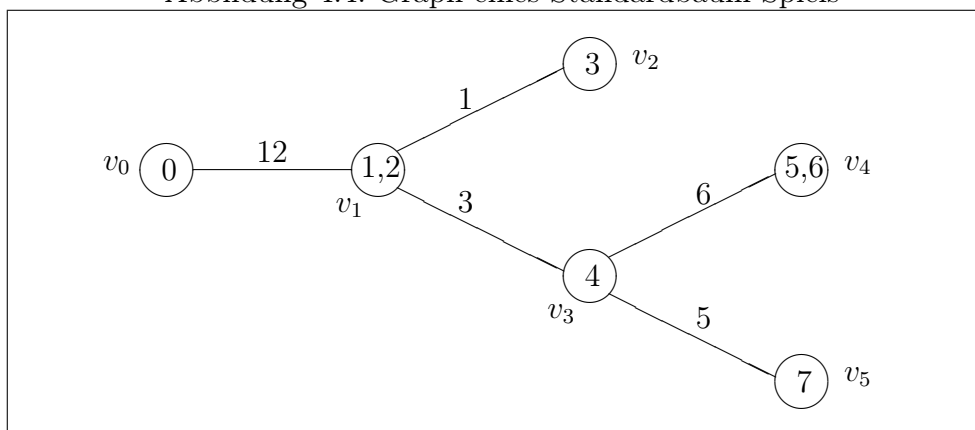
Das Standardbaum-Spiel

Eine in der Literatur im Zusammenhang mit der Berechnung des Nukleolus diskutierte spezielle Form der kooperativen Spiele bezüglich kostenminimaler spannender Bäume ist das sog. *Standardbaum-Spiel* (engl. *standard tree game*). Zum Beispiel stellt die folgende Situation ein Standardbaum-Spiel dar (vgl. Raghavan [108], S. 232). Wir können uns ein Stadtzentrum vorstellen, in das täglich über eine Autobahn Einwohner anderer Städte fahren. Jede Stadt ist durch diese Autobahn mit dem erwähnten Stadtzentrum verbunden, wobei es für Nachbarstädte, die relativ nahe beieinanderliegen, nur einen Autobahnanschluss gibt. Die Unterhaltskosten sowohl für die gesamte Autobahn wie auch für die Teile zwischen zwei Anschlüssen seien bekannt. Zudem seien die gesamten Unterhaltskosten der Autobahn von den Städten zu finanzieren. Die Aufteilung der Gesamtkosten auf die Städte ist somit eine Frage, die mit Hilfe der Theorie der kooperativen Spiele beantwortet werden kann. Jede derartige Situation können wir durch einen Graphen repräsentieren. Zu diesem Zweck nehmen wir an, dass ein Baum mit einer Wurzel 0, der Knotenmenge

4.6 Kooperative Spiele über kostenminimale spannende Bäume

V und der Kantenmenge E gegeben ist, wobei die Wurzel 0 der Endknoten nur einer Kante ist. Es sei $N = \{1, 2, \dots, n\}$ die Menge der Spieler. Jeder Knoten des Baumes beinhaltet mindestens einen Spieler. Wir nehmen an, dass einige Spieler zusammen den selben Knoten besetzen dürfen, wobei jeder Spieler an genau einem Knoten des Baumes auftreten darf. Wenn zudem jede Kante des Baumes ein nicht-negatives Gewicht hat, wird dieses Spiel *Standardbaum-Spiel* genannt. Im obigen Beispiel besteht die Spielermenge aus den Städten. Das Stadtzentrum wird durch die Wurzel 0 des Baumes repräsentiert. Ferner entsprechen die Knoten des Baumes den Autobahnanschlüssen. Die Kanten des Baumes werden durch die Autobahnabschnitte zwischen zwei benachbarten (adjazenten) Knoten dargestellt. Des Weiteren entsprechen die Gewichte der Kanten den Unterhaltskosten der Autobahnabschnitten. Demnach gibt der Wert einer Koalition $S \subset N$ die minimalen Kosten an, die aus der Verbindung aller Mitglieder der Koalition S zur Wurzel des Baumes entstehen.

Abbildung 4.4: Graph eines Standardbaum-Spiels



Quelle: Granot et al. [48], S. 230.

Der Graph (Baum) in der Abbildung 4.4 gibt ein Beispiel eines Standardbaum-Spiels wieder. In diesem Spiel sind die Spielermenge $N = \{1, 2, \dots, 7\}$ und die Knotenmenge $V = \{v_0, v_1, v_2, v_3, v_4, v_5\}$, wobei die Spieler 1 und 2 gemeinsam an dem Knoten v_1 und die Spieler 5 und 6 gemeinsam an dem Knoten v_4 abgebildet sind. Die Gewichte werden über den Kanten wiedergegeben.

4 Ausgewählte spezielle kooperative Spiele

Der erste Algorithmus zur Berechnung des Nukleolus des Standardbaum-Spiels stammt von Megiddo [85]. Der Algorithmus erfordert $\mathcal{O}(n^3)$ elementare Operationen für die Ermittlung des Nukleolus eines Standardbaum-Spiels, wobei n die Anzahl der Knoten des Baumes ist. Dieser Algorithmus wurde dann von Galil [43] verbessert, sodass die Berechnung des Nukleolus in $\mathcal{O}(n \log n)$ elementaren Operationen erfolgt.

Ein anderer Algorithmus zur Berechnung des Nukleolus des Standardbaum-Spiels wurde von Granot et al. [48] beschrieben. Die Autoren beweisen, dass der Aufwand des von ihnen vorgeschlagenen Algorithmus $\mathcal{O}(n)$ beträgt, wenn der Baum eine Kette ist. Diese spezielle Spielsituation wurde in der Literatur intensiv diskutiert und auf empirische Fälle angewandt. In diesem Zusammenhang spricht man in der Literatur von sog. *Flughafen-Spielen*.

Das Flughafen-Spiel

Das Flughafen-Spiel wurde von Littlechild/Owen [79] in die Literatur eingeführt. Danach wurden dieser Spieltyp und seine speziellen Formen unter anderem in Littlechild/Thompson [78], Dubey [36], Potters/Sudhölter [106] und Brânzei et al. [19] diskutiert. Zur Illustration betrachten wir nun einen Flughafen, auf dem n verschiedene Flugzeugtypen landen. Jeder Flugzeugtyp $i \in N = \{1, 2, \dots, n\}$ erfordere eine minimale Länge der Landebahn. Wir nehmen an, dass die Flugzeugtypen nach der von ihnen benötigten Länge der Landebahn in steigender Reihenfolge geordnet sind, so dass der Flugzeugtyp 1 die kürzeste Landebahn und der Flugzeugtyp n die längste Landebahn benötigt, um zu landen bzw. zu starten. Diese Anordnung impliziert, dass ein Flugzeugtyp $j \in N$ ebenfalls eine Landebahn anfliegen kann, welche für die Landung von Flugzeugtypen $i \geq j$, $i \in N$ geeignet ist. Des Weiteren seien die Unterhaltskosten einer Landebahn umso höher, je länger die Landebahn ist. Die Unterhaltskosten einer Landebahn für ein Flugzeug vom Typ $i \in N$ seien $c_i \in \mathbb{R}_{\geq 0}$, so dass gilt

$$c_1 \leq c_2 \leq \dots \leq c_n.$$

Damit stellt sich die Frage nach der Aufteilung der Unterhaltskosten einer Landebahn, die von allen Flugzeugtypen verwendet werden kann. Dieses Problem wird

4.6 Kooperative Spiele über kostenminimale spannende Bäume

wie oben erwähnt in der Literatur (vgl. z. B. Borm et al. [14], S. 13) *Flughafen-Problem* (engl. *airport problem*) genannt. Jedes Flughafen-Problem können wir als ein kooperatives Spiel darstellen, indem wir den Wert einer „Koalition“ $S \subseteq N$ von Flugzeugtypen wie folgt definieren

$$c(S) := \max_{i \in S} c_i \quad \text{für alle } S \subseteq N. \quad (4.36)$$

Der Ausdruck (4.36) besagt, dass der Wert $c(S)$ einer Koalition S den Unterhaltskosten der längsten Landebahn entspricht, die von allen Flugzeugtypen $i \in S$ angeflogen werden kann. Einen Algorithmus zur Berechnung des Nukleolus eines Flughafen-Spiels finden wir in Littlechild [77]. Der Algorithmus beruht auf dem Verfahren von Maschler et al. [82].

4.7 Zusammenfassung

In diesem Kapitel haben wir zunächst eines der in der Literatur entwickelten Verfahren zur Berechnung des Nukleolus einer speziellen Klasse kooperativer Spiele vorgestellt. Das hier ausgewählte Verfahren von Wolsey [150] berechnet den Nukleolus einfacher kooperativer Spiele. Einfache Spiele sind dadurch gekennzeichnet, dass der Wert jeder Koalition entweder Eins (Gewinnkoalition) oder Null (Verlustkoalition) ist. Das Verfahren von Wolsey beruht auf dem Verfahren von Maschler et al. [82] und führt folglich zum Nukleolus, indem eine Folge linearer Programme gelöst wird. Somit zählt das Verfahren von Wolsey zu den Methoden der dritten Generation. Ein Vorteil des Verfahrens besteht darin, dass für die Berechnung des Nukleolus einfacher kooperativer Spiele die Verlustkoalitionen des Spiels nicht berücksichtigt werden. Darüber hinaus geht das Verfahren zu Beginn der Berechnung nur von einer Teilmenge der Gewinnkoalitionen aus und bezieht im Laufe der Iterationen nötigenfalls neue Gewinnkoalitionen in die Berechnung ein. Demzufolge sind im Verfahren von Wolsey lineare Programme mit einer vergleichsweise geringeren Anzahl an Nebenbedingungen zu lösen. Im Unterschied zu den in dieser Arbeit vorgestellten anderen Verfahren werden im Verfahren von Wolsey zudem weitere binäre lineare Programme gelöst. Mit Hilfe dieser linearen Programme wird einerseits überprüft, ob der Nukleolus des Spiels erreicht ist. Andererseits werden neue Gewinnkoalitionen gewonnen, die in der nächsten Iteration berücksichtigt werden sollen, falls der Nukleolus noch nicht ermittelt ist.

In den darauf folgenden Abschnitten des Kapitels haben wir weitere spezielle kooperative Spiele vorgestellt. Hierfür haben wir zunächst an Hand von Beispielen die Charakteristika dieser Spiele illustriert und anschliessend gezeigt, wie die Koalitionswerte berechnet werden können. Danach haben wir einen Literaturüberblick über die wichtigsten Algorithmen gegeben, die zur Berechnung des Nukleolus des jeweiligen speziellen Spiels entwickelt worden sind.

Im Abschnitt 4.2 haben wir das Zuordnungsspiel beschrieben, bei dem die Spielermenge in zwei disjunkte Gruppen, z. B. in Anbieter und Nachfrager, aufgeteilt wird.

Kooperation ist nur über die verschiedenen Gruppen hinweg und zwischen Paaren von jeweils zwei Spielern möglich. Anschliessend haben wir in diesem Abschnitt das Matchingspiel, das Permutationsspiel und das Nachbarspiel vorgestellt, welche je eine besondere Form der Zuordnungsspiele bilden. Im Matchingspiel ist nicht erforderlich, dass die oben erwähnten disjunkten Gruppen eine Partition der Spielermenge sind. Zudem wird nicht vorausgesetzt, dass jedem Spieler ein anderer Spieler zugeordnet wird. Im Permutationsspiel hingegen wird die Spielermenge nicht unterteilt. Ferner kann ein Spieler i einem Spieler j zugeordnet werden, während der Spieler j einem anderen Spieler k zugeordnet ist. Das Nachbarspiel ist wiederum dadurch gekennzeichnet, dass nur bestimmte Paare von jeweils zwei Spielern, nämlich die sog. benachbarten Spieler, einander zugeordnet werden können. Im Abschnitt 4.3 haben wir das balancierte zusammenhängende kooperative Spiel eingeführt. In diesem Spieltyp sind die Spieler gemäss einer gegebenen Reihenfolge geordnet. Für die Berechnung des Nukleolus dieses Spiels werden nur jene Zweierkoalitionen herangezogen, die jeweils zwei in der gegebenen Reihenfolge nacheinander stehende Spieler enthalten. Das bilaterale Marktspiel mit komplementären Gütern ist Inhalt des Abschnitts 4.4. In diesem Spiel wird die Spielermenge erneut in zwei disjunkte Mengen aufgeteilt. Ein Koalitionsgewinn kann nur dann erzielt werden, wenn die jeweilige Koalition aus Spielern beider Gruppen besteht. Im Abschnitt 4.5 haben wir das Peer-Group-Spiel und das Veto-Rich-Spiel vorgestellt. Im Peer-Group-Spiel sind die Spieler durch eine gegebene Hierarchie miteinander verbunden. Hierbei realisiert eine beliebige Koalition S nur dann einen Gewinn, wenn sie den in der Hierarchie ganz oben stehenden Spieler, also den Leiter, enthält. Zudem soll die Koalition S mindestens für einen Spieler $i \in S$ alle in der Hierarchie zwischen dem Leiter und dem Spieler i stehenden Spieler enthalten. Im Veto-Rich-Spiel können mehrere Spieler in der Hierarchie ganz oben stehen. Deswegen ist in diesem Spieltyp ein Koalitionsgewinn nur dann möglich, wenn alle Leiter der Koalition angehören. Im Abschnitt 4.6 haben wir das kooperative Spiel über kostenminimale spannende Bäume vorgestellt. Dieses Spiel kann durch einen Graphen in der Form eines spannenden Baumes repräsentiert werden. Der Wert jeder Koalition S entspricht den Kosten des Teilgraphen, der die Mitglie-

4 Ausgewählte spezielle kooperative Spiele

der der Koalition S mit minimalen Kosten zur sog. Quelle verbindet. Eine besondere Form dieser Spiele ist das Standardbaum-Spiel. In diesem Spiel ist die Quelle nur mit einem Knoten des Graphen verbunden. Zudem können mehrere Spieler an einem Knoten abgebildet werden.

5 Implementation zweier Verfahren

5.1 Wahl der Verfahren

In diesem Kapitel beschreiben wir eine Implementation zweier Algorithmen zur Berechnung des Nukleolus kooperativer Spiele und vergleichen diese Algorithmen. Für diesen Vergleich standen grundsätzlich alle in dieser Arbeit vorgestellten Verfahren zur Auswahl. Wir haben uns für das Verfahren von Potters et al. [105] (vgl. S. 130 ff.) und für das Verfahren von Wolsey [150] (vgl. S. 173 ff.) entschieden. Beide Verfahren erfordern zur Berechnung des Nukleolus die Ermittlung einer Lösung einer Folge linearer Programme und beruhen somit auf der konzeptionellen Idee von Maschler et al. [82].

Der Algorithmus von Potters et al. [105] ist für uns von Interesse, weil er einerseits einer der neuesten Algorithmen zur Berechnung des Nukleolus allgemeiner kooperativer Spiele ist. Andererseits haben wir den Algorithmus von Potters et al. dem Algorithmus von Derks/Kuipers [32] gegenüber gestellt, der ebenfalls zu den neuesten, leistungsfähigen Algorithmen zählt (vgl. Tabelle 3.3, S. 170). Wie wir gleich darlegen werden, hat unser Vergleich ergeben, dass der Algorithmus von Potters et al., zumindest für die von uns gewählten kooperativen Spiele mit n Spielern ($n = 3, 4, \dots, 17$),¹ hinsichtlich der Rechenzeit bessere Ergebnisse aufweist. Wir haben für jedes Spiel mit n Spielern ($n = 3, 4, \dots, 17$) je fünf Spiele herangezogen und den Nukleolus jedes dieser Spiele sowohl mit Hilfe des Algorithmus von Potters et al. wie auch mit Hilfe des Algorithmus von Derks/Kuipers bestimmt. Hierfür haben wir unsere Implementation des Verfahrens von Potters et al. bzw. das von Derks/Kuipers im Internet zur

¹Diese Testspiele sowie deren Nukleoli werden im Anhang H wiedergegeben.

5 Implementation zweier Verfahren

Tabelle 5.1: Vergleich der Rechenzeiten \bar{t}_{DK} und \bar{t}_{PRA}

Spielerzahl n	$\bar{t}_{DK}/\bar{t}_{PRA}$
3	22.03
4	12.24
5	11.48
6	8.45
7	8.37
8	8.51
9	5.87
10	5.52
11	4.12
12	2.17
13	1.76
14	4.02
15	5.68
16	10.87
17	11.89

Verfügung gestellte Computerprogramm `cg_sol` eingesetzt.² Wir haben für die jeweiligen fünf Testspiele jedes Spiels mit n Spielern ($n = 3, 4, \dots, 17$) die durchschnittlichen Rechenzeiten \bar{t}_{PRA} und \bar{t}_{DK} ermittelt, die von den Verfahren von Potters et al. bzw. Derks/Kuipers in Anspruch genommen wurden.³ Anschliessend haben wir die gefundene jeweilige durchschnittliche Rechenzeit \bar{t}_{DK} durch die entsprechende Rechenzeit \bar{t}_{PRA} dividiert. In der Tabelle 5.1 werden die Quotienten $\bar{t}_{DK}/\bar{t}_{PRA}$ für alle Testspiele wiedergegeben. Aus der Tabelle 5.1 geht hervor, dass das Verfahren von Derks/Kuipers für die Berechnungen im Durchschnitt stets eine längere Rechenzeit in Anspruch nimmt als das Verfahren von Potters et al., da alle Quotienten $\bar{t}_{DK}/\bar{t}_{PRA}$ einen grösseren Wert aufweisen als den Wert Eins. Beispielsweise erfordert das Verfahren von Derks/Kuipers für $n = 17$ das 11.89fache mehr Zeit als das Verfahren von Potters et al.

²Die Implementation des Verfahrens von Potters et al. wird im Abschnitt 5.2, S. 214 ff. ausführlich beschrieben. Das Computerprogramm `cg_sol` ist erhältlich unter www.math.unimaas.nl/PERSONAL/jeand/home1.htm (28.11.2005).

³Die Berechnungen wurden mit einem Pentium 4 (2.8 GHz) Computer durchgeführt.

5.1 Wahl der Verfahren

Wie wir im Abschnitt 3.3.3, S. 130 ff., bereits erläutert haben, ist der Algorithmus von Potters et al. zur Berechnung des Nukleolus jedes kooperativen Spiels geeignet. Folglich lässt er mögliche spezielle Eigenschaften von Spielen ausser Acht, auch dann, wenn diese die Berechnung des Nukleolus vereinfachen würden. Diese Vereinfachung kann z. B. in einer reduzierten Anzahl der Nebenbedingungen der zu lösenden linearen Programme bestehen. Das Verfahren von Wolsey [150] hingegen berechnet den Nukleolus einfacher kooperativer Spiele. Der Algorithmus nutzt bestimmte Eigenschaften von einfachen kooperativen Spielen, indem er zur Berechnung des Nukleolus nicht alle möglichen Koalitionen, sondern nur eine Teilmenge der Gewinnkoalitionen heranzieht. Somit reduziert der Algorithmus die Anzahl der Nebenbedingungen von linearen Programmen, die für die Berechnung des Nukleolus gelöst werden. Der Algorithmus verlangt aber in jeder Iteration zusätzlich die Lösung eines binären linearen Programms, nämlich eines Rucksackproblems (vgl. Anhang A.4). Die Tabelle 5.2 fasst die Merkmale der beiden Verfahren zusammen. Darin ist n die Anzahl der Spieler.

Tabelle 5.2: Gegenüberstellung der implementierten Verfahren

	Potters et al. [105]	Wolsey [150]
Grundidee der Verfahren	Berechnung einer Lösung einer Folge linearer Programme	
Optimierungsaufgaben	lineares Programm	<ul style="list-style-type: none"> • lineares Programm • binäres lineares Programm
Maximale Zeilenzahl im linearen Programm	$2^n - 1$	Anzahl der Gewinnkoalitionen ($\leq 2^n - 1$)

Im Allgemeinen, wenn verschiedene Verfahren zur Lösung eines Problems verfügbar sind und eines dieser Verfahren eine mögliche spezielle Struktur des Problems ausnutzt, ist es naheliegend anzunehmen, dass dieses Verfahren eine bessere Leistung (z. B. Laufzeitverhalten) erbringt als die allgemeinen Verfahren. Das ist aber

5 Implementation zweier Verfahren

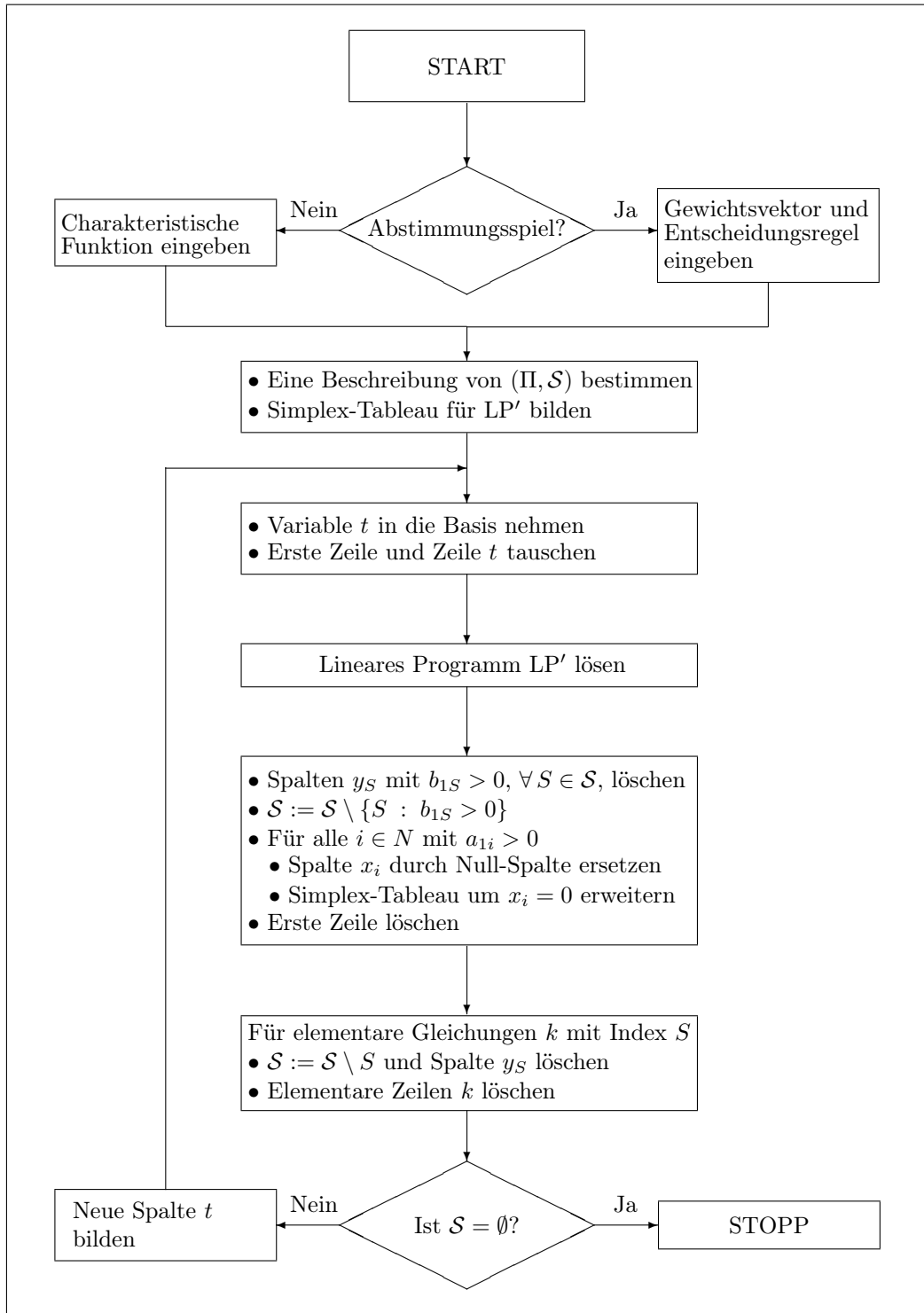
nicht immer der Fall. Beispielsweise hat man in der Literatur bei der Entwicklung der Verfahren zur Lösung linearer Programme andere Erfahrungen gemacht: Man hat nämlich zur Lösung linearer Programme die sog. Interior Point-Methoden entwickelt, die theoretisch hinsichtlich des Rechenaufwands in ungünstigsten Fällen besser als die Simplex-Methode sind. Die gemachten Erfahrungen deuten darauf hin, dass die Interior Point-Methoden hinsichtlich des durchschnittlichen Rechenaufwands eine schlechtere Leistung aufweisen als die Simplex-Methode und für die Lösung praktischer Probleme weniger geeignet sind (vgl. dazu Anhang A.1, S. 295). Um zu prüfen, ob ein Verfahren, das zur Berechnung des Nukleolus nur einer Klasse kooperativer Spiele entwickelt worden ist, ein besseres Laufzeitverhalten als ein anderes Verfahren aufweist, das den Nukleolus allgemeiner kooperativer Spiele berechnet, haben wir exemplarisch das Verfahren von Wolsey [150] und das Verfahren von Potters et al. [105] ausgewählt. Für die Wahl des Algorithmus von Wolsey [150] als ein Lösungsverfahren für spezielle Spiele spricht, dass es sich bei unserer Anwendung, die wir im Kapitel 6 vorstellen werden, um ein einfaches kooperatives Spiel handelt. In der Literatur wurden die Verfahren von Potters et al. [105] und von Wolsey [150] bisher noch nicht miteinander verglichen.

5.2 Implementation des Algorithmus von Potters et al.

Im Abschnitt 3.3.3 haben wir den Algorithmus von Potters et al. [105] bereits ausführlich vorgestellt. Bevor wir nun die Implementation dieses Algorithmus im Detail diskutieren, wollen wir mit seiner Hilfe den Nukleolus des Einführungsbeispiels berechnen. Die im Beispiel durchzuführenden Rechenschritte des Verfahrens haben wir bereits auf S. 142 dargestellt. In Abbildung 5.1 (S. 215) wird jeder Schritt in einem Ablaufdiagramm wiedergegeben. Dieses Diagramm stellt auch die Grundlage unserer konkreten Implementation des Algorithmus von Potters et al. [105] dar.

5.2 Implementation des Algorithmus von Potters et al.

Abbildung 5.1: Ablaufdiagramm des Verfahrens von Potters et al. [105]



5 Implementation zweier Verfahren

Zur Erinnerung geben wir zunächst nochmals die charakteristische Funktion des Einführungsbeispiels an:

$$\begin{aligned} v(\{1, 2\}) &= 6, & v(\{1, 3\}) &= 16, & v(\{2, 3\}) &= 26, \\ v(\{1, 2, 3\}) &= 42, & v(\{i\}) &= 0, & i &= 1, 2, 3. \end{aligned}$$

Um den Nukleolus dieses Spiels mit Hilfe des Verfahrens von Potters et al. [105] zu berechnen, definieren wir $\mathcal{S} := \{\{1\}, \{1, 2\}, \{1, 3\}, \{2\}, \{2, 3\}, \{3\}\}^4$ mit $|\mathcal{S}| = 2^3 - 2 = 6$, $\Pi := \mathcal{I}(N, v)$, $c := \max_{S \in \mathcal{S}} v(S) = 6$. Für den Spieler $i' := 3$ haben wir die Beschreibung des Paares (Π, \mathcal{S}) auf S. 134 aufgestellt:

$$\begin{aligned} x_1 + x_2 + x_3 &= 42 \\ y_{\{1\}} - x_1 &= 26 \\ y_{\{1,2\}} - x_1 - x_2 &= 20 \\ y_{\{1,3\}} + x_2 &= 52 \\ y_{\{2\}} - x_2 &= 26 \\ y_{\{2,3\}} + x_1 &= 42 \\ y_{\{3\}} + x_1 + x_2 &= 68. \end{aligned} \tag{5.1}$$

Das Simplex-Tableau des auf S. 143 vorgestellten linearen Programms LP' , dessen Nebenbedingungen die Gleichungen (5.1) sind, wird in der Tabelle 5.3 dargestellt (vgl. dazu auch die Tabelle 3.2, S. 3.2). In diesem Tableau gibt die erste Spalte,

Tabelle 5.3: Simplex-Tableau des linearen Programms LP'

d	t	x_1	x_2	x_3	$y_{\{1\}}$	$y_{\{1,2\}}$	$y_{\{1,3\}}$	$y_{\{2\}}$	$y_{\{2,3\}}$	$y_{\{3\}}$
42	0	1	1	1	0	0	0	0	0	0
26	1	-1	0	0	1	0	0	0	0	0
20	1	-1	-1	0	0	1	0	0	0	0
52	1	0	1	0	0	0	1	0	0	0
26	1	0	-1	0	0	0	0	1	0	0
42	1	1	0	0	0	0	0	0	1	0
68	1	1	1	0	0	0	0	0	0	1

also die Spalte d , die rechte Seite des Gleichungssystems (5.1) wieder. Die restlichen Spalten enthalten die Koeffizienten der Variablen des linearen Programms, wobei t ,

⁴Die Koalitionen sind in lexikographischer Ordnung dargestellt.

5.2 Implementation des Algorithmus von Potters et al.

x_1, x_2 Nichtbasisvariablen und $x_3, y_{\{1\}}, y_{\{1,2\}}, y_{\{1,3\}}, y_{\{2\}}, y_{\{2,3\}}, y_{\{3\}}$ Basisvariablen sind. Hierbei heisst jede Variable y_S , $S \in \mathcal{S}$, Schlupfvariable der zugehörigen Nebenbedingung. Um den Algorithmus von Potters et al. implementieren zu können, haben wir jeder Variablen des linearen Programms einen Index zugewiesen. Hierzu haben wir für die Variablen x_i , $i \in N$, den Index i und für t den Index $n + 1$ verwendet. Den Variablen y_S , $S \in \mathcal{S}$, haben wir die Indizes $n + 2, n + 3, \dots$ zugeordnet, wobei die jeweiligen Koalitionen in lexikographischer Reihenfolge aufgelistet werden. Ist also ein Index grösser als $n + 1$, so ist die zugehörige Variable eine Variable y_S . Demnach sehen für das in der Tabelle 5.3 dargestellte Simplex-Tableau die Variablen und die ihnen zugeordneten Indizes wie folgt aus:

Variable	t	x_1	x_2	x_3	$y_{\{1\}}$	$y_{\{1,2\}}$	$y_{\{1,3\}}$	$y_{\{2\}}$	$y_{\{2,3\}}$	$y_{\{3\}}$
Index	4	1	2	3	5	6	7	8	9	10

Um während der Lösung der jeweiligen linearen Programme zwischen Basisvariablen und Nichtbasisvariablen unterscheiden zu können, haben wir in unserer Implementation einen $(2^n - 1)$ -dimensionalen Index-Vektor „**BasInd**“ für die Basisvariablen und einen n -dimensionalen Index-Vektor „**NoBasInd**“ für die Nichtbasisvariablen des linearen Programms eingeführt. Für die Tabelle 5.3 sind **BasInd** = (3, 5, 6, 7, 8, 9, 10) und **NoBasInd** = (4, 1, 2).

Um Speicherplatz zu sparen, haben wir in unserer Implementation nicht die ganze Tabelle 5.3 gespeichert, sondern nur die erste Spalte und die Spalten der Nichtbasisvariablen. Die Spalten der Basisvariablen sind Einheitsspalten. Folglich genügt es, für die Spalte einer Basisvariablen nur die Stelle der Eins zu speichern, weil jedes andere Element einer Einheitsspalte eine Null ist. Zu diesem Zweck speichern wir den Index einer Basisvariablen im Vektor **BasInd** an der Stelle j , wenn die Komponente j in der Spalte dieser Variablen Eins ist. Beispielsweise wird der Index 10 in **BasInd** an der siebten Stelle gespeichert, weil die siebte Komponente in der Spalte der Basisvariablen y_3 mit dem Index 10 eine Eins ist und die anderen Einträge Nullen sind. Somit besteht das Simplex-Tableau, das wir für unsere Implementation speichern, aus den ersten vier Spalten der Tabelle 5.3. In der Tabelle 5.4 sind die in unserer

5 Implementation zweier Verfahren

Implementation zu speichernden Spalten dargestellt. Die erste Spalte der Tabelle 5.4 enthält den Index-Vektor **BasInd** und die erste Zeile ab der dritten Spalte den Index-Vektor **NoBasInd**. Die letzten vier Spalten ohne die erste Zeile sind die Spalten des Simplex-Tableaus.

Tabelle 5.4: Simplex-Tableau für die Implementation

	<i>d</i>	<i>4</i>	<i>1</i>	<i>2</i>
<i>3</i>	42	0	1	1
<i>5</i>	26	1	-1	0
<i>6</i>	20	1	-1	-1
<i>7</i>	52	1	0	1
<i>8</i>	26	1	0	-1
<i>9</i>	42	1	1	0
<i>10</i>	68	1	1	1

Wir lösen nun das lineare Programm LP' . Hierzu führen wir verschiedene Pivotoperationen durch.⁵ Da wir zunächst die Variable t durch eine Pivotoperation in die Basis nehmen, ist die Spalte der Variablen t mit Index 4 die Pivotspalte. Also ist die zweite Spalte des Simplex-Tableaus die Pivotspalte ps . Demnach ist $ps = 2$. Die Anwendung der Regel zur Bestimmung der Pivotzeile (man suche den minimalen nicht-negativen Quotient der Spalte d und der Spalte ps) ergibt, dass die dritte Zeile die Pivotzeile pz ist. Folglich gilt $pz = 3$. Das Pivotelement ist im Simplex-Tableau in der Tabelle 5.4 in fetter Druckschrift dargestellt. Nach der Pivotoperation wird die Variable t eine Basisvariable und die der Pivotzeile 3 entsprechende Basisvariable mit dem Index 6 eine Nichtbasisvariable. Die dritte Zeile ist nun diejenige Zeile, die die Variable t beinhaltet. Diese Zeile ist jetzt die Zielfunktion des linearen Programms. Um bei den nächsten Schritten die Festlegung des Pivotelements zu vereinfachen, tauschen wir die erste und die dritte ($pz = 3$) Zeile des Simplex-Tableaus. In der Tabelle 5.5 stellen wir das nach der Durchführung der Pivotoperation und dem Zeilentausch erhaltene Simplex-Tableau dar, wobei **BasInd** = ($4, 5, 3, 7, 8, 9, 10$) und **NoBasInd** = ($6, 1, 2$) sind.

⁵Für die Lösung linearer Programme und die Durchführung von Pivotoperationen vgl. Anhang A.

5.2 Implementation des Algorithmus von Potters et al.

Tabelle 5.5: Variable t wird Basisvariable (Iteration 1)

	d	6	1	2
4	20	1	-1	-1
5	6	-1	0	1
3	42	0	1	1
7	32	-1	1	2
8	6	-1	1	0
9	22	-1	2	1
10	48	-1	2	2

Da wir in der Tabelle 5.5 in der Zielfunktionszeile (erste Zeile) noch negative Werte finden, müssen wir mindestens noch eine Pivotoperation durchführen. Die beiden negativen Werte in der ersten Zeile sind gleich gross. Wir wählen einen Wert als Pivotspalte, z. B. die dritte Spalte, $ps = 3$. Folglich ist die fünfte Zeile Pivotzeile, $pz = 5$. Das neue Simplex-Tableau wird in Tabelle 5.6 wiedergegeben.

Tabelle 5.6: Aktuelles Simplex-Tableau nach $ps = 3$, $pz = 5$

	d	6	8	2
4	26	0	1	-1
5	6	-1	0	1
3	36	1	-1	1
7	26	0	-1	2
1	6	-1	1	0
9	10	1	-2	1
10	36	1	-2	2

Die aktuellen Index-Vektoren der Basis- und Nichtbasisvariablen sind nun $\text{BasInd} = (4, 5, 3, 7, 1, 9, 10)$ und $\text{NoBasInd} = (6, 8, 2)$. Da das Optimum noch nicht erreicht ist, fahren wir mit der nächsten Pivotoperation fort. Nach der Durchführung von weiteren zwei Pivotoperationen, die wir hier nicht vorführen, erreichen wir das in der Tabelle 5.7 angegebene optimale Simplex-Tableau des linearen Programms mit $\text{BasInd} = (4, 2, 3, 7, 1, 6, 10)$ und $\text{NoBasInd} = (9, 8, 5)$. Somit ist der erste Schritt der ersten Iteration beendet.

Im zweiten Schritt des Algorithmus bestimmen wir diejenigen Koalitionen, deren Schlupfvariablen in der Zielfunktionszeile des optimalen Simplex-Tableaus positive

5 Implementation zweier Verfahren

Tabelle 5.7: Optimales Simplex-Tableau der ersten Iteration

	d	9	8	5
4	34	0.5	0	0.5
2	8	0.5	-1	0.5
3	26	-1	1	0
7	10	-1	1	-1
1	8	0.5	0	-0.5
6	2	0.5	-1	-0.5
10	18	-1.5	1	-0.5

Koeffizienten aufweisen. Da die Indizes der Schlupfvariablen grösser als $n + 1 = 4$ sind und die Koeffizienten aller Basisvariablen in der Zielfunktionszeile des optimalen Simplex-Tableaus Nullen sind, interessieren uns diejenigen Nichtbasisvariablen, deren Indizes grösser als $n + 1 = 4$ sind. Die Koalitionen mit dieser Eigenschaft sind $\{1\}$ und $\{2, 3\}$, weil deren Schlupfvariablen $y_{\{1\}}$ und $y_{\{2,3\}}$, die durch die Indizes 5 und 9 vertreten sind, in der Zielfunktionszeile positive Koeffizienten besitzen. Deshalb entfernen wir die zweite und vierte Spalte aus dem Simplex-Tableau, die Indizes 9 und 5 aus dem Index-Vektor **NoBasInd** und die Koalitionen $\{1\}$ sowie $\{2, 3\}$ aus der Menge \mathcal{S} , d. h. $\mathcal{S} := \mathcal{S} \setminus \{\{1\}, \{2, 3\}\} = \{\{1, 2\}, \{1, 3\}, \{2\}, \{3\}\}$. Anschliessend entfernen wir die erste Zeile (die Zeile der Zielfunktion) aus dem optimalen Simplex-Tableau und den zu der Variablen t gehörenden Index 4 aus dem Index-Vektor **BasInd**. Das Simplex-Tableau in der Tabelle 5.7 beinhaltet keine elementaren Gleichungen. Da $\mathcal{S} \neq \emptyset$ ist, beginnen wir mit der zweiten Iteration.

Die Tabelle 5.8 gibt das aktuelle Simplex-Tableau zu Beginn der neuen Iteration wieder. Die aktuellen Index-Vektoren sind **BasInd** = $(2, 3, 7, 1, 6, 10)$ und **NoBasInd** = (8) . Wir bestimmen zunächst die neue Spalte t für das nächste lineare Programm LP' , indem wir in jeder Zeile die Koeffizienten der Variablen $y_{\mathcal{S}}$ addieren. Um die Bestimmung der neuen Spalte t zu illustrieren, geben wir in der Tabelle 5.9 das optimale Simplex-Tableau erneut an, in dem nun auch die Spalten der Basisvariablen ersichtlich sind. Die Indexzahl jeder Variablen ist in der Tabelle 5.9 unterhalb der jeweiligen Variablen angegeben.

5.2 Implementation des Algorithmus von Potters et al.

Tabelle 5.8: Simplex-Tableau am Ende der ersten Iteration

	d	δ
2	8	-1
3	26	1
7	10	1
1	8	0
6	2	-1
10	18	1

Tabelle 5.9: Simplex-Tableau nach Schritt 2 der ersten Iteration

d	$y_{\{2\}}$	x_2	x_3	$y_{\{1,3\}}$	x_1	$y_{\{1,2\}}$	$y_{\{3\}}$
	8	2	3	7	1	6	10
8	-1	1	0	0	0	0	0
26	1	0	1	0	0	0	0
10	1	0	0	1	0	0	0
8	0	0	0	0	1	0	0
2	-1	0	0	0	0	1	0
18	1	0	0	0	0	0	1

Nachdem wir die neue Spalte t berechnet haben, können wir das Simplex-Tableau des linearen Programms der zweiten Iteration aufstellen. In der Tabelle 5.10 geben wir dieses Tableau ohne die Spalten der Basisvariablen an.

Tabelle 5.10: Simplex-Tableau mit der neuen Spalte t

	d	t	$y_{\{2\}}$
		4	8
2	8	-1	-1
3	26	1	1
7	10	2	1
1	8	0	0
6	2	0	-1
10	18	2	1

Um dieses zweite lineare Programm zu lösen, bringen wir wiederum zunächst die Variable t in die Basis. Spalte t wird zur Pivotspalte und die dritte Zeile zur Pivotzeile. Nachdem wir die erste Pivotoperation durchgeführt haben, tauschen wir die dritte Zeile und die erste Zeile. Die erste Zeile des daraus entstandenen Simplex-Tableaus

5 Implementation zweier Verfahren

Tabelle 5.11: Variable t wird Basisvariable (Iteration 2)

	d	γ	δ
4	5	0.5	0.5
3	21	-0.5	0.5
2	13	0.5	-0.5
1	8	0	0
6	2	0	1
10	8	-1	0

(vgl. Tabelle 5.11) ist die Zielfunktion des zweiten linearen Programms LP' . Wir stellen fest, dass alle Koeffizienten der Zielfunktionszeile nicht-negativ sind. Demnach ist das Optimum des linearen Programms gefunden. Im optimalen Simplex-Tableau sind γ und δ die Indizes der Nichtbasisvariablen. Da diese Indizes grösser als $n+1 = 4$ sind und in der Zielfunktionszeile positive Werte stehen ($0.5 > 0$ und $0.5 > 0$), entfernen wir im Schritt 2 der zweiten Iteration die beiden Spalten der Nichtbasisvariablen aus dem optimalen Simplex-Tableau. Folglich bleiben nur noch die Basisvariablen übrig. In diesem Fall ist in jeder Zeile nur ein Eintrag von null verschieden. Somit ist jede zu überprüfende Zeile eine elementare Gleichung. Durch Entfernung der den elementaren Gleichungen entsprechenden Koalitionen aus \mathcal{S} erhalten wir $\mathcal{S} = \emptyset$. Somit endet der Algorithmus. Wir lesen den Nukleolus des Spiels aus dem optimalen Simplex-Tableau ab: $\mathcal{N} = (x_1, x_2, x_3) = (8, 13, 21)$. Für die Berechnung des Nukleolus haben wir zwei lineare Programme gelöst und insgesamt fünf Pivotoperationen durchgeführt.

Den Algorithmus haben wir in der Programmiersprache Delphi 6 realisiert, die eine Weiterentwicklung der Programmiersprache PASCAL darstellt. Für die Implementierung haben wir den Algorithmus in Teilprobleme zerlegt und für jedes dieser Teilprobleme ein Unterprogramm, eine sog. *Prozedur* (engl. procedure) oder eine sog. *Funktion* (engl. function), aufgestellt. Die Unterprogramm-Technik hat den Vorteil, dass sie einerseits das Verständnis des Hauptprogramms verbessert. Andererseits können die Unterprogramme nötigenfalls im Hauptprogramm mehrfach verwendet werden. Für die Implementation des Algorithmus von Potters et al. haben wir ver-

5.2 Implementation des Algorithmus von Potters et al.

schiedene Unterprogramme erstellt. Unten beschreiben wir die einzelnen Aufgaben dieser Unterprogramme kurz im Überblick.⁶ Den gesamten Quellcode unserer Implementation findet der interessierte Leser in der in dieser Arbeit beigelegten CD:

Procedure DelSpalte löscht eine angegebene Spalte s aus dem optimalen Simplex-Tableau (aus der Matrix \mathbf{A}) und den zu dieser Spalte gehörenden Index aus dem Index-Vektor der Nichtbasisvariablen **NoBasInd**.

Procedure DelZeile löscht eine angegebene Zeile k aus dem optimalen Simplex-Tableau (aus der Matrix \mathbf{A}) und das Element k aus dem Index-Vektor der Basisvariablen **BasInd**.

Procedure Eingabe liest die Daten aus der Datei ab, in der die zum Start benötigten Angaben gespeichert sind.

Procedure ElementaryRows überprüft die Zeilen des optimalen Simplex-Tableaus darauf, ob sie elementare Gleichungen sind oder nicht. Nach der Bestimmung der elementaren Gleichungen überprüft dieses Unterprogramm, ob alle Zeilen der Matrix \mathbf{A} elementar sind. Wenn dies der Fall ist, dann ist der Nukleolus erreicht. Sonst entfernt dieses Unterprogramm die elementaren Gleichungen aus der Matrix \mathbf{A} .

Function Hoch berechnet die Anzahl der Koalitionen der Spielermenge N , also $2^{|N|}$. Denn zur Berechnung des Nukleolus eines kooperativen Spiels zieht der Algorithmus alle möglichen Koalitionen ohne die leere Menge heran. Folglich beträgt die Anzahl der Nebenbedingungen des ersten linearen Programms $2^{|N|} - 1$.

Procedure PivotOp führt eine Pivotoperation durch, nachdem die Pivotspalte und die Pivotzeile festgelegt worden sind.

Procedure PivotSpalte findet die Pivotspalte ps . Die zuerst gefundene Spalte, die in der Zielfunktionszeile den grössten negativen Wert hat, wird als Pivotspalte ausgewählt.

Procedure PivotZeile stellt die Pivotzeile pz fest. Diejenige Zeile ist die Pivotzeile, für die der nicht-negative Quotient von den Einträgen der Spalte d (erste Spalte in \mathbf{A}) und den Einträgen der Pivotspalte ps minimal ist. Damit die erste Zeile die Zielfunktionszeile ist, tauscht dieses Unterprogramm die erste Zeile mit der Zeile t

⁶Die Unterprogramme werden in alphabetischer Reihenfolge beschrieben.

5 Implementation zweier Verfahren

in jeder Iteration nach der ersten Pivotoperation.

Procedure PosSpalten entfernt aus der Matrix **A** diejenigen Spalten, deren Variablen einen Index haben, der grösser als $Sp + 1$ ist,⁷ und die in der Zielfunktionszeile (in der ersten Zeile der Matrix **A**) positive Koeffizienten besitzen (vgl. z. B. Tabelle 5.7, S. 220). Falls eine Variable x_i mit einem Index, der kleiner als $Sp + 1$ ist, einen positiven Eintrag in der Zielfunktionszeile besitzt, wird die zugehörige Spalte durch eine Null-Spalte ersetzt und die Matrix **A** um die Zeile $x_i = 0$ erweitert.

Procedure Powerset findet alle echten Teilmengen der Menge N und bildet das zugehörige Gleichungssystem gemäss der Beschreibung des Spiels (vgl. S. 142) durch eine Matrix **A**, wobei **A** die Gleichung $x(N) = v(N)$ noch nicht beinhaltet. Das Programm ist ein rekursives Programm und verwendet zwei Unterprogramme **NextSub** und **KoalAlsNB**. Das Unterprogramm **NextSub** bestimmt die echten Teilmengen der Menge N und das Unterprogramm **KoalAlsNB** die zur jeweiligen Teilmenge gehörende Zeile für die Matrix **A**.

Procedure Simplex führt Pivotoperationen solange durch, bis das Optimum, wenn es existiert, gefunden wird.

Procedure StartMatrix fügt zu der unter **Powerset** gefundenen Matrix **A** die Spalte t und die Zeile $x(N) = v(N)$ hinzu und bildet somit die für die Berechnung des Nukleolus benötigte Matrix (vgl. z. B. Tabelle 5.4, S. 218). Das Unterprogramm speichert die Indizes der Basis- und Nichtbasisvariablen der aktuellen Matrix **A** in **BasInd** bzw. **NoBasInd**. Zu Beginn des Algorithmus gibt es insgesamt $2^{|N|} - 1$ (Anzahl der Koalitionen ohne die leere Menge) Basisvariablen und Sp (Anzahl der Spieler) Nichtbasisvariablen.

Procedure t_ColumnPivot bringt die Variable t durch eine Pivotoperation in die Basis.

Procedure t_SpalteBilden bildet für das nächste lineare Programm die neue Spalte t .

Procedure ZeitAufw berechnet den Zeitaufwand der Berechnung in Millisekunden.

⁷In unseren Implementationen bezeichnen wir die Anzahl der Spieler aus programmtechnischen Gründen als Sp .

5.3 Implementation des Algorithmus von Wolsey

Ein letztes Unterprogramm, das wir für die Implementation des Verfahrens von Potters et al. erstellt haben, ist **Procedure VetoGame**. Da dieses Unterprogramm auch bei der Implementation des Algorithmus von Wolsey Verwendung findet, stellen wir es im nächsten Abschnitt auf S. 238 vor.

5.3 Implementation des Algorithmus von Wolsey

Wir haben im Abschnitt 4.1 das Verfahren von Wolsey [150] zur Berechnung des Nukleolus einfacher kooperativer Spiele beschrieben. In diesem Abschnitt werden wir unsere Implementation des Verfahrens erläutern und den Nukleolus eines einfachen gewichteten Abstimmungsspiels mit Hilfe des Verfahrens von Wolsey berechnen. Wir fassen zunächst die einzelnen Schritte des Algorithmus von Wolsey nochmals zusammen, wobei die Durchführung der Schritte 1, 2 und 3 jeweils eine Iteration des Algorithmus bildet.⁸

Initialisierung

$k := 0, r := 0, \mathcal{V}_0 := \emptyset$.

Schritt 1

Wir lösen das folgende lineare Programm LP

$$\begin{aligned} \text{LP :} \quad & \max_{u, \mathbf{x}} u \\ & \text{unter den Nebenbedingungen} \\ & x(S) \geq u, \quad \forall S \in \mathcal{W}_{k,r} \subseteq \mathcal{W} \setminus \bigcup_{q=0}^k \mathcal{V}_q \\ & x(S) = x_0^q, \quad \forall S \in \mathcal{V}_q, \quad q = 0, 1, \dots, k \\ & x(N) = 1 \\ & x_i \geq 0, \quad i = 1, 2, \dots, n. \end{aligned}$$

Es sei $(x_1^{k,r}, x_2^{k,r}, \dots, x_n^{k,r}, u^*) =: (\mathbf{x}^{k,r}, x_0^{k,r})$ eine optimale Lösung von LP, wobei $x_0^{k,r}$ der optimale Zielfunktionswert ist. Zudem sei $n^k := |\bigcup_{q=0}^k \mathcal{V}_q|$.

⁸Für die nähere Beschreibung der einzelnen Schritte vgl. Abschnitt 4.1.

5 Implementation zweier Verfahren

Schritt 2

Wir lösen das folgende binäre lineare Programm IP

$$\begin{aligned} \text{IP :} \quad & (n^k + 1) - \text{bestes Minimum von } \mathbf{x}^{k,r} \mathbf{y}^T \\ & \text{unter der Nebenbedingung} \\ & \mathbf{y} \in Y(\Gamma). \end{aligned} \tag{5.2}$$

Es sei $(\mathbf{y}^{k,r}; z^{k,r})$ eine optimale Lösung von IP, wobei $z^{k,r}$ der Wert der Zielfunktion im Optimum ist.

Schritt 3

Falls $z^{k,r} < x_0^{k,r}$ ist, setzen wir $\hat{S} := S(\mathbf{y}^{k,r})$, $r := r + 1$, $\mathcal{W}_{k,r} := \mathcal{W}_{k,r-1} \cup \hat{S}$ und gehen zu Schritt 1. Sonst ist $z^{k,r} \geq x_0^{k,r}$. Ist in diesem Fall die Lösung von LP eindeutig, so ist $\mathbf{x}^{k,r}$ der Nukleolus des Spiels. Wir beenden dann die Berechnung. Falls die Lösung von LP nicht-eindeutig ist, bestimmen wir die Menge

$$\mathcal{V}_{k+1} := \left\{ S \mid x(S) = x_0^{k,r} \ \forall \text{ optimalen Lösungen von LP, } \forall S \in \mathcal{W}_{k,r} \right\},$$

setzen $x_0^k := x_0^{k,r}$, $r := 0$, $k := k + 1$ und gehen zurück zu Schritt 1.

In der Abbildung 5.2 wird das Ablaufdiagramm des oben beschriebenen Algorithmus angegeben.

Zur Illustration des Algorithmus von Wolsey und unserer Implementation dieses Verfahrens berechnen wir nun den Nukleolus eines gewichteten Abstimmungsspiels mit $Sp = 4$ Spielern. Das gewichtete Abstimmungsspiel sei

$$\Gamma = [24; 15, 14, 9, 8].$$

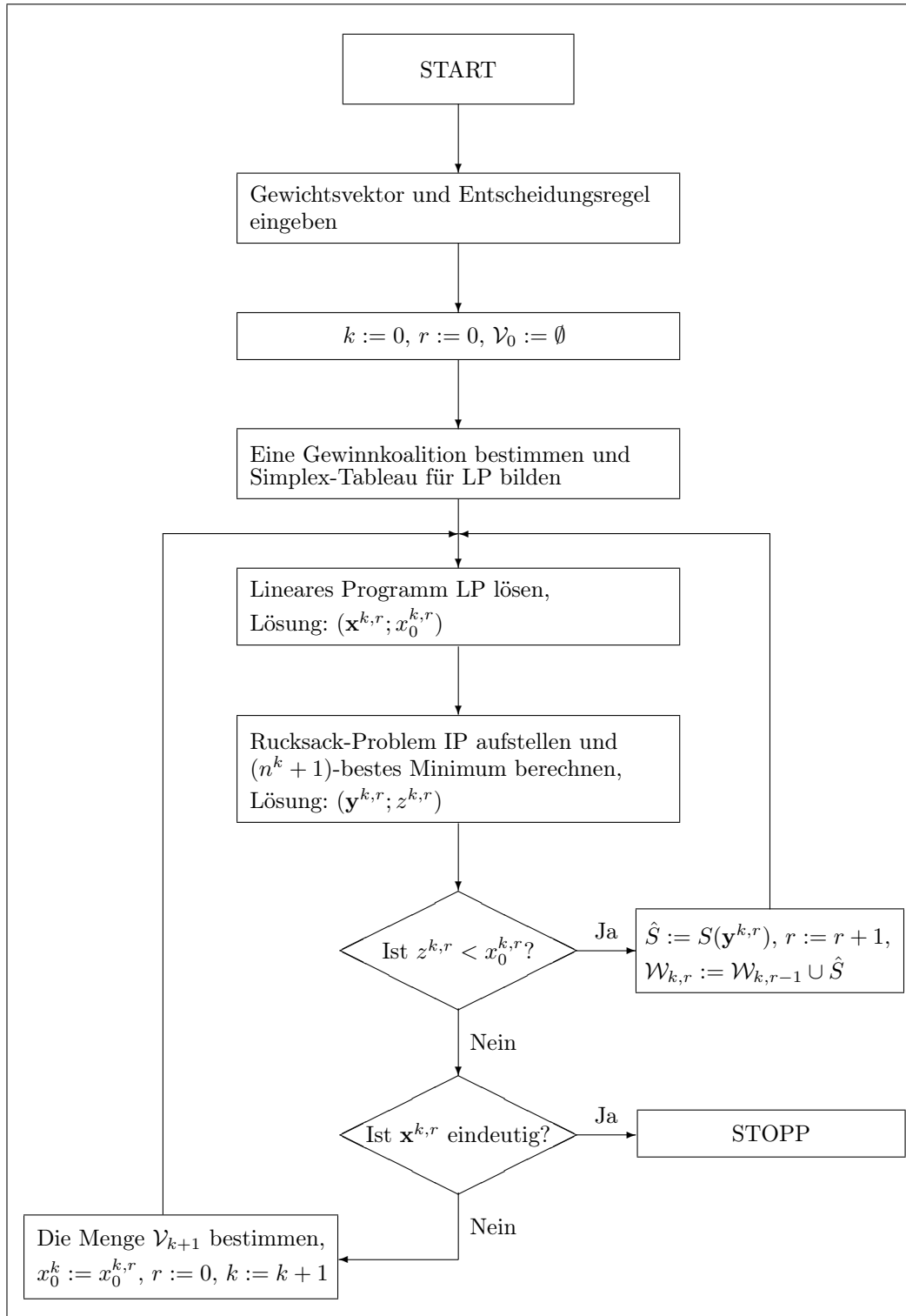
Die Menge der Gewinnkoalitionen dieses gewichteten Abstimmungsspiels ist:

$$\mathcal{W} = \{\{1, 2\}, \{1, 3\}, \{1, 2, 3\}, \{1, 2, 4\}, \{1, 3, 4\}, \{2, 3, 4\}, \{1, 2, 3, 4\}\}.$$

Als Initialisierung definieren wir $k := 0$, $r := 0$ und $\mathcal{V} := \emptyset$. Um mit Hilfe des Algorithmus von Wolsey den Nukleolus dieses einfachen kooperativen Spiels zu berechnen, wählen wir zunächst eine beliebige Teilmenge der Menge aller Gewinnkoalitionen \mathcal{W} . Da Wolsey nicht explizit angibt, wie viele Gewinnkoalitionen zu Beginn

5.3 Implementation des Algorithmus von Wolsey

Abbildung 5.2: Ablaufdiagramm des Verfahrens von Wolsey [150]



5 Implementation zweier Verfahren

der Berechnung herangezogen werden sollen, haben wir uns für die minimale Anzahl der Gewinnkoalitionen, also für nur eine Gewinnkoalition, entschieden. Das heisst $|\mathcal{W}_{k=0,r=0}| = 1$. Für unser Beispiel wählen wir die Gewinnkoalition $\{1, 2\}$. Folglich ist $\mathcal{W}_{k=0,r=0} := \{\{1, 2\}\}$. Das nun zu lösende lineare Programm LP hat die Gestalt:

$$\begin{aligned} \text{LP : } \quad & \max_{u, \mathbf{x}} u \\ & \text{unter den Nebenbedingungen} \\ & x_1 + x_2 - u \geq 0 \end{aligned} \tag{5.3}$$

$$x_1 + x_2 + x_3 + x_4 = 1 \tag{5.4}$$

$$x_i \geq 0, \quad i = 1, 2, 3, 4. \tag{5.5}$$

Die Anzahl der Nebenbedingungen vom \geq -Typ (5.3) bezeichnen wir in der Implementation als $M2$ und die Anzahl der Nebenbedingungen vom $=$ -Typ als $M3$. Wir führen nun eine Schlupfvariable $s_{\{1,2\}}$ in das lineare Programm LP ein und erhalten somit

$$\begin{aligned} \text{LP : } \quad & \max_{u, \mathbf{x}, s_{\{1,2\}}} u \\ & \text{unter den Nebenbedingungen} \\ & x_1 + x_2 - u - s_{\{1,2\}} = 0 \\ & x_1 + x_2 + x_3 + x_4 = 1 \\ & s_{\{1,2\}} \geq 0, \quad x_i \geq 0, \quad i = 1, 2, 3, 4. \end{aligned} \tag{5.6}$$

In unserer Implementation ordnen wir jeder Variablen einen Index zu, um im Optimum zwischen den Basis-, Nichtbasis-, und Schlupfvariablen unterscheiden zu können. Den Variablen von LP, also x_1, x_2, \dots, x_{Sp} und u , ordnen wir Indizes von eins bis $Sp + 1$ zu. Den Schlupfvariablen weisen wir dann die Indizes beginnend mit $Sp + 2$ zu. Für das obige lineare Programm stellen sich die Indizes der Variablen folgendermassen dar:

Variable	x_1	x_2	x_3	x_4	u	$s_{\{1,2\}}$
Index	1	2	3	4	5	6

5.3 Implementation des Algorithmus von Wolsey

Die Lösung des linearen Programms LP erfolgt in unserer Implementation durch das Unterprogramm **RevSimplex**. Um Speicherplatz zu sparen, haben wir, wie in der Implementation des Algorithmus von Potters et al. [105], die Spalten der Schlupfvariablen nicht gespeichert. Im Unterprogramm **RevSimplex** nehmen für das lineare Programm (5.3) die Koeffizientenmatrix **A** der Nebenbedingungen, der Vektor **B** der rechten Seite der Nebenbedingungen und der Vektor **C** der Zielfunktionskoeffizienten die folgende Gestalt an:⁹

$$\mathbf{A} = \begin{pmatrix} 1 & 1 & 0 & 0 & -1 \\ 1 & 1 & 1 & 1 & 0 \end{pmatrix}, \mathbf{B} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \mathbf{C} = (0 \ 0 \ 0 \ 0 \ 1)^T.$$

Als Lösung von LP erhalten wir

$$x_0^{0,0} := u = 1 \text{ und } \mathbf{x}^{0,0} = (x_1^{0,0}, x_2^{0,0}, x_3^{0,0}, x_4^{0,0}) = (0, 1, 0, 0).$$

Da $\mathcal{V} := \emptyset$ ist, gilt $n^k = 0$. Folglich stellen wir das folgende binäre lineare Programm IP auf:

$$\begin{aligned} \text{IP :} \quad & \min \ 0 y_1 + 1 y_2 + 0 y_3 + 0 y_4 \\ & \text{unter der Nebenbedingung} \\ & 15y_1 + 14y_2 + 9y_3 + 8y_4 \geq 24 \\ & y_j \in \{0, 1\}, \ j = 1, 2, 3, 4. \end{aligned}$$

Ein binäres lineares Programm der Gestalt IP ist ein sog. Rucksackproblem, das wir im Anhang A.4 vorstellen. In unserer Implementation haben wir zur Lösung von Rucksackproblemen der Gestalt (5.2) (vgl. S. 226) die Unterprogramme **Knapsack** und **Kth_Best** erstellt. Wir berechnen mit Hilfe dieser Unterprogramme eine optimale Lösung der obigen Optimierungsaufgabe IP. Die Lösung ergibt den optimalen Zielfunktionswert $z^{0,0} := 0$ und $\mathbf{y}^{0,0} = (y_1^{0,0}, y_2^{0,0}, y_3^{0,0}, y_4^{0,0}) = (1, 0, 1, 1)$. Im nächsten Schritt (Schritt 3 des Algorithmus) vergleichen wir den berechneten op-

⁹Aus programmtechnischen Gründen bezeichnen wir in unserer Implementation auch die Vektoren mit grossen Buchstaben.

5 Implementation zweier Verfahren

timale Zielfunktionswert von IP mit dem optimalen Zielfunktionswert von LP. Da $z^{0,0} = 0 < x_0^{0,0} = 1$ ist, setzen wir

$$\begin{aligned} r &:= r + 1 = 1, \\ \hat{S} &:= S(\mathbf{y}^{0,0}) = \{1, 3, 4\}, \\ \mathcal{W}_{0,1} &:= \mathcal{W}_{0,0} \cup \hat{S} = \{\{1, 2\}, \{1, 3, 4\}\}. \end{aligned}$$

Nun beginnen wir mit der nächsten Iteration $(k, r) = (0, 1)$. Wir bilden ein neues lineares Programm LP, indem wir die der Gewinnkoalition $\hat{S} = \{1, 3, 4\}$ zugehörige Nebenbedingung dem ersten linearen Programm hinzufügen:

$$\begin{aligned} \text{LP : } \quad & \max_{u, \mathbf{x}} u \\ & \text{unter den Nebenbedingungen} \\ & x_1 + x_2 - u \geq 0 \\ & x_1 + x_3 + x_4 - u \geq 0 \\ & x_1 + x_2 + x_3 + x_4 = 1 \\ & x_i \geq 0, \quad i = 1, 2, 3, 4. \end{aligned} \tag{5.7}$$

Als Lösung erhalten wir $x_0^{0,1} := u = 1$ und $\mathbf{x}^{0,1} = (x_1^{0,1}, x_2^{0,1}, x_3^{0,1}, x_4^{0,1}) = (1, 0, 0, 0)$. Da die Menge \mathcal{V} immer noch leer ist, gilt erneut $n^k = 0$. Demzufolge stellen wir die folgende binäre lineare Optimierungsaufgabe IP auf:

$$\begin{aligned} \text{IP : } \quad & \min 1 y_1 + 0 y_2 + 0 y_3 + 0 y_4 \\ & \text{unter der Nebenbedingung} \\ & 15y_1 + 14y_2 + 9y_3 + 8y_4 \geq 24 \\ & y_j \in \{0, 1\}, \quad j = 1, 2, 3, 4. \end{aligned}$$

Als eine optimale Lösung erhalten wir $\mathbf{y}^{0,1} = (y_1^{0,1}, y_2^{0,1}, y_3^{0,1}, y_4^{0,1}) = (0, 1, 1, 1)$ mit dem optimalen Zielfunktionswert $z^{0,1} = 0$. Da wiederum $z^{0,1} = 0 < x_0^{0,1} = 1$ ist, setzen wir $r := r + 1 = 2$, $\hat{S} := S(\mathbf{y}^{0,1}) = \{2, 3, 4\}$ und $\mathcal{W}_{0,2} := \mathcal{W}_{0,1} \cup \hat{S} = \{\{1, 2\}, \{1, 3, 4\}, \{2, 3, 4\}\}$. Somit gelangen wir zur Iteration $(k, r) = (0, 2)$. Das nun zu lösende lineare Programm ist:

5.3 Implementation des Algorithmus von Wolsey

$$\begin{aligned}
\text{LP : } \quad & \max_{u, \mathbf{x}} u \\
& \text{unter den Nebenbedingungen} \\
& x_1 + x_2 - u \geq 0 \\
& x_1 + x_3 + x_4 - u \geq 0 \\
& x_2 + x_3 + x_4 - u \geq 0 \\
& x_1 + x_2 + x_3 + x_4 = 1 \\
& x_i \geq 0, \quad i = 1, 2, 3, 4.
\end{aligned} \tag{5.8}$$

Eine optimale Lösung des linearen Programms (5.8) lautet $x_0^{0,2} := u = \frac{2}{3}$ und $\mathbf{x}^{0,2} = (x_1^{0,2}, x_2^{0,2}, x_3^{0,2}, x_4^{0,2}) = (\frac{1}{3}, \frac{1}{3}, 0, \frac{1}{3})$. Da $|\mathcal{V}| = 0$ ist, gilt in dieser Iteration immer noch $n^k = 0$. Wir lösen also die folgende binäre lineare Optimierungsaufgabe:

$$\begin{aligned}
\text{IP: } \quad & \min \frac{1}{3} y_1 + \frac{1}{3} y_2 + 0 y_3 + \frac{1}{3} y_4 \\
& \text{unter der Nebenbedingung} \\
& 15y_1 + 14y_2 + 9y_3 + 8y_4 \geq 24 \\
& y_j \in \{0, 1\}, \quad j = 1, 2, 3, 4.
\end{aligned}$$

Eine optimale Lösung von IP lautet $\mathbf{y}^{0,2} = (y_1^{0,2}, y_2^{0,2}, y_3^{0,2}, y_4^{0,2}) = (1, 0, 1, 0)$, wobei $z^{0,2} := \frac{1}{3}$ der optimale Zielfunktionswert ist. Da nun erneut $z^{0,2} = \frac{1}{3} < x_0^{0,2} = \frac{2}{3}$ gilt, setzen wir $r := r + 1 = 3$, $\hat{S} := S(\mathbf{y}^{0,2}) = \{1, 3\}$ und $\mathcal{W}_{0,3} := \mathcal{W}_{0,2} \cup \hat{S} = \{\{1, 2\}, \{1, 3, 4\}, \{2, 3, 4\}, \{1, 3\}\}$. Das lineare Programm für die nächste Iteration $(k, r) = (0, 3)$ ist:

$$\begin{aligned}
\text{LP : } \quad & \max_{u, \mathbf{x}} u \\
& \text{unter den Nebenbedingungen} \\
& x_1 + x_2 - u \geq 0 \\
& x_1 + x_3 + x_4 - u \geq 0 \\
& x_2 + x_3 + x_4 - u \geq 0 \\
& x_1 + x_3 - u \geq 0 \\
& x_1 + x_2 + x_3 + x_4 = 1 \\
& x_i \geq 0, \quad i = 1, 2, 3, 4.
\end{aligned} \tag{5.9}$$

5 Implementation zweier Verfahren

Als eine optimale Lösung dieses linearen Programms erhalten wir $x_0^{0,3} := u = \frac{2}{3}$ und $\mathbf{x}^{0,3} = (x_1^{0,3}, x_2^{0,3}, x_3^{0,3}, x_4^{0,3}) = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0)$. Somit erhalten wir die folgende Optimierungsaufgabe IP:

$$\begin{aligned} \text{IP :} \quad & \min \frac{1}{3} y_1 + \frac{1}{3} y_2 + \frac{1}{3} y_3 + 0 y_4 \\ & \text{unter der Nebenbedingung} \\ & 15y_1 + 14y_2 + 9y_3 + 8y_4 \geq 24 \\ & y_j \in \{0, 1\}, \quad j = 1, 2, 3, 4. \end{aligned}$$

Die Lösung der Optimierungsaufgabe IP ergibt $\mathbf{y}^{0,3} = (y_1^{0,3}, y_2^{0,3}, y_3^{0,3}, y_4^{0,3}) = (1, 1, 0, 1)$ mit dem optimalen Zielfunktionswert $z^{0,3} = \frac{2}{3}$. Da nun $z^{0,3} = \frac{2}{3} \geq x_0^{0,2} = \frac{2}{3}$ ist, überprüfen wir gemäss dem Algorithmus, ob die Lösung von LP eindeutig ist.

Die Lösung eines linearen Programms ist eindeutig, wenn im Optimum keiner der Koeffizienten der Schlupfvariablen, die in der optimalen Lösung des linearen Programms Nichtbasisvariablen sind, in der Zielfunktionszeile null ist (vgl. dazu Anhang A.2, S. 291). Also erfordert die Überprüfung der Eindeutigkeit einer Lösung eines linearen Programms in unserer Implementation, dass wir zwischen den Basis- und Nichtbasisvariablen unterscheiden können und die Koeffizienten in der Zielfunktionszeile des optimalen Simplex-Tableaus kennen. Zu diesem Zweck definieren wir im Unterprogramm `RevSimplex` eine Menge `BasVar` und speichern die Indizes der Basisvariablen in dieser Menge. Eine Variable, deren Index nicht in `BasVar` ist, ist dann eine Nichtbasisvariable. Überdies definieren wir im Unterprogramm `RevSimplex` einen $(Sp + 1 + M2)$ -dimensionalen Vektor `OptZielFktZeile` und speichern die Zielfunktionskoeffizienten der Variablen $x_1, x_2, \dots, x_{Sp}, u$ als die ersten $Sp + 1$ Einträge und die insgesamt $M2$ Schlupfvariablen als die letzten $M2$ Einträge in diesem Vektor. Gemäss unserer Berechnung lauten in der Iteration $(k, r) = (0, 3)$ im Optimum des linearen Programms LP `BasVar` = $(1, 2, 3, 4, 5)$ und `OptZielFktZeile` = $(0, 0, 0, 0, 0, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0)$. Das heisst, dass die Variablen x_1, x_2, x_3, x_4 und u im Optimum Basisvariablen und die anderen Variablen Nichtbasisvariablen sind. Zudem besitzen die Koeffizienten der Variablen $x_1, x_2, \dots, x_{Sp}, u$ und der Schlupfvariablen in der Zielfunktionszeile des optimalen Simplex-Tableaus die folgenden Werte:

5.3 Implementation des Algorithmus von Wolsey

Variable	x_1	x_2	x_3	x_4	u	$s_{\{1,2\}}$	$s_{\{1,3,4\}}$	$s_{\{2,3,4\}}$	$s_{\{1,3\}}$
Koeffizient in OptZielFktZeile	0	0	0	0	0	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	0

Da der Koeffizient der Schlupfvariablen $s_{\{1,3\}}$, die im Optimum eine Nichtbasisvariable ist, in der Zielfunktionszeile des optimalen Simplex-Tableaus den Wert Null aufweist, ist die Lösung des linearen Programms nicht-eindeutig.

Folglich bestimmen wir die Menge $\mathcal{V}_{k+1} = \mathcal{V}_1$, also die Menge derjenigen Koalitionen, die für alle optimalen Lösungen des linearen Programms konstante Koalitionsüberschüsse haben. Das sind Koalitionen in $\mathcal{W}_{k,r}$, deren zugehörige Nebenbedingungen in LP für alle optimalen Lösungen als Gleichung erfüllt sind. Dies ist der Fall, wenn die diesen Nebenbedingungen zugehörigen Schlupfvariablen für alle optimalen Lösungen den Wert Null haben. Gemäss der Komplementärschlupfbedingung (vgl. Anhang A.1, S. 287) haben jene Schlupfvariablen, die in der Zielfunktionszeile des optimalen Simplex-Tableaus positive Koeffizienten besitzen, für alle optimalen Lösungen den Wert Null.¹⁰ Da die Koeffizienten der Basisvariablen in der Zielfunktionszeile immer Nullen sind, untersuchen wir in unserer Implementation nur diejenigen Schlupfvariablen, die Nichtbasisvariablen sind. In unserem Beispiel besitzen die Schlupfvariablen $s_{\{1,2\}}$, $s_{\{1,3,4\}}$ und $s_{\{2,3,4\}}$ in der Zielfunktionszeile positive Koeffizienten. Demzufolge gilt

$$\mathcal{V}_{k+1} = \mathcal{V}_1 := \{\{1, 2\}, \{1, 3, 4\}, \{2, 3, 4\}\}.$$

Die ersten drei Nebenbedingungen von (5.9) werden nun im nächsten linearen Programm zu Nebenbedingungen mit Gleichheitszeichen.

Gemäss dem Algorithmus von Wolsey setzen wir jetzt zunächst $x_0^0 := x_0^{0,3} = \frac{2}{3}$, $r := 0$ und $k := k + 1 = 1$. Anschliessend stellen wir das folgende lineare Programm LP für $(k, r) = (1, 0)$ auf:

¹⁰Für weitere Erläuterungen dieser Aussage vgl. z. B. Gass [44], S. 126, Bloech [10], S. 86, und Krelle/Künzi [67], S. 73 f.

5 Implementation zweier Verfahren

$$\begin{aligned}
\text{LP : } \quad & \max_{u, \mathbf{x}} u \\
& \text{unter den Nebenbedingungen} \\
& x_1 + x_3 - u \geq 0 \\
& x_1 + x_2 = 2/3 \\
& x_1 + x_3 + x_4 = 2/3 \\
& x_2 + x_3 + x_4 = 2/3 \\
& x_1 + x_2 + x_3 + x_4 = 1 \\
& x_i \geq 0, \quad i = 1, 2, 3, 4.
\end{aligned} \tag{5.10}$$

Als Lösung erhalten wir $x_0^{1,0} := u = \frac{2}{3}$ und $\mathbf{x}^{1,0} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0)$. Für $n^1 := |\mathcal{V}_1| = 3$ lautet nun das zu lösende binäre lineare Programm IP

$$\begin{aligned}
\text{IP: } \quad & 4. \text{ Minimum von } \frac{1}{3} y_1 + \frac{1}{3} y_2 + \frac{1}{3} y_3 + 0 y_4 \\
& \text{unter der Nebenbedingung} \\
& 15y_1 + 14y_2 + 9y_3 + 8y_4 \geq 24 \\
& y_j \in \{0, 1\}, \quad j = 1, 2, 3, 4.
\end{aligned}$$

Unsere Berechnung ergibt als eine optimale Lösung $\mathbf{y}^{1,0} = (1, 1, 0, 1)$ mit dem optimalen Zielfunktionswert $z^{1,0} := \frac{2}{3}$. Es gilt $z^{1,0} = \frac{2}{3} \geq x_0^{1,0} = \frac{2}{3}$. Mit Hilfe des Unterprogramms **RevSimplex** stellen wir fest, dass die Variablen x_1, x_2, x_3, x_4, u im Optimum Basisvariablen sind und **OptZielFktZeile** = $(0, 0, 0, 0, 0, 1)$ ist. Das heisst, dass die einzige Schlupfvariable $s_{\{1,3\}}$ im Optimum eine Nichbasisvariable ist und in der Zielfunktionszeile den Koeffizienten 1 besitzt. Da der Koeffizient von $s_{\{1,3\}}$ nicht null ist, ist die Lösung des letzten linearen Programms LP eindeutig. Folglich ist der Nukleolus des gewichteten Abstimmungsspiels $\Gamma = [24; 15, 14, 9, 8]$ gefunden. Er lautet $\mathcal{N} = \mathbf{x}^{1,0} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0)$.

Wir stellen fest, dass der Algorithmus von Wolsey den Nukleolus des gewichteten Abstimmungsspiels $\Gamma = [24; 15, 14, 9, 8]$ mit Hilfe von lediglich fünf Gewinnkoalitionen bestimmt hat. Es ist also nicht notwendig, zur Berechnung des Nukleolus alle auf S. 226 angegebenen Gewinnkoalitionen heranzuziehen.

5.3 Implementation des Algorithmus von Wolsey

Im Folgenden geben wir zunächst einige Hinweise auf die in unserer Implementation verwendeten Computerprogramme zur Lösung von linearen und binären linearen Programmen. Anschliessend erläutern wir zwei Besonderheiten unserer Implementation. Zum Schluss des Abschnitts beschreiben wir die in der Implementation des Algorithmus von Wolsey verwendeten Unterprogramme.

Die Berechnung des $(n^k + 1)$ -besten Minimums von binären linearen Programmen erfolgt in unserer Implementation mit Hilfe des Algorithmus von Lawler [72].¹¹ Das in unserer Implementation dieses Algorithmus verwendete Computerprogramm zur Berechnung einer optimalen Lösung der Rucksackprobleme stammt aus Parson [100], S. 361 ff. Zur Lösung linearer Programme LP haben wir die revidierte Simplex-Methode verwendet. Das Computerprogramm dazu beruht zum grössten Teil auf Syslo/Deo/Kowalik [139], S. 3 ff., aber auch auf Gass [44], S. 107 ff., sowie auf Niemeyer [91], S. 159 ff., und S. 213 ff. Wir haben das Computerprogramm von Syslo/Deo/Kowalik [139] folgendermassen erweitert, um alle für den Algorithmus von Wolsey erforderlichen Informationen über optimale Lösungen von linearen Programmen zu gewinnen und den für die jeweiligen Berechnungen erforderlichen Speicherplatz zu reduzieren: Unsere in der beigelegten CD angegebene Implementation der revidierten Simplex-Methode liefert neben einer Lösung des jeweiligen linearen Programms auch die Information, ob diese optimale Lösung eindeutig ist. Zudem bestimmt unsere Implementation alle Nebenbedingungen des linearen Programms LP, die für alle optimalen Lösungen von LP als Gleichung erfüllt sind. Wie wir oben bereits erwähnt haben, erfolgt die Bestimmung dieser Nebenbedingungen über die Komplementärschlupfbedingung. Ferner speichern wir in unserer Implementation keine Koeffizienten der Schlupfvariablen im Simplex-Tableau ab, um weniger Speicherplatz in Anspruch zu nehmen.

Nun erläutern wir zwei besondere Merkmale unserer Implementation. Wir haben nämlich festgestellt, dass im Schritt 3 des Algorithmus von Wolsey die Menge \mathcal{V}_{k+1} alle Koalitionen in $\mathcal{W}_{k,r}$ beinhalten kann, d. h. die Situation $\mathcal{V}_{k+1} = \mathcal{W}_{k,r}$ eintreten kann. In diesem Fall sind alle Nebenbedingungen von LP in der Iteration $(k + 1, r)$

¹¹Diesen Algorithmus stellen wir im Anhang A.5 vor.

5 Implementation zweier Verfahren

Gleichungen, wobei die einzige Zielfunktionsvariable u in keiner dieser Gleichungen erscheint. Das impliziert, dass wir die Zielfunktion beliebig erhöhen können, wenn das Gleichungssystem der Nebenbedingungen eine Lösung besitzt. Folglich gilt im Schritt 3 des Algorithmus $z^{k+1,r} < x_0^{k+1,r}$. Um in diesem Fall mit der Berechnung fortfahren zu können, muss gemäss dem Algorithmus von Wolsey eine binäre lineare Optimierungsaufgabe IP aufgestellt werden. Die Aufstellung einer solchen Optimierungsaufgabe erfordert aber eine optimale Lösung des linearen Programms LP. Für $\mathcal{V}_{k+1} = \mathcal{W}_{k,r}$ existiert jedoch keine solche Lösung. In unserer Implementation haben wir in diesem Fall mit Hilfe des Lösungsvektors des Gleichungssystems der Nebenbedingungen von LP die binäre lineare Optimierungsaufgabe IP aufgestellt. Falls das lineare Programm LP eine zulässige Lösung besitzt, dann ist diese gleichzeitig eine Lösung des Gleichungssystems der Nebenbedingungen. Wir bestimmen diese zulässige Lösung von LP mit Hilfe der von uns implementierten revidierten Simplex-Methode. In einem speziellen Fall, in dem die Koeffizientenmatrix des Gleichungssystems der Nebenbedingungen von LP eine quadratische Matrix ist, überprüfen wir die Eindeutigkeit der Lösung des Gleichungssystems. Ist nämlich diese Lösung eindeutig, so ist sie gleichzeitig der Nukleolus des Spiels. Zu diesem Zweck berechnen wir die Determinante dieser quadratischen Koeffizientenmatrix der Nebenbedingungen. Wenn die Determinante von null verschieden ist, dann ist die Lösung eindeutig.¹²

Nun erläutern wir eine andere Besonderheit unserer Implementation. Wir haben oben bereits dargelegt, dass wir in der Iteration (k, r) im Schritt 3 des Algorithmus von Wolsey eine Koalition $\hat{S} := S(\mathbf{y}^{k,r})$ bestimmen, falls $z^{k,r} < x_0^{k,r}$ vorliegt. Anschliessend stellen wir das neue lineare Programm LP auf, indem wir $\mathcal{W}_{k,r} := \mathcal{W}_{k,r-1} \cup \hat{S}$ setzen, also die der Koalition \hat{S} zugehörige Nebenbedingung als neue Nebenbedingung dem linearen Programm LP hinzufügen. Wie wir während unserer

¹²Ein Gleichungssystem hat dann und nur dann eine eindeutige Lösung, wenn der Rang der Koeffizientenmatrix des Gleichungssystems mit der Anzahl der Unbekannten des Gleichungssystems übereinstimmt. In einem Gleichungssystem mit n Gleichungen und n Unbekannten ist die Koeffizientenmatrix des Gleichungssystems quadratisch. In diesem Fall besitzt das Gleichungssystem eine eindeutige Lösung, wenn der Rang der Koeffizientenmatrix des Gleichungssystems n ist, also die Determinante der Koeffizientenmatrix des Gleichungssystems nicht null ist. Wenn die Determinante null ist, dann existiert entweder eine nicht-eindeutige Lösung oder es gibt unendlich viele Lösungen, und man sagt, die Matrix ist singulär (vgl. dazu Anhang D und Padberg/Kütting [99], S. 216 f.).

5.3 Implementation des Algorithmus von Wolsey

Implementation festgestellt haben, kann es vorkommen, dass die der Koalition \hat{S} zugehörige Restriktion bereits eine Nebenbedingung des linearen Programms ist.¹³ Um dies zu umgehen, können wir in der Iteration $(k + 1, r)$ die Nebenbedingungen $\mathbf{y} \in Y(\Gamma)$ von IP durch

$$\mathbf{y} \in Y(\Gamma) \setminus \{\mathbf{y} \mid S(\mathbf{y}) \in \mathcal{V}_k \cup \mathcal{W}_{k,r}\} \quad (5.11)$$

ersetzen. Die Umsetzung der Bedingung 5.11 in unserer Implementation erfolgt dadurch, dass wir die Nebenbedingungen $\mathbf{y} \in Y(\Gamma)$ beibehalten und nach jeder Lösung von IP überprüfen, ob die gefundene Lösung schon eine Nebenbedingung von LP ist oder nicht.

Zum Schluss des Abschnitts beschreiben wir die Unterprogramme, die wir in der Implementation des Algorithmus von Wolsey verwendet haben. Auch diesen Algorithmus haben wir in der Programmiersprache Delphi 6 implementiert. Der gesamte Quellcode unserer Implementation wird in der beigelegten CD angegeben. Wie in der Implementation des Verfahrens von Potters et al. haben wir auch hier verschiedene Unterprogramme erstellt:

Procedure Austausch stellt bei der Berechnung der Determinante fest, ob ein Zeilentausch nötig ist und führt diesen Tausch gegebenenfalls durch.

Procedure Determinante berechnet die Determinante einer quadratischen Matrix durch Gauss-Elimination.¹⁴

Procedure Eingabe liest die Daten aus der Datei ab, in der die zum Start benötigten Angaben gespeichert sind.

Procedure Eingabe_Kth_Best bereitet die für die Aufstellung des binären linearen Programms benötigten Werte vor.

Procedure EingabeRevSimp stellt die Koeffizientenmatrix **A** der Nebenbedingungen, den Vektor der rechten Seite der Nebenbedingungen **B** und den Vektor der

¹³Dieser Fall kann vorliegen, weil die Nebenbedingung jedes binären linearen Programms IP in jeder Iteration immer dieselbe, nämlich $\mathbf{y} \in Y(\Gamma)$, ist.

¹⁴Dies erfolgt dadurch, dass wir zunächst die Matrix in eine obere Dreiecksmatrix transformieren. Die Determinante ist dann das Produkt der Hauptdiagonalelemente. Sobald während dieser Transformation ermittelt wird, dass ein Hauptdiagonalelement den Wert Null besitzt, wird im Unterprogramm **Determinante** die Berechnung abgebrochen, weil in diesem Fall die Determinante null und somit die Matrix singulär ist.

5 Implementation zweier Verfahren

Zielfunktionskoeffizienten \mathbf{C} des linearen Programms LP auf.

Procedure Knapsack formuliert die im Algorithmus von Wolsey beschriebene binäre lineare Minimierungsaufgabe IP als eine äquivalente Maximierungsaufgabe und berechnet deren optimale Lösung.

Procedure Kth_Best berechnet die k -beste Lösung eines Rucksack-Problems gemäss dem Verfahren von Lawler [72] und überprüft, ob diese Lösung bereits einer Zeile der Matrix \mathbf{A} entspricht. Wenn dies der Fall ist, sucht dieses Unterprogramm eine andere optimale Lösung, die mit keiner Zeile der Matrix \mathbf{A} übereinstimmt.

Procedure RevSimplex löst ein lineares Optimierungsproblem mit Hilfe der revidierten Simplex-Methode.

Procedure Ueberpruefen untersucht, ob der gefundene optimale Zielfunktionswert von IP mindestens so gross wie der gefundene optimale Zielfunktionswert von LP ist. Wenn dies zutrifft und die Lösung von LP eindeutig ist, liefert das Unterprogramm als Ausgabe den Nukleolus des Spiels.

Procedure ZeitAufw stellt die Laufzeit des Algorithmus in Millisekunden fest.

Procedure VetoGame ist ein Unterprogramm, das wir bei der Implementierung des Verfahrens sowohl von Potters et al. als auch von Wolsey verwenden. Dieses Unterprogramm kommt nur bei Abstimmungsspielen zur Anwendung. Bevor die eigentlichen Berechnungen beginnen, überprüft es, ob es im gegebenen einfachen kooperativen Spiel Vetospieler gibt. Der Nukleolus einfacher kooperativer Spiele, die Vetospieler enthalten, kann nämlich ohne grossen Rechenaufwand berechnet werden. Um das zu erläutern, nehmen wir an, dass \mathcal{V} die Menge der Vetospieler sei. Die Elemente dieser Menge haben wir auf S. 32 wie folgt definiert:

$$\mathcal{V} := \{i : v(N \setminus \{i\}) = 0\}. \quad (5.12)$$

Ist die Menge \mathcal{V} nicht-leer, so sind alle Vetospieler Mitglieder jeder Gewinnkoalition. Das heisst, dass nur die der jeweiligen Gewinnkoalition zugehörige Restriktion eine Nebenbedingung des linearen Programms LP sein kann. Dies impliziert, dass bei der Berechnung des Nukleolus jeder Vetospieler in allen Nebenbedingungen des linearen Programms LP erscheinen muss. Ferner ist der Kern jedes einfachen kooperativen

5.4 Vergleich der implementierten Verfahren

Spiels nicht-leer, falls es in diesem Spiel Vetospieler gibt (vgl. Beispiel 2.2.1, S. 31).¹⁵ Zudem ist jede nicht-negative Aufteilung von $v(N) = 1$ zwischen den Vetospielern eine Kernallokation (vgl. dazu Curiel [28], S. 16). Da der Nukleolus eines Spiels stets im nicht-leeren Kern des Spiels liegt, ist er eine der Kernallokationen. Gemäss der Anonymitätseigenschaft des Nukleolus, wonach der Nukleolus unabhängig von der Nummerierung der Spieler ist, gilt folglich, dass der Nukleolus eines einfachen kooperativen Spiels mit nicht-leerer Vetospielmengemenge \mathcal{V} eine Gleichverteilung von $v(N) = 1$ zwischen den Vetospielern ist. Für Abstimmungsspiele mit $\mathcal{V} \neq \emptyset$ berechnet das Unterprogramm `VetoGame` in der Implementation beider Algorithmen den Nukleolus als Gleichverteilung des Wertes der grossen Koalition.

5.4 Vergleich der implementierten Verfahren

In diesem Abschnitt vergleichen wir das Verfahren von Wolsey mit dem Verfahren von Potters et al. an Hand unserer Implementationen. Da wir mit Hilfe des Verfahrens von Wolsey lediglich den Nukleolus einfacher kooperativer Spiele berechnen können, betrachten wir für den Vergleich ausschliesslich gewichtete Abstimmungsspiele mit n ($n = 3, 4, \dots, 25$) Spielern. Für die obere Grenze der Anzahl der Spieler im jeweiligen Verfahren ist der Speicherbedarf des jeweiligen Algorithmus ausschlaggebend, weil bei der Programmiersprache Delphi eine Matrix höchstens ein Gigabyte Speicherplatz beanspruchen kann. Das Verfahren von Potters et al. zieht für die Lösung des ersten linearen Programms ein Simplex-Tableau mit $(2^n - 1) \times (n + 1)$ Einträgen (ohne die Spalten der Basisvariablen) heran und reduziert nach jeder Iteration die Dimension des Simplex-Tableaus. Derweil geht das Verfahren von Wolsey in unserer Implementation für das erste lineare Programm von einem Simplex-Tableau mit $2 \times (n + 1)$ Einträgen (ohne die Spalten der Basisvariablen) aus und erhöht in jeder Iteration die Anzahl der Zeilen höchstens um eine neue Zeile. In unserer Implementation haben wir die Einträge der Simplex-Tableaus als Variablen mit 19- bis 20-stelliger Genauigkeit definiert (sog. Variablentyp „extended“). Die Programmiersprache Del-

¹⁵Eine Beweisführung dieser Aussage finden wir in Fargó et al. [38], S. 231.

5 Implementation zweier Verfahren

phi reserviert für jede solche Variable zehn Byte Speicherplatz. Demzufolge können wir mit unserer Implementation des Algorithmus von Potters et al. den Nukleolus von kooperativen Spielen mit praktisch maximal ca. 20 Spielern berechnen. Diese Zahl kann erhöht werden, wenn die Variablen mit einer niedrigeren Genauigkeit definiert sind. Werden z. B. die Einträge mit 15- bis 16-stelliger Genauigkeit (sog. Variablentyp „double“) oder mit 7- bis 8-stelliger Genauigkeit (sog. Variablentyp „real“) definiert, so nimmt jede Variable nur noch acht bzw. vier Byte Speicherplatz in Anspruch. Die Erhöhung der Spielerzahl auf diese Weise kann aber in der Praxis problematisch werden, weil diese Technik aufgrund der niedrigeren Rechengenauigkeit der Berechnungen zu falschen Ergebnissen führen kann. Im Gegensatz zu der Implementation des Verfahrens von Potters et al. können wir für die Berechnung des Nukleolus mit unserer Implementation des Verfahrens von Wolsey a priori keine Aussage über die effektive maximale Anzahl der Spieler machen. Denn die Anzahl der zur Berechnung des Nukleolus eines einfachen kooperativen Spiels erforderlichen Gewinnkoalitionen ist im Voraus nicht bekannt. Wir haben in dieser Arbeit mit unserer Implementation des Verfahrens von Wolsey den Nukleolus gewichteter Abstimmungsspiele mit bis zu 25 Spielern berechnet.

Um die implementierten Verfahren miteinander zu vergleichen, haben wir den Nukleolus verschiedener zufällig ausgewählter gewichteter Abstimmungsspiele mit Hilfe der beiden Verfahren berechnet.¹⁶ Für jedes gewichtete Abstimmungsspiel mit n ($n = 3, 4, \dots, 25$) Spielern haben wir je fünf Spiele für den Vergleich herangezogen. Diese Testspiele sowie deren Nukleoli werden im Anhang H angegeben. Wie wir noch begründen werden, haben wir mit Hilfe des Verfahrens von Potters et al. den Nukleolus kooperativer Spiele mit bis zu 17 Spielern berechnet. Die Berechnung des Nukleolus jedes dieser Spiele mit den beiden Verfahren hat stets dieselben Resultate ergeben.

Wir gehen zuerst auf die Anzahl der linearen Optimierungsaufgaben ein, die jeweils zur Berechnung des Nukleolus der Testspiele gelöst werden. Anschliessend analysie-

¹⁶Alle Berechnungen wurden mit demselben Computer, nämlich mit einem COMPAQ Deskpro EP 6/400 II (Pentium 2, 400 MHz), durchgeführt.

5.4 Vergleich der implementierten Verfahren

Tabelle 5.12: Anzahl linearer Programme

Spielerzahl	Potters et al.			Wolsey		
	max.	min.	arith. Mittel	max.	min.	arith. Mittel
3	1	1	1	4	3	3.6
4	2	1	1.2	6	5	5.2
5	1	1	1	16	6	7.4
6	1	1	1	18	10	13.4
7	1	1	1	22	16	18.8
8	3	1	1.6	44	27	27.6
9	4	1	1.6	49	25	31.6
10	2	1	1.2	58	28	36.8
11	5	1	2.2	45	29	38.2
12	7	1	2.4	56	33	38.0
13	11	1	3.6	71	42	54.4
14	1	1	1	88	41	55.6
15	3	1	1.4	91	73	79.8
16	1	1	1	108	48	82.4
17	5	1	1.8	97	51	79.0
18	-	-	-	114	57	71.8
19	-	-	-	137	88	118.6
20	-	-	-	138	104	117.0
21	-	-	-	139	83	115.2
22	-	-	-	147	88	113.8
23	-	-	-	154	87	133.0
24	-	-	-	158	93	128.6
25	-	-	-	156	107	136.2

ren wir die Anzahl der Nebenbedingungen dieser linearen Programme. Diese beiden Aspekte haben einen wesentlichen Einfluss auf die Länge der Rechenzeit. Die Tabelle 5.12 gibt für beide Verfahren die maximale, die minimale und die durchschnittliche Anzahl der jeweiligen linearen Programme wieder. Tabelle 5.12 zeigt, dass das Verfahren von Potters et al. zur Berechnung der Nukleoli die Lösung von maximal elf

5 Implementation zweier Verfahren

linearen Programmen erfordert.¹⁷ Zudem nimmt das Verfahren von Wolsey immer die Lösung einer grösseren Anzahl linearer Programme als das Verfahren von Potters et al. in Anspruch.¹⁸ Hierbei ist zu beachten, dass die Anzahl der gelösten linearen Programme im Verfahren von Wolsey [150] unter anderem auch davon abhängt, von wie vielen Gewinnkoalitionen zu Beginn des Verfahrens ausgegangen wird. Da wir bei unseren Berechnungen zu Beginn von nur einer einzigen Gewinnkoalition ausgehen, stellt die Anzahl der linearen Programme für das Verfahren von Wolsey [150] eine obere Grenze dar.

Nun vergleichen wir die beiden Verfahren hinsichtlich der Anzahl der Nebenbedingungen der zu lösenden linearen Programme. Die Anzahl der Nebenbedingungen entspricht zugleich der Anzahl der bei der jeweiligen Berechnung herangezogenen Koalitionen. Während das Verfahren von Potters et al. zur Berechnung des Nukleolus eines kooperativen Spiels mit n Spielern alle möglichen $2^n - 1$ Koalitionen berücksichtigt, geht das Verfahren von Wolsey nur von Gewinnkoalitionen aus und ermittelt den Nukleolus des Spiels in der Regel mit Hilfe lediglich einer Teilmenge aller Gewinnkoalitionen. Die Anzahl der Gewinnkoalitionen, die ausreicht, um den Nukleolus eines einfachen kooperativen Spiels mit Hilfe des Algorithmus von Wolsey zu berechnen, stimmt mit der Anzahl der Nebenbedingungen des zuletzt gelösten linearen Programms überein. Die Tabelle 5.13 gibt für beide Verfahren die Anzahl der Nebenbedingungen zur Bestimmung der Nukleoli der Testspiele wieder. Da das Verfahren von Potters et al. immer von allen möglichen Koalitionen ausgeht, entsprechen die Werte in der zweiten Spalte der Tabelle 5.13 für $n = 3, 4, \dots, 25$ immer der Zahl $2^n - 1$. Vergleichen wir die Spalte 3 (die maximale Anzahl der Nebenbedingungen im Verfahren von Wolsey) mit der Spalte 2, so wird deutlich, dass das Verfahren von Wolsey im Vergleich zu dem Verfahren von Potters et al. bedeutend weniger Nebenbedingungen setzt, also Koalitionen heranzieht. Während im Algorith-

¹⁷Potters et al. [105] zeigen in ihrer Arbeit, dass ihr Verfahren den Nukleolus eines kooperativen Spiels mit n Spielern im Wege der Lösung höchstens $n - 1$ linearer Programme berechnet. Unsere Testspiele haben dies exemplarisch bestätigt.

¹⁸Auf den expliziten Ausweis derselben Anzahl der zusätzlich zu lösenden binären linearen Programme im Algorithmus von Wolsey haben wir verzichtet.

5.4 Vergleich der implementierten Verfahren

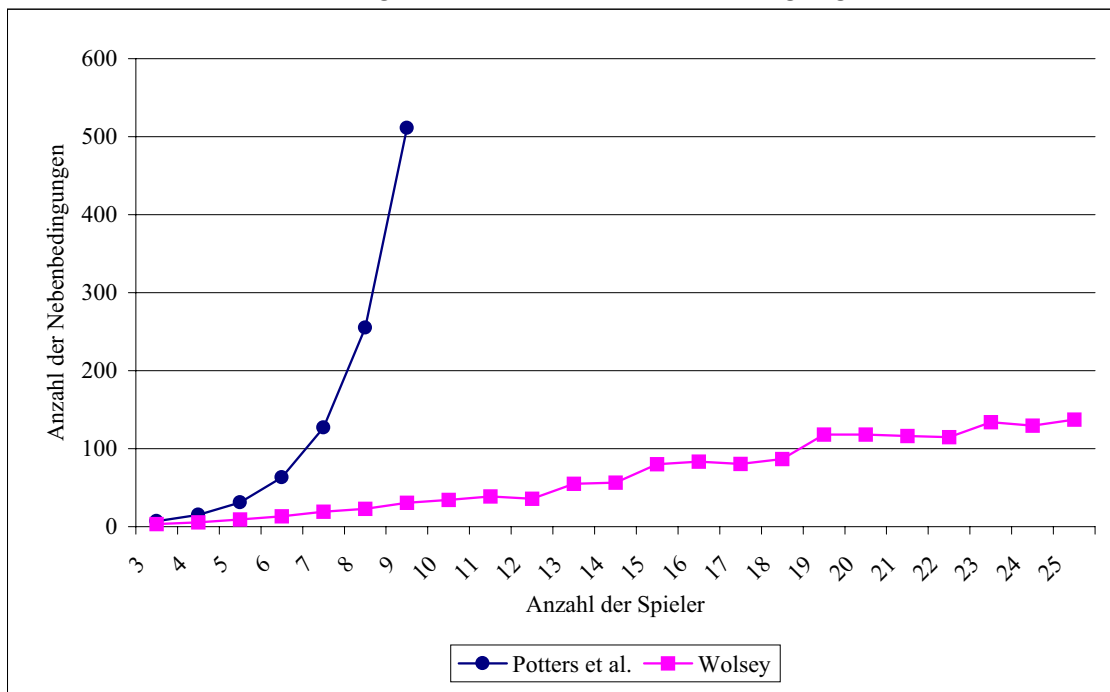
Tabelle 5.13: Anzahl der Nebenbedingungen

Spielerzahl	Potters et al.	Wolsey		
		max.	min.	arith. Mittel
3	7	4	3	3.4
4	15	6	5	5.4
5	31	12	7	9.4
6	63	19	10	13.4
7	127	22	16	19.2
8	255	35	16	23.0
9	511	41	26	30.6
10	1 023	42	29	34.2
11	2 047	45	29	38.8
12	4 095	41	30	35.8
13	8 191	71	43	55.0
14	16 383	88	36	56.4
15	32 767	91	72	80.2
16	65 535	109	49	83.4
17	131 071	98	52	80.6
18	262 143	114	58	86.6
19	524 287	134	89	118.0
20	1 048 575	139	105	118.0
21	2 097 151	140	84	116.2
22	4 194 303	148	89	114.8
23	8 388 607	155	88	134.0
24	16 777 215	159	94	129.6
25	33 554 431	157	108	137.2

mus von Wolsey die durchschnittliche Anzahl der Nebenbedingungen mit wachsender Spielerzahl nahezu linear steigt, erhöht sich die Anzahl der Nebenbedingungen im Algorithmus von Potters et al. exponentiell. Erhöhen wir beispielsweise die Anzahl der Spieler von sieben auf acht, steigt die durchschnittliche Anzahl der Nebenbedingungen im Verfahren von Wolsey (Tabelle 5.13, Spalte 5) noch von 19.2 auf 23 und im Verfahren von Potters et al. (Tabelle 5.13, Spalte 2) aber bereits von 127 auf 255. Demnach zieht das Verfahren von Wolsey in diesem Fall im Durchschnitt zusätzlich 3.8 Koalitionen heran, während das Verfahren von Potters et al. die Berücksichtigung

5 Implementation zweier Verfahren

Abbildung 5.3: Anzahl der Nebenbedingungen



von insgesamt 128 neuen Koalitionen erfordert. Folglich ist die Anzahl der zusätzlich benötigten Koalitionen im Verfahren von Potters et al. das $128/3.8 = 33.68$ fache der entsprechenden Koalitionen im Verfahren von Wolsey. Der Faktor ist bei einer Zunahme der Spielerzahl z. B. von 17 auf 18 noch grösser, nämlich das 19 859fache. In der Abbildung 5.3 wird die Entwicklung der Anzahl der Nebenbedingungen mit steigender Spielerzahl für beide Algorithmen graphisch illustriert.

Wir vergleichen nun den Zeitbedarf der beiden Algorithmen zur Berechnung des Nukleolus unserer Testspiele in Abhängigkeit von der Anzahl der Spieler. Tabelle 5.14 gibt die von uns ermittelten Rechenzeiten zur Berechnung des Nukleolus der Testspiele an, wobei die Berechnung der Nukleoli unserer Testspiele im RAM unseres Rechners erfolgt ist. Die Tabelle beinhaltet die jeweilige maximale, minimale und durchschnittliche Laufzeit beider Verfahren für jedes Spiel mit n Spielern.¹⁹ Tabelle

¹⁹Wegen der explodierenden Rechenzeit für das Verfahren von Potters et al. ab 17 Spieler haben wir die Berechnung abgebrochen. Beispielsweise haben wir die Berechnung des Nukleolus des im Anhang H angegebenen Spiels Γ_5^{18} mit 18 Spielern nach über 14 Stunden abgebrochen.

5.4 Vergleich der implementierten Verfahren

Tabelle 5.14: Rechenzeit in Sekunden

n	Potters et al.			Wolsey		
	max.	min.	arith. Mittel	max.	min.	arith. Mittel
3	0.220	0.060	0.132	0.000	0.000	0.000
4	0.270	0.000	0.130	0.060	0.000	0.012
5	0.330	0.000	0.130	0.000	0.000	0.000
6	0.270	0.050	0.110	0.600	0.000	0.022
7	0.490	0.170	0.352	0.110	0.000	0.044
8	0.390	0.000	0.242	0.666	0.110	0.330
9	0.610	0.330	0.506	0.710	0.220	0.306
10	0.710	0.330	0.556	0.820	0.220	0.428
11	0.770	0.710	0.748	0.770	0.280	0.562
12	0.930	0.720	0.846	0.870	0.490	0.622
13	6.700	1.930	3.022	1.420	0.710	1.068
14	7.300	2.750	4.392	3.350	0.610	1.310
15	12.910	7.690	9.274	5.830	2.140	3.064
16	22.850	16.970	19.534	10.320	0.760	4.424
17	85.360	32.180	54.768	5.440	0.930	3.406
18	-	-	-	12.740	1.100	5.778
19	-	-	-	35.700	4.450	22.574
20	-	-	-	48.100	9.290	24.024
21	-	-	-	74.210	3.570	29.902
22	-	-	-	74.700	4.230	28.322
23	-	-	-	114.960	4.620	51.214
24	-	-	-	120.450	4.610	57.704
25	-	-	-	118.740	11.640	64.356

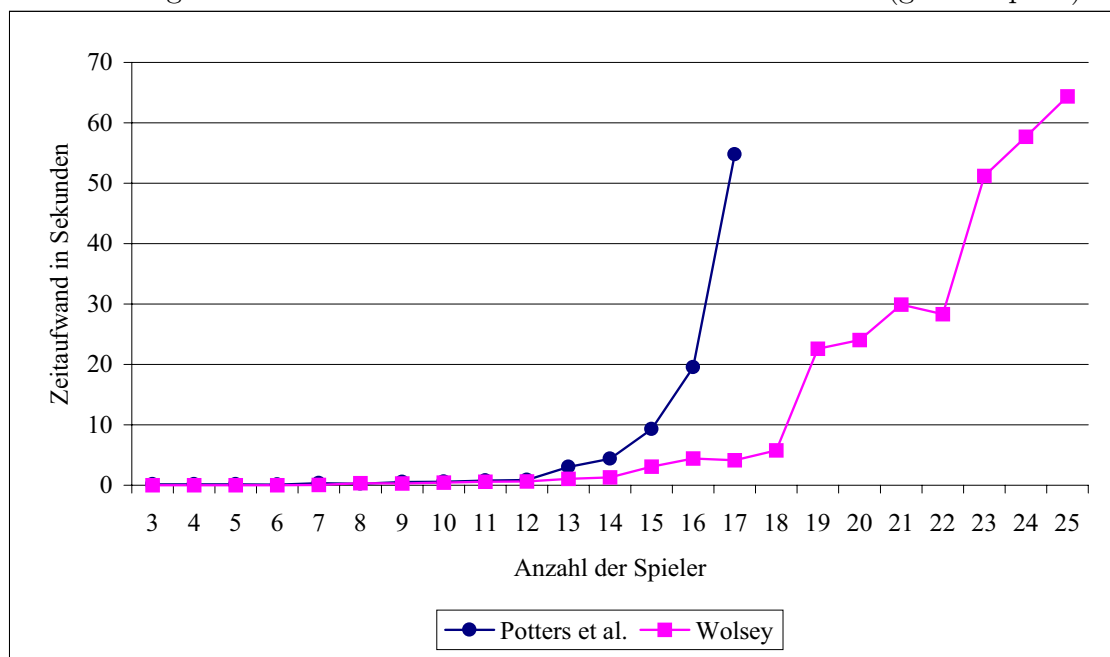
5.14 zeigt, dass das Verfahren von Wolsey hinsichtlich der durchschnittlichen Rechenzeit bei nahezu allen Testspielen dem Verfahren von Potters et al. überlegen war. Eine Ausnahme bilden lediglich die Spiele mit 8 Spielern. Die Abbildungen 5.4 und 5.5 illustrieren diesen Tatbestand. Ferner nimmt die Laufzeit im Verfahren von Potters et al. exponentiell zu. Während eine Erhöhung der Anzahl der Spieler von 3 auf 12 die Laufzeit nur um wenige Millisekunden steigen lässt, nimmt sie bereits um mehrere Stunden zu, wenn wir die Anzahl der Spieler von 12 auf 18 erhöhen. Hingegen berechnet das Verfahren von Wolsey den Nukleolus jedes unserer Test-

5 Implementation zweier Verfahren

Abbildung 5.4: Durchschnittlicher Zeitaufwand der Verfahren (kleine Spiele)



Abbildung 5.5: Durchschnittlicher Zeitaufwand der Verfahren (grosse Spiele)



5.4 Vergleich der implementierten Verfahren

spiele mit einem Zeitaufwand von wenigen Sekunden oder Minuten.²⁰ Vergleichen wir beispielsweise den durchschnittlichen Zeitbedarf der beiden Verfahren für Spiele mit 10 Spielern, so stellen wir fest, dass der Algorithmus von Wolsey um den Faktor $0.556/0.428 = 1.29$ schneller konvergiert als der von Potters et al. Vergleichen wir die Laufzeit der Verfahren bei den Spielen mit 14 Spielern, beträgt dieser Faktor schon $4.392/1.310 = 3.35$ und bei den Spielen mit 17 Spielern bereits $54.768/4.124 = 13.28$.

Stellen wir die Abbildungen 5.3 und 5.5 einander gegenüber, so erkennen wir, dass im Verfahren von Wolsey die Anzahl der Nebenbedingungen mit steigender Spielerzahl einen annähernd linearen Verlauf aufweist (vgl. Abbildung 5.3), der Zeitbedarf (vgl. Abbildung 5.5) dabei aber überlinear ansteigt.

Aus unserem Vergleich der beiden Verfahren lässt sich somit Folgendes Fazit ziehen: Der Algorithmus von Potters et al. erfordert im Vergleich zu demjenigen von Wolsey die Lösung von erheblich weniger linearen Programmen. Die linearen Programme, die gemäss dem Algorithmus von Potters et al. zu lösen sind, enthalten aber eine sehr hohe Anzahl an Nebenbedingungen, z. B. für Spiele mit 25 Spielern über 33 Mio. Dies führt dazu, dass einerseits die Speicherung dieser Nebenbedingungen schnell einen sehr hohen Bedarf an Arbeitsspeicher generiert und andererseits die jeweils erforderlichen Pivotoperationen sehr viel Zeit in Anspruch nehmen. Folglich ist die Berechnung des Nukleolus von Spielen mit einer hohen Anzahl von Spielern mit diesem Algorithmus in der Praxis problematisch. Hingegen enthalten die linearen Programme, die gemäss dem Algorithmus von Wolsey gelöst werden, eine relativ geringe Anzahl an Nebenbedingungen. Bei unseren Testspielen beträgt z. B. die maximale Anzahl der Nebenbedingungen für Spiele mit 25 Spielern 156. Daher hat das Verfahren von Wolsey einen relativ niedrigen Speicherbedarf und gestattet es insofern, den Nukleolus einfacher Spiele mit grosser Spielerzahl (z. B. mit 25 Spielern) zu berechnen. Zudem erfordert das Verfahren von Wolsey einen relativ geringeren Rechenaufwand als das Verfahren von Potters et al. Folglich weist der zur Berechnung des Nukleolus einer speziellen Klasse entwickelte spezifische Algorithmus von

²⁰Unsere Berechnungen einiger hier nicht angegebenen Spiele mit 30 Spielern haben einen durchschnittlichen Zeitaufwand in Höhe von etwa fünf Minuten ergeben.

5 Implementation zweier Verfahren

Wolsey tatsächlich eine bessere Leistung als der für die Berechnung des Nukleolus allgemeiner kooperativer Spiele entwickelte Algorithmus von Potters et al. auf. Somit haben wir anhand unserer Erfahrungen mit unseren Testspielen exemplarisch dargestellt, dass es sich für die Berechnung des Nukleolus spezieller Spiele lohnt, spezifische Algorithmen zu verwenden.

5.5 Zusammenfassung

Im Kapitel 5 haben wir uns mit der Implementation und dem Vergleich zweier Verfahren zur Berechnung des Nukleolus kooperativer Spiele befasst. Für unsere Analyse haben wir die Verfahren von Potters et al. [105] und Wolsey [150] ausgewählt. Im Abschnitt 5.1 haben wir die Wahl dieser Verfahren begründet. Anhand unserer Analyse wollten wir exemplarisch prüfen, ob es sich lohnt, zur Berechnung des Nukleolus spezieller kooperativer Spiele spezifische Verfahren zu entwickeln. In der Literatur wurde eine ähnliche Fragestellung im Kontext der Lösung linearer Programme diskutiert. Hierbei haben die praktischen Erfahrungen darauf hingedeutet, dass die sog. Interior Point-Methoden, die als Alternative zu der Simplex-Methode entwickelt worden sind, zur Lösung praktischer Probleme weniger geeignet sind. In diesem Kapitel haben wir uns im Zusammenhang mit der Berechnung des Nukleolus mit dieser offenen Frage exemplarisch auseinander gesetzt. Das Verfahren von Wolsey [150] wurde gewählt, da es auf einfache kooperative Spiele angewendet werden kann, zu denen insbesondere unser Anwendungsbeispiel zählen wird. Für die Wahl des Verfahrens von Potters et al. [105] spricht, dass es eines der neuesten Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele ist. Im Abschnitt 5.2 haben wir unsere Implementation dieses Verfahrens an Hand des Einführungsbeispiels vorgestellt. Anschliessend haben wir im Abschnitt 5.3 die Implementation des Verfahrens von Wolsey [150] an Hand eines einfachen kooperativen Spiels mit vier Spielern beschrieben. Im Abschnitt 5.4 haben wir die beiden Verfahren miteinander verglichen. Wir haben einfache kooperative Spiele mit bis zu 25 Spielern betrachtet, wobei wir für jede Spieleranzahl n ($n = 3, 4, \dots, 25$) jeweils fünf Spiele aufgestellt haben. Wir haben unter anderem festgestellt, dass das Verfahren von Potters et al. zwar die Lösung von weniger linearen Programmen als das Verfahren von Wolsey erfordert, dass aber die Anzahl der Nebenbedingungen im Verfahren von Potters et al. erheblich höher ist. Demzufolge kann mit Hilfe des Verfahrens von Wolsey praktisch der Nukleolus von Spielen mit einer grösseren Anzahl von Spielern berechnet werden. Zudem ist die Rechenzeit des Verfahrens von Wolsey erheblich niedriger. Am Schluss des Kapitels

5 Implementation zweier Verfahren

haben wir das Fazit gezogen, dass es sich für die Berechnung des Nukleolus spezieller Spiele lohnt, spezifische Verfahren zu entwickeln.

6 Anwendungsbeispiel: Die Zusammensetzung des schweizerischen Bundesrates

6.1 Streit um die Zauberformel

Nach den Parlamentswahlen 1999 begannen in der Schweiz intensive Diskussionen um die Zusammensetzung des schweizerischen Bundesrates, der die Regierung der Schweiz bildet.¹ Damals wollte die Schweizerische Volkspartei (SVP) nach ihrem Wahlerfolg ihre Sitzanzahl im Bundesrat um einen Sitz auf zwei erhöhen. Dieses Ziel erreichte die SVP bei der Ersatzwahl in den Bundesrat vom 6. Dezember 2000 nicht, sodass die seit 1959 geltende parteipolitische Zusammensetzung des Bundesrates unverändert blieb: Freisinnig-Demokratische Partei (FDP), Christlich-Demokratische Volkspartei (CVP) und Sozialdemokratische Partei (SP) stellten je zwei Mitglieder und die Schweizerische Volkspartei (SVP) ein Mitglied im Bundesrat (die sog. Zauberformel).² Die Debatte wurde aber auch nach der Ersatzwahl weitergeführt.³ Schliesslich löste der erneute Wahlerfolg der SVP im Oktober 2003 einen heftigeren Disput über die Aufteilung der Bundesratssitze zwischen der vier grössten Parteien aus. Die SVP wiederholte ihre Forderung nach einem zweiten Sitz im Bundesrat und konnte sich nun damit durchsetzen. Seit Dezember 2003 gilt eine neue Sitzverteilung. Während FDP und SP ihre zwei Sitze behalten haben, musste die CVP nach ihrer Wahlniederlage im Oktober 2003 einen Sitz zugunsten der SVP abtreten.

Die geschilderte Debatte über die parteipolitische Zusammensetzung des schweizerischen Bundesrates bildet eine Konfliktsituation, die durch die Parteien kooperativ gelöst werden musste. In diesem Kapitel wollen wir diesen Tatbestand ex post aus

¹Vgl. z. B. Altermatt [1], Saxer [114] und Breitenstein [17].

²Die in dieser Arbeit verwendete Reihenfolge der Parteien beruht auf der in der Literatur üblichen Reihung.

³Vgl. etwa Neidhard [89].

6 Anwendungsbeispiel: Die Zusammensetzung des schweizerischen Bundesrates

dem Blickwinkel der Theorie der kooperativen Spiele evaluieren. Hierbei werden wir das Konzept des Nukleolus heranziehen. Wir werden erläutern, dass der konzeptionelle Grundgedanke des Nukleolus Ähnlichkeiten mit dem schweizerischen Demokratieverständnis aufweist. Deshalb werden wir das Konzept des Nukleolus auf die schweizerische Politik anwenden. Wie wir noch erörtern werden, besteht eine Anwendungsmöglichkeit darin, dass wir die theoretische Aufteilung der Bundesratssitze unter den Parteien gemäss dem Lösungskonzept des Nukleolus ermitteln und unser Ergebnis mit der tatsächlichen Sitzverteilung vergleichen.

Zu diesem Zweck geben wir im Folgenden einen Überblick über die schweizerische Demokratie. Die Schweiz ist föderalistisch aufgebaut. Der Föderalismus zielt darauf ab, die Vielfalt einer Gesellschaft (religiös, kulturell, ethnisch usw.) innerhalb eines Staates zu bewahren. Der Bundesstaat Schweiz besteht aus 26 Kantonen, davon sind sechs Halbkantone.⁴ Alle Rechte und Aufgaben werden zwischen Bund und Kantonen aufgeteilt. In der Bundespolitik gilt das Zweikammersystem. Die beiden Kammern sind der Nationalrat (Grosse Kammer) und der Ständerat (Kleine Kammer). Diese beiden Räte werden auch Eidgenössische Räte genannt. Der Nationalrat repräsentiert die Gesamtbevölkerung. Die 200 Sitze im Nationalrat werden im Verhältnis zur Wohnbevölkerung auf die Kantone verteilt. Die Bundesverfassung garantiert jedem Kanton mindestens einen Sitz. In den Kantonen, die in der Grossen Kammer mehr als einen Sitz innehaben, werden die Abgeordneten im Proporzverfahren⁵ bestimmt. Demnach sind die Parteien im Nationalrat entsprechend ihrem Stimmenanteil vertreten. Der Ständerat repräsentiert die 26 Kantone. Jeder Kanton ist, unabhängig von der Bevölkerungszahl, im Ständerat mit zwei Sitzen und jeder Halbkanton mit einem Sitz vertreten. Die Kleine Kammer zählt also 46 Sitze. Die Vertreter der Kantone im Ständerat werden im Majorzverfahren bestimmt,

⁴Halbkanton ist die Bezeichnung für die sechs Schweizer Kantone, die nur mit einem Sitz im Ständerat vertreten sind. In der Jurisprudenz bezeichnet man sie als Kantone mit halber Stimmstimme.

⁵Beim Proporzverfahren werden die Wähleranteile der Parteien im Parlament möglichst genau abgebildet. Proportional zur erreichten Stimmenzahl wird jeder Partei die Anzahl an Mandaten zugeteilt. So haben auch Minderheiten die Möglichkeit, im Parlament vertreten zu sein. Beim Majorzverfahren hingegen werden die Kandidaten gewählt, die die meisten Stimmen erhalten haben. Eindeutige Mehrheitsverhältnisse im Parlament werden dadurch begünstigt (vgl. Linder [76], S. 92).

6.1 Streit um die Zauberformel

mit Ausnahme des Kantons Jura, der seine beiden Abgeordneten im Proporzverfahren bestellt. Die beiden Eidgenössischen Räte werden unabhängig voneinander alle vier Jahre vom Volk gewählt. Bezüglich Kompetenzen und Entscheidungsmacht sind die beiden Räte gleichgestellt und verfügen über die selben parlamentarischen Mittel (Motion, Postulat, Interpellation, einfache Anfrage, parlamentarische Initiative). Auch im Gesetzgebungsverfahren sind die beiden Kammern gleichberechtigt: Jedes neue Gesetz bedarf der Zustimmung beider Räte.

Der Nationalrat und der Ständerat bilden gemeinsam die Vereinigte Bundesversammlung. Die Vereinigte Bundesversammlung wählt den Bundesrat, den Bundeskanzler⁶ und die Mitglieder des Bundesgerichts. Sie entscheidet auch über Begnadigungen und Kompetenzstreitigkeiten (vgl. Jegher [55], S. 58 f.). Der Bundesrat ist die Regierung der Schweiz. Er besteht aus sieben Mitgliedern. Bei der Wahl der Bundesräte wird auf die folgenden verfassungsrechtlichen und traditionsbedingten Kriterien geachtet (vgl. Linder [76], S. 222 f.):⁷

- Die grossen Parteien sollten proportional zu ihren Wähleranteilen im Bundesrat vertreten sein.
- Die drei grossen Kantone Bern, Waadt und Zürich sollten möglichst ständige Vertretungen im Bundesrat haben.
- Zwei oder drei Bundesratssitze sollten nicht-deutschsprachigen Kantonen zugewiesen werden.
- Die Frauen sollten im Bundesrat angemessen repräsentiert werden.

Die 41 Mitglieder des Bundesgerichts werden ebenfalls durch die Vereinigte Bundesversammlung gewählt. Für die Wahl der Bundesrichter achten die Parteien auf sprachliche und regionale Kriterien. Bisher haben sie auch auf die Proporzansprüche der grossen politischen Parteien Rücksicht genommen. Zur Zeit setzt sich das

⁶Die in dieser Arbeit verwendeten Funktionsbezeichnungen gelten ohne Unterschied für Personen beider Geschlechter.

⁷Die Kantonsklausel, wonach aus dem gleichen Kanton nur ein Mitglied im Bundesrat Einsitz nehmen darf, ist seit Februar 1999 nicht mehr verfassungsmässig verankert.

6 Anwendungsbeispiel: Die Zusammensetzung des schweizerischen Bundesrates

Bundesgericht aus drei Italienisch, zwölf Französisch und 25 Deutsch sprechenden Richtern sowie einem Rätoromanisch sprechenden Richter zusammen. Gegenwärtig verfügen SP und FDP über acht Sitze, CVP über sieben Sitze, SVP über fünf Sitze und die Liberale Partei (LP) sowie die Grüne Partei (GP) über je einen Sitz.⁸

6.2 Schweizerische Demokratie als Konsensdemokratie

Bei der Erläuterung der Konzeption des Nukleolus im Abschnitt 2.4 dieser Arbeit haben wir dargelegt, dass der Nukleolus als „Minimalkonsens“ der Koalitionspartner interpretiert werden kann. In diesem Abschnitt erörtern wir, inwieweit auch die schweizerische Demokratie auf der Konsensfindung beruht.

Linder [76], S. 359 ff., stellt fest,⁹ dass in der politischen Geschichte der modernen Schweiz die Bemühung, auch die Minderheiten an der politischen Macht zu beteiligen, eine wichtige Rolle spielt. Der Weg dazu führt über die Machtteilung in den politischen Institutionen: Alle Kantone haben ohne jeglichen Vorrang einer Kultur, einer Religion oder einer Ethnie gleiche Rechte (multikulturelle Staatsgründung und föderalistischer Kompromiss). Ausserdem ermöglicht die proportionale Repräsentation der Bevölkerung im Nationalrat, im Bundesrat, im Bundesgericht, in den Expertenkommissionen und in der Verwaltung den Minderheiten den anteilmässigen Zugang zur politischen Macht. Zudem führt die Möglichkeit des Referendums in der schweizerischen Politik dazu, dass am Entscheidungsprozess auf den Politikstil des Verhandelns und des Kompromisses ein grosses Gewicht gelegt wird (Konkordanzsystem). Diese Grundsteine der Machtteilung führen dazu, dass der Bundesrat in der Schweiz beinahe alle Gruppierungen beinhaltet und es deswegen keine regelmässige Machtablösung zwischen Regierung und Opposition gibt. Dieses Modell des Regierens, das Konsensdemokratie genannt wird, unterscheidet sich von der sog. Mehrheitsdemokratie.

⁸Stand der Angaben: November 2004.

⁹Linder gibt in seinem Buch „Schweizerische Demokratie: Institutionen – Prozesse – Perspektiven“ einen umfassenden Überblick über Eigenschaften und Funktionsweise der schweizerischen Demokratie. Der Abschnitt 6.2 dieser Arbeit basiert auf Kapitel 14 in Linder [76].

6.2 Schweizerische Demokratie als Konsensdemokratie

Tabelle 6.1: Länder mit Mehrheits- und Konsensdemokratie

Dimension I (Politikprozess)	Dimension II (Föderalistische Struktur)		
	<i>Unitarische Mehrheitsstruktur</i>	<i>Mischform</i>	<i>Föderalistische Konsensualst.</i>
<i>Mehrheitsprozesse</i>	Neuseeland Grossbritannien	Irland	Australien Österreich Kanada Deutschland USA
<i>Mischform</i>	Island Luxemburg	Norwegen Schweden	Italien Japan
<i>Konsensualprozesse</i>	Dänemark Israel	Belgien Finnland Niederlande	Schweiz

Quelle: Vgl. Lijphart [75], S. 219 (Zitiert nach Linder [76], S. 361).

In einer empirischen Untersuchung legt Lijphart [75], S. 219, (zitiert nach Linder [76], S. 361) dar, wie weit westliche Demokratien eher der Mehrheits- oder der Konsensdemokratie angehören (vgl. Tabelle 6.1). Die Dimension I in der Tabelle 6.1 umfasst den Politikprozess eines Landes. Dieser beinhaltet alle Prozessmerkmale der Regierung und des Parlaments, die zur Konsens- bzw. Mehrheitsdemokratie beitragen. Die Dimension II beinhaltet hingegen den Grad des Föderalismus eines Landes. Die geschriebene Verfassung mit Minderheitsrechten, die Dezentralisierung und das Zweikammersystem werden in Lijphart [75] als Merkmale der Dimension II angegeben. Wie in der Tabelle 6.1 dargelegt, entspricht die schweizerische Demokratie der „reinen“ Konsensdemokratie (Linder [76], S. 362).

Die Feststellung, dass in der schweizerischen Politik die Konsensfindung eine zentrale Rolle spielt, wirft die Frage auf, inwieweit sich das ebenfalls auf der Konsensfindung beruhende Konzept des Nukleolus auf die politischen Entscheidungen in der Schweiz übertragen lässt. Um diese Frage zu beantworten, betrachten wir im nächsten Abschnitt ausgewählte Aspekte der schweizerischen Politik als ein kooperatives Spiel.

6.3 Spieltheoretische Betrachtung der Eidgenössischen Räte

Mit Hilfe der Spieltheorie können wir die Politik eines Gremiums als Abstimmungsspiel darstellen und sie spieltheoretisch analysieren. Wird dabei das Parlament eines Landes betrachtet, werden in der Regel die Parteien als die Spieler des Spiels bezeichnet. Handelt es sich um internationale Gemeinschaften wie z. B. die Europäische Union (EU) oder die Organisation der Vereinten Nationen (UNO), dann werden die jeweiligen Mitgliedsländer als die Spieler des Abstimmungsspiels interpretiert. In der Literatur finden wir verschiedene Anwendungen der Spieltheorie, die die a-priori-Abstimmungsmacht der Spieler in einem Gremium berechnen. Dabei werden Lösungskonzepte der Theorie der kooperativen Spiele wie z. B. der sog. Shapley-Shubik-Index, der sog. Banzhaf-Index usw., verwendet.¹⁰ In dieser Arbeit wenden wir hingegen das Konzept des Nukleolus auf die schweizerische Politik an. Wir stellen in diesem Abschnitt die Eidgenössischen Räte als Abstimmungsspiele dar, wobei die Parteien die Spieler sind.

Wie wir in den Abschnitten 6.1 und 6.2 erläutert haben, versucht man in der Schweiz auf Bundesebene bei der Zusammensetzung der Regierung, des Bundesgerichts und der verschiedenen Kommissionen bestimmte Kriterien einzuhalten, um möglichst einen breiten Konsens zu finden. Wegen dieser Parallelität der Konzeption des Nukleolus und der Funktionsweise der schweizerischen Politik scheint es uns sinnvoll zu sein, das Konzept des Nukleolus auf die hypothetische Zusammensetzung des schweizerischen Bundesrates anzuwenden.

Der schweizerische Bundesrat, der die Regierung der Schweiz bildet, setzt sich aus sieben Bundesräten aus den vier grössten Parteien Freisinnig-Demokratische Partei, Christlich-Demokratische Volkspartei, Sozialdemokratische Partei und Schweizeri-

¹⁰Die Arbeit von Shapley/Shubik [124] ist die erste Anwendung der Theorie der kooperativen Spiele zur Untersuchung von a-priori-Abstimmungsstärken. Seither gibt es eine Vielzahl von Arbeiten, die sich mit der Verteilung von Abstimmungsmacht in verschiedenen Gremien beschäftigen. Beispielsweise untersucht Wildgrén [146] die a-priori-Abstimmungsmacht im Ministerrat der Europäischen Union (EU) und die Konsequenzen der EU-Osterweiterung mit Hilfe des Shapley-Shubik-Index und des Banzhaf-Index. Für weitere Beispiele über die Anwendung der Spieltheorie auf die EU vgl. Bindseil/Hantke [8], Colomer/Hosli [27], Brückner [16], Sutter [137]. Für die Definition von Shapley-Shubik-Index und Banzhaf-Index vgl. etwa Holler/Illing [53], S. 299 ff.

6.3 Spieltheoretische Betrachtung der Eidgenössischen Räte

sche Volkspartei zusammen. Die Mitglieder des Bundesrates werden in der Vereinigten Bundesversammlung (VBV) einzeln gewählt, wobei darauf geachtet wird, dass die zu wählenden Bundesräte bestimmten Anforderungen genügen (vgl. S. 253). Bei der Aufteilung der Sitze im Bundesrat spielt eine möglichst proportionale Vertretung der Parteien eine wichtige Rolle. Wie wir bereits erwähnt haben, blieb die parteipolitische Zusammensetzung des Bundesrates zwischen 1959 und 2003 unverändert. Ein Blick auf die Entwicklung der Parteienstärke über die Zeit hinweg zeigt jedoch, dass die Wähleranteile der FDP und CVP ab 1979 bis 2003 stetig abgenommen haben (FDP von 24% auf 17.3%, CVP von 21.3% auf 14.4%), die SVP dagegen ihren Wähleranteil im selben Zeitraum von 11.1% auf 26.7% gesteigert hat (vgl. Seitz et al. [119], S. 33). Deswegen ist die Forderung der SVP nach einem zweiten Sitz im Bundesrat zunehmend stärker geworden. Bei den Bundesratswahlen im Dezember 2003 erhielt die SVP auf Kosten der CVP einen zweiten Sitz.

Bevor wir nun die Eidgenössischen Räte der Schweiz als ein kooperatives Spiel darstellen, rufen wir nochmals die Struktur eines Abstimmungsspiels in Erinnerung. Wie wir auf S. 176 ausgeführt haben, bezeichnen wir ein Abstimmungsspiel mit $\Gamma = [q : \mathbf{w}]$. Darin sind q die Entscheidungsregel des Spiels und $\mathbf{w} = (w_1, w_2, \dots, w_n)$ die Gewichte der Spieler. Die Gewichte der Spieler entsprechen bei einem Abstimmungsspiel beispielsweise der Sitzanzahl der Parteien in einem Parlament. Bei den Abstimmungsspielen nennt man eine Koalition S Gewinnkoalition, wenn $\sum_{i \in S} w_i \geq q$ gilt. Sonst heisst sie Verlustkoalition. Die charakteristische Funktion eines Abstimmungsspiels ordnet jeder Gewinnkoalition den Wert Eins und jeder Verlustkoalition den Wert Null zu. Formal lässt sich die charakteristische Funktion eines Abstimmungsspiels $\Gamma = [q : \mathbf{w}]$ wie folgt ausdrücken:

$$v(S) = \begin{cases} 1, & \text{falls } \sum_{i \in S} w_i \geq q, \\ 0, & \text{sonst} \end{cases}$$

für alle $S \in 2^N$.

In der Schweiz waren zur Zeit der Wahl der Bundesräte im Dezember 2003 insgesamt 14 Parteien im Parlament vertreten. Diese interpretieren wir als Spieler. Die Anzahl ihrer Sitze sind die Gewichte der Parteien im jeweiligen Gremium: National-

6 Anwendungsbeispiel: Die Zusammensetzung des schweizerischen Bundesrates

Tabelle 6.2: Wahlergebnisse und Sitzverteilung der Parteien 2003

	Partei	%	NR	SR	VBV
1	Freisinnig-Dem. Partei (FDP)	17.3	36	14	50
2	Christlich-Dem. Volkspartei (CVP)	14.4	28	15	43
3	Sozialdemokratische Partei (SP)	23.3	52	9	61
4	Schweizerische Volkspartei (SVP)	26.7	55	8	63
5	Liberale Partei (LP)	2.2	4	0	4
6	Evangelische Volkspartei (EVP)	2.3	3	0	3
7	Christlichsoziale Partei (CSP)	0.4	1	0	1
8	Partei der Arbeit (PdA)	0.7	2	0	2
9	Solidarität (Sol.)	0.5	1	0	1
10	Fem. und grün-alt. Gruppierungen (FGA) ¹⁾	0.5	1	0	1
11	Grüne Partei (GP)	7.4	13	0	13
12	Schweizer Demokraten (SD)	1.0	1	0	1
13	Eidgenössisch-Dem. Union (EDU)	1.3	2	0	2
14	Freiheitspartei (FP)	0.2	0	0	0
15	Lega dei Ticinesi (Lega)	0.4	1	0	1
16	Übrige	1.4	0	0	0
	Total	100	200	46	246

Quelle: Bundesamt für Statistik [119], www.parlament.ch, Stand: 15.12.2003.

¹⁾ Sammelbezeichnung für feministische Gruppierungen und die alternativen Grünen.

rat (NR), Ständerat (SR) sowie Vereinigte Bundesversammlung (VBV). Die Tabelle 6.2 gibt die Sitzverteilung und die Wähleranteile (%) der Parteien nach den Wahlen im Oktober 2003 an. In der Schweiz gilt als Entscheidungsregel sowohl in National- und Ständerat als auch in der Vereinigten Bundesversammlung die einfache Mehrheitsregel. Die Mehrheit wird im Nationalrat mit 101, im Ständerat mit 24 und in der Vereinigten Bundesversammlung mit 124 erreicht.¹¹ Mit Hilfe dieser Angaben stellen wir nun jeweils ein Abstimmungsspiel für den Nationalrat, den Ständerat und die Vereinigte Bundesversammlung auf, wobei die Parteien die Spieler des jeweiligen Abstimmungsspiels sind (vgl. Tabelle 6.2):

¹¹Die aufgeführten Entscheidungsregeln gelten bei vollständiger Anwesenheit und gültiger Stimmabgabe aller Ratsmitglieder.

6.3 Spieltheoretische Betrachtung der Eidgenössischen Räte

Der Nationalrat:

- $$\Gamma_{\text{NR}} : \begin{array}{l} \bullet \quad |N| = n = 14 \\ \bullet \quad \Gamma = [101 : 36, 28, 52, 55, 4, 3, 1, 2, 1, 1, 13, 1, 2, 1] \\ \bullet \quad \text{Für alle } S \subseteq N \setminus \emptyset, \quad v(S) = \begin{cases} 1, & \text{falls } \sum_{i \in S} w_i \geq 101, \\ 0, & \text{sonst.} \end{cases} \end{array}$$

Der Ständerat:

- $$\Gamma_{\text{SR}} : \begin{array}{l} \bullet \quad |N| = n = 4 \\ \bullet \quad \Gamma = [24 : 14, 15, 9, 8] \\ \bullet \quad \text{Für alle } S \subseteq N \setminus \emptyset, \quad v(S) = \begin{cases} 1, & \text{falls } \sum_{i \in S} w_i \geq 24, \\ 0, & \text{sonst.} \end{cases} \end{array}$$

Die Vereinigte Bundesversammlung:

- $$\Gamma_{\text{VBV}} : \begin{array}{l} \bullet \quad |N| = n = 14 \\ \bullet \quad \Gamma = [124 : 50, 43, 61, 63, 4, 3, 1, 2, 1, 1, 13, 1, 2, 1] \\ \bullet \quad \text{Für alle } S \subseteq N \setminus \emptyset, \quad v(S) = \begin{cases} 1, & \text{falls } \sum_{i \in S} w_i \geq 124, \\ 0, & \text{sonst.} \end{cases} \end{array}$$

Neben den oben vorgestellten Spielen können wir andere Abstimmungsspiele aufstellen, indem wir z. B. nicht mehr die Parteien, sondern die Fraktionen als Spieler betrachten. In der Schweiz können sich die im Parlament vertretenen Parteien zu Fraktionen zusammenschliessen, falls die jeweiligen Parteien gemeinsam mindestens fünf Sitze besitzen. Zur Zeit der Bundesratswahlen im Dezember 2003 waren im Parlament insgesamt sechs Fraktionen tätig. In der Tabelle 6.3 werden die Fraktionen, die sie konstituierenden Parteien und die Sitzverteilung der Fraktionen in Nationalrat, Ständerat und der Vereinigten Bundesversammlung wiedergegeben. Hierbei werden die Parteien, die keiner Fraktion angehören, als fraktionslos bezeichnet. Betrachten wir nun anstelle der Parteien die Fraktionen als Spieler, so ergibt sich beispielsweise

6 Anwendungsbeispiel: Die Zusammensetzung des schweizerischen Bundesrates

Tabelle 6.3: Zusammensetzung und Sitzverteilung der Fraktionen 2003

	Fraktion	Zusammensetzung	NR	SR	VBV
1	Freisinnig-Dem. Fraktion	FDP, LP	40	14	54
2	Christlich-Dem. Fraktion	CVP	28	15	43
3	Sozialdemokratische Fraktion	SP	52	9	61
4	Fraktion der Schw. Volkspartei	SVP, Lega	56	8	64
5	Grüne Fraktion	GP, CSP, FGA	15	0	15
6	EVP-EDU Fraktion	EVP, EDU	5	0	5
	Fraktionslos	PdA, SD, Sol.	4	0	4

Quelle: www.parlament.ch, Stand: 15.12.2003.

für die Vereinigte Bundesversammlung das folgende Abstimmungsspiel (vgl. Tabelle 6.3):

$$\Gamma_F : \begin{array}{l} \bullet \quad |N| = n = 7 \\ \bullet \quad \Gamma = [124 : 54, 43, 61, 64, 15, 5, 4] \\ \bullet \quad \text{Für alle } S \subseteq N \setminus \emptyset, \quad v(S) = \begin{cases} 1, & \text{falls } \sum_{i \in S} w_i \geq 124, \\ 0, & \text{sonst.} \end{cases} \end{array}$$

Für die Aufstellung aller oben aufgeführten Abstimmungsspiele nehmen wir an, dass die Mitglieder der Parteien bzw. der Fraktionen als Block stimmen. Die Erfahrung zeigt, dass diese Annahme bei der Wahl der Bundesräte und der Bundesrichter plausibel ist. Bei Abstimmungen über Sachfragen können häufiger abweichende Meinungen auftreten.

Im nächsten Abschnitt werden wir die oben definierten Spiele Γ_{NR} und Γ_{VBV} für das Wahljahr 2003 ausführlich diskutieren. Die Abstimmungsspiele aus früheren Wahljahren lassen sich analog aufstellen. Dafür ist lediglich eine Information über die Anzahl der Sitze der Parteien in der jeweiligen Legislaturperiode erforderlich. Die Tabelle 6.4 gibt die Anzahl der Sitze der Parteien im Nationalrat und in der Vereinigten Bundesversammlung in den Wahljahren von 1987 bis 1999 an. Die Anzahl der Sitze im Ständerat entspricht der Differenz zwischen den Spalten VBV und NR.

6.4 Der Nukleolus des Anwendungsbeispiels

Tabelle 6.4: Anzahl der Mandate der Parteien zw. 1987-1999

	Partei	1987		1991		1995		1999	
		NR	VBV	NR	VBV	NR	VBV	NR	VBV
1	FDP	51	65	44	62	45	62	43	61
2	CVP	42	61	36	52	34	50	35	50
3	SP	42	47	43	46	54	59	51	57
4	SVP	25	29	25	29	29	34	44	51
5	LP	9	12	10	13	7	9	6	6
6	EVP	3	3	3	3	2	2	3	3
7	CSP	0	0	1	1	1	1	1	1
8	PdA	1	1	2	2	3	3	3	3
9	Sol.	-	-	-	-	0	0	0	0
10	FGA	0	0	1	1	2	2	0	0
11	GP	9	9	14	14	8	8	9	9
12	SD	3	3	5	5	3	3	1	1
13	EDU	-	-	1	1	1	1	1	1
14	FP	2	2	8	8	7	7	0	0
15	Lega	0	0	2	3	1	1	2	2
16	LDU ¹⁾	9	10	5	6	3	4	1	1
17	POCH ²⁾	3	3	-	-	-	-	-	-
18	Übrige	1	1	-	-	-	-	-	-

Quelle: www.parlament.ch, Stand: 08.01.2004.

¹⁾ Landesring der Unabhängigen.

²⁾ Progressive Organisationen der Schweiz.

6.4 Der Nukleolus des Anwendungsbeispiels

In diesem Abschnitt legen wir zunächst dar, inwieweit der Nukleolus den Parteien eine ihren Wähleranteilen entsprechende Verteilung der Sitze im Nationalrat zuweist. Anschliessend berechnen wir eine dem Nukleolus entsprechende theoretische Aufteilung der Bundesratssitze und vergleichen diese in verschiedener Hinsicht mit der tatsächlichen Sitzverteilung.

Um zu analysieren, ob zwischen dem Nukleolus und den Wähleranteilen der Parteien ein Zusammenhang besteht, ziehen wir aus dem folgenden Grund das Abstimmungsspiel Γ_{NR} in Betracht: Während sich die Zahl der Abgeordneten eines Kantons

6 Anwendungsbeispiel: Die Zusammensetzung des schweizerischen Bundesrates

im Nationalrat nach seiner Einwohnerzahl richtet, stehen im Ständerat jedem Kanton zwei Sitze und jedem Halbkanton ein Sitz zur Verfügung, unabhängig von der jeweiligen Einwohnerschaft. Zudem gilt bei den Nationalratswahlen das Proporzverfahren, wonach jede Partei im Verhältnis zu ihren Wähleranteilen ihre Sitze erhält. Da nun in der Vereinigten Bundesversammlung auch die Mitglieder des Ständerats Einsitz nehmen, ist die Vereinigte Bundesversammlung kein geeignetes Gremium, um die Wähleranteile und die Sitzanteile zu vergleichen. In der Presse wird deswegen für die Berechnung der Wähleranteile die Nationalratswahl herangezogen.¹² Folglich gehen wir von dem Abstimmungsspiel Γ_{NR} aus, um zu untersuchen, ob der Nukleolus eine mit den Wähleranteilen bzw. Sitzanteilen der Parteien übereinstimmende theoretische Sitzverteilung im Nationalrat beinhaltet. Zu diesem Zweck betrachten wir die Tabelle 6.5. Darin steht „%“ für die prozentualen Wähleranteile der Parteien. Die prozentualen Sitzanteile der Parteien im Nationalrat werden unter „%-Sitz“ wiedergegeben. Diese lassen sich jeweils im Wege der Division der Anzahl der Parteisitze durch die gesamte Anzahl der Sitze im NR, also durch 200, berechnen. In der letzten Spalte geben wir den Nukleolus des Abstimmungsspiels Γ_{NR} an.¹³

Aus der Tabelle 6.5 erkennen wir, dass die Wähleranteile und die Sitzanteile der Parteien im Nationalrat nicht stark voneinander abweichen. Die höchste Differenz beträgt 2.7 Prozentpunkte für die SP. Die relativ geringe Differenz zwischen den Wähleranteilen und den Sitzanteilen der Parteien im Nationalrat ist auch zu erwarten, weil bei den Nationalratswahlen das Proporzverfahren gilt. Vergleichen wir nun die Wähleranteile bzw. die Sitzanteile der Parteien mit dem Nukleolus der Abstimmungsspiele, so stellen wir folgendes fest: Die maximale absolute Abweichung des Nukleolus von den Wähleranteilen beträgt 4.3 Prozentpunkte für die CVP. Für alle anderen Parteien liegt diese Abweichung unter 3 Prozentpunkten. Die absolute Differenz des Nukleolus zu den Sitzanteilen beträgt 3.9 Prozentpunkte, wiederum für die CVP, wobei diese Abweichung für alle anderen Parteien niedriger ist. Beispielsweise beträgt der Wähleranteil der FDP 17.3% und der Nukleolus für sie 16.406%.

¹²Vgl. dazu z. B. Neue Zürcher Zeitung NZZ [93].

¹³Alle in dieser Arbeit angegebenen Nukleoli wurden mittels unserer Programme berechnet, die in der beigelegten CD gespeichert sind.

6.4 Der Nukleolus des Anwendungsbeispiels

Tabelle 6.5: Nukleolus des Spiels Γ_{NR} 2003

	Partei	%	%-Sitz	Nukleolus (\mathcal{N})
1	FDP	17.3	18.0	0.16406
2	CVP	14.4	14.0	0.10156
3	SP	23.3	26.0	0.24219
4	SVP	26.7	27.5	0.26562
5	LP	2.2	2.0	0.03125
6	EVP	2.3	1.5	0.02344
7	CSP	0.4	0.5	0.00781
8	PdA	0.7	1.0	0.01562
9	Sol.	0.5	0.5	0.00781
10	FGA	0.5	0.5	0.00781
11	GP	7.4	6.5	0.10156
12	SD	1.0	0.5	0.00781
13	EDU	1.3	1.0	0.01562
14	FP	0.2	0.0	0
15	Lega	0.4	0.5	0.00781
16	Übrige	1.4	0.0	0

Quelle: Bundesamt für Statistik [119], eigene Berechnungen.

Wir interpretieren diese Abweichungen als relativ klein. Folglich zeigt sich in der Tabelle 6.5 eine näherungsweise Übereinstimmung zwischen dem Nukleolus des Spiels Γ_{NR} und den Wähleranteilen sowie den Sitzanteilen der Parteien. Also generiert der Nukleolus des Spiels Γ_{NR} für die Parteien eine etwa ihren Wähler- und Sitzanteilen im Nationalrat entsprechende Verteilung als Spiellösung. Um zu überprüfen, ob sich dieses Resultat auch in anderen Wahlperioden einstellt, haben wir den Nukleolus des jeweiligen Abstimmungsspiels für den Nationalrat von 1987 bis 1999 berechnet (vgl. Tabelle 6.6). Für jedes dieser Spiele haben wir die Gewichte der Spieler, also die Anzahl der Sitze der Parteien im Nationalrat, in den Spalten NR der Tabelle 6.4 (vgl. S. 261) aufgeführt. Für das jeweilige Jahr werden in der Tabelle 6.6 die prozentualen Wähleranteile der Parteien (Spalte %) und der Nukleolus des Abstimmungsspiels Γ_{NR} in Prozent (Spalte \mathcal{N}^*) angegeben. Aus dem Vergleich der Werte in den Spalten % und \mathcal{N}^* für die jeweiligen Jahre geht hervor, dass die näherungsweise Übereinstimmung des Nukleolus mit den Wähleranteilen der Parteien nicht nur für

6 Anwendungsbeispiel: Die Zusammensetzung des schweizerischen Bundesrates

Tabelle 6.6: Wähleranteile und der Nukleolus des Spiels Γ_{NR} zw. 1987-1999

	Partei	1987		1991		1995		1999	
		%	\mathcal{N}^*	%	\mathcal{N}^*	%	\mathcal{N}^*	%	\mathcal{N}^*
1	FDP	22.9	25.3	21.0	21.8	20.2	22.1	19.9	18.9
2	CVP	19.6	20.1	18.0	17.6	16.8	15.7	15.9	12.1
3	SP	18.4	20.1	18.5	21.3	21.8	27.3	22.5	25.9
4	SVP	11.0	10.4	11.9	11.7	14.9	12.8	22.5	19.8
5	LP	2.7	5.2	3.0	5.3	2.7	4.1	2.3	5.2
6	EVP	1.9	1.9	1.9	1.6	1.8	1.2	1.8	2.6
7	CSP	0.3	0	0.4	0.5	0.3	0.6	0.4	0.9
8	PdA	0.8	0.6	0.8	1.1	1.2	1.7	1.0	2.6
9	Sol.	0	0	0	0	0.3	0	0.5	0
10	FGA	2.4	0	1.3	0.5	1.5	1.2	0.3	0
11	GP	4.9	5.2	6.1	7.4	5.0	4.7	5.0	7.7
12	SD	2.5	1.9	3.4	2.7	3.1	1.7	1.8	0.9
13	EDU	0.9	0	1.0	0.5	1.3	0.6	1.3	0.9
14	FP	2.6	1.3	5.1	4.3	4.0	4.1	0.9	0
15	Lega	0	0	1.4	1.1	0.9	0.6	0.9	1.7
16	LDU	4.2	5.2	2.8	2.7	1.8	1.7	0.7	0.9
17	POCH	1.3	1.9	0	0	0	0	0	0
18	Übrige	2.6	0.6	0	0	0	0	0	0

Quelle: Bundesamt für Statistik [119], eigene Berechnungen.

2003, sondern auch für die jeweiligen Legislaturperioden zwischen 1987 und 1999 gilt. Die höchste Differenz zwischen dem Nukleolus und den Wähleranteilen beträgt 5.5 Prozentpunkte für die SP im Jahr 1995. Bei allen anderen Parteien beträgt die Abweichung weniger als 4 Prozentpunkte.

Im Folgenden werden wir nun die Verteilung der Bundesratssitze zwischen den Parteien bzw. Fraktionen gemäss dem Konzept des Nukleolus berechnen und die Resultate in verschiedener Hinsicht interpretieren.¹⁴ Da in der Schweiz die Wahl der Bundesräte in der Vereinigten Bundesversammlung erfolgt, ziehen wir das Abstim-

¹⁴In der Literatur finden wir nur eine Arbeit, nämlich die von Frey [41], die sich mit der Anwendung der Theorie der kooperativen Spiele auf den schweizerischen Bundesrat beschäftigt. Während Frey mit Hilfe des sog. Shapley-Shubik-Index die Machtverteilung der Parteien im Bundesrat analysiert, ermitteln wir mit Hilfe des Nukleolus eine theoretische Aufteilung der Sitze im Bundesrat und vergleichen diese mit der vorherrschenden Aufteilung.

6.4 Der Nukleolus des Anwendungsbeispiels

Tabelle 6.7: Nukleolus des Abstimmungsspiels Γ_{VBV}

	Partei	\mathcal{N}
1	Freisinnig-Dem. Partei (FDP)	0.15873
2	Christlich-Dem. Volkspartei (CVP)	0.10317
3	Sozialdemokratische Partei (SP)	0.24603
4	Schweizerische Volkspartei (SVP)	0.26190
5	Liberale Partei (LP)	0.03175
6	Evangelische Volkspartei (EVP)	0.02381
7	Christlichsoziale Partei (CSP)	0.00794
8	Partei der Arbeit (PdA)	0.01587
9	Solidarität (Sol.)	0.00794
10	Fem. und grün-alt. Gruppierungen (FGA)	0.00794
11	Grüne Partei (GP)	0.10317
12	Schweizer Demokraten (SD)	0.00794
13	Eidgenössisch-Dem. Union (EDU)	0.01587
14	Freiheitspartei (FP)	0
15	Lega dei Ticinesi (Lega)	0.00794
16	Übrige	0

mungsspiel Γ_{VBV} (vgl. S. 259) heran. Unsere Berechnungen haben für das Abstimmungsspiel Γ_{VBV} die in der Tabelle 6.7 aufgeführten Resultate erbracht. In der Tabelle 6.7 entspricht die Spalte \mathcal{N} dem Nukleolus des Spiels Γ_{VBV} . Da bei den Abstimmungsspielen der Koalitionswert jeder Gewinnkoalition den Wert Eins annimmt, beinhaltet der Nukleolus die Aufteilung des Koalitionswertes in der Höhe von Eins zwischen den Spielern. Somit können wir den Nukleolus eines Abstimmungsspiels als die Gesamtheit der prozentualen Anteile der Spieler am gesamten Koalitionswert interpretieren. In unserem Fall entspricht der Nukleolus konkret den theoretischen prozentualen Sitzanteilen der Parteien im Bundesrat. Beispielsweise besagt der Wert 0.15873 für die FDP, dass bei einer Zuteilung der Bundesratssitze gemäss dem Konzept des Nukleolus dieser Partei 15.873% der Sitze zustehen. Da im Bundesrat sieben Sitze zu verteilen sind, würde die FDP gemäss dem Nukleolus 1.11 Sitze erhalten. Entsprechend enthält die Spalte (4) der Tabelle 6.8 die theoretische Verteilung der Bundesratssitze auf alle Parteien. In der Spalte (1) werden die aktuellen prozentualen Sitzanteile der Bundesratsparteien im Bundesrat wiedergegeben. Diese Werte lassen

6 Anwendungsbeispiel: Die Zusammensetzung des schweizerischen Bundesrates

Tabelle 6.8: Verteilung der Bundesratssitze nach Parteien

	Partei	Praxis		Theoretisch		Differenz
		%-Sitz (1)	Sitzanzahl (2)	%-Sitz (\mathcal{N}) (3)	Sitzanzahl (4)	(1) – (3) (5)
1	FDP	28.571	2	15.873	1.11	12.698
2	CVP	14.286	1	10.317	0.72	3.969
3	SP	28.571	2	24.603	1.72	3.968
4	SVP	28.571	2	26.190	1.83	2.381
5	LP	0	0	3.175	0.22	–3.175
6	EVP	0	0	2.381	0.17	–2.381
7	CSP	0	0	0.794	0.05	–0.794
8	PdA	0	0	1.587	0.11	–1.587
9	Sol.	0	0	0.794	0.05	–0.794
10	FGA	0	0	0.794	0.05	–0.794
11	GP	0	0	10.317	0.72	–10.317
12	SD	0	0	0.794	0.05	–0.794
13	EDU	0	0	1.587	0.11	–1.587
14	Lega	0	0	0.794	0.05	–0.794

sich jeweils mittels Division der Anzahl der aktuellen Sitze (Spalte (2)) durch die gesamte Anzahl der Bundesratssitze, also durch sieben, berechnen. Die Spalte (3) in der Tabelle 6.8 ist der Nukleolus des Abstimmungsspiels Γ_{VBV} in Prozent. Für jede Partei gilt, dass sie über eine im Vergleich zu der gemäss dem Nukleolus ermittelten Sitzverteilung vorteilhaftere Sitzverteilung verfügt, wenn der entsprechende Wert in der letzten Spalte der Tabelle 6.8 positiv ist. Vergleichen wir nun die Spalte (1) mit der Spalte (3), so erkennen wir, dass die FDP in der Praxis mit dem Sitzanteil von 28.57% im schweizerischen Bundesrat im Vergleich zu dem theoretischen Sitzanteil von 15.873% den grössten Vorteil in der Höhe von $28.571\% - 15.873\% = 12.698\%$ hat. Die Vorteile der anderen Bundesratsparteien CVP, SP und SVP sind in dieser Reihenfolge 3.969%, 3.968% und 2.381%. Das heisst, dass die theoretischen Sitzansprüche der Parteien, die im aktuellen Bundesrat über keinen Sitz verfügen, in der Praxis auf die vier Bundesratsparteien aufgeteilt worden sind, wobei die FDP diejenige Partei ist, die aus dieser Aufteilung den höchsten Vorteil zieht. Entsprechend

6.4 Der Nukleolus des Anwendungsbeispiels

verhält es sich bei den Parteien, die in der aktuellen Aufteilung der Bundesratssitze keinen Sitz innehaben. Ihnen stünden die theoretischen Sitzanteile in der Spalte (3) der Tabelle 6.8 zu. Die Partei, die am meisten durch die gegenwärtige Verteilung benachteiligt wird, ist die GP. Obwohl ihr theoretischer Sitzanteil 10.317% mit dem Sitzanteil der CVP exakt übereinstimmt, ist die GP nicht im Bundesrat vertreten.

Wir interpretieren nun die Differenz zwischen den Spalten (1) und (3) gemäss der Definition des Nukleolus. Wie wir im Abschnitt 2.4 ausgeführt haben, minimiert der Nukleolus jedes kooperativen Spiels die Unzufriedenheiten der Koalitionen des Spiels lexikographisch. Wenn nun für eine Partei (Einerkoalition) die in der Spalte (5) der Tabelle 6.8 aufgeführte Differenz positiv ist, dann ist die Zufriedenheit dieser Einerkoalition in der Praxis höher als gemäss dem Nukleolus. Das ist für die Bundesratsparteien FDP, CVP, SP und SVP der Fall. Ist hingegen der Wert in der Spalte (5) negativ, so ist die Zufriedenheit der entsprechenden Einerkoalition bezüglich der aktuellen Sitzverteilung der Bundesratssitze niedriger als ihre Zufriedenheit, die aus dem theoretischen Sitzanteil im Bundesrat entstehen würde.

Die Realisierung der Sitzverteilung gemäss dem Konzept des Nukleolus ist nur beschränkt möglich, da die Sitzanzahl der Parteien in der Praxis ganzzahlig sein muss. Um eine ganzzahlige Aufteilung der Sitze gemäss dem Nukleolus zu ermitteln, gibt es zwei Vorgehensweisen, wobei das Resultat je nach der verwendeten Vorgehensweise anders ausfallen kann: Erstens können wir zunächst den Nukleolus des Spiels berechnen und anschliessend die berechneten Sitzanteile entsprechend auf- oder abrunden. In dieser Arbeit verwenden wir der Einfachheit halber diese erste Methode. Zweitens können wir bei der Lösung der zur Berechnung des Nukleolus erforderlichen linearen Programme für die Variablen voraussetzen, dass sie ganzzahlig sind.

Würde man in der Praxis die Bundesratssitze zwischen den Parteien gemäss dem Konzept des Nukleolus verteilen, so sollte man versuchen, diejenige Koalition, deren Unzufriedenheit bezüglich der aktuellen Sitzverteilung in der Praxis am höchsten ist zu minimieren. Hierbei ist zu beachten, dass die Verteilung ganzzahlig ist. Wie wir bereits erörtert haben, ist die Differenz zwischen der aktuellen und der theoretischen Sitzverteilung für die FDP am grössten und für die GP am kleinsten. Das Konzept

6 Anwendungsbeispiel: Die Zusammensetzung des schweizerischen Bundesrates

des Nukleolus würde also der FDP weniger und der GP mehr Sitze zuteilen. Zudem ordnet der Nukleolus den beiden Parteien CVP und GP den gleichen Sitzanteil zu (vgl. Tabelle 6.8, Spalte (3)). Folglich ist zu erwarten, dass der Nukleolus diesen beiden Parteien jeweils einen Sitz zuteilt, wobei der Sitz für die GP zu Lasten der FDP geht. Demnach führt der Nukleolus zu der folgenden ganzzahligen Sitzverteilung im Bundesrat: FDP, CVP und GP erhalten je einen Sitz, SVP und SP erhalten je zwei Sitze. Wie in Tabelle 6.9 ersichtlich wird, entspricht diese neue parteipolitische Zusammensetzung des Bundesrates dem Konzept des Nukleolus besser als

Tabelle 6.9: Theoretische Verteilung der Bundesratssitze nach Parteien

	Partei	Ganzzahlig		Nicht-ganzzahlig		Differenz
		%-Sitz (1)	Sitzanzahl (2)	%-Sitz (\mathcal{N}) (3)	Sitzanzahl (4)	(1) – (3) (5)
1	FDP	14.286	1	15.873	1.11	–1.587
2	CVP	14.286	1	10.317	0.72	3.969
3	SP	28.571	2	24.603	1.72	3.968
4	SVP	28.571	2	26.190	1.83	2.381
5	GP	14.286	1	10.317	0.72	3.969

die aktuelle Sitzverteilung.¹⁵ Die absoluten Werte der in der Spalte (5) der Tabelle 6.9 aufgeführten Differenzen für die fünf grössten Parteien sind nicht grösser als die entsprechenden absoluten Werte in der Spalte (5) der Tabelle 6.8.

Wir führen nun die obige Analyse für jede Legislaturperiode zwischen 1987 und 1999 durch. Um den Nukleolus der Abstimmungsspiele für die Vereinigte Bundesversammlung zwischen 1987 und 1999 zu berechnen, betrachten wir die Abstimmungsspiele Γ_{VBV} in dieser Zeitspanne. Jedes Spiel können wir mit Hilfe der Angaben in der Tabelle 6.4 (vgl. S. 261) aufstellen, wobei wiederum die Parteien die Spieler sind. Die Gewichte dieser Spieler entsprechen der Anzahl der Sitze der Parteien in der Vereinigten Bundesversammlung in der jeweiligen Legislaturperiode. Die Entscheidungsregel ist die einfache Mehrheitsregel, also 124. Die Tabelle 6.10 gibt den

¹⁵In der Tabelle 6.9 sind nur die Werte der fünf grössten Parteien aufgeführt. Für die übrigen Parteien würden sich in den Spalten (3) und (4) der Tabelle 6.9 dieselben Werte wie in den entsprechenden Spalten der Tabelle 6.8 ergeben.

6.4 Der Nukleolus des Anwendungsbeispiels

Tabelle 6.10: Nukleolus der Abstimmungsspiele für VBV zw. 1987-2003

	Partei	1987	1991	1995	1999	2003
1	FDP	0.26316	0.25213	0.25294	0.24545	0.15873
2	CVP	0.24211	0.20940	0.18235	0.14545	0.10317
3	SP	0.17895	0.18376	0.23529	0.20909	0.24603
4	SVP	0.08421	0.11111	0.08824	0.15454	0.26190
5	LP	0.06316	0.05556	0.05294	0.05455	0.03175
6	EVP	0.01579	0.01282	0.01177	0.02727	0.02381
7	CSP	-	0.00427	0.00588	0.00909	0.00794
8	PdA	0.00526	0.00855	0.01765	0.02727	0.01587
9	Sol.	-	-	0	0	0.00794
10	FGA	-	0.00427	0.01176	0	0.00794
11	GP	0.04737	0.05983	0.04706	0.08182	0.10317
12	SD	0.01579	0.02137	0.01765	0.00909	0.00794
13	EDU	-	0.00427	0.00588	0.00909	0.01587
14	FP	0.01053	0.03419	0.04118	0	0
15	Lega	0	0.01282	0.00588	0.01818	0.00794
16	LDU	0.05263	0.02564	0.02353	0.00909	0
17	POCH	0.01579	-	-	-	-
18	Übrige	0.0053	-	-	-	-

Nukleolus der verschiedenen Abstimmungsspiele wieder. Der Vollständigkeit halber haben wir den Nukleolus des Abstimmungsspiels Γ_{VBV} für das Jahr 2003 nochmals aufgeführt. Wir multiplizieren nun wiederum jeden Wert in der Tabelle 6.10 mit der Anzahl der Bundesratssitze (sieben) und erhalten somit die theoretischen Verteilungen der Bundesratssitze gemäss dem Konzept des Nukleolus. In der Tabelle 6.11 werden diese theoretischen Verteilungen angegeben.

Diese vergleichen wir nun mit der Verteilung der Bundesratssitze in der Praxis. Wie wir bereits auf S. 251 erläutert haben, blieb die parteipolitische Zusammensetzung des Bundesrates in der Praxis zwischen 1959 und 2003 unverändert, sodass die Parteien FDP, CVP und SP je zwei Sitze und die SVP einen Sitz im Bundesrat besaßen. Aus dem Vergleich der Verteilung der Bundesratssitze in der Praxis mit der theoretischen Sitzverteilung gemäss dem Konzept des Nukleolus (Tabelle 6.11) geht hervor, dass der Nukleolus den vier grössten Parteien FDP, CVP, SP und SVP im

6 Anwendungsbeispiel: Die Zusammensetzung des schweizerischen Bundesrates

Tabelle 6.11: Sitzverteilung der Parteien im Bundesrat gemäss dem Nukleolus

	Partei	1987	1991	1995	1999	2003
1	FDP	1.84	1.76	1.77	1.72	1.11
2	CVP	1.69	1.47	1.28	1.02	0.72
3	SP	1.25	1.29	1.65	1.46	1.72
4	SVP	0.59	0.78	0.62	1.08	1.83
5	LP	0.44	0.39	0.37	0.38	0.22
6	EVP	0.11	0.09	0.08	0.19	0.17
7	CSP	-	0.03	0.04	0.06	0.05
8	PdA	0.04	0.06	0.12	0.19	0.11
9	Sol.	-	-	0	0	0.05
10	FGA	-	0.03	0.08	0	0.05
11	GP	0.33	0.42	0.33	0.57	0.72
12	SD	0.11	0.15	0.12	0.06	0.05
13	EDU	-	0.03	0.04	0.06	0.11
14	FP	0.07	0.24	0.29	0	0
15	Lega	0	0.09	0.04	0.13	0.05
16	LDU	0.37	0.18	0.16	0.06	0
17	POCH	0.11	-	-	-	-
18	Übrige	0.04	-	-	-	-

Zeitraum von 1987 bis 1999 eine der Praxis entsprechende Verteilung zuordnet. Nach den Wahlen 1999 ändert sich dieses Bild, da der SVP mit dem theoretischen Sitzanteil 1.08 rechnerisch ein leicht höherer Sitzanteil im Bundesrat zugeteilt wird als der CVP mit dem theoretischen Sitzanteil 1.02. Somit würde sich bei einer ganzzahligen Aufteilung der Bundesratssitze zwischen den vier grössten Parteien die Frage stellen, ob der SVP oder der CVP im Bundesrat zwei Sitze zustehen.¹⁶ Wollte man neben den vier grössten Parteien auch die anderen Parteien bei der theoretischen ganzzahligen Aufteilung der Bundesratssitze in Betracht ziehen, dann sollte ab 1999 auch die GP mitberücksichtigt werden, da die Anzahl ihrer Sitze im Bundesrat von 0.57 im Jahre 1999 und 0.72 im Jahre 2003 auf 1 aufgerundet wird.

¹⁶Wie wir bereits im Abschnitt 6.1 erläutert haben, hat die SVP nach den Wahlen 1999 ihren Anspruch auf einen zweiten Sitz im Bundesrat bekundet.

6.4 Der Nukleolus des Anwendungsbeispiels

Tabelle 6.12: Nukleolus des Abstimmungsspiels Γ_F für 2003

	Fraktion	\mathcal{N}
1	Freisinnig-Dem. Fraktion (R)	0.15789
2	Christlich-Dem. Fraktion (C)	0.10526
3	Sozialdemokratische Fraktion (S)	0.26316
4	Fraktion der Schw. Volkspartei (V)	0.26316
5	Grüne Fraktion (G)	0.10526
6	EVP-EDU Fraktion (E)	0.05263
	Fraktionslos (O)	0.05263

Bisher haben wir die Parteien als Spieler betrachtet. Im folgenden gehen wir davon aus, dass stattdessen die Fraktionen die Spieler sind. Wir haben den Nukleolus des Spiels Γ_F (vgl. S. 260) exemplarisch für das Jahr 2003 berechnet (vgl. Tabelle 6.12). Für dieses Spiel beträgt die Anzahl der Spieler sieben. Die zugehörige Tabelle 6.13 gibt ein gegenüber Tabelle 6.8 wenig verändertes Bild wieder: Die in der Spalte (5) der Tabelle 6.13 angegebene Differenz der Spalten (1) und (3) zeigt, dass der Nukleolus den Fraktionen C, S und V eine der gegenwärtigen Verteilung der Bundesratssitze angenäherte Sitzverteilung zuordnen würde. Demnach würde die Bundesratsfraktion R in der Praxis zu Lasten der kleinen Fraktionen über einen höheren Sitzanteil als gemäss dem Nukleolus verfügen.

Tabelle 6.13: Verteilung der Bundesratssitze nach Fraktionen

	Fraktion	Praxis		Theoretisch		Differenz
		%-Sitz (1)	Sitzanzahl (2)	%-Sitz (\mathcal{N}) (3)	Sitzanzahl (4)	(1) – (3) (5)
1	R	28.571	2	15.789	1.11	12.782
2	C	14.286	1	10.526	0.74	3.760
3	S	28.571	2	26.316	1.84	2.255
4	V	28.571	2	26.316	1.84	2.255
5	G	0	0	10.526	0.74	–10.526
6	E	0	0	5.263	0.37	–5.263
	O	0	0	5.263	0.37	–5.263

6 Anwendungsbeispiel: Die Zusammensetzung des schweizerischen Bundesrates

Mit Hilfe gleicher Überlegungen wie zu der Tabelle 6.9 geben wir nun in der Tabelle 6.14 eine ganzzahlige Sitzverteilung im Bundesrat an. Diese ganzzahlige Verteilung teilt der Fraktion R einen Sitz weniger als in der gegenwärtigen Sitzverteilung zu. Dafür erhält die Fraktion G neu einen Sitz.

Tabelle 6.14: Theoretische Verteilung der Bundesratssitze nach Fraktionen

	Fraktion	Ganzzahlig		Nicht-ganzzahlig		Differenz
		%-Sitz (1)	Sitzanzahl (2)	%-Sitz (\mathcal{N}) (3)	Sitzanzahl (4)	(1) – (3) (5)
1	R	14.286	1	15.789	1.11	–1.503
2	C	14.286	1	10.526	0.74	3.760
3	S	28.571	2	26.316	1.84	2.255
4	V	28.571	2	26.316	1.84	2.255
5	G	14.286	1	10.526	0.74	3.760
6	E	0	0	5.263	0.37	–5.263
	O	0	0	5.263	0.37	–5.263

In diesem Abschnitt haben wir oben die parteipolitische Zusammensetzung des schweizerischen Bundesrates untersucht und die gegenwärtige Verteilung der Bundesratssitze mit der theoretischen, gemäss dem Nukleolus berechneten Sitzverteilung im Bundesrat verglichen. Dieselbe Analyse können wir prinzipiell auch für die Sitzverteilung der Parteien im schweizerischen Bundesgericht durchführen.¹⁷ Da die Wahl der Bundesrichter in der Vereinigten Bundesversammlung erfolgt, ziehen wir als Abstimmungsspiel das Spiel Γ_{VBV} heran. Den Nukleolus dieses Spiels haben wir in der Tabelle 6.7 (vgl. S. 265) angegeben. Wir vergleichen nun die Sitzverteilung der Parteien in der Praxis mit der theoretischen Sitzverteilung gemäss dem Konzept des Nukleolus (vgl. Tabelle 6.15). In der Spalte (2) der Tabelle 6.15 werden die aktuelle parteipolitische Zusammensetzung des Bundesgerichts und in der Spalte (1) die entsprechenden prozentualen Sitzanteile der Parteien wiedergegeben. Diese Anteile lassen sich im Wege der Division der jeweiligen Werte in der Spalte (2) durch die

¹⁷Nach den Wahlen im Oktober 2003 wurde wegen der veränderten Sitzverteilung der Parteien im schweizerischen Parlament die Diskussion über die Neuverteilung der Sitze im Bundesgericht intensiver. Vgl. dazu z. B. NZZ [92] und Strebel [135].

6.4 Der Nukleolus des Anwendungsbeispiels

Tabelle 6.15: Verteilung der Bundesgerichtssitze nach Parteien für 2003

	Partei	Praxis		Theoretisch		Differenz
		%-Sitz (1)	Sitzanzahl (2)	%-Sitz (\mathcal{N}) (3)	Sitzanzahl (4)	(1) – (3) (5)
1	FDP	26.666	8	15.873	4.76	10.793
2	CVP	23.333	7	10.317	3.10	13.016
3	SP	26.666	8	24.603	7.38	2.063
4	SVP	16.666	5	26.190	7.86	−9.524
5	LP	3.333	1	3.175	0.95	0.158
6	EVP	0	0	2.381	0.71	−2.381
7	CSP	0	0	0.794	0.24	−0.794
8	PdA	0	0	1.587	0.48	−1.587
9	Sol.	0	0	0.794	0.24	−0.794
10	FGA	0	0	0.794	0.24	−0.794
11	GP	3.333	1	10.317	3.10	−6.984
12	SD	0	0	0.794	0.24	−0.794
13	EDU	0	0	1.587	0.48	−1.587
14	Lega	0	0	0.794	0.24	−0.794

gesamte Anzahl der 30 Sitze im Bundesgericht berechnen. Die Spalte (3) gibt den Nukleolus des Abstimmungsspiels Γ_{VBV} wieder. Multiplizieren wir die Werte der Spalte (3) jeweils mit 30, so erhalten wir die theoretische Anzahl der Sitze der Parteien gemäss dem Konzept des Nukleolus (Spalte (4)). Vergleichen wir die prozentualen Sitzanteile (Spalte (1)) bzw. die Sitze (Spalte (2)) der Parteien in der Praxis mit den theoretischen Sitzanteilen (Spalte (3)) bzw. den Sitzen (Spalte (4)) gemäss dem Nukleolus, so stellen wir fest, dass die Parteien FDP und CVP im Bundesgericht stark übervertreten, die Parteien SVP und GP dagegen entsprechend unterrepräsentiert sind.

6.5 Zusammenfassung

In diesem Kapitel haben wir dargelegt, dass der Nukleolus als ein Lösungskonzept der Theorie der kooperativen Spiele auf die schweizerische Politik angewandt werden kann. Konkret haben wir insbesondere die Verteilung der Bundesratssitze zwischen den politischen Parteien gemäss dem Konzept des Nukleolus analysiert. Im Abschnitt 6.1 haben wir zunächst die Debatte über die parteipolitische Zusammensetzung des schweizerischen Bundesrates erläutert. Diese Debatte hatte nach dem Wahlerfolg der Schweizerischen Volkspartei (SVP) im Jahre 1999 begonnen. Anschliessend haben wir einen Überblick über das schweizerische Politiksystem gegeben. Im Abschnitt 6.2 haben wir dargelegt, dass die schweizerische Demokratie in der Literatur als eine Konsensdemokratie charakterisiert wird und folglich der Nukleolus auf die schweizerische Politik angewandt werden kann. Im Abschnitt 6.3 haben wir die schweizerische Politik als ein kooperatives Spiel in der Gestalt eines Abstimmungsspiels zwischen den politischen Parteien modelliert.

Hierfür haben wir für den Nationalrat, den Ständerat und die Vereinigte Bundesversammlung je ein Abstimmungsspiel vorgestellt, wobei das Gewicht jedes Spielers der Sitzzahl der jeweiligen Partei im jeweiligen Parlament entspricht. Anhand des Abstimmungsspiels für den Nationalrat haben wir im Abschnitt 6.4 zunächst festgestellt, dass der Nukleolus den Wähleranteilen der Parteien annähernd entsprechende Resultate beinhaltet. Wir haben zur Berechnung der Verteilung der Bundesratssitze gemäss dem Konzept des Nukleolus das Abstimmungsspiel für die Vereinigte Bundesversammlung herangezogen, weil die Bundesräte in diesem Gremium gewählt werden. Unsere Analyse hat gezeigt, dass der Nukleolus für die Parteien im Zeitraum von 1987 bis 1999 eine der Praxis näherungsweise entsprechende theoretische Verteilung der Bundesratssitze generieren würde. Nach den Wahlen 1999 würde aber der Nukleolus der SVP und der GP höhere Sitzanteile zuteilen, als diesen Parteien in der gegenwärtigen Verteilung der Bundesratssitze zugestanden wurden. Die berechneten theoretischen Sitzanteile der Parteien für das Jahr 2003 haben gezeigt, dass der Nukleolus der GP einen Bundesratssitz zusprechen würde. Der FDP hingegen

6.5 Zusammenfassung

stünde ein Sitz weniger zu. Darüber hinaus würde der Nukleolus der SVP zwei Sitze und der CVP nur noch einen Sitz zuordnen. In den nach den Wahlen 2003 stattfindenden Bundesratswahlen in der Vereinigten Bundesversammlung fand eine teilweise Angleichung der aktuellen Sitzverteilung im Bundesrat an die gemäss dem Nukleolus berechnete theoretische Sitzverteilung statt. Die SVP verfügt auf Kosten der CVP im Bundesrat neu über zwei Sitze.

Anschliessend haben wir ein neues Abstimmungsspiel aufgestellt, indem wir nicht mehr die Parteien, sondern die Fraktionen als Spieler definiert haben. Aus unserer Untersuchung haben wir Ergebnisse erhalten, die die oben aufgeführten Aussagen bestätigen. Zum Schluss des Abschnitts 6.4 haben wir gezeigt, dass eine entsprechende Analyse für die Verteilung der Bundesgerichtssitze möglich ist.

7 Zusammenfassung

Im Kapitel 1 der vorliegenden Arbeit haben wir eine einfache Systematik strategischer Spiele vorgestellt. Insbesondere haben wir die kooperativen Spiele eingeführt. Ein kooperatives Spiel wird durch die sog. Spielermenge und die sog. charakteristische Funktion definiert. In kooperativen Spielen können die Spieler bindende Abmachungen treffen und somit Koalitionen bilden. Die charakteristische Funktion ordnet jeder möglichen Koalition eine positive reellwertige Zahl zu, die für den Wert der jeweiligen Koalition steht. In kooperativen Spielen bestehen für jeden Spieler mehrere Möglichkeiten zur Koalitionsbildung. Anhand von Beispielen haben wir gezeigt, dass sich die Kooperation für die einzelnen Spieler lohnen kann und das Zustandekommen einer Koalition von der Verteilung des Koalitionsgewinns abhängt. Hierbei werden in der Theorie der kooperativen Spiele als Spiellösung jene Auszahlungsvektoren herangezogen, die jedem Spieler der zustande gebrachten Koalition Anreize geben, diese Koalition nicht zu verlassen.

Im Kapitel 2 haben wir zunächst einige Grundbegriffe der Theorie der kooperativen Spiele eingeführt. Insbesondere haben wir wichtige Eigenschaften der charakteristischen Funktion erörtert. Hierbei haben wir angenommen, dass die charakteristische Funktion der von uns betrachteten Spiele stets superadditiv ist. In diesem Fall haben die Spieler einen Anreiz, die grosse Koalition zu bilden. Die zentrale Frage besteht dann in der Aufteilung des aus einer Koalitionsbildung resultierenden Gewinns zwischen den einzelnen Spielern. Dazu gibt es in der Literatur unterschiedliche Lösungskonzepte. Jedes Lösungskonzept genügt plausiblen begründbaren Anforderungen. Wir haben uns auf die Lösungskonzepte des Kerns und insbesondere des Nukleolus konzentriert. Der Kern eines kooperativen Spiels enthält alle Auszahlungsvektoren, die effizient, individuell rational und gruppenrational sind. Somit ist jeder Auszahlungsvektor im Kern eines Spiels insbesondere anreizkompatibel. Der Kern weist aber zwei

7 Zusammenfassung

Schwächen auf: Erstens ist die Existenz des Kerns nicht für jedes kooperative Spiel gewährleistet, selbst wenn sich die Kooperation lohnt. Zweitens enthält der nicht-leere Kern eines Spiels in der Regel mehrere Elemente, sodass nicht ohne weiteres bestimmt werden kann, welcher Auszahlungsvektor als Spiellösung ausgewählt werden soll. Weiter haben wir die Begriffe der balancierten Koalitionssysteme und der balancierten Spiele eingeführt. Ein Koalitionssystem heisst balanciert, falls in einem kooperativen Spiel jeder Spieler seine Gesamtressourcen, z. B. seine Arbeitszeit, vollumfänglich auf die Koalitionen des Koalitionssystems aufteilt. Ist für ein kooperatives Spiel die gewichtete Summe aller Koalitionen jedes balancierten Koalitionssystems nicht grösser als der Wert der grossen Koalition, so nennt man dieses Spiel ein balanciertes kooperatives Spiel. Der Kern eines Spiels ist dann und nur dann nicht-leer, wenn das Spiel balanciert ist.

In den weiteren Abschnitten des Kapitels 2 haben wir das Lösungskonzept des Nukleolus vorgestellt. Der Nukleolus geht von effizienten und individuell rationalen Auszahlungsvektoren aus und generiert als Spiellösung den Auszahlungsvektor, der den sog. Vektor der geordneten Überschüsse lexikographisch minimiert. Der Überschuss einer Koalition ist als die Differenz zwischen dem Koalitionswert und der Auszahlung der Koalition definiert und kann als Mass der Unzufriedenheit der Koalition gegenüber dem Auszahlungsvektor interpretiert werden. Demnach wäre die Unzufriedenheit einer Koalition umso höher, je grösser der Koalitionüberschuss ist. Im Unterschied zum Kern existiert der Nukleolus stets und enthält einen einzigen Auszahlungsvektor. Weiter haben wir das Kohlberg-Kriterium vorgestellt, das eine notwendige und hinreichende Bedingung dafür angibt, dass ein Auszahlungsvektor der Nukleolus des Spiels ist. Anschliessend haben wir die wichtigsten Eigenschaften des Nukleolus erläutert und den Nukleolus axiomatisch begründet. Wir haben ausgeführt, dass die Berechnung des Nukleolus kooperativer Spiele bereits mit wenigen Spielern ohne ein spezielles, zu diesem Zweck entwickeltes mathematisches Verfahren im Allgemeinen nicht mehr möglich ist.

Zur Berechnung des Nukleolus allgemeiner kooperativer Spiele wurden von verschiedenen Autoren diverse Verfahren entwickelt. Im Kapitel 3 haben wir die in der

Literatur zugänglichen Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele ausführlich vorgestellt. Auf Grund der jeweils verwendeten Methoden haben wir diese Verfahren in drei Gruppen zusammengefasst. Der ersten Gruppe gehören die Verfahren von Kohlberg [65] und Owen [96] an, die wir als Methoden der ersten Generation bezeichnet haben. Diese Verfahren berechnen den Nukleolus mit Hilfe eines einzigen linearen Programms. Das von Kohlberg vorgeschlagene lineare Programm besitzt eine grosse Anzahl an Nebenbedingungen, in denen zudem sehr grosse Koeffizienten auftreten. Das gemäss dem Verfahren von Owen zu lösende lineare Programm ist eine äquivalente Darstellung des linearen Programms im Verfahren von Kohlberg. Im Vergleich zu dem von Kohlberg vorgeschlagenen linearen Programm besitzt das gemäss dem Verfahren von Owen zu lösende lineare Programm zwar eine geringere Anzahl an Nebenbedingungen, beinhaltet aber mehr Variablen. Auf Grund der erläuterten Nachteile stellen die beiden Verfahren keine praktikablen Methoden zur Berechnung des Nukleolus allgemeiner kooperativer Spiele dar. Die Verfahren von Kohlberg und Owen sind aber bei der Entwicklung anderer fortgeschrittener Verfahren zur Berechnung des Nukleolus zweifach von besonderer Bedeutung: Erstens zeigen die beiden Verfahren auf, dass für die Berechnung des Nukleolus die Methoden der linearen Optimierung eingesetzt werden können. Zweitens legen diese Verfahren dar, dass die Berechnung des Nukleolus theoretisch im Wege der Lösung eines einzigen linearen Programms erfolgen kann.

Weiter haben wir in diesem Kapitel die Methoden der zweiten Generation vorgestellt. Dazu zählen das Verfahren von Bruyneel [22] und das Verfahren von Dragan [35]. Diese Autoren berechnen den Nukleolus mit Hilfe minimal balancierter Koalitionssysteme. Das Verfahren von Bruyneel erfordert zur Berechnung des Nukleolus zunächst die rechnerisch sehr aufwändige Bestimmung von allen minimal balancierten Koalitionssystemen und deren Gewichtsvektoren. Die Koalitionssysteme und deren Gewichtsvektoren für eine Spielermenge mit n Spielern müssen aber nur ein Mal festgelegt werden, weil sie für die Berechnung des Nukleolus jedes weiteren kooperativen Spiels mit n Spielern eingesetzt werden können. Das Verfahren von Dragan geht von der konzeptionellen Idee des Verfahrens von Bruyneel aus und berechnet den Pre-

7 Zusammenfassung

nukleolus. Der Prenukleolus unterscheidet sich vom Nukleolus dadurch, dass die zur Berechnung des Prenukleolus herangezogenen Auszahlungsvektoren nicht individuell rational sein müssen. In der direkten Form des Verfahrens von Dragan ist ebenfalls die Bestimmung von allen minimal balancierten Koalitionssystemen und deren Gewichtsvektoren erforderlich. Im Wege der Lösung eines linearen Programms bestimmt jedoch die modifizierte Form des Verfahrens von Dragan in jeder Iteration nur ein balanciertes Koalitionssystem, sodass nicht mehr alle balancierten Koalitionssysteme und deren Gewichtsvektoren ermittelt werden müssen.

Neben den Methoden der ersten und zweiten Generationen finden wir in der Literatur mehrere Verfahren, die den Nukleolus berechnen, indem sie eine Folge linearer Programme lösen. Diese Verfahren haben wir als Methoden der dritten Generation bezeichnet. Den Verfahren dieser Generation liegt das Verfahren von Maschler et al. [82] zu Grunde, das wir als erstes vorgestellt haben. Anschliessend haben wir das Verfahren von Sankaran [113] dargestellt. Dieses Verfahren verbessert das Verfahren von Maschler et al. insofern, als es die Anzahl der zu lösenden linearen Programme reduziert. Danach haben wir das Verfahren von Potters et al. [105] erörtert. In diesem Verfahren bilden die Nebenbedingungen der jeweils zu lösenden linearen Programme ein Gleichungssystem, das bereits eine Basis besitzt. Insbesondere wird jedes lineare Programm jeweils aus dem ihm vorangehenden linearen Programm hergeleitet und enthält eine geringere Anzahl an Nebenbedingungen. Im Anschluss daran haben wir das Verfahren von Derks/Kuipers [32] zur Berechnung des Prenukleolus vorgestellt. Dieses Verfahren nützt bestimmte Eigenschaften der Koeffizientenmatrix der Nebenbedingungen der entsprechenden linearen Programmen aus, um die Anzahl der Nebenbedingungen zu reduzieren. Zur Berechnung einer Lösung der linearen Programme setzt das Verfahren von Derks/Kuipers eine spezielle Simplex-Methode ein.

In der Literatur sind verschiedene spezielle kooperative Spiele zu finden, deren charakteristische Funktion besondere Charakteristika aufweist. Bei solchen Spielen kann die Berechnung des Nukleolus in verschiedener Hinsicht vereinfacht werden, indem die besonderen Eigenschaften der charakteristischen Funktion ausgenutzt werden. Im

Kapitel 4 haben wir zunächst exemplarisch das Verfahren von Wolsey [150] vorgestellt, das den Nukleolus einfacher kooperativer Spiele berechnet. Dieses Verfahren berücksichtigt lediglich Gewinnkoalitionen und geht zu Beginn nur von einer beliebig gewählten Teilmenge von Gewinnkoalitionen aus. Zudem erfordert das Verfahren von Wolsey die Lösung von bestimmten binären linearen Programmen. Diese ermöglichen einerseits zu überprüfen, ob der Nukleolus des Spiels gefunden ist und andererseits nötigenfalls eine neue Gewinnkoalition festzustellen. Anschliessend haben wir anhand von Beispielen einige ausgewählte spezielle Spiele erläutert und einen Literaturüberblick über die in der Literatur entwickelten Verfahren zur Berechnung des Nukleolus dieser Spiele gegeben.

Im Kapitel 5 haben wir unsere Implementation zweier Verfahren zur Berechnung des Nukleolus vorgestellt. Mittels Berechnungen des Nukleolus mehrerer Spiele haben wir die Verfahren bezüglich des Zeitaufwands, der Anzahl der Nebenbedingungen und der Anzahl der gelösten linearen Programme miteinander verglichen. Um zu untersuchen, ob ein Verfahren zur Berechnung des Nukleolus spezieller kooperativer Spiele eine bessere Leistung (z. B. eine geringere Rechenzeit) aufweist als ein Verfahren zur Berechnung des Nukleolus allgemeiner kooperativer Spiele, haben wir das Verfahren von Wolsey und das Verfahren von Potters et al. ausgewählt. Unser Ziel bestand darin, exemplarisch darzulegen, ob es sich lohnt, zur Berechnung des Nukleolus spezieller kooperativer Spiele spezifische Verfahren zu entwickeln. Beispielsweise hat man in der Literatur die Erfahrung gemacht, dass die Anwendung der spezifischen sog. Interior Point-Methoden zur Lösung linearer Programme in der Regel weniger geeignet sind als die Simplex-Methode. Die Analyse unserer Berechnungen hat ergeben, dass das Verfahren von Wolsey dem Verfahren von Potters et al. in verschiedener Hinsicht überlegen ist. Wir haben also exemplarisch dargelegt, dass zur Berechnung des Nukleolus spezieller Spiele die Anwendung spezifischer Verfahren von Vorteil sein kann.

Im letzten Kapitel der Arbeit haben wir unser Anwendungsbeispiel vorgestellt. Wir haben gezeigt, dass die schweizerische Demokratie in der Literatur als eine Konsensdemokratie betrachtet wird. Da nun die Konsensfindung auch der Grundgedan-

7 Zusammenfassung

ke des Nukleolus ist, haben wir das Konzept des Nukleolus auf die schweizerische Demokratie angewandt. Insbesondere haben wir uns auf die parteipolitische Zusammensetzung des schweizerischen Bundesrates konzentriert. Zu diesem Zweck haben wir die Eidgenössischen Räte als ein einfaches kooperatives Spiel dargestellt. Da die Bundesräte in der Vereinigten Bundesversammlung gewählt werden, haben wir den Nukleolus jedes einfachen kooperativen Spiels für die jeweilige Vereinigte Bundesversammlung von 1987 bis 2003 mit Hilfe unserer Implementation des Verfahrens von Wolsey berechnet. Anschliessend haben wir für jede Legislaturperiode die theoretische, gemäss dem Nukleolus vorgenommene, Verteilung der Bundesratssitze und die jeweilige aktuelle Verteilung der Bundesratssitze miteinander verglichen. Unsere Untersuchungen haben ergeben, dass die theoretische Verteilung und die tatsächliche Verteilung der Bundesratssitze zwischen den Parteien im Zeitraum von 1987 bis 1999 näherungsweise übereinstimmen. Demgegenüber haben wir festgestellt, dass der Nukleolus nach den Wahlen 1999 den politischen Parteien eine von der vorherrschenden Praxis abweichende Verteilung der Bundesratssitze zuordnen würde. Nach der neuen parteipolitischen Zusammensetzung des schweizerischen Bundesrates nach den Wahlen 2003 hat sich die Praxis an die dem Nukleolus entsprechende Sitzverteilung angenähert. In diesem Kapitel haben wir auch Abstimmungsspiele aufgestellt, in denen nicht die einzelnen Parteien, sondern die Fraktionen die Spieler sind. Die Analyse dieser Spiele haben die angeführten Ergebnisse bestätigt. Zum Schluss des Kapitels haben wir erläutert, dass entsprechende Untersuchungen auch zur Berechnung einer theoretischen Verteilung der Bundesgerichtssitze zwischen den Parteien durchgeführt werden können.

A Lineare Optimierung

A.1 Grundlagen

Unter einer *linearen Optimierungsaufgabe*, auch lineares Programm (LP) genannt, versteht man ein bestimmtes Optimierungsproblem. In diesem Problem maximiert oder minimiert man eine lineare *Zielfunktion* z von p Variablen x_1, \dots, x_p unter Berücksichtigung von linearen *Nebenbedingungen* (*Restriktionen*) und zumeist unter der weiteren Voraussetzung, dass die Variablen x_1, \dots, x_p nicht-negativ sein sollen (*Nichtnegativitätsbedingung*). Formal kann ein lineares Programm wie folgt dargestellt werden:¹

$$\begin{aligned} \max \quad & z(\mathbf{x}) := c_1x_1 + c_2x_2 + \dots + c_px_p \\ \text{unter den Nebenbedingungen} \\ & a_{i1}x_1 + a_{i2}x_2 + \dots + a_{ip}x_p \leq b_i \quad \text{für } i = 1, 2, \dots, m_1, \\ & a_{i1}x_1 + a_{i2}x_2 + \dots + a_{ip}x_p = b_i \quad \text{für } i = m_1 + 1, m_1 + 2, \dots, m, \\ & x_1 \geq 0, \quad x_2 \geq 0, \quad \dots, \quad x_p \geq 0. \end{aligned} \tag{A.1}$$

Die Variablen x_1, x_2, \dots, x_p heissen *Strukturvariablen* oder *Entscheidungsvariablen*. Falls in einem linearen Programm für bestimmte Variablen die Nichtnegativität nicht verlangt wird, so bezeichnet man diese Variablen als *freie Variablen*. Die Menge aller die Nebenbedingungen erfüllenden Vektoren $(x_1 \dots x_p)$ wird als *zulässiger Bereich* und jedes Element dieser Menge als *zulässige Lösung* bezeichnet. Falls eine zulässige Lösung $(x_1 \dots x_p)$ den Wert der Zielfunktion maximiert, ist $(x_1 \dots x_p)$ eine *optimale Lösung* des linearen Optimierungsproblems. Die Einführung der sog. *Schlupfvariablen* x_{p+1}, \dots, x_n in die Nebenbedingungen ermöglicht es uns, jede Ungleichung der

¹Ein lineares Programm kann auch ein Minimierungsproblem beinhalten. In diesem Fall kann aber die Minimierungsaufgabe als eine Maximierungsaufgabe umgeschrieben werden, indem die zu minimierende Zielfunktion mit -1 multipliziert wird. Zudem vernachlässigen wir im linearen Programm die Nebenbedingungen vom \geq -Typ. Durch Multiplikation mit -1 kann jede solche Nebenbedingung als \leq -Restriktion dargestellt werden.

A Lineare Optimierung

Nebenbedingungen als Gleichung abzubilden:

$$\begin{aligned}
 \max \quad & z(\mathbf{x}) := c_1x_1 + c_2x_2 + \dots + c_px_p + 0x_{p+1} + \dots + 0x_n \\
 \text{unter den Nebenbedingungen} \\
 & a_{i1}x_1 + a_{i2}x_2 + \dots + a_{ip}x_p + x_{p+i} = b_i \quad \text{für } i = 1, 2, \dots, m_1, \\
 & a_{i1}x_1 + a_{i2}x_2 + \dots + a_{ip}x_p = b_i \quad \text{für } i = m_1 + 1, m_1 + 2, \dots, m, \\
 & x_1 \geq 0, \quad x_2 \geq 0, \quad \dots, \quad x_n \geq 0.
 \end{aligned} \tag{A.2}$$

Das lineare Programm (A.2) besitzt nun insgesamt p Strukturvariablen und m_1 Schlupfvariablen, sodass $n = p + m_1$ ist. Dieses Optimierungsproblem kann durch Verwendung der Matrixschreibweise wie folgt wiedergegeben werden:²

$$\begin{aligned}
 \text{LP :} \quad & \max \quad z(\mathbf{x}) := \mathbf{c}^T \mathbf{x} \\
 & \text{unter den Nebenbedingungen} \\
 & \mathbf{Ax} = \mathbf{b} \\
 & \mathbf{x} \geq \mathbf{0}.
 \end{aligned} \tag{A.3}$$

Die Schreibweise (A.3) wird als *Normalform* oder auch *Standardform* eines linearen Optimierungsproblems bezeichnet. Hierbei sind \mathbf{x} und \mathbf{c} jeweils n -dimensionale Spaltenvektoren, \mathbf{b} ist ein m -dimensionaler Spaltenvektor und \mathbf{A} ist eine $(m \times n)$ -Matrix. Die Schreibweise \mathbf{c}^T bedeutet, dass der Vektor \mathbf{c} transponiert wird. Die Matrix \mathbf{A} nennen wir die *Koeffizientenmatrix* und den Vektor \mathbf{b} die *rechte Seite* des linearen Programms. Die Nichtnegativitätsbedingungen $x_i \geq 0$, $i = 1, 2, \dots, n$, des linearen Optimierungsproblems werden durch $\mathbf{x} \geq \mathbf{0}$ abgebildet. Im Weiteren nehmen wir ohne Beschränkung der Allgemeinheit an, dass die Elemente der rechten Seite nicht-negativ sind. Zudem sei m ($\leq n$) der Rang von \mathbf{A} .³ Durch entsprechende Ummummerierung der Variablen bilden wir die aus den linear unabhängigen m Zeilen und Spalten der Matrix \mathbf{A} gebildete reguläre Matrix \mathbf{A}_B . Die Matrix der restlichen Spalten von \mathbf{A} sei \mathbf{A}_N . Wir permutieren die Spalten der Matrix \mathbf{A} derart, dass wir

²Wir benutzen für die Darstellung von Matrizen und von Vektoren Grossbuchstaben bzw. Kleinbuchstaben in jeweils fetter Druckschrift. Eine Ausnahme bildet der Nullvektor, den wir durch $\mathbf{0}$ repräsentieren.

³Diese Annahme impliziert, dass keine Zeile der Matrix \mathbf{A} redundant ist, also nicht durch Linearkombination anderer Zeilen gewonnen werden kann. Vgl. dazu Anhang D und Lawler [73], S. 41.

die Matrix \mathbf{A} als $\mathbf{A} = (\mathbf{A}_B, \mathbf{A}_N)$ darstellen können. Unter diesen Annahmen können wir das Gleichungssystem $\mathbf{Ax} = \mathbf{b}$ als $\mathbf{A}_B \mathbf{x}_B + \mathbf{A}_N \mathbf{x}_N = \mathbf{b}$ umschreiben, wobei $\mathbf{x}^T = (\mathbf{x}_B^T, \mathbf{x}_N^T)$ ist. Entsprechend teilen wir den Vektor \mathbf{c} in der Form $\mathbf{c}^T = (\mathbf{c}_B^T, \mathbf{c}_N^T)$ auf. Die Matrix \mathbf{A}_B heisst *Basismatrix* oder auch *Basis* des linearen Programms in Normalform. Eine Lösung von $\mathbf{Ax} = \mathbf{b}$ mit $\mathbf{x}_N = \mathbf{0}$ und $\mathbf{x}_B = \mathbf{A}_B^{-1} \mathbf{b}$ nennt man *Basislösung des Gleichungssystems*.⁴ Eine Basislösung \mathbf{x} , die alle Nichtnegativitätsbedingungen erfüllt, d. h. $\mathbf{x}_B \geq \mathbf{0}$ und $\mathbf{x}_N = \mathbf{0}$, nennt man eine *zulässige* Basislösung des Gleichungssystems $\mathbf{Ax} = \mathbf{b}$. Die m linear unabhängigen Spaltenvektoren der Matrix \mathbf{A}_B nennt man *Basisvektoren* und die zugehörigen Variablen x_j , $j = 1, \dots, m$, *Basisvariablen*. Der Vektor $\mathbf{x}_B^T := (x_1 \dots x_m)$ heisst Vektor der Basisvariablen. Die restlichen Spaltenvektoren nennt man *Nichtbasisvektoren* und die zugehörigen Variablen x_j , $j = m + 1, \dots, n$, *Nichtbasisvariablen*. Der Vektor $\mathbf{x}_N^T := (x_{m+1} \dots x_n)$ heisst Vektor der Nichtbasisvariablen. Ist ein Vektor \mathbf{x} eine zulässige Basislösung des Gleichungssystems $\mathbf{Ax} = \mathbf{b}$, so bezeichnet man \mathbf{x} als eine *Ecke* oder auch als einen *Eckpunkt* des zulässigen Bereichs. Falls der zulässige Bereich beschränkt und nicht-leer ist, kann man zeigen, dass das Optimierungsproblem eine Lösung in einer Ecke des zulässigen Bereichs besitzt.⁵ Da ein Eckpunkt gleichzeitig eine zulässige Basislösung ist und für eine zulässige Basislösung $\mathbf{x}_B \geq \mathbf{0}$ und $\mathbf{x}_N = \mathbf{0}$ gilt, nehmen die Nichtbasisvariablen im Maximum der Zielfunktion den Wert Null an.

Um ein weiteres in dieser Arbeit verwendetes Instrumentarium einzuführen, betrachten wir das folgende lineare Programm

⁴Die Schreibweise $\mathbf{x}_N = \mathbf{0}$ soll anzeigen, dass alle Komponenten des Vektors \mathbf{x}_N Null sind.

⁵Ein Beweis dieser Aussage ist z. B. in Bomze/Grossmann [11], S. 87 f. zu finden.

A Lineare Optimierung

$$\begin{aligned}
 \text{PLP :} \quad & \max z(\mathbf{x}) := \sum_{j=1}^p c_j x_j \\
 & \text{unter den Nebenbedingungen} \\
 & \sum_{j=1}^p a_{ij} x_j \leq b_i \quad \text{für } i = 1, 2, \dots, m_1, \\
 & \sum_{j=1}^p a_{ij} x_j = b_i \quad \text{für } i = m_1 + 1, m_1 + 2, \dots, m, \\
 & x_j \geq 0 \quad \text{für } j = 1, 2, \dots, k, \\
 & x_j \quad \text{frei} \quad \text{für } j = k + 1, k + 2, \dots, p.
 \end{aligned} \tag{A.4}$$

Das lineare Programm

$$\begin{aligned}
 \text{DLP :} \quad & \min z_D(\mathbf{y}) := \sum_{i=1}^m b_i y_i \\
 & \text{unter den Nebenbedingungen} \\
 & \sum_{i=1}^m a_{ij} y_i \geq c_j \quad \text{für } j = 1, 2, \dots, k, \\
 & \sum_{i=1}^m a_{ij} y_i = c_j \quad \text{für } j = k + 1, k + 2, \dots, p, \\
 & y_i \geq 0 \quad \text{für } i = 1, 2, \dots, m_1, \\
 & y_i \quad \text{frei} \quad \text{für } i = m_1 + 1, m_1 + 2, \dots, m
 \end{aligned} \tag{A.5}$$

nennt man das zu PLP *duale* Problem. Das Ausgangsproblem PLP heisst *primales* Problem. Die Variablen y_i in DLP bezeichnet man als *duale Variablen*. Ein zu einem primalen linearen Programm PLP duales Problem DLP beinhaltet für jede Restriktion in PLP eine nicht-negative oder freie duale Variable, je nachdem, ob die Restriktion eine Ungleichung oder eine Gleichung ist. Zusätzlich wird für jede primale Variable x_j in PLP eine duale Restriktion in DLP als Ungleichung bzw. Gleichung angesetzt, je nachdem, ob x_j in PLP eine nicht-negative oder freie Variable ist. Jedes primale lineare Programm steht in einem wohl bestimmten Zusammenhang zu seinem dualen linearen Programm. Die nachfolgenden beiden Sätze, deren Beweise in Bomze/Grossmann [11], S. 130 bzw. S. 137, zu finden sind, geben diesen Zusammenhang wieder:

Satz A.1.1 (Dualitätssatz)

- (a) Ist \mathbf{x} eine zulässige Lösung von PLP und \mathbf{y} eine zulässige Lösung von DLP, dann gilt $z(\mathbf{x}) \leq z_D(\mathbf{y})$.
- (b) Sind \mathbf{x}^* und \mathbf{y}^* optimale Lösungen von PLP bzw. DLP, dann gilt $z(\mathbf{x}^*) = z_D(\mathbf{y}^*)$.

Den Satz A.1.1 (a) nennt man in der Literatur den schwachen Dualitätssatz und den Satz A.1.1 (b) den starken Dualitätssatz.

Satz A.1.2 (Komplementärschlupfbedingungen) Sind \mathbf{x} und \mathbf{y} optimale Lösungen von PLP bzw. DLP, so gelten die folgenden beiden sog. Komplementärschlupfbedingungen gleichzeitig:

$$\begin{aligned}
 (i) \quad y_i > 0, \quad i \in \{1, 2, \dots, m_1\} &\Rightarrow \sum_{j=1}^p a_{ij}x_j = b_i, \\
 (ii) \quad x_j > 0, \quad j \in \{1, 2, \dots, k\} &\Rightarrow \sum_{i=1}^m a_{ij}y_i = c_j.
 \end{aligned}$$

Die Bedingung (i) besagt, dass bei positivem y_i , $i \in \{1, 2, \dots, m_1\}$, die Schlupfvariable in der Nebenbedingung i von PLP den Wert Null annimmt.⁶ Die Bedingung (ii) hingegen impliziert, dass bei positivem x_j , $j \in \{1, 2, \dots, k\}$, die Schlupfvariable in der Nebenbedingung j des DLP den Wert Null hat.

A.2 Die Simplex-Methode

Die Simplex-Methode wurde im Jahre 1947 von Dantzig [29] entwickelt. Sie ist ein bewährtes Standardverfahren der Linearen Optimierung. Die Simplex-Methode beschränkt sich bei der Lösung linearer Programme auf die Untersuchung der Eckpunkte des zulässigen Bereichs, weil eine optimale Lösung stets in einer Ecke gefunden

⁶Diese Eigenschaft ist für uns von besonderer Bedeutung. Denn wie wir bei der Beschreibung der Verfahren zur Berechnung des Nukleolus erläutert haben, verlangen einige der Verfahren die Bestimmung der Nebenbedingungen, deren Schlupfvariablen im Optimum null sind. Auch bei unserer Implementation des Verfahrens von Wolsey [150] verwenden wir die Komplementärschlupfbedingungen zur Feststellung dieser Nebenbedingungen.

A Lineare Optimierung

werden kann. Hierzu startet der Algorithmus mit einem beliebigen Eckpunkt des zulässigen Bereichs, also mit einer zulässigen Basislösung. Falls kein Start-Eckpunkt existiert, besitzt das lineare Programm keine optimale Lösung. In jedem Schritt versucht der Algorithmus, zu einem Nachbareckpunkt mit einem höheren Zielfunktionswert zu gelangen. Das nennt man *Basistausch*. Dadurch kommt genau eine Nichtbasisvariable neu in die Basis und dafür verlässt genau eine Basisvariable die Basis. Dieser Vorgang wird so lange wiederholt, bis eine optimale Lösung gefunden wird oder festgestellt wird, dass der Zielfunktionswert unbeschränkt ist. Der Simplex-Algorithmus sucht also systematisch Eckpunkte im zulässigen Bereich ab. Im Folgenden gehen wir in Anlehnung an Krelle/Künzi [67], S. 28 f., kurz auf die rechnerische Darstellung der Simplex-Methode ein. Wir leiten zunächst das Abbruchkriterium des Simplex-Algorithmus her. Zu diesem Zweck geben wir zuerst das lineare Programm in Normalform an, wobei wir explizit zwischen den Basis- und Nichtbasisvariablen unterscheiden:

$$\begin{aligned} \max \quad & z(\mathbf{x}) := \mathbf{c}^T \mathbf{x} = \mathbf{c}_B^T \mathbf{x}_B + \mathbf{c}_N^T \mathbf{x}_N \\ \text{unter den Nebenbedingungen} \\ & \mathbf{A} \mathbf{x} = \mathbf{A}_B \mathbf{x}_B + \mathbf{A}_N \mathbf{x}_N = \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0}. \end{aligned} \tag{A.6}$$

Wir lösen nun das Gleichungssystem $\mathbf{A}_B \mathbf{x}_B + \mathbf{A}_N \mathbf{x}_N = \mathbf{b}$ nach \mathbf{x}_B auf. Dies ergibt $\mathbf{x}_B = \mathbf{A}_B^{-1} \mathbf{b} - \mathbf{A}_B^{-1} \mathbf{A}_N \mathbf{x}_N$. Setzen wir dieses Resultat in die Zielfunktion ein, so erhalten wir

$$\begin{aligned} z(\mathbf{x}) &= \mathbf{c}_B^T (\mathbf{A}_B^{-1} \mathbf{b} - \mathbf{A}_B^{-1} \mathbf{A}_N \mathbf{x}_N) + \mathbf{c}_N^T \mathbf{x}_N \\ &= \mathbf{c}_B^T \mathbf{A}_B^{-1} \mathbf{b} + (\mathbf{c}_N^T - \mathbf{c}_B^T \mathbf{A}_B^{-1} \mathbf{A}_N) \mathbf{x}_N. \end{aligned}$$

Wann immer $(\mathbf{c}_N^T - \mathbf{c}_B^T \mathbf{A}_B^{-1} \mathbf{A}_N) > \mathbf{0}$ ist, können wir durch Einsetzen genügend kleiner, positiver Werte für \mathbf{x}_N den Wert der Zielfunktion $z(\mathbf{x})$ noch erhöhen, solange $\mathbf{x}_B \geq \mathbf{0}$ ist.⁷ Somit muss im Maximum $(\mathbf{c}_N^T - \mathbf{c}_B^T \mathbf{A}_B^{-1} \mathbf{A}_N) \leq \mathbf{0}$ gelten. Diese Bedingung ist das sog. *Simplex-Kriterium* für eine Optimallösung eines linearen Programms.

⁷Für einen beliebigen Vektor \mathbf{y} soll $\mathbf{y} > \mathbf{0}$ anzeigen, dass die Komponenten des Vektors \mathbf{y} nicht-negativ sind, wobei mindestens eine Komponente streng positiv ist.

A.2 Die Simplex-Methode

Die Simplex-Methode zur Lösung linearer Programme wird zweckmässig in sog. *Simplex-Tabellen* (Simplex-Tableaus) durchgeführt. In der Abbildung A.1 zeigen wir die Simplex-Tabelle eines linearen Programms in Normalform.

Abbildung A.1: Simplex-Tableau

	Nichtbasisvariablen					Basisvariablen						rechte Seite
	x_1	\dots	x_t	\dots	x_p	x_{p+1}	\dots	x_s	\dots	x_n	z	b_i
x_{p+1}	a_{11}	\dots	a_{1t}	\dots	a_{1p}	1	\dots	0	\dots	0	0	b_1
\vdots	\vdots	\ddots	\vdots	\ddots	\vdots	\vdots	\ddots	\vdots	\ddots	\vdots	\vdots	\vdots
x_s	a_{s1}	\dots	a_{st}	\dots	a_{sp}	0	\dots	1	\dots	0	0	b_s
\vdots	\vdots	\ddots	\vdots	\ddots	\vdots	\vdots	\ddots	\vdots	\ddots	\vdots	\vdots	\vdots
x_n	a_{n1}	\dots	a_{nt}	\dots	a_{np}	0	\dots	0	\dots	1	0	b_m
	$-c_1$	\dots	$-c_t$	\dots	$-c_p$	0	\dots	0	\dots	0	1	0
	$= -(\mathbf{c}_N^T - \mathbf{c}_B^T \mathbf{A}_B^{-1} \mathbf{A}_N)$											$= \mathbf{c}_B^T \mathbf{A}_B^{-1} \mathbf{b}$

Diese Tabelle ist eine Darstellung des Gleichungssystems $\mathbf{Ax} = \mathbf{b}$ in einem Tableau, das für die Zielfunktion um eine Zeile (letzte Zeile) und eine Spalte (zweitletzte Spalte) erweitert ist.⁸ Die Zielfunktion $z(\mathbf{x})$ wird als Basisvariable interpretiert. Da sie die Basis nie verlassen wird, können wir auf die Spalte z verzichten. Die letzte Zeile nennen wir *Zielfunktionszeile* oder Zeile z . Zweckmässigerweise werden in der Literatur die Koeffizienten c_j , $j = 1, \dots, p$, mit negativem Vorzeichen im Simplex-Tableau eingetragen. Folglich ist das Maximum der Zielfunktion erreicht, wenn durch eventuell mehrmaligen Basiswechsel alle Koeffizienten c_j nicht-negativ geworden sind.

Der Simplex-Algorithmus, der zur Bestimmung einer optimalen Lösung eines Optimierungsproblems in der Regel ein mehrmaliges Umrechnen der Simplex-Tabelle erfordert, besteht aus drei Schritten (vgl. Abbildung A.1):⁹

Schritt 1 (Wahl der Pivotspalte)

Wenn die Zielfunktionszeile nur nicht-negative Koeffizienten enthält, ist die aktuelle Basislösung optimal. Der Algorithmus wird beendet. Sonst wird in der Zeile z die

⁸In der Abbildung A.1 sind die Variablen so unnummeriert, dass die ersten p Spalten Nichtbasisvariablen und die restlichen $n - p$ Variablen Basisvariablen sind.

⁹Vgl. Zimmermann [156], S. 51 ff., und Domschke/Drexel [34], S. 22 f.

Spalte t mit dem kleinsten (negativen) Wert gesucht. Die so gefundene Ecke generiert den grössten Zuwachs der Zielfunktion. Falls mehrere Spalten mit kleinstem Wert zur Auswahl stehen, wählt man unter diesen eine beliebige Spalte. Die zu der Spalte t zugehörige Nichtbasisvariable x_t wird neu in die Basis aufgenommen. Die Spalte t nennt man *Pivotspalte*.

Schritt 2 (Wahl der Pivotzeile)

Falls kein Element der Pivotspalte t positiv ist, also $a_{it} \leq 0$ für $i = 1, 2, \dots, m$ gilt, existiert keine optimale Lösung. In diesem Fall ist die Zielfunktion im zulässigen Bereich des linearen Programms unbeschränkt. Der Algorithmus wird beendet. Ansonsten bestimmt man eine Zeile s , die die folgende Eigenschaft besitzt:

$$\frac{b_s}{a_{st}} = \min \left\{ \frac{b_i}{a_{it}} \mid i = 1, 2, \dots, m \text{ mit } a_{it} > 0 \right\}.$$

Dieses Kriterium gewährleistet, dass keine der Variablen des linearen Programms negativ wird.¹⁰ Die Basisvariable, die zu der Zeile s gehört, verlässt die Basis. Die Zeile s nennt man *Pivotzeile*, das Element a_{st} heisst *Pivotelement*.

Schritt 3 (Basiswechsel)

Durch lineare Transformation des Simplex-Tableaus wird eine neue zulässige Basislösung geschaffen, die einen grösseren (im Grenzfall unveränderten) Zielfunktionswert besitzt. Der Basistausch führt dazu, dass die Pivotspalte t zu einer Einheitsspalte mit $a_{st} = 1$ wird. Hierzu wird die Pivotzeile mit $\frac{1}{a_{st}}$ multipliziert, und die anderen Elemente der Zeile i ($i \neq s$, $i = 1, 2, \dots, m$) des Simplex-Tableaus werden durch $a'_{ij} = a_{ij} - a_{it} \left(\frac{a_{sj}}{a_{st}} \right)$ (sog. Dreiecksregel) ersetzt.

Wie wir zu Beginn dieses Abschnitts erwähnt haben, startet die Simplex-Methode mit einer beliebigen zulässigen Basislösung. Eine solche Startecke ist typischerweise der Ursprungspunkt, in dem alle Variablen null sind. Falls der Ursprungspunkt keine zulässige Lösung ist, berechnen wir eine solche. Hierfür gibt es verschiedene Methoden. Eine davon beinhaltet, in einer ersten Phase ein lineares Hilfsprogramm zu lösen. Dies führt dann entweder zu einer zulässigen Basislösung des eigentlichen linearen Programms oder zu dem Ergebnis, dass das eigentliche Problem keine zulässige Basis

¹⁰Den Beweis dieser Aussage finden wir z. B. in Bloech [10], S. 69 f., oder in Krelle/Künzi [67], S. 47 f.

und somit keine Lösung besitzt. Falls eine zulässige Basislösung gefunden ist, wird in der zweiten Phase das eigentliche Optimierungsproblem gelöst, wobei nun mit der zuvor bestimmten Startecke begonnen wird. Man spricht in diesem Zusammenhang von der sog. *Zwei-Phasen-Methode*. Zur Veranschaulichung der Zwei-Phasen-Methode nehmen wir an, dass ein lineares Optimierungsproblem in Standardform gegeben ist. In der ersten Phase fügen wir dem Gleichungssystem $\mathbf{Ax} = \mathbf{b}$ die sog. *künstlichen Schlupfvariablen* $\mathbf{z}^T = (z_1 \ z_2 \ \dots \ z_m)$ hinzu. Die Bestimmung einer Startecke erfolgt dann im Wege der Lösung des folgenden linearen Hilfsprogramms:

$$\begin{aligned} \text{LP}_H: \quad & \min \sum_{j=1}^m z_j \\ & \text{unter den Nebenbedingungen} \\ & \mathbf{Ax} + \mathbf{Iz} = \mathbf{b} \\ & \mathbf{x}, \mathbf{z} \geq \mathbf{0}, \end{aligned} \tag{A.7}$$

wobei hier \mathbf{I} eine quadratische $(m \times m)$ -Identitätsmatrix (Einheitsmatrix) ist. Man kann zeigen, dass der zulässige Bereich jedes linearen Programms LP genau dann nicht-leer ist, wenn das zugehörige LP_H den minimalen Zielfunktionswert Null hat. Ist hingegen dieser minimale Zielfunktionswert grösser als Null, so ist der zulässige Bereich von LP leer.¹¹ Falls im Optimum die Zielfunktion von LP_H den Wert Null besitzt, sind alle künstlichen Variablen Null. In diesem Fall bilden die Werte der Variablen \mathbf{x} eine Startecke.

Zum Schluss dieses Abschnitts stellen wir zwei Spezialfälle vor, die im Optimum eines linearen Programms auftreten können. Erstens kann ein lineares Optimierungsproblem mehrere optimale Basislösungen haben. Das ist immer dann der Fall, wenn im optimalen Simplex-Tableau mindestens eine Nichtbasisvariable in der Zeile z den Wert Null besitzt. Zweitens kann eine der Basisvariablen einer Basislösung den Wert Null haben (sog. *Degeneration*). Eine solche Ecke nennt man *entartete Ecke*. In diesem Fall ändert sich der Zielfunktionswert beim Übergang zur neuen Basis nicht und

¹¹Der Beweis ist z. B. in Bomze/Grossmann [11], S. 112 zu finden.

es kann vorkommen, dass eine Ecke mehrmals besucht wird, was im Extremfall zu einem zyklischen Verhalten des Simplex-Algorithmus führen kann.¹²

A.3 Die Revidierte Simplex-Methode

Wie wir im vorangegangenen Abschnitt erläutert haben, rechnen wir bei der Simplex-Methode für jeden Basistausch das gesamte Simplex-Tableau um, obwohl für jede Iteration lediglich folgende Informationen erforderlich sind: Für die Bestimmung der Pivotspalte t und die Überprüfung, ob das Optimum erreicht worden ist, sind in jeder Iteration die aktuellen Koeffizienten der Nichtbasisvariablen, die wir als \hat{c}_j , $j = 1, \dots, p$, bezeichnen, in der Zielfunktionszeile zu berechnen. Ferner sind für die Bestimmung der Pivotzeile s die aktuelle rechte Seite, d. h. die Werte der momentan zulässigen Basislösung, sowie der Spaltenvektor der neu in die Basis aufzunehmenden Nichtbasisvariablen, also die Spalte t , zu bestimmen. Die revidierte Simplex-Methode generiert diese Informationen, ohne dass sie das gesamte Simplex-Tableau umrechnet. Statt dessen wird in jeder Iteration nur die Inverse der jeweils aktuellen Basismatrix \mathbf{A}_B berechnet. Die Koeffizientenmatrix \mathbf{A} und den Koeffizientenvektor der Zielfunktion \mathbf{c} benötigt man als Anfangsdaten für die Bestimmung der aktuellen Werte von \mathbf{A}_B , \mathbf{b} ($= \mathbf{x}_B$) und \mathbf{c} . Sie werden deswegen in einem anderen Tableau separat gespeichert und bleiben während der ganzen Berechnung unverändert.

Zur Erläuterung der revidierten Simplex-Methode gehen wir von einem linearen Optimierungsproblem LP in Standardform aus und nehmen an, dass eine zulässige Basislösung und die zugehörige Basismatrix \mathbf{A}_B gegeben sind.

- Die neuen Koeffizienten $\hat{\mathbf{c}}_N$ der Nichtbasisvariablen in der Zeile z werden durch $(\hat{\mathbf{c}}_N)^T = (\mathbf{c}_N^T - \mathbf{c}_B^T \mathbf{A}_B^{-1} \mathbf{A}_N)$ gewonnen. In diesem Ausdruck sind ausser \mathbf{A}_B^{-1} alle Terme bekannt. Nach der Berechnung von \mathbf{A}_B^{-1} können wir die neuen Werte \hat{c}_j , $j = 1, 2, \dots, p$, durch $\hat{c}_j = c_j - \mathbf{u}^T \mathbf{a}_{.j}$ mit $\mathbf{u} := (\mathbf{A}_B^{-1})^T \mathbf{c}_B$ festlegen, wobei $\mathbf{a}_{.j}$

¹²Dieser Fall ist aber in der Praxis sehr selten, so dass wir bei unserer Implementation auf mögliche Vorkehrungen wie z. B. die Regel von Bland verzichten. Die Anwendung der Regel von Bland garantiert, dass solche Zyklen nicht auftreten. Für eine Definition der Regel von Bland vgl. z. B. Beisel/Mendel [5], S. 34.

der Spaltenvektor j , $j \in \{1, 2, \dots, p\}$, der *ursprünglichen* Matrix \mathbf{A} ist. Eine der Nichtbasisvariablen t , für die $\hat{c}_t < 0$ gilt, kann in der nächsten Iteration in die Basis genommen werden

- Zur Bestimmung der Pivotzeile sind der aktuelle Spaltenvektor $\hat{\mathbf{a}}_t$ der Pivotspalte t und die aktuelle rechte Seite $\hat{\mathbf{b}}$ erforderlich. Wir erhalten diese durch $\hat{\mathbf{a}}_t = \mathbf{A}_B^{-1} \mathbf{a}_t$ und $\hat{\mathbf{b}} = \mathbf{A}_B^{-1} \mathbf{b}$. Diese zwei Gleichungen können wir wie folgt herleiten: Es gilt $\mathbf{x}_B = \mathbf{A}_B^{-1} \mathbf{b} - \mathbf{A}_B^{-1} \mathbf{A}_N \mathbf{x}_N$. Da nun für eine zulässige Basislösung $\mathbf{x}_N = \mathbf{0}$ gilt, ist $\mathbf{x}_B = \hat{\mathbf{b}} = \mathbf{A}_B^{-1} \mathbf{b}$. Die $(m \times p)$ -Matrix $\mathbf{A}_B^{-1} \mathbf{A}_N$ beinhaltet die Spalten der Nichtbasisvariablen. Somit kann eine Spalte t explizit dargestellt werden als $\hat{\mathbf{a}}_t = \mathbf{A}_B^{-1} \mathbf{a}_t$, wobei hier zu beachten ist, dass \mathbf{a}_t die Spalte t in der Anfangsmatrix \mathbf{A}_N ist.

Da wir in unserer Implementation des Verfahrens nach Wolsey [150] zur Lösung der linearen Programme die revidierte Simplex-Methode verwendet haben, geben wir nachfolgend in Anlehnung an Syslo et al. [139], S. 5 f., und Neumann/Morlock [90], S. 112 f., die einzelnen Schritte dieser Methode für ein lineares Programm in Standardform an.

Es sei eine Basismatrix \mathbf{A}_B gegeben, wobei $\mathbf{x}_0 := \mathbf{x}_B = \hat{\mathbf{b}} = \mathbf{A}_B^{-1} \mathbf{b} \geq \mathbf{0}$ gilt. Wir berechnen zunächst den Vektor der Dualvariablen $\mathbf{u} := (\mathbf{A}_B^{-1})^T \mathbf{c}_B$ und anschliessend die neuen Koeffizienten der Zeile z

$$(\hat{\mathbf{c}}_N)^T = (\mathbf{c}_N^T - \mathbf{c}_B^T \mathbf{A}_B^{-1} \mathbf{A}_N) = (\mathbf{c}_N^T - \mathbf{u}^T \mathbf{A}_N).$$

Falls $(\hat{\mathbf{c}}_N)^T = (\mathbf{c}_N^T - \mathbf{u}^T \mathbf{A}_N) \geq \mathbf{0}$ gilt, ist die aktuelle zulässige Basislösung optimal. Wir beenden die Berechnung. Sonst bestimmen wir die Pivotspalte und die Pivotzeile. Als Pivotspalte wählen wir in der Nichtbasismatrix \mathbf{A}_N die Spalte \mathbf{a}_t mit der Eigenschaft, dass \hat{c}_t der kleinste (negative) Wert ist. Wir berechnen sodann die Koeffizienten der Pivotspalte t durch $\hat{\mathbf{a}}_t = \mathbf{A}_B^{-1} \mathbf{a}_t$, wobei \mathbf{a}_t die Spalte t in der Anfangsmatrix \mathbf{A}_N ist. Anschliessend bestimmen wir die Pivotzeile s durch

$$\theta := \frac{x_{0s}}{\hat{a}_{st}} = \min \left\{ \frac{x_{0i}}{\hat{a}_{it}} \mid i = 1, 2, \dots, m \text{ mit } \hat{a}_{it} > 0 \right\}.$$

A Lineare Optimierung

Die zu der Pivotzeile s gehörende Basisvariable x_{Bs} verlässt die Basis. Falls keines der Elemente a_{it} positiv ist, kann die Zielfunktion $z(\mathbf{x})$ beliebig erhöht werden; die Zielfunktion ist also unbeschränkt und folglich beenden wir die Berechnung, weil in diesem Fall keine Lösung existiert. Andernfalls aktualisieren wir die zulässige Basislösung:

$$x_{Bi} = \hat{b}_i = \begin{cases} x_{0i} - \theta \hat{a}_{it} & \text{für } i \neq t \\ \theta & \text{für } i = t, \end{cases}$$

wobei x_{Bt} die neue Basisvariable ist. Wir bestimmen die Basismatrix \mathbf{A}_B und berechnen erneut den Vektor der Dualvariablen \mathbf{u} usw.

Die obige Beschreibung der revidierten Simplex-Methode erfordert, dass wir in jeder Iteration die aktuelle Basismatrix \mathbf{A}_B invertieren. Die Inverse \mathbf{A}_B^{-1} können wir z. B. mit Hilfe des Gauss-Jordan-Verfahrens¹³ in jeder Iteration explizit bestimmen. Da sich aber die Matrix \mathbf{A}_B in jeder Iteration nur in einer Spalte, nämlich in der Pivotspalte, ändert, können wir die Berechnung von \mathbf{A}_B^{-1} durch die sog. *Produktform der Inversen* vereinfachen.¹⁴

Im Allgemeinen steigt die Anzahl der Austauschschritte der Simplex-Methode und ihrer Varianten (z. B. die revidierte Simplex-Methode und die hier nicht behandelte duale Simplex-Methode) bei der Lösung grosser Optimierungsaufgaben linear mit der Anzahl der Zeilen und der Spalten der Koeffizientenmatrix \mathbf{A} . In der Regel wächst hingegen der Rechenaufwand eines Austauschschritts, d. h. die Anzahl der durchgeführten arithmetischen Rechenoperationen in einem Austauschschritt, quadratisch mit m und linear mit n .¹⁵ Klee und Minty [61] haben jedoch in ihrer Arbeit am Beispiel eines speziellen Typs von linearen Optimierungsaufgaben gezeigt, dass die

¹³Durch das Gauss-Jordan-Verfahren kann man die Inverse einer $(m \times m)$ -Matrix \mathbf{A} durch elementare Zeilenumformung von $(\mathbf{A}|\mathbf{I})$ in $(\mathbf{I}|\mathbf{A}^{-1})$ berechnen. Hierbei ist \mathbf{I} eine $(m \times m)$ -Einheitsmatrix. Vgl. dazu z. B. Schaback/Werner [115], S. 52–54.

¹⁴Die Produktform der Inversen einer Basismatrix \mathbf{A}_B ist wie folgt definiert (vgl. z. B. Syslo et al. [139], S. 29 f., oder Neumann/Morlock [90], S. 116 f.): Es seien $\mathbf{A}_B = [\mathbf{a}_1 \ \mathbf{a}_2 \ \dots \ \mathbf{a}_s \ \dots \ \mathbf{a}_m]$ und $\hat{\mathbf{A}}_B = [\hat{\mathbf{a}}_1 \ \hat{\mathbf{a}}_2 \ \dots \ \hat{\mathbf{a}}_s \ \dots \ \hat{\mathbf{a}}_m]$ zwei Basismatrizen, die sich in der Spalte s unterscheiden. Es sei $\mathbf{A}_B^{-1} \hat{\mathbf{a}}_s = (y_1 \ y_2 \ \dots \ y_s \ \dots \ y_m)^T$. Dann gilt $\hat{\mathbf{A}}_B^{-1} = \mathbf{E} \mathbf{A}_B^{-1}$, wobei $\mathbf{E} = (\mathbf{e}^1 \ \dots \ \mathbf{g} \ \dots \ \mathbf{e}^m)$. Die Spalte \mathbf{g} ist definiert durch $\mathbf{g} := (-\frac{y_1}{y_s} \ -\frac{y_2}{y_s} \ \dots \ \frac{1}{y_s} \ \dots \ -\frac{y_m}{y_s})$. Die anderen Spaltenvektoren \mathbf{e}^i , $i = 1, 2, \dots, m$, sind jeweils Einheitsvektoren mit einer Eins an der Stelle i . Da \mathbf{A}_B eine Einheitsmatrix ist, gilt in der ersten Iteration der revidierten Simplex-Methode $\hat{\mathbf{A}}_B^{-1} = \mathbf{E} \mathbf{A}_B^{-1} = \mathbf{E} \mathbf{I} = \mathbf{E}$.

¹⁵Für eine ausführliche Behandlung des Rechenaufwandes der Simplex-Methode vgl. z. B. Shamir [121] oder Borgwardt [13]. Für einen Überblick vgl. Neumann/Morlock [90], S. 160.

Anzahl der Austauschschritte im ungünstigsten Fall exponentiell mit m und n steigt (vgl. dazu Beisel/Mendel [5], S. 146). Deshalb hat man Alternativen zur Simplex-Methode mit einem potentiell geringeren Rechenaufwand entwickelt. Die zwei bekanntesten sog. *Interior Point-Methoden* sind die *Ellipsoid-Methode* von Khachijan [60] und die *Projektionsmethode* von Karmarkar [57]. Während bei der Simplex-Methode der Rand des zulässigen Bereichs nach einer optimalen Lösung durchsucht wird, geht man bei den Interior Point-Methoden von einer im Inneren des zulässigen Bereichs liegenden Lösung aus.¹⁶ In der Literatur (vgl. z. B. Godfarb/Todd [46], Abschnitte II.8 und II.9 oder Schrijver [117], Abschnitte 13–15) hat man gezeigt, dass die beiden Interior Point-Methoden im Gegensatz zu der Simplex-Methode einen polynomialen Aufwand besitzen. Die Analyse der beiden Methoden bezüglich des *durchschnittlichen* Rechenaufwandes (engl. average performance) hat jedoch gezeigt, dass die Ellipsoid-Methode im Vergleich zur Simplex-Methode einen sehr hohen durchschnittlichen Rechenaufwand erfordert, während der Aufwand der Projektionsmethode von Karmarkar mit der Simplex-Methode vergleichbar ist. Zudem wurde festgestellt, dass die beiden Interior Point-Methoden zur Lösung praktischer Probleme keine bessere Alternative zur Simplex-Methode darstellen.¹⁷

A.4 Rucksackprobleme

Die *Rucksackprobleme* (Knapsack-Probleme) sind eine spezielle Form von ganzzahligen linearen Optimierungsaufgaben. Ganzzahlige lineare Optimierungsaufgaben sind dadurch charakterisiert, dass die Variablen nur ganzzahlige Werte annehmen dürfen.¹⁸ Um die Rucksackprobleme vorzustellen, betrachten wir bildlich einen Rucksack, in den Gegenstände verschiedenen Gewichts eingepackt werden sollen. Es stehen insgesamt n Gegenstände x_j , $j = 1, 2, \dots, n$, mit den Werten (den Nutzen) $v_j \in \mathbb{R}_{>0}$

¹⁶Eine genaue Beschreibung dieser Algorithmen ist z. B. in Beisel/Mendel [5], S. 146 ff., oder Neumann/Morlock [90], S. 161 ff. zu finden.

¹⁷Für weiter gehende Ausführungen hierzu vgl. z. B. Shamir [121] oder Neumann/Morlock [90], S. 160 ff.

¹⁸Für eine vollständige Darstellung von ganzzahligen linearen Programmen und deren Lösungen vgl. z. B. Wolsey [151] oder Walukiewicz [145].

A Lineare Optimierung

und den Gewichten $w_j \in \mathbb{R}_{>0}$ zur Auswahl. Das zu lösende Problem lautet: Welche der Gegenstände sollen eingepackt werden, damit das zulässige Gesamtgewicht (die Kapazität des Rucksacks) C nicht überschritten und der Wert der eingepackten Gegenstände maximal wird? Da es hier nur darum geht, ob ein Gegenstand in den Rucksack eingepackt werden soll oder nicht, verwenden wir Binärvariablen der Form

$$x_j = \begin{cases} 1, & \text{falls der Gegenstand } j \text{ eingepackt wird,} \\ 0, & \text{sonst.} \end{cases}$$

Dieses Rucksackproblem können wir als eine sog. *binäre* lineare Optimierungsaufgabe wie folgt formulieren:

$$\begin{aligned} \text{IP}_B : \quad & \max \sum_{j=1}^n v_j x_j \\ & \text{unter der Nebenbedingung} \\ & \sum_{j=1}^n w_j x_j \leq C \\ & x_j \in \{0, 1\} \text{ für } j = 1, 2, \dots, n. \end{aligned} \tag{A.8}$$

Wir nehmen an, dass $\sum_{j=1}^n w_j > C$ gilt, da sonst das Problem die triviale Lösung $x_j = 1$ für alle $j = 1, 2, \dots, n$ besitzt.

Es gibt verschiedene Methoden zur Lösung des IP_B . Eine davon ist die Methode der dynamischen Optimierung. Wir stellen hier in Anlehnung an Burkard [24], S. 426 f., Wolsey [151], S. 73 f., und Parsons [100], S. 361 ff., ein rekursives dynamisches Verfahren zur Lösung des IP_B vor. Zu diesem Zweck definieren wir zunächst für $k = 0, 1, \dots, n$ und $c = 0, 1, \dots, C$ die sog. *Rucksackfunktion* $f_k(c)$, die den optimalen Wert des Rucksackproblems

$$\begin{aligned} P_k(c) : \quad & \max \sum_{j=1}^k v_j x_j \\ & \text{unter der Nebenbedingung} \\ & \sum_{j=1}^k w_j x_j \leq c \\ & x_j \in \{0, 1\} \text{ für } j = 1, 2, \dots, k \end{aligned} \tag{A.9}$$

bezüglich der ersten k Variablen ($k \leq n$) und der rechten Seite c ($c \leq C$) angibt. Zudem setzen wir $f_{k=0}(c) = 0$ für alle c und $f_k(0) = 0$ für alle $k = 1, 2, \dots, n$. Aus der Definition der Rucksackfunktion folgt, dass $z^* := f_n(C)$ den optimalen Wert des Rucksackproblems IP_B angibt. Wir definieren nun eine rekursive Funktion, welche es uns ermöglicht, $f_k(c)$ mit Hilfe der Werte $f_s(\tilde{c})$ für $s \leq k$ und $\tilde{c} < c$ zu berechnen. Es sei $(x_1^* \ x_2^* \ \dots \ x_k^*)$ eine optimale Lösung des Problems $P_k(c)$, wobei $f_k(c)$ der optimale Wert ist. Jede Variable x_k nimmt im Optimum entweder den Wert Null oder den Wert Eins an. Falls $x_k^* = 0$ ist, gilt $f_k(c) = f_{k-1}(c)$. Ist hingegen $x_k^* = 1$, so gilt

$$\begin{aligned} f_k(c) &= v_k x_k^* + f_{k-1}(c - w_k x_k^*) \\ &= v_k + f_{k-1}(c - w_k). \end{aligned}$$

Folglich erhalten wir

$$f_k(c) = \max\{f_{k-1}(c), \ v_k + f_{k-1}(c - w_k)\}. \quad (\text{A.10})$$

Die Gleichung (A.10) besagt, dass wir den Wert der Rucksackfunktion $f_k(c)$ berechnen können, wenn wir den Wert $f_{k-1}(c - w_k)$ kennen. Wir können diesen Tatbestand ausnutzen, um eine Rekursion zu bilden, die mit $f_{k=0}(c) = 0$ für alle $0 \leq c \leq C$ beginnt oder alternativ mit

$$f_1(c) = \begin{cases} 0, & \text{falls } 0 \leq c < w_1, \\ v_1, & \text{falls } c \geq w_1. \end{cases} \quad (\text{A.11})$$

Mit Hilfe dieser Rekursion können wir nun die Werte der restlichen Rucksackfunktionen f_2, f_3, \dots, f_n für alle ganzzahligen Werte von c , $0 \leq c \leq C$, berechnen und sie in einer $(n+1) \times (C+1)$ -Matrix speichern. Nachdem wir diese Matrix berechnet haben, bestimmen wir in der nächsten Phase eine optimale Lösung $(x_1^*, x_2^*, \dots, x_n^*)$ des ursprünglichen Rucksackproblems IP_B . Zu diesem Zweck iterieren wir zurück, beginnend mit dem optimalen Wert $f_n(C)$. Wir vergleichen zunächst $f_n(C)$ mit $f_{n-1}(C)$, um den optimalen Wert von x_n^* zu bestimmen:

$$x_n^* = \begin{cases} 0, & \text{falls } f_n(C) = f_{n-1}(C), \\ 1, & \text{sonst.} \end{cases}$$

Im nächsten Schritt berechnen wir x_{n-1}^* . Falls wir im ersten Schritt berechnet haben, dass $x_n^* = 0$ ist, dann steht uns für die Bestimmung von x_{n-1}^* die gesamte

A Lineare Optimierung

Kapazität C zur Verfügung. Wir vergleichen also $f_{n-1}(C)$ mit $f_{n-2}(C)$. Falls aber $x_n^* = 1$ ist, ist der Gegenstand n mit dem Gewicht w_n in den Rucksack einzupacken. Somit steht uns für die Bestimmung von x_{n-1}^* nicht mehr die volle Kapazität C zur Verfügung, sondern nur noch $C - w_n$. Wir streben also weiterhin eine optimale Lösung von $f_{n-1}(C - w_n)$ an. In diesem Fall vergleichen wir $f_{n-1}(C - w_n)$ mit $f_{n-2}(C - w_n)$, um zu beurteilen, ob der Gegenstand $n - 1$ eingepackt werden soll oder nicht. Nach n Schritten erhalten wir die optimale Lösung des Rucksackproblems. Der Algorithmus besteht also aus einer Vorwärtsrekursion, in der wir die Werte $f_j(c)$ berechnen und aus einer Rückwärtsrekursion, in der wir einen optimalen Lösungsvektor $\mathbf{x}^* = (x_1 \dots x_n)$ bestimmen. Zur Illustration des Verfahrens betrachten wir das nachfolgende Rucksackproblem:

$$\begin{aligned} \max \quad & z(\mathbf{x}) := 8x_1 + 12x_2 + 5x_3 + 9x_4 \\ \text{unter der Nebenbedingung} \quad & \\ & 2x_1 + 5x_2 + 4x_3 + 3x_4 \leq 8 \\ & x_j \in \{0, 1\}, \quad j = 1, 2, 3, 4. \end{aligned} \tag{A.12}$$

Die Werte $f_1(c)$, $c = 1, 2, \dots, 8$, haben wir mit Hilfe der Gleichung (A.11) und die anderen Werte $f_k(c)$, $k = 2, 3, 4$, $c = 1, 2, \dots, 8$, mit Hilfe der Rekursionsgleichung (A.10) berechnet. Diese Werte sind in der Tabelle A.1 angegeben.

Tabelle A.1: Alle Werte von $f_n(c)$

	Die Werte von c								
	$c = 0$	$c = 1$	$c = 2$	$c = 3$	$c = 4$	$c = 5$	$c = 6$	$c = 7$	$c = 8$
f_0	0	0	0	0	0	0	0	0	0
f_1	0	0	8	8	8	8	8	8	8
f_2	0	0	8	8	8	12	12	20	20
f_3	0	0	8	8	8	12	13	20	20
f_4	0	0	8	9	9	17	17	20	21

Zum Beispiel gilt für $n = 2$ und $c = 8$

$$f_2(8) = \max \{f_1(8), 12 + f_1(8 - 5)\} = \max \{8, 12 + 8\} = 20.$$

A.5 Berechnung der K -besten Lösung eines binären linearen Optimierungsproblems

Mit Hilfe der Werte von $f_j(c)$ in der Tabelle A.1 können wir nun durch Rückwärtsrekursion einen optimalen Vektor $\mathbf{x}^* = (x_1^* \ x_2^* \ x_3^* \ x_4^*)$ bestimmen. Da $f_4(8) = 21 \neq f_3(8) = 20$ ist, gilt $x_4^* = 1$. Die Restkapazität ist nun $C - w_4 = 7 - 3 = 4$. Für die Berechnung von x_3^* vergleichen wir $f_3(4)$ mit $f_2(4)$. Da $f_2(4) = f_3(4) = 8$ ist, gilt $x_3^* = 0$. Analog stellen wir fest, dass $x_2^* = 1$ und $x_1^* = 0$ ist. Somit lautet eine optimale Lösung des Rucksackproblems $\mathbf{x}^* = (0 \ 1 \ 0 \ 1)$, mit dem maximalen Zielfunktionswert $z^* = 21$.

A.5 Berechnung der K -besten Lösung eines binären linearen Optimierungsproblems

Lawler [72] schlägt einen Algorithmus vor, mit Hilfe dessen wir eine sog. K -beste Lösung, z. B. eine zweitbeste oder eine fünftbeste Lösung, eines binären linearen Optimierungsproblems

$$\begin{aligned} \text{IP:} \quad & \max \mathbf{c}^T \mathbf{x} \\ & \text{unter der Nebenbedingung} \\ & \mathbf{Ax} \leq \mathbf{b} \\ & x_j \in \{0, 1\} \text{ für } j = 1, 2, \dots, n \end{aligned} \tag{A.13}$$

berechnen können.¹⁹ In (A.13) sind \mathbf{c} und \mathbf{x} Spaltenvektoren mit n Komponenten, \mathbf{A} eine $(m \times n)$ -Matrix und \mathbf{b} ein m -dimensionaler Spaltenvektor.

Um eine K -beste Lösung einer Optimierungsaufgabe IP mit Hilfe des Verfahrens von Lawler [72] zu berechnen, bestimmen wir für $k = 1, 2, \dots, K$ schrittweise eine k -beste Lösung. Also berechnen wir nacheinander zunächst eine beste Lösung ($k = 1$), dann mit Hilfe dieser eine zweitbeste Lösung ($k = 2$), usw., und am Schluss eine K -beste Lösung. Wir nehmen nun an, dass X die Menge der zulässigen Lösungen des binären linearen Programms IP ist und $\mathbf{x}^{(1)} = (x_1^{(1)} \ x_2^{(1)} \ \dots \ x_n^{(1)})$ als eine optimale Lösung des IP für $k = 1$ bestimmt worden ist.²⁰ Für die Berechnung einer zweit-

¹⁹Ein Rucksackproblem ist eine Optimierungsaufgabe der Form IP mit nur einer Nebenbedingung.

²⁰Für Rucksackprobleme kann die Lösung für $k = 1$ mit Hilfe des im vorangegangenen Abschnitt vorgestellten rekursiven dynamischen Verfahrens berechnet werden. Für die Lösung allgemeiner Optimierungsaufgaben IP vgl. z. B. Syslo et al. [139], S. 100 ff.

A Lineare Optimierung

besten Lösung schliessen wir nun den Vektor $\mathbf{x}^{(1)}$ aus der Menge X aus. Zu diesem Zweck definiert Lawler n Optimierungsaufgaben IP^i , $i = 1, 2, \dots, n$, die sich vom binären linearen Programm IP dadurch unterscheiden, dass für jedes IP^i insgesamt t Variablen, $t = 1, 2, \dots, i$, wie folgt fixiert werden:

$$\begin{aligned} \text{Für } \text{IP}^1 : & \quad x_1 := 1 - x_1^{(1)}, \\ \text{Für } \text{IP}^2 : & \quad x_1 := x_1^{(1)}, \quad x_2 := 1 - x_2^{(1)}, \\ \text{Für } \text{IP}^3 : & \quad x_1 := x_1^{(1)}, \quad x_2 := x_2^{(1)}, \quad x_3 := 1 - x_3^{(1)}, \\ & \quad \vdots \\ \text{Für } \text{IP}^n : & \quad x_1 := x_1^{(1)}, \quad x_2 := x_2^{(1)}, \dots, \quad x_{n-1} := x_{n-1}^{(1)}, \quad x_n := 1 - x_n^{(1)}. \end{aligned} \tag{A.14}$$

Es seien $X^{(1)}$, $X^{(2)}$, ..., $X^{(n)}$ die Mengen der zulässigen Lösungen der Probleme IP^1 , IP^2 , ..., IP^n . Demnach gilt per Definition der Optimierungsprobleme IP^i , $i = 1, 2, \dots, n$, dass

$$X^{(1)} \cup X^{(2)} \cup \dots \cup X^{(n)} = X \setminus \{\mathbf{x}^{(1)}\}.$$

Folglich schliesst (A.14) die Lösung $\mathbf{x}^{(1)}$ im weiteren Verlauf der Berechnungen aus. Wir bestimmen nun eine optimale Lösung jedes Problems IP^i , $i = 1, 2, \dots, n$, dessen zulässiger Bereich $X^{(i)}$ nicht-leer ist. Die Lösung eines Problems, dessen Zielfunktionswert maximal ist, ist dann die zweitbeste Lösung $\mathbf{x}^{(2)} = (x_1^{(2)} \ x_2^{(2)} \ \dots \ x_n^{(2)})$ des IP . Für die Bestimmung der drittbesten Lösung ($k = 3$) gehen wir nun von der zweitbesten Lösung $\mathbf{x}^{(2)}$ aus. Wir nehmen an, dass der Vektor $\mathbf{x}^{(2)}$ nach der Fixierung der Variablen $x_1 \ x_2 \ \dots \ x_s$, $s \leq n$ gefunden worden ist. Analog zur Berechnung der zweitbesten Lösung konstruieren wir diesmal $n - s$ neue Optimierungsprobleme IP^i , $i = 1, 2, \dots, n - s$, wobei in jedem Optimierungsproblem IP^i die ersten s Variablen ihren bereits fixierten Wert bewahren und die Variablen x_{s+t} , $t = 1, 2, \dots, i$, wie folgt zu fixieren sind:

$$\begin{aligned} \text{Für } \text{IP}^1 : & \quad x_{s+1} := 1 - x_{s+1}^{(2)}, \\ \text{Für } \text{IP}^2 : & \quad x_{s+1} := x_{s+1}^{(2)}, \quad x_{s+2} := 1 - x_{s+2}^{(2)}, \\ \text{Für } \text{IP}^3 : & \quad x_{s+1} := x_{s+1}^{(2)}, \quad x_{s+2} := x_{s+2}^{(2)}, \quad x_{s+3} := 1 - x_{s+3}^{(2)}, \\ & \quad \vdots \\ \text{Für } \text{IP}^{n-s} : & \quad x_{s+1} := x_{s+1}^{(2)}, \quad x_{s+2} := x_{s+2}^{(2)}, \dots, \quad x_{n-1} := x_{n-1}^{(2)}, \quad x_n := 1 - x_n^{(2)}. \end{aligned}$$

A.5 Berechnung der K -besten Lösung eines binären linearen Optimierungsproblems

Anschliessend berechnen wir für jede Optimierungsaufgabe IP^i , $i = 1, 2, \dots, n - s$, mit nicht-leerem zulässigen Bereich eine optimale Lösung. Die Lösung eines Problems, das den maximalen Zielfunktionswert besitzt, ist die drittbeste Lösung $\mathbf{x}^{(3)} = (x_1^{(3)} \ x_2^{(3)} \ \dots \ x_n^{(3)})$ des IP. Diese Schritte wiederholen wir bis $k = K$.

Zur Illustration des oben vorgestellten Algorithmus berechnen wir die zweitbeste Lösung ($K = 2$) des auf S. 298 vorgeführten Rucksackproblems

$$\begin{aligned} \max \quad & z(\mathbf{x}) := 8x_1 + 12x_2 + 5x_3 + 9x_4 \\ \text{unter der Nebenbedingung} \\ & 2x_1 + 5x_2 + 4x_3 + 3x_4 \leq 8 \\ & x_j \in \{0, 1\}, \quad j = 1, 2, 3, 4. \end{aligned}$$

Um die zweitbeste Lösung zu berechnen, bestimmen wir zunächst eine optimale Lösung des Problems ($k = 1$). Eine solche Lösung haben wir bereits auf S. 298 berechnet:

$$\mathbf{x}^{(1)} = (x_1^{(1)} \ x_2^{(1)} \ x_3^{(1)} \ x_4^{(1)}) = (0 \ 1 \ 0 \ 1) \quad \text{mit} \quad z^{(1)} = 21.$$

Wir definieren nun Optimierungsaufgaben IP^i , $i = 1, 2, 3, 4$, indem wir die in den Gleichungen (A.14) angegebenen Variablen fixieren. Also setzen wir $x_1 := 1 - x_1^{(1)} = 1 - 0 = 1$ und stellen das Optimierungsproblem

$$\begin{aligned} \text{IP}^1 : \quad & \max \quad 8 + 12x_2 + 5x_3 + 9x_4 \\ & \text{unter der Nebenbedingung} \\ & 5x_2 + 4x_3 + 3x_4 \leq 6 \\ & x_j \in \{0, 1\}, \quad j = 2, 3, 4 \end{aligned} \tag{A.15}$$

auf. Für IP^2 legen wir $x_1 := x_1^{(1)} = 0$ und $x_2 := 1 - x_2^{(1)} = 1 - 1 = 0$ fest:

$$\begin{aligned} \text{IP}^2 : \quad & \max \quad 5x_3 + 9x_4 \\ & \text{unter der Nebenbedingung} \\ & 4x_3 + 3x_4 \leq 8 \\ & x_j \in \{0, 1\}, \quad j = 3, 4. \end{aligned} \tag{A.16}$$

Durch Fixierung von $x_1 := x_1^{(1)} = 0$, $x_2 := x_2^{(1)} = 1$ und $x_3 := 1 - x_3^{(1)} = 1 - 0 = 1$

A Lineare Optimierung

erhalten wir

$$\begin{aligned}
 \text{IP}^3 : \quad & \max \quad 12 + 5 + 9x_4 \\
 & \text{unter der Nebenbedingung} \\
 & 3x_4 \leq -1 \\
 & x_4 \in \{0, 1\}.
 \end{aligned} \tag{A.17}$$

Für IP^4 definieren wir zum Schluss $x_1 := x_1^{(1)} = 0$, $x_2 := x_2^{(1)} = 1$, $x_3 := x_3^{(1)} = 0$ und $x_4 := 1 - x_4^{(1)} = 1 - 1 = 0$.

In der Tabelle A.2 sind die mit Hilfe unserer Implementation des Algorithmus von Lawler berechneten Lösungen der vier Rucksackprobleme IP^i , $i = 1, 2, 3, 4$, angegeben. Eine Zahl $p \leq n$ in der Spalte der Anzahl der fixierten Variablen besagt, dass bei der entsprechenden Lösung die ersten p Variablen fixiert worden sind. Da der Zielfunktionswert des Problems IP^1 im Optimum den maximalen Wert 20 besitzt, ist der zugehörige Lösungsvektor $\mathbf{x}^* = (1 \ 1 \ 0 \ 0)$ eine zweitbeste Lösung des ursprünglichen Rucksackproblems.

Tabelle A.2: Optimale Werte von IP^i für die zweitbeste Lösung

IP_B^i	z^*	Anzahl fixierter Variablen	x_1^*	x_2^*	x_3^*	x_4^*
IP^1	20	1	1	1	0	0
IP^2	9	2	0	0	0	1
IP^3	-	3	Keine Lösung			
IP^4	12	4	0	1	0	0

B Minimierung des Maximums linearer Funktionen

Gegeben seien die linearen Funktionen $z_k = \sum_{j=1}^n c_j^k x_j$, $k = 1, 2, \dots, K$. Wir betrachten das folgende Problem (vgl. Dantzig/Thapa [30], S. 152 f.):

$$\begin{aligned} & \min_{\mathbf{z}} \max_{\mathbf{x}} \{z_1, z_2, \dots, z_K\} \\ & \text{unter den Nebenbedingungen} \\ & z_k = \sum_{j=1}^n c_j^k x_j, \quad k = 1, 2, \dots, K \\ & \mathbf{Ax} = \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0}, \end{aligned} \tag{B.1}$$

wobei \mathbf{A} eine $(m \times n)$ -Matrix und \mathbf{b} ein $(m \times 1)$ -Spaltenvektor sind. Das Problem (B.1) ist kein lineares Programm. Um es als ein lineares Programm zu formulieren, definieren wir eine neue Variable z mit $z \geq z_k$, $k = 1, 2, \dots, K$. Demzufolge gilt $z - z_k \geq 0$ bzw. $z - \sum_{j=1}^n c_j^k x_j \geq 0$, $k = 1, 2, \dots, K$. Somit können wir das Problem (B.1) als ein lineares Programm darstellen:

$$\begin{aligned} & \min_{\mathbf{x}, z} z \\ & \text{unter den Nebenbedingungen} \\ & z - \sum_{j=1}^n c_j^k x_j \geq 0, \quad k = 1, 2, \dots, K \\ & \mathbf{Ax} = \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0}. \end{aligned}$$

C Umwandlung eines Sattelpunktproblems in eine lineare Optimierungsaufgabe

In der Literatur nennt man eine Aufgabe der nachstehenden Form (C.1)-(C.5) ein Sattelpunktproblem. Darin sind \mathbf{x} und \mathbf{f} Spaltenvektoren mit m Komponenten, \mathbf{A} , \mathbf{B} und \mathbf{E} Matrizen mit m Zeilen und n Spalten und \mathbf{y} und \mathbf{c} zwei n -dimensionale Spaltenvektoren:

$$\max_{\mathbf{x}} \left\{ \min_{\mathbf{y}} \mathbf{x}^T \mathbf{A} \mathbf{y} \right\} \quad (\text{C.1})$$

unter den Nebenbedingungen

$$\mathbf{x}^T \mathbf{B} \leq \mathbf{c}^T \quad (\text{C.2})$$

$$\mathbf{x} \geq \mathbf{0} \quad (\text{C.3})$$

$$\mathbf{y}^T \mathbf{E}^T \leq \mathbf{f}^T \quad (\text{C.4})$$

$$\mathbf{y} \geq \mathbf{0}, \quad (\text{C.5})$$

Wir beschreiben nun in Anlehnung an Owen [97], S. 59 f., eine Methode, die das obige Sattelpunktproblem mit Hilfe der Dualitätssätze der linearen Optimierung in ein lineares Programm umwandelt. In (C.1) ist der Ausdruck in der geschweiften Klammer eine Minimierungsaufgabe bezüglich der Variablen \mathbf{y} . Folglich sind die Grössen $\mathbf{x}^T \mathbf{A}$ die Koeffizienten der Variablen \mathbf{y} und können somit wie Konstanten behandelt werden. Zudem sind die linearen Optimierungsaufgaben

$$\min_{\mathbf{y}} (\mathbf{x}^T \mathbf{A}) \mathbf{y} \quad (\text{C.6})$$

unter den Nebenbedingungen

$$\mathbf{y}^T \mathbf{E}^T \leq \mathbf{f}^T \quad (\text{C.7})$$

$$\mathbf{y} \geq \mathbf{0} \quad (\text{C.8})$$

C Umwandlung eines Sattelpunktproblems in eine lineare Optimierungsaufgabe

und

$$\max_{\mathbf{z}} \mathbf{z}^T \mathbf{f} \quad (\text{C.9})$$

unter den Nebenbedingungen

$$\mathbf{z}^T \mathbf{E} \leq \mathbf{x}^T \mathbf{A} \quad (\text{C.10})$$

$$\mathbf{z} \geq \mathbf{0} \quad (\text{C.11})$$

zueinander dual. Hierbei beinhaltet der m -dimensionale Spaltenvektor \mathbf{z} die Variablen des Dualproblems von (C.7). Da nun gemäss dem starken Dualitätssatz (vgl. Satz A.1.1, S. 287) die beiden zueinander dualen linearen Programme den selben optimalen Wert besitzen, können wir im ursprünglichen Sattelpunktproblem die Optimierungsaufgabe (C.6)–(C.8) durch (C.9)–(C.11) ersetzen und somit das Sattelpunktproblem als ein lineares Programm wie folgt darstellen:

$$\max_{\mathbf{x}, \mathbf{z}} \mathbf{z} \mathbf{f}^T$$

unter den Nebenbedingungen

$$\mathbf{z} \mathbf{E} - \mathbf{x} \mathbf{A} \leq \mathbf{0}$$

$$\mathbf{x} \mathbf{B} \leq \mathbf{c}$$

$$\mathbf{z}, \mathbf{x} \geq \mathbf{0}.$$

D Lineare Gleichungs- und Ungleichungssysteme

Die Vektoren $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_m$ seien n -dimensionale Zeilenvektoren. Ein n -dimensionaler Zeilenvektor \mathbf{c} ist genau dann eine *Linearkombination* der Vektoren $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_m$, wenn es reellwertige Zahlen $\alpha_1, \alpha_2, \dots, \alpha_m$ gibt, so dass $\mathbf{c} = \alpha_1 \mathbf{a}_1 + \alpha_2 \mathbf{a}_2 + \dots + \alpha_m \mathbf{a}_m$ erfüllt ist. Die Vektoren $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_m$ heissen *linear abhängig*, wenn mindestens ein Vektor von ihnen als Linearkombination der übrigen Vektoren darstellbar ist. Man sagt in diesem Fall auch, dass dieser Vektor von den übrigen Vektoren linear abhängig ist. Die Vektoren $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_m$ heissen *linear unabhängig*, wenn kein Vektor von ihnen als Linearkombination der übrigen Vektoren darstellbar ist. Man kann zeigen (vgl. z. B. Padberg/Kütting [99], S. 128 f.), dass die Vektoren $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_m$ genau dann linear unabhängig sind, wenn aus $\alpha_1 \mathbf{a}_1 + \alpha_2 \mathbf{a}_2 + \dots + \alpha_m \mathbf{a}_m = \mathbf{0}$ stets $\alpha_1 = \alpha_2 = \dots = \alpha_m = 0$ folgt. Für eine $(m \times n)$ -Matrix \mathbf{A} nennt man die Maximalzahl linear unabhängiger Zeilen- oder Spaltenvektoren den Rang $\text{Rg}(\mathbf{A})$ der Matrix \mathbf{A} . Mit Hilfe des Rangs einer Matrix kann überprüft werden, ob ein lineares Gleichungssystem eine Lösung besitzt: Gegeben seien eine $(m \times n)$ -Matrix \mathbf{A} , ein $(m \times 1)$ -Vektor \mathbf{b} und ein $(n \times 1)$ -Vektor \mathbf{x} . Jedes lineare Gleichungssystem $\mathbf{Ax} = \mathbf{b}$ hat genau dann eine Lösung, wenn der Rang der erweiterten $m \times (n + 1)$ -Koeffizientenmatrix $(\mathbf{A}, \mathbf{b}) = [\mathbf{a}_1 \dots \mathbf{a}_n \mathbf{b}]$ und der Rang der ursprünglichen Koeffizientenmatrix \mathbf{A} übereinstimmen (für den Beweis vgl. Bomze/Grossmann [11], S. 15). Das bedeutet, dass jedes Gleichungssystem $\mathbf{Ax} = \mathbf{b}$ genau dann eine Lösung besitzt, falls die rechte Seite des Gleichungssystems, der Spaltenvektor \mathbf{b} , als Linearkombination der Spalten von \mathbf{A} dargestellt werden kann. Demzufolge existiert eine Lösung des Gleichungssystems $\mathbf{Ax} = \mathbf{b}$ genau dann, wenn \mathbf{b} von den Spalten der Matrix \mathbf{A} linear abhängig ist. Die Tabelle D.1 gibt einen Überblick über die möglichen Fälle:

Zudem gilt für jede $(m \times n)$ -Matrix \mathbf{A} und jeden $(m \times 1)$ -Vektor \mathbf{b} , dass entweder das Gleichungssystem $\mathbf{Ax} = \mathbf{b}$ oder das Gleichungssystem $\mathbf{A}^T \mathbf{y} = \mathbf{0}$, $\mathbf{b}^T \mathbf{y} \neq 0$ eine

D Lineare Gleichungs- und Ungleichungssysteme

Tabelle D.1: Lösbarkeit linearer Gleichungssysteme

	Gleichungssystem	
	$\mathbf{Ax} = \mathbf{b}$ (inhomogenes System)	$\mathbf{Ax} = \mathbf{0}$ ($\mathbf{b} = \mathbf{0}$) (homogenes System)
1. $\text{Rg}(\mathbf{A}, \mathbf{b}) \neq \text{Rg}(\mathbf{A})$	System unlösbar	System lösbar
2. $\text{Rg}(\mathbf{A}, \mathbf{b}) = \text{Rg}(\mathbf{A}) = r$	System lösbar	System lösbar
a) $r = n$	Lösung eindeutig	Nur triviale Lösung
b) $r < n$	Lösung nicht-eindeutig	Nicht-triviale Lösungen

Quelle: Vgl. Bronstein/Semendjajew [21], S. 157.

Lösung besitzt. Hierbei sind \mathbf{x} ein $(n \times 1)$ -Vektor und \mathbf{y} ein $(m \times 1)$ -Vektor. Falls wir Systeme von Ungleichungen anstelle von Gleichungen betrachten, gilt der folgende Satz:

Satz D.0.1 (Farkas-Lemma) Für jede $(m \times n)$ -Matrix \mathbf{A} und jeden $(m \times 1)$ -Vektor \mathbf{b} gilt, dass entweder das Gleichungssystem $\mathbf{Ax} = \mathbf{b}$, $\mathbf{x} \geq \mathbf{0}$ oder das Ungleichungssystem $\mathbf{A}^T \mathbf{y} \leq \mathbf{0}$, $\mathbf{b}^T \mathbf{y} \geq 0$ eine Lösung hat.

In der Literatur finden wir verschiedene Varianten des Lemmas von Farkas. Eine davon besagt, dass für die Matrix \mathbf{A} und den Vektor \mathbf{b} entweder $\mathbf{Ax} = \mathbf{b}$, $\mathbf{x} \in \mathbb{R}^n$ oder $\mathbf{A}^T \mathbf{y} = \mathbf{0}$, $\mathbf{b}^T \mathbf{y} > 0$, $\mathbf{y} \in \mathbb{R}^m$ eine Lösung aufweist.¹

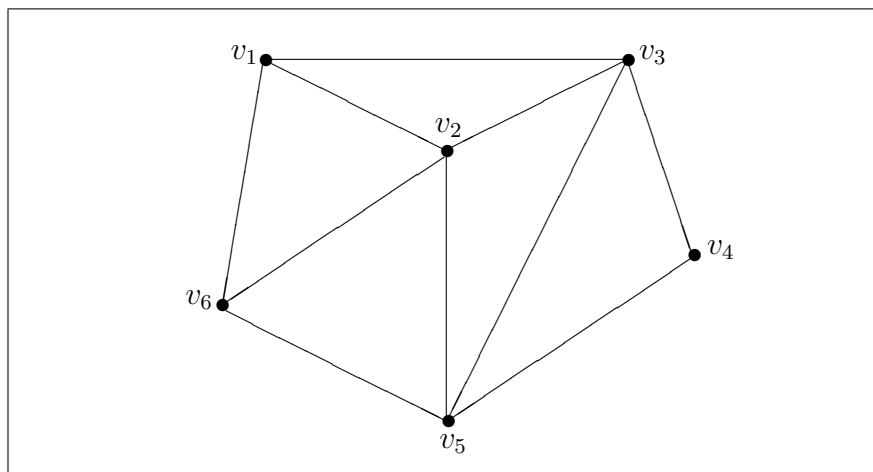
¹Ein Beweis der hier gemachten Aussagen ist z. B. in Jungnickel [55], S. 35 ff., zu finden.

E Einige Begriffe der Graphentheorie

In diesem Anhang definieren wir einige Grundbegriffe der Graphentheorie. Hierbei beziehen wir uns auf Domschke/Drexel [34], Kapitel 3, und Biess [6]. Ein *Graph* $G = (V, E)$ besteht aus einer nicht-leeren Knotenmenge V und einer Menge E , wobei jedem Element von E in eindeutiger Weise ein geordnetes oder ungeordnetes Paar von Knotenpunkten $v, w \in V$ zugeteilt ist (sog. Inzidenzabbildung). Die Elemente von E bezeichnen wir als $e := (v, w)$. Falls jedem $e \in E$ ein ungeordnetes Paar von Knotenpunkten zugeordnet ist, heisst der Graph *ungerichtet*, sonst heisst er *gerichtet*.

Wir gehen im Folgenden von einem ungerichteten Graphen $G = (V, E)$ aus (vgl. Abbildung E.1), da wir in dieser Arbeit ausschliesslich ungerichtete Graphen verwendet haben. Die Elemente von E nennen wir die *Kanten* des Graphen. Zum Beispiel sind $V = \{v_1, v_2, v_3, v_4, v_5, v_6\}$ die Knotenmenge und $E = \{(v_1, v_2), (v_1, v_3), (v_1, v_6), (v_2, v_3), (v_2, v_5), (v_2, v_6), (v_3, v_4), (v_3, v_5), (v_4, v_5), (v_5, v_6)\}$ die Kantenmenge des in der Abbildung E.1 wiedergegebenen Graphen. Die einer Kante $e \in E$ zugeordneten Knoten v und w heissen *Endpunkte* von e . Zwei Knotenpunkte v und w heissen *adjazent*, falls sie Endpunkte derselben Kante e sind. Eine Kante $e \in E$ ist mit dem Knoten v *inzident*, wenn v ein Endpunkt von e ist. Beispielsweise sind in der Abbildung E.1 für $e = (v_1, v_3)$ die Knoten v_1 und v_3 adjazent und die Kante $e = (v_1, v_3)$ ist mit v_1 sowie mit v_3 inzident. Eine *Kantenfolge* ist eine Folge e_1, e_2, \dots, e_n von (nicht notwendigerweise verschiedenen) Kanten, so dass eine Folge von (nicht notwendigerweise verschiedenen) Knoten v_1, v_2, \dots, v_n existiert, wobei $e_i = (v_i, v_{i+1})$, $i = 1, 2, \dots, n - 1$. Gilt $v_1 \neq v_n$, so nennt man die Kantenfolge *offen*, andernfalls *geschlossen*. In der Abbildung E.1 ist z. B. $(v_1, v_2), (v_2, v_3), (v_3, v_4)$ eine offene und $(v_1, v_2), (v_2, v_3), (v_3, v_5), (v_5, v_2), (v_2, v_1)$ eine geschlossene Kantenfolge. Ist eine Kantenfolge offen, dann heissen v_1 und v_n die Endknoten der offenen Kantenfolge. Man sagt, dass diese Kantenfolge die Endknoten v_1 und v_n *verbindet*, oder auch, dass die Knoten v_1 und

Abbildung E.1: Ein ungerichteter Graph



v_n miteinander *verbunden* sind. Eine *Kette* oder ein *Weg* zwischen zwei Knoten v_1 und v_n ist eine Kantenfolge, in der alle Knoten v_1, v_2, \dots, v_n und alle Kanten verschieden sind. Ein *Kreis* ist eine Kette mit $v_1 = v_n$. Ein Kreis ist somit eine Kette, die einen Knoten mit sich selbst verbindet, ohne dass hierfür eine Kante zweimal durchlaufen wird. Beispielsweise stellt in der Abbildung E.1 die Kantenfolge (v_1, v_2) , (v_2, v_6) , (v_6, v_5) eine Kette und (v_2, v_3) , (v_3, v_4) , (v_4, v_5) , (v_5, v_2) einen Kreis dar.

Einen Graphen G nennen wir *zusammenhängend* oder *verbunden*, wenn jedes Knotenpaar von G durch mindestens eine Kette verbunden ist. Der Graph in der Abbildung E.1 ist zusammenhängend, weil jedes Knotenpaar des Graphen z. B. durch die beiden Ketten (v_1, v_2) , (v_2, v_6) , (v_6, v_5) und (v_3, v_4) verbunden ist. Ein zusammenhängender kreisloser ungerichteter Graph heisst *Baum*. Es sei $G = (V, E)$ ein zusammenhängender ungerichteter Graph. Einen Graphen $G' = (V', E')$ mit $V' \subseteq V$ und $E' \subseteq E$ nennen wir *Teilgraphen* von G . Einen zusammenhängenden kreislosen Teilgraphen $G' = (V, E')$ von G nennt man *spannenden Baum* oder *Gerüst* von G . In der Abbildung E.1 ist der Teilgraph G' mit $V' = \{v_1, v_2, v_3, v_4, v_5\}$ und $E' = \{(v_1, v_2), (v_2, v_5), (v_2, v_3), (v_3, v_4)\}$ ein Baum. Der Teilgraph G' mit $V = V' = \{v_1, v_2, v_3, v_4, v_5, v_6\}$ und $E' = \{(v_1, v_2), (v_2, v_6), (v_2, v_5), (v_2, v_3), (v_3, v_4)\}$ ist ein spannender Baum von G .

Ein *Matching* M in G ist eine Teilmenge von E , so dass keine zwei verschiedenen Kanten von M mit ein und demselben Knoten von G inzident sind. Das heisst, dass

keine zwei Kanten von M einen gemeinsamen Endknoten besitzen. Ein Matching M in G heisst *vollständig* oder *perfekt*, wenn M alle Knoten von G überdeckt, d. h. wenn jeder Knoten von G ein Endknoten einer Matching-Kante ist. Zum Beispiel ist $M = \{(v_1, v_6), (v_2, v_5), (v_3, v_4)\} \subset E$ ein vollständiges Matching. Ein Matching M in G heisst *maximal*, falls es in G kein Matching M' mit $|M'| > |M|$ gibt. Ein Graph $G = (V, E)$ heisst *bipartit*, wenn sich seine Knotenmenge V in zwei nicht-leere disjunkte Teilmengen R und T aufteilen lässt, d. h. $V = R \cup T$, $R \cap T = \emptyset$, so dass jede Kante von E jeweils einen Endknoten in R und den anderen in T hat. Der Teilgraph $G' = (V, E')$ mit $E' = \{(v_1, v_3), (v_4, v_3), (v_2, v_3), (v_2, v_5), (v_2, v_6)\}$ des in der Abbildung E.1 dargestellten Graphen ist bipartit, wobei hier $R = \{v_1, v_2, v_4\}$ und $T = \{v_3, v_5, v_6\}$ ist.

Ein ungerichteter Graph $G = (V, E)$, dessen sämtliche Kanten $e = (i, j)$ eine Bewertung $c_e := c(i, j)$ besitzen, wird *kantenbewerteter Graph* genannt. Die Abbildung $c : E \rightarrow \mathbb{R}$ heisst *Bewertungsfunktion* oder *Gewichtsfunktion* von G . Den Funktionswert $c_{(e)} = c(i, j)$ nennt man *Bewertung* oder *Gewicht* der Kante $e = (i, j)$. Ist G ein bewerteter Graph mit der Bewertungsfunktion c , dann schreiben wir $G = (V, E, c)$. Es sei $G = (V, E, c)$ ein bewerteter zusammenhängender ungerichteter Graph. Einen spannenden Baum $T = (V', E')$ von G mit minimaler Summe der Kantenbewertungen bezeichnen wir als *(kosten-)minimalen spannenden Baum* von G .

F Komplexität von Algorithmen

Die Komplexität eines Algorithmus wird an der Anzahl der elementaren Operationen gemessen, die für die Ausführung des Algorithmus erforderlich sind.¹ In einem Algorithmus ist eine elementare Operation, auch Elementarschritt genannt, z. B. eine Addition oder ein Vergleich. In der Literatur bezeichnet man eine elementare Operation auch als eine „Zeiteinheit“ und spricht in diesem Zusammenhang von der *(Zeit-)Komplexität* oder vom *Rechenaufwand* eines Algorithmus.

Um die Komplexität eines Algorithmus zu bestimmen, definiert man eine sog. (Größen-)Ordnung innerhalb der reellwertigen Funktionen. Es seien $f(n)$ und $g(n)$ zwei reellwertige Funktionen. Man sagt, dass die Funktion $f(n)$ von der Ordnung $g(n)$ ist, falls $f(n)$ für ein hinreichend grosses n proportional zur Funktion $g(n)$ ist. Formal kann man dies wie folgt ausdrücken:

$$\exists c \in \mathbb{R}_+ \text{ und } \exists n_0, \text{ sodass } \forall n \geq n_0, f(n) \leq cg(n). \quad (\text{F.1})$$

Ist für $f(n)$ und $g(n)$ die Bedingung (F.1) erfüllt, so bildet $g(n)$ eine obere Schranke kleinster Ordnung für $f(n)$, wobei die Konstante c vernachlässigt wird. Dies wird durch $f(n) \in \mathcal{O}(g(n))$ ausgedrückt. Gilt (F.1) für den Rechenaufwand eines Algorithmus, so sagt man, dass der Rechenaufwand des Algorithmus von der Ordnung $g(n)$ ist, oder auch, dass die Komplexität des Algorithmus $\mathcal{O}(g(n))$ ist. Die Funktion $g(n)$ nennt man die *Komplexitätsfunktion* des Algorithmus. Ist die Funktion $g(n)$ ein Polynom in n , so nennt man den Rechenaufwand *polynomial*, sonst nennt man ihn *exponentiell*. Als ein Beispiel betrachten wir einen Algorithmus, der zur Lösung eines Problems genau $3n^2 + 5n$ elementare Operationen benötigt. Da $3n^2 + 5n \leq 4n^2$ für alle $n \geq n_0 = 5$ ist, ist die Komplexität des Algorithmus $\mathcal{O}(n^2)$.

¹In diesem Abschnitt stützen wir uns auf Domschke/Drexel [34], S. 115 f., Herrmann [52], S. 43–52, und Reischuk [111], S. XV–XVI.

G Aufbau der beigelegten Software

Die im Kapitel 5 dieser Arbeit vorgestellten Verfahren von Potters et al. und von Wolsey wurden von uns in der Programmiersprache Delphi 6 programmiert, die eine Weiterentwicklung der Programmiersprache PASCAL darstellt. Die Berechnung der Nukleoli unserer Testspiele (vgl. Anhang H) erfolgte mit Hilfe unserer Programme. Dieser Arbeit liegt eine CD mit den vollständigen Programm-Codes der implementierten Verfahren bei. Die auf der CD angegebenen Programme wurden unter den Betriebssystemen Microsoft Windows 98 und Microsoft Windows XP erfolgreich getestet.

Der Inhalt der beigelegten CD ist wie folgt aufgebaut. Die CD enthält drei Ordner. Im Ordner **Potters et al.** befinden sich ein Unterordner und eine PDF-Datei. Der Unterordner **Programm-Code Delphi 6** beinhaltet den Quelltext unserer Implementation des Verfahrens von Potters et al. in der Programmiersprache Delphi 6. Der Programm-Code als PDF-Datei befindet sich in der Datei **Programm-Code PDF**. Der für das Verfahren von Wolsey erstellte Ordner **Wolsey** hat die gleiche Struktur wie der Ordner **Potters et al.** Der dritte angelegte Ordner **Ausführbare Dateien** enthält die Ausführungsdateien **PottersReijnierseAnsing.exe** und **Wolsey.exe**. Diesem Ordner gehören auch die Textdateien **Beschreibung (PottersReijnierseAnsing)** und **Beschreibung (Wolsey)** an. Die Textdateien zeigen, wie man für die Berechnung des Nukleolus kooperativer Spiele die Programme **PottersReijnierseAnsing** und **Wolsey** verwenden kann.

H Testspiele zum Vergleich der implementierten Verfahren

In diesem Anhang geben wir die für den Vergleich der implementierten Algorithmen verwendeten Testspiele mit n Spielern, $n = 3, 4, \dots, 25$, und den Nukleolus jedes dieser Spiele an. Für jedes n haben wir insgesamt fünf Spiele herangezogen, wobei wir für jedes dieser Spiele einen Laufindex k , $k \in \{1, 2, 3, 4, 5\}$, verwenden. Die unten verwendete Notation Γ_k^n besagt beispielsweise für $n = 3$ und $k = 1$, dass es sich um das erste Spiel der Testspiele mit $n = 3$ Spielern handelt. Alle Spiele sind zufällig konstruierte gewichtete Abstimmungsspiele.

Gewichtete Abstimmungsspiele mit 3 Spielern:

- $\Gamma_1^3 = [4; 3, 2, 1]$, $\mathcal{N} = (1, 0, 0)$
- $\Gamma_2^3 = [6; 5, 4, 1]$, $\mathcal{N} = (1, 0, 0)$
- $\Gamma_3^3 = [2; 1, 1, 1]$, $\mathcal{N} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$
- $\Gamma_4^3 = [11; 8, 4, 4]$, $\mathcal{N} = (1, 0, 0)$
- $\Gamma_5^3 = [8; 5, 5, 3]$, $\mathcal{N} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$.

Gewichtete Abstimmungsspiele mit 4 Spielern:

- $\Gamma_1^4 = [3; 2, 1, 1, 1]$, $\mathcal{N} = (\frac{2}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5})$
- $\Gamma_2^4 = [5; 3, 2, 2, 1]$, $\mathcal{N} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0)$
- $\Gamma_3^4 = [10; 5, 4, 4, 4]$, $\mathcal{N} = (\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$
- $\Gamma_4^4 = [7; 5, 4, 2, 1]$, $\mathcal{N} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0)$
- $\Gamma_5^4 = [5; 3, 3, 1, 1]$, $\mathcal{N} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{6}, \frac{1}{6})$.

Gewichtete Abstimmungsspiele mit 5 Spielern:

- $\Gamma_1^5 = [7; 5, 4, 3, 2, 1]$, $\mathcal{N} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{6}, \frac{1}{6}, 0)$
- $\Gamma_2^5 = [4; 3, 1, 1, 1, 1]$, $\mathcal{N} = (\frac{3}{7}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7})$
- $\Gamma_3^5 = [6; 4, 3, 2, 2, 1]$, $\mathcal{N} = (\frac{1}{3}, \frac{1}{4}, \frac{1}{6}, \frac{1}{6}, \frac{1}{12})$
- $\Gamma_4^5 = [10; 7, 4, 4, 3, 1]$, $\mathcal{N} = (\frac{2}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}, 0)$
- $\Gamma_5^5 = [24; 16, 10, 8, 4, 2]$, $\mathcal{N} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0, 0)$.

H Testspiele zum Vergleich der implementierten Verfahren

Gewichtete Abstimmungsspiele mit 6 Spielern:

- $\Gamma_1^6 = [5; 2, 2, 2, 1, 1, 1], \quad \mathcal{N} = \frac{1}{9}(2, 2, 2, 1, 1, 1)$
- $\Gamma_2^6 = [13; 9, 7, 3, 3, 2, 1], \quad \mathcal{N} = \frac{1}{13}(4, 3, 2, 2, 1, 1)$
- $\Gamma_3^6 = [5; 4, 3, 2, 1, 1, 1], \quad \mathcal{N} = \frac{1}{12}(4, 3, 2, 1, 1, 1)$
- $\Gamma_4^6 = [24; 15, 10, 7, 5, 5, 4], \quad \mathcal{N} = \frac{1}{9}(3, 2, 1, 1, 1, 1)$
- $\Gamma_5^6 = [18; 10, 10, 8, 6, 4, 2], \quad \mathcal{N} = \frac{1}{4}(1, 1, 1, 1, 0, 0).$

Gewichtete Abstimmungsspiele mit 7 Spielern:

- $\Gamma_1^7 = [6; 5, 4, 3, 2, 1, 1, 1], \quad \mathcal{N} = \frac{1}{14}(4, 3, 2, 2, 1, 1, 1)$
- $\Gamma_2^7 = [5; 4, 3, 2, 1, 1, 1, 1], \quad \mathcal{N} = \frac{1}{13}(4, 3, 2, 1, 1, 1, 1)$
- $\Gamma_3^7 = [35; 20, 18, 14, 11, 8, 6, 4], \quad \mathcal{N} = \frac{1}{13}(3, 3, 2, 2, 1, 1, 1)$
- $\Gamma_4^7 = [22; 12, 10, 10, 7, 7, 5, 3], \quad \mathcal{N} = \frac{1}{17}(4, 3, 3, 2, 2, 2, 1)$
- $\Gamma_5^7 = [15; 6, 6, 5, 5, 4, 3, 3], \quad \mathcal{N} = \frac{1}{21}(4, 4, 3, 3, 3, 2, 2).$

Gewichtete Abstimmungsspiele mit 8 Spielern:

- $\Gamma_1^8 = [22; 12, 10, 5, 3, 2, 2, 1, 1], \quad \mathcal{N} = \frac{1}{3}(1, 1, 1, 0, 0, 0, 0, 0)$
- $\Gamma_2^8 = [9; 6, 5, 5, 4, 4, 3, 3, 2], \quad \mathcal{N} = \frac{1}{21}(4, 3, 3, 3, 3, 2, 2, 1)$
- $\Gamma_3^8 = [124; 61, 57, 51, 50, 9, 6, 3, 2], \quad \mathcal{N} = \frac{1}{7}(2, 2, 1, 1, 1, 0, 0, 0)$
- $\Gamma_4^8 = [7; 6, 5, 4, 3, 2, 1, 1, 1], \quad \mathcal{N} = \frac{1}{23}(6, 5, 4, 3, 2, 1, 1, 1)$
- $\Gamma_5^8 = [42; 18, 16, 14, 14, 11, 10, 5, 2], \quad \mathcal{N} = \frac{1}{6}(1, 1, 1, 1, 1, 1, 0, 0).$

Gewichtete Abstimmungsspiele mit 9 Spielern:

- $\Gamma_1^9 = [50; 30, 25, 20, 18, 16, 12, 9, 7, 5], \quad \mathcal{N} = \frac{1}{28}(6, 5, 4, 4, 3, 2, 2, 1, 1)$
- $\Gamma_2^9 = [45; 18, 15, 13, 13, 10, 8, 6, 6, 4], \quad \mathcal{N} = \frac{1}{35}(7, 6, 5, 5, 4, 3, 2, 2, 1)$
- $\Gamma_3^9 = [32; 14, 11, 9, 9, 7, 6, 5, 4, 2], \quad \mathcal{N} = \frac{1}{23}(5, 4, 3, 3, 2, 2, 2, 1, 1)$
- $\Gamma_4^9 = [25; 12, 12, 10, 10, 8, 8, 6, 6, 6], \quad \mathcal{N} = \frac{1}{63}(9, 9, 8, 8, 7, 7, 5, 5, 5)$
- $\Gamma_5^9 = [18; 9, 7, 5, 5, 4, 4, 3, 2, 1], \quad \mathcal{N} = \frac{1}{40}(9, 7, 5, 5, 4, 4, 3, 2, 1).$

Gewichtete Abstimmungsspiele mit 10 Spielern:

- $\Gamma_1^{10} = [45; 22, 20, 15, 10, 9, 6, 6, 4, 3, 2], \quad \mathcal{N} = \frac{1}{97}(22, 20, 15, 10, 9, 6, 6, 4, 3, 2)$
- $\Gamma_2^{10} = [149; 100, 60, 50, 40, 20, 12, 6, 4, 3, 2], \quad \mathcal{N} = \frac{1}{103}(33, 19, 19, 14, 5, 5, 3, 2, 2, 1)$
- $\Gamma_3^{10} = [60; 35, 31, 28, 20, 16, 14, 10, 7, 7, 7], \quad \mathcal{N} = \frac{1}{51}(10, 9, 8, 6, 5, 4, 3, 2, 2, 2)$
- $\Gamma_4^{10} = [50; 14, 14, 12, 11, 8, 8, 7, 6, 6, 4], \quad \mathcal{N} = \frac{1}{90}(14, 14, 12, 11, 8, 8, 7, 6, 6, 4)$
- $\Gamma_5^{10} = [35; 15, 14, 13, 10, 8, 7, 7, 6, 5, 3], \quad \mathcal{N} = \frac{1}{88}(15, 14, 13, 10, 8, 7, 7, 6, 5, 3).$

Gewichtete Abstimmungsspiele mit 11 Spielern:

$$\begin{aligned}
\bullet \Gamma_1^{11} &= [100; 40, 38, 35, 27, 25, 22, 20, 15, 14, 10, 8], \\
\mathcal{N} &= \frac{1}{152} (24, 23, 21, 16, 15, 13, 12, 9, 8, 6, 5) \\
\bullet \Gamma_2^{11} &= [70; 22, 20, 16, 16, 16, 12, 10, 8, 6, 4, 2], \\
\mathcal{N} &= \frac{1}{66} (11, 10, 8, 8, 8, 6, 5, 4, 3, 2, 1) \\
\bullet \Gamma_3^{11} &= [60; 25, 20, 18, 18, 18, 16, 14, 12, 10, 8, 6], \\
\mathcal{N} &= \frac{1}{82} (12, 10, 9, 9, 9, 8, 7, 6, 5, 4, 3) \\
\bullet \Gamma_4^{11} &= [42; 19, 17, 15, 13, 13, 10, 10, 8, 8, 6, 6], \\
\mathcal{N} &= \frac{1}{125} (19, 17, 15, 13, 13, 10, 10, 8, 8, 6, 6) \\
\bullet \Gamma_5^{11} &= [30; 9, 8, 7, 7, 6, 5, 5, 5, 4, 3, 2], \\
\mathcal{N} &= \frac{1}{61} (9, 8, 7, 7, 6, 5, 5, 5, 4, 3, 2).
\end{aligned}$$

Gewichtete Abstimmungsspiele mit 12 Spielern:

$$\begin{aligned}
\bullet \Gamma_1^{12} &= [80; 17, 14, 11, 9, 9, 9, 7, 7, 7, 5, 4, 2], \\
\mathcal{N} &= \frac{1}{88} (15, 12, 10, 8, 8, 8, 6, 6, 6, 4, 3, 2) \\
\bullet \Gamma_2^{12} &= [70; 20, 16, 15, 11, 10, 9, 8, 7, 6, 5, 4, 3], \\
\mathcal{N} &= \frac{1}{114} (20, 16, 15, 11, 10, 9, 8, 7, 6, 5, 4, 3) \\
\bullet \Gamma_3^{12} &= [62; 30, 20, 15, 14, 13, 12, 10, 8, 6, 5, 4, 4], \\
\mathcal{N} &= \frac{1}{141} (30, 20, 15, 14, 13, 12, 10, 8, 6, 5, 4, 4) \\
\bullet \Gamma_4^{12} &= [43; 21, 11, 9, 9, 7, 7, 7, 4, 4, 4, 2, 2], \\
\mathcal{N} &= \frac{1}{87} (21, 11, 9, 9, 7, 7, 7, 4, 4, 4, 2, 2) \\
\bullet \Gamma_5^{12} &= [30; 18, 12, 8, 8, 6, 5, 4, 3, 2, 1, 1, 1], \\
\mathcal{N} &= \frac{1}{69} (18, 12, 8, 8, 6, 5, 4, 3, 2, 1, 1, 1).
\end{aligned}$$

Gewichtete Abstimmungsspiele mit 13 Spielern:

$$\begin{aligned}
\bullet \Gamma_1^{13} &= [25; 7, 7, 7, 6, 6, 6, 4, 4, 4, 4, 3, 3, 1], \\
\mathcal{N} &= \frac{1}{62} (7, 7, 7, 6, 6, 6, 4, 4, 4, 4, 3, 3, 1) \\
\bullet \Gamma_2^{13} &= [10; 4, 4, 4, 4, 4, 3, 3, 3, 2, 2, 1, 1, 1], \\
\mathcal{N} &= \frac{1}{36} (4, 4, 4, 4, 4, 3, 3, 3, 2, 2, 1, 1, 1) \\
\bullet \Gamma_3^{13} &= [20; 10, 10, 8, 7, 7, 7, 5, 5, 5, 3, 3, 3, 3], \\
\mathcal{N} &= \frac{1}{76} (10, 10, 8, 7, 7, 7, 5, 5, 5, 3, 3, 3, 3) \\
\bullet \Gamma_4^{13} &= [40; 20, 16, 14, 14, 14, 10, 10, 8, 6, 4, 3, 3, 2], \\
\mathcal{N} &= \frac{1}{124} (20, 16, 14, 14, 14, 10, 10, 8, 6, 4, 3, 3, 2) \\
\bullet \Gamma_5^{13} &= [33; 12, 11, 11, 8, 6, 6, 6, 6, 4, 4, 4, 4, 4], \\
\mathcal{N} &= \frac{1}{44} (6, 6, 6, 4, 3, 3, 3, 3, 2, 2, 2, 2, 2).
\end{aligned}$$

H Testspiele zum Vergleich der implementierten Verfahren

Gewichtete Abstimmungsspiele mit 14 Spielern:

$$\begin{aligned}
 \bullet \Gamma_1^{14} &= [124; 61, 57, 51, 50, 9, 6, 3, 2, 2, 1, 1, 1, 1], \\
 \mathcal{N} &= \frac{1}{110} (27, 23, 17, 16, 9, 6, 3, 2, 2, 1, 1, 1, 1) \\
 \bullet \Gamma_2^{14} &= [101; 51, 44, 43, 35, 9, 6, 3, 2, 2, 1, 1, 1, 1], \\
 \mathcal{N} &= \frac{1}{116} (30, 23, 22, 14, 9, 6, 3, 2, 2, 1, 1, 1, 1) \\
 \bullet \Gamma_3^{14} &= [80; 42, 35, 31, 28, 26, 24, 19, 16, 11, 11, 8, 6, 6], \\
 \mathcal{N} &= \frac{1}{169} (26, 22, 19, 17, 16, 15, 12, 10, 7, 7, 5, 5, 4, 4) \\
 \bullet \Gamma_4^{14} &= [50; 20, 16, 14, 14, 10, 9, 9, 7, 6, 6, 5, 5, 5, 5], \\
 \mathcal{N} &= \frac{1}{131} (20, 16, 14, 14, 10, 9, 9, 7, 6, 6, 5, 5, 5, 5) \\
 \bullet \Gamma_5^{14} &= [20; 11, 11, 10, 10, 8, 8, 8, 8, 6, 6, 6, 6, 4, 3], \\
 \mathcal{N} &= \frac{1}{31} (3, 3, 3, 3, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1).
 \end{aligned}$$

Gewichtete Abstimmungsspiele mit 15 Spielern:

$$\begin{aligned}
 \bullet \Gamma_1^{15} &= [25; 14, 10, 9, 9, 9, 9, 7, 7, 7, 5, 5, 4, 3, 3, 2], \\
 \mathcal{N} &= \frac{1}{103} (14, 10, 9, 9, 9, 9, 7, 7, 7, 5, 5, 4, 3, 3, 2) \\
 \bullet \Gamma_2^{15} &= [30; 12, 12, 11, 10, 9, 8, 8, 6, 6, 6, 6, 4, 3, 2, 1], \\
 \mathcal{N} &= \frac{1}{104} (12, 12, 11, 10, 9, 8, 8, 6, 6, 6, 6, 4, 3, 2, 1) \\
 \bullet \Gamma_3^{15} &= [60; 24, 23, 22, 21, 18, 16, 15, 13, 12, 10, 9, 6, 6, 4, 4], \\
 \mathcal{N} &= \frac{1}{203} (24, 23, 22, 21, 18, 16, 15, 13, 12, 10, 9, 6, 6, 4, 4) \\
 \bullet \Gamma_4^{15} &= [18; 10, 9, 8, 8, 8, 7, 7, 6, 6, 6, 5, 4, 4, 3, 3], \\
 \mathcal{N} &= \frac{1}{94} (10, 9, 8, 8, 8, 7, 7, 6, 6, 6, 5, 4, 4, 3, 3) \\
 \bullet \Gamma_5^{15} &= [14; 8, 6, 5, 4, 4, 4, 4, 3, 3, 3, 3, 2, 2, 2, 1], \\
 \mathcal{N} &= \frac{1}{54} (8, 6, 5, 4, 4, 4, 4, 3, 3, 3, 3, 2, 2, 2, 1).
 \end{aligned}$$

Gewichtete Abstimmungsspiele mit 16 Spielern:

$$\begin{aligned}
 \bullet \Gamma_1^{16} &= [155; 55, 50, 47, 42, 35, 21, 15, 12, 12, 12, 9, 6, 5, 4, 4, 2], \\
 \mathcal{N} &= \frac{1}{331} (55, 50, 47, 42, 35, 21, 15, 12, 12, 12, 9, 6, 5, 4, 4, 2) \\
 \bullet \Gamma_2^{16} &= [80; 35, 32, 32, 29, 26, 22, 22, 18, 18, 14, 14, 10, 9, 8, 8, 6], \\
 \mathcal{N} &= \frac{1}{303} (35, 32, 32, 29, 26, 22, 22, 18, 18, 14, 14, 10, 9, 8, 8, 6) \\
 \bullet \Gamma_3^{16} &= [63; 28, 26, 25, 22, 19, 17, 15, 15, 15, 11, 8, 6, 4, 4, 3, 2], \\
 \mathcal{N} &= \frac{1}{220} (28, 26, 25, 22, 19, 17, 15, 15, 15, 11, 8, 6, 4, 4, 3, 2) \\
 \bullet \Gamma_4^{16} &= [24; 8, 8, 6, 6, 6, 6, 5, 5, 4, 4, 4, 3, 3, 3, 2, 2], \\
 \mathcal{N} &= \frac{1}{75} (8, 8, 6, 6, 6, 6, 5, 5, 4, 4, 4, 3, 3, 3, 2, 2) \\
 \bullet \Gamma_5^{16} &= [16; 8, 7, 7, 7, 6, 6, 5, 5, 4, 4, 3, 3, 3, 1, 1, 1], \\
 \mathcal{N} &= \frac{1}{71} (8, 7, 7, 7, 6, 6, 5, 5, 4, 4, 3, 3, 3, 1, 1, 1).
 \end{aligned}$$

Gewichtete Abstimmungsspiele mit 17 Spielern:

$$\begin{aligned}
\bullet \Gamma_1^{17} &= [155; 55, 50, 47, 42, 35, 21, 15, 12, 12, 12, 9, 6, 5, 4, 4, 2, 2], \\
\mathcal{N} &= \frac{1}{333} (55, 50, 47, 42, 35, 21, 15, 12, 12, 12, 9, 6, 5, 4, 4, 2, 2) \\
\bullet \Gamma_2^{17} &= [100; 44, 42, 37, 33, 31, 28, 27, 25, 17, 14, 12, 10, 8, 6, 4, 3, 2], \\
\mathcal{N} &= \frac{1}{343} (44, 42, 37, 33, 31, 28, 27, 25, 17, 14, 12, 10, 8, 6, 4, 3, 2) \\
\bullet \Gamma_3^{17} &= [50; 24, 24, 18, 16, 16, 14, 13, 12, 11, 10, 9, 8, 6, 6, 4, 3, 3], \\
\mathcal{N} &= \frac{1}{197} (24, 24, 18, 16, 16, 14, 13, 12, 11, 10, 9, 8, 6, 6, 4, 3, 3) \\
\bullet \Gamma_4^{17} &= [20; 5, 5, 5, 5, 5, 4, 4, 4, 4, 3, 3, 3, 2, 2, 1, 1, 1], \\
\mathcal{N} &= \frac{1}{57} (5, 5, 5, 5, 5, 4, 4, 4, 4, 3, 3, 3, 2, 2, 1, 1, 1) \\
\bullet \Gamma_5^{17} &= [18; 6, 6, 6, 6, 6, 6, 5, 5, 5, 5, 5, 4, 4, 4, 4, 3, 3], \\
\mathcal{N} &= \frac{1}{83} (6, 6, 6, 6, 6, 6, 5, 5, 5, 5, 5, 4, 4, 4, 4, 3, 3).
\end{aligned}$$

Gewichtete Abstimmungsspiele mit 18 Spielern:

$$\begin{aligned}
\bullet \Gamma_1^{18} &= [155; 55, 50, 47, 42, 35, 21, 15, 12, 12, 12, 9, 6, 5, 4, 4, 2, 1, 1], \\
\mathcal{N} &= \frac{1}{333} (55, 50, 47, 42, 35, 21, 15, 12, 12, 12, 9, 6, 5, 4, 4, 2, 1, 1) \\
\bullet \Gamma_2^{18} &= [32; 7, 7, 7, 7, 7, 7, 6, 6, 6, 6, 6, 5, 5, 5, 5, 5], \\
\mathcal{N} &= \frac{1}{110} (7, 7, 7, 7, 7, 7, 6, 6, 6, 6, 6, 5, 5, 5, 5, 5) \\
\bullet \Gamma_3^{18} &= [25; 16, 13, 13, 12, 10, 9, 9, 8, 7, 6, 5, 4, 4, 4, 3, 3, 3], \\
\mathcal{N} &= \frac{1}{133} (16, 13, 13, 12, 10, 9, 9, 8, 7, 6, 5, 4, 4, 4, 3, 3, 3) \\
\bullet \Gamma_4^{18} &= [102; 33, 33, 27, 25, 20, 20, 20, 15, 15, 10, 9, 7, 7, 6, 4, 3, 1], \\
\mathcal{N} &= \frac{1}{262} (33, 33, 27, 25, 20, 20, 20, 15, 15, 10, 9, 7, 7, 6, 4, 3, 1) \\
\bullet \Gamma_5^{18} &= [88; 42, 40, 35, 35, 27, 25, 21, 21, 18, 16, 14, 10, 9, 9, 9, 6, 6, 6], \\
\mathcal{N} &= \frac{1}{349} (42, 40, 35, 35, 27, 25, 21, 21, 18, 16, 14, 10, 9, 9, 9, 6, 6, 6).
\end{aligned}$$

Gewichtete Abstimmungsspiele mit 19 Spielern:

$$\begin{aligned}
\bullet \Gamma_1^{19} &= [160; 67, 64, 62, 62, 60, 55, 50, 45, 45, 40, 35, 30, 28, 24, 22, 17, 14, 13, 11], \\
\mathcal{N} &= \frac{1}{744} (67, 64, 62, 62, 60, 55, 50, 45, 45, 40, 35, 30, 28, 24, 22, 17, 14, 13, 11) \\
\bullet \Gamma_2^{19} &= [180; 64, 62, 62, 62, 55, 50, 44, 40, 37, 32, 30, 28, 25, 21, 20, 20, 15, 10, 10], \\
\mathcal{N} &= \frac{1}{412} (38, 37, 37, 37, 33, 30, 27, 24, 22, 19, 18, 17, 15, 13, 12, 12, 9, 6, 6) \\
\bullet \Gamma_3^{19} &= [90; 28, 28, 28, 24, 24, 24, 22, 22, 22, 19, 17, 15, 15, 16, 13, 12, 10, 9, 5], \\
\mathcal{N} &= \frac{1}{353} (28, 28, 28, 24, 24, 24, 22, 22, 22, 19, 17, 15, 15, 16, 13, 12, 10, 9, 5) \\
\bullet \Gamma_4^{19} &= [72; 32, 30, 26, 24, 20, 16, 14, 13, 11, 9, 8, 7, 6, 5, 4, 4, 4, 3, 3], \\
\mathcal{N} &= \frac{1}{239} (32, 30, 26, 24, 20, 16, 14, 13, 11, 9, 8, 7, 6, 5, 4, 4, 4, 3, 3) \\
\bullet \Gamma_5^{19} &= [53; 32, 30, 28, 26, 24, 23, 20, 18, 16, 14, 13, 11, 9, 7, 6, 5, 4, 3, 2], \\
\mathcal{N} &= \frac{1}{291} (32, 30, 28, 26, 24, 23, 20, 18, 16, 14, 13, 11, 9, 7, 6, 5, 4, 3, 2).
\end{aligned}$$

H Testspiele zum Vergleich der implementierten Verfahren

Gewichtete Abstimmungsspiele mit 20 Spielern:

- $\Gamma_1^{20} = [93; 41, 38, 31, 28, 25, 25, 21, 18, 16, 16, 13, 10, 8, 8, 6, 5, 4, 2, 1, 1],$
 $\mathcal{N} = \frac{1}{317}(41, 38, 31, 28, 25, 25, 21, 18, 16, 16, 13, 10, 8, 8, 6, 5, 4, 2, 1, 1)$
- $\Gamma_2^{20} = [80; 35, 30, 30, 25, 24, 22, 18, 14, 14, 14, 8, 6, 4, 4, 4, 3, 3, 2, 2],$
 $\mathcal{N} = \frac{1}{265}(35, 30, 30, 25, 24, 22, 18, 14, 14, 14, 8, 6, 4, 4, 4, 3, 3, 2, 2)$
- $\Gamma_3^{20} = [32; 15, 14, 13, 12, 12, 10, 10, 8, 8, 8, 8, 7, 6, 5, 5, 4, 4, 3, 3, 2],$
 $\mathcal{N} = \frac{1}{157}(15, 14, 13, 12, 12, 10, 10, 8, 8, 8, 8, 7, 6, 5, 5, 4, 4, 3, 3, 2)$
- $\Gamma_4^{20} = [60; 30, 28, 26, 25, 23, 21, 17, 15, 14, 12, 10, 7, 7, 7, 5, 5, 5, 1, 1, 1],$
 $\mathcal{N} = \frac{1}{260}(30, 28, 26, 25, 23, 21, 17, 15, 14, 12, 10, 7, 7, 7, 5, 5, 5, 1, 1, 1)$
- $\Gamma_5^{20} = [160; 67, 64, 62, 62, 60, 55, 50, 45, 45, 40, 35, 30, 28, 24, 22, 17, 14, 13, 11, 11],$
 $\mathcal{N} = \frac{1}{755}(67, 64, 62, 62, 60, 55, 50, 45, 45, 40, 35, 30, 28, 24, 22, 17, 14, 13, 11, 11).$

Gewichtete Abstimmungsspiele mit 21 Spielern:

- $\Gamma_1^{21} = [97; 48, 43, 41, 36, 30, 25, 21, 20, 18, 16, 16, 14, 14, 14, 11, 11, 10, 8, 6, 6, 4],$
 $\mathcal{N} = \frac{1}{412}(48, 43, 41, 36, 30, 25, 21, 20, 18, 16, 16, 14, 14, 14, 11, 11, 10, 8, 6, 6, 4)$
- $\Gamma_2^{21} = [48; 16, 14, 13, 12, 11, 11, 11, 11, 8, 8, 8, 7, 7, 6, 6, 5, 5, 5, 4, 3, 2],$
 $\mathcal{N} = \frac{1}{173}(16, 14, 13, 12, 11, 11, 11, 11, 8, 8, 8, 7, 7, 6, 6, 5, 5, 5, 4, 3, 2)$
- $\Gamma_3^{21} = [183; 58, 56, 50, 47, 45, 40, 38, 37, 20, 15, 10, 5, 5, 5, 5, 5, 4, 4, 4, 4, 4],$
 $\mathcal{N} = \frac{1}{461}(58, 56, 50, 47, 45, 40, 38, 37, 20, 15, 10, 5, 5, 5, 5, 5, 4, 4, 4, 4, 4)$
- $\Gamma_4^{21} = [300; 122, 110, 89, 80, 75, 70, 67, 63, 58, 55, 43, 40, 35, 30, 27, 20, 16, 14, 10, 8, 6],$
 $\mathcal{N} = \frac{1}{1038}(122, 110, 89, 80, 75, 70, 67, 63, 58, 55, 43, 40, 35, 30, 27, 20, 16, 14, 10, 8, 6)$
- $\Gamma_5^{21} = [24; 10, 10, 10, 10, 8, 8, 8, 7, 7, 6, 6, 6, 6, 6, 5, 4, 4, 4, 4, 2],$
 $\mathcal{N} = \frac{1}{136}(10, 10, 10, 10, 8, 8, 8, 7, 7, 6, 6, 6, 6, 6, 5, 4, 4, 4, 4, 2).$

Gewichtete Abstimmungsspiele mit 22 Spielern:

- $\Gamma_1^{22} = [70; 40, 29, 27, 25, 20, 19, 16, 14, 14, 12, 12, 10, 9, 9, 8, 8, 7, 6, 6, 4, 3, 3],$
 $\mathcal{N} = \frac{1}{301}(40, 29, 27, 25, 20, 19, 16, 14, 14, 12, 12, 10, 9, 9, 8, 8, 7, 6, 6, 4, 3, 3)$
- $\Gamma_2^{22} = [50; 18, 15, 12, 11, 10, 10, 10, 10, 7, 7, 7, 6, 6, 5, 5, 4, 4, 4, 3, 2, 1, 1],$
 $\mathcal{N} = \frac{1}{158}(18, 15, 12, 11, 10, 10, 10, 10, 7, 7, 7, 6, 6, 5, 5, 4, 4, 4, 3, 2, 1, 1)$
- $\Gamma_3^{22} = [190; 60, 55, 52, 50, 48, 45, 43, 40, 35, 32, 30, 21, 16, 15, 12, 10, 8, 8, 6, 6, 4, 4],$
 $\mathcal{N} = \frac{1}{600}(60, 55, 52, 50, 48, 45, 43, 40, 35, 32, 30, 21, 16, 15, 12, 10, 8, 8, 6, 6, 4, 4)$
- $\Gamma_4^{22} = [250; 101, 85, 62, 58, 50, 43, 35, 28, 24, 20, 20, 20, 15, 11, 10, 9, 7, 5, 3, 1, 1, 1],$
 $\mathcal{N} = \frac{1}{609}(101, 85, 62, 58, 50, 43, 35, 28, 24, 20, 20, 20, 15, 11, 10, 9, 7, 5, 3, 1, 1, 1)$
- $\Gamma_5^{22} = [30; 10, 10, 10, 10, 9, 9, 8, 8, 8, 7, 7, 6, 6, 5, 5, 5, 5, 4, 4, 3, 2, 2],$
 $\mathcal{N} = \frac{1}{143}(10, 10, 10, 10, 9, 9, 8, 8, 8, 7, 7, 6, 6, 5, 5, 5, 5, 4, 4, 3, 2, 2).$

Gewichtete Abstimmungsspiele mit 23 Spielern:

- $\Gamma_1^{23} = [122; 57, 53, 45, 41, 38, 33, 27, 22, 21, 20, 18, 15, 15, 13, 10, 10, 10, 9, 8, 7, 7, 5, 3],$
 $\mathcal{N} = \frac{1}{487} (57, 53, 45, 41, 38, 33, 27, 22, 21, 20, 18, 15, 15, 13, 10, 10, 10, 9, 8, 7, 7, 5, 3)$
- $\Gamma_2^{23} = [70; 40, 29, 27, 25, 20, 19, 16, 14, 14, 12, 12, 10, 9, 9, 8, 8, 7, 6, 6, 4, 3, 3, 2],$
 $\mathcal{N} = \frac{1}{303} (40, 29, 27, 25, 20, 19, 16, 14, 14, 12, 12, 10, 9, 9, 8, 8, 7, 6, 6, 4, 3, 3, 2)$
- $\Gamma_3^{23} = [86; 25, 24, 22, 18, 16, 14, 12, 12, 8, 8, 8, 8, 6, 6, 5, 5, 5, 5, 4, 4, 2, 2, 2],$
 $\mathcal{N} = \frac{1}{221} (25, 24, 22, 18, 16, 14, 12, 12, 8, 8, 8, 8, 6, 6, 5, 5, 5, 5, 4, 4, 2, 2, 2)$
- $\Gamma_4^{23} = [123; 48, 44, 33, 30, 25, 25, 22, 22, 22, 19, 19, 17, 17, 17, 13, 13, 10, 10, 9, 9, 2, 2, 2],$
 $\mathcal{N} = \frac{1}{430} (48, 44, 33, 30, 25, 25, 22, 22, 22, 19, 19, 17, 17, 17, 13, 13, 10, 10, 9, 9, 2, 2, 2)$
- $\Gamma_5^{23} = [95; 43, 41, 40, 35, 35, 30, 27, 25, 25, 25, 20, 20, 18, 16, 14, 10, 10, 8, 6, 6, 4, 4, 4],$
 $\mathcal{N} = \frac{1}{466} (43, 41, 40, 35, 35, 30, 27, 25, 25, 25, 20, 20, 18, 16, 14, 10, 10, 8, 6, 6, 4, 4, 4).$

Gewichtete Abstimmungsspiele mit 24 Spielern:

- $\Gamma_1^{24} = [105; 47, 44, 41, 35, 35, 30, 27, 25, 25, 25, 21, 20, 20, 18, 18, 16, 14, 10, 8, 6, 6, 4, 4, 4],$
 $\mathcal{N} = \frac{1}{503} (47, 44, 41, 35, 35, 30, 27, 25, 25, 25, 21, 20, 20, 18, 18, 16, 14, 10, 8, 6, 6, 4, 4, 4)$
- $\Gamma_2^{24} = [153; 35, 33, 30, 26, 23, 20, 20, 20, 20, 16, 16, 13, 13, 13, 10, 10, 9, 7, 7, 5, 5, 5, 4, 2],$
 $\mathcal{N} = \frac{1}{362} (35, 33, 30, 26, 23, 20, 20, 20, 20, 16, 16, 13, 13, 13, 10, 10, 9, 7, 7, 5, 5, 5, 4, 2)$
- $\Gamma_3^{24} = [67; 20, 16, 13, 10, 10, 10, 10, 8, 8, 8, 8, 6, 6, 6, 6, 4, 4, 4, 4, 3, 3, 2, 2, 1],$
 $\mathcal{N} = \frac{1}{172} (20, 16, 13, 10, 10, 10, 10, 8, 8, 8, 8, 6, 6, 6, 6, 4, 4, 4, 4, 3, 3, 2, 2, 1)$
- $\Gamma_4^{24} = [88; 31, 28, 28, 25, 21, 20, 18, 18, 16, 14, 12, 12, 12, 11, 10, 9, 9, 9, 8, 6, 4, 4, 4, 3],$
 $\mathcal{N} = \frac{1}{332} (31, 28, 28, 25, 21, 20, 18, 18, 16, 14, 12, 12, 12, 11, 10, 9, 9, 9, 8, 6, 4, 4, 4, 3)$
- $\Gamma_5^{24} = [121; 48, 44, 42, 40, 40, 35, 32, 30, 29, 29, 27, 26, 24, 24, 21, 19, 16, 14, 14, 10, 8, 6, 5, 5],$
 $\mathcal{N} = \frac{1}{588} (48, 44, 42, 40, 40, 35, 32, 30, 29, 29, 27, 26, 24, 24, 21, 19, 16, 14, 14, 10, 8, 6, 5, 5).$

Gewichtete Abstimmungsspiele mit 25 Spielern:

- $\Gamma_1^{25} = [70; 40, 29, 27, 25, 20, 19, 16, 14, 14, 12, 12, 10, 9, 9, 8, 8, 7, 6, 6, 4, 3, 3, 2, 2, 2],$
 $\mathcal{N} = \frac{1}{307} (40, 29, 27, 25, 20, 19, 16, 14, 14, 12, 12, 10, 9, 9, 8, 8, 7, 6, 6, 4, 3, 3, 2, 2, 2)$
- $\Gamma_2^{25} = [250; 130, 120, 100, 85, 60, 58, 52, 40, 35, 24, 20, 16, 14, 10, 8, 8, 8, 6, 6, 6, 4, 4, 3, 2, 2],$
 $\mathcal{N} = \frac{1}{821} (130, 120, 100, 85, 60, 58, 52, 40, 35, 24, 20, 16, 14, 10, 8, 8, 8, 6, 6, 6, 4, 4, 3, 2, 2)$
- $\Gamma_3^{25} = [135; 50, 50, 45, 40, 35, 32, 28, 24, 24, 20, 16, 14, 13, 11, 11, 10, 9, 9, 6, 6, 4, 3, 2, 1, 1],$
 $\mathcal{N} = \frac{1}{464} (50, 50, 45, 40, 35, 32, 28, 24, 24, 20, 16, 14, 13, 11, 11, 10, 9, 9, 6, 6, 4, 3, 2, 1, 1)$
- $\Gamma_4^{25} = [53; 21, 20, 15, 15, 15, 15, 13, 11, 10, 9, 9, 9, 9, 7, 7, 7, 6, 6, 4, 4, 4, 4, 2, 2, 2],$
 $\mathcal{N} = \frac{1}{226} (21, 20, 15, 15, 15, 15, 13, 11, 10, 9, 9, 9, 9, 7, 7, 7, 6, 6, 4, 4, 4, 4, 2, 2, 2)$
- $\Gamma_5^{25} = [84; 21, 20, 19, 16, 15, 15, 13, 13, 13, 10, 10, 10, 7, 7, 7, 6, 6, 6, 4, 2, 2, 2, 2, 2],$
 $\mathcal{N} = \frac{1}{230} (21, 20, 19, 16, 15, 15, 13, 13, 13, 10, 10, 10, 7, 7, 7, 6, 6, 6, 4, 2, 2, 2, 2, 2).$

Literaturverzeichnis

- [1] Altermatt, U. Das lange Warten auf einen Sitz im Bundesrat. Historische Reminiszenzen zur Zauberformel. *Neue Zürcher Zeitung*, 9. November 1999.
- [2] Arin, J. and V. Feltkamp. The Nucleolus and Kernel of Veto-Rich Transferable Utility Games. *International Journal of Game Theory*, 26:61–73, 1997.
- [3] Aumann, R. J. and J. Dreze. Cooperative Games with Coalition Structures. *International Journal of Game Theory*, 4:217–237, 1975.
- [4] Bamberg, G. and F. Baur. *Statistik*. R. Oldenbourg Verlag, München, Wien, 12. Aufl., 2002.
- [5] Beisel, E.-P. and M. Mendel. *Optimierungsmethoden des Operations Research*. Vol. 1. Friedr. Vieweg & Sohn, Braunschweig, Wiesbaden, 1987.
- [6] Biess, G. *Graphentheorie*. Teubner Verlagsgesellschaft, Leipzig, 3. Aufl., 1988.
- [7] Bilbao, J. M. *Cooperative Games on Combinatorial Structures*, volume 26 of *Series C: Game Theory, Mathematical Programming and Operations Research*. Kluwer Academic Publishers, Boston, Dordrecht, London, 2000.
- [8] Bindseil, U. and C. Hantke. The Power Distribution in Decision Making Among EU Member States. *European Journal of Political Economy*, 13:171–185, 1997.
- [9] Bird, C. On Cost Allocation for a Spanning Tree: A Game Theoretic Approach. *Networks*, 6:335–350, 1976.
- [10] Bloech, J. *Lineare Optimierung für Wirtschaftswissenschaftler*. Westdeutscher Verlag, Opladen, 1974.

Literaturverzeichnis

- [11] Bomze, I. M. and W. Grossmann. *Optimierung - Theorie und Algorithmen: Eine Einführung in Operations Research für Wirtschaftsinformatiker*. Wissenschaftsverlag, Mannheim, Leipzig, Wien, 1993.
- [12] Bondavera, O. N. Teoriia iadra v igre n lits. *Vestnik Leningrad University, Mathematics, Mechanics, Astronomy*, 13:141–142, 1962.
- [13] Borgwardt, K. H. *The Simplex Method: A Probabilistic Analysis*. Springer Verlag, Berlin, New York, 1987.
- [14] Borm, P., H. Hamers, and R. Hendrickx. Operations Research Games. CentER Discussion Paper 2001-45, Tilburg University, Netherlands, 2001.
- [15] Bortz, J. *Lehrbuch für Statistik: Für Sozialwissenschaftler*. Springer Verlag, Berlin, Heidelberg, New York, 2. Aufl., 1985.
- [16] Brückner, M. Voting Power in the European Central Bank. In Holler, J. and G. Owen, editors, *Power Indices and Coalition Formation*, chapter 4. Kluwer Academic Publishers, Boston, 2001.
- [17] Breitenstein, M. Normaldruck nach den letzten Positionsbezügen. *Neue Zürcher Zeitung*, 15. Dezember 1999.
- [18] Brânzei, R., V. Fragnelli, and S. Tijs. Tree-Connected Peer Group Situations and Peer Group Games. *Mathematical Methods of Operations Research*, 55:93–106, 2002.
- [19] Brânzei, R., E. Iñarra, S. Tijs, and J. M. Zarzuelo. An Algorithm for the Nucleolus of Airport Profit Problems. CentER Discussion Paper 2003-50, Tilburg University, Netherlands, 2003.
- [20] Brânzei, R., T. Solymosi, and S. Tijs. Strongly Essential Coalitions and the Nucleolus of Peer Group Games. CentER Discussion Paper 2003-19, Tilburg University, Netherlands, 2003.

- [21] Bronstein, I. N. and K. A. Semendjajew. *Taschenbuch der Mathematik*. B. G. Teubner Verlagsgesellschaft, Stuttgart, Leipzig, 25. Aufl., 1991.
- [22] Bruyneel, G. G. Computation of the Nucleolus of a Game by Means of Minimal Balanced Sets. *Operations Research-Verfahren (Später: Methods of Operations Research, MOR)*, 34:35–51, 1979.
- [23] Bösel, M. *Statistik*. Oldenbourg, München, Wien, 1992.
- [24] Burkard, R. H. Ganzzahlige Optimierung. In Gal, T., editor, *Grundlagen des Operations Research 2: Graphen und Netzwerke, Netzplantechnik, Transportprobleme, ganzzahlige Optimierung*, chapter 9, pages 361–444. Springer Verlag, Berlin, Heidelberg, New York, 1992.
- [25] Charnes, A. and K. O. Kortanek. On Balanced Sets, Cores, and Linear Programming. *Cahiers du Centre d'Etudes de Recherche Opérationnelle*, 9:32–43, 1967.
- [26] Claus, A. and D. Kleitman. Cost Allocation for a Spanning Tree. *Networks*, 3:289–304, 1973.
- [27] Colomer, J. M. and M. O. Hosli. Decision-Making in the European Union: The Power of Political Parties. *Aussenwirtschaft*, 52:255–280, 1997.
- [28] Curiel, E. *Cooperative Game Theory and Applications – Cooperative Games Arising from Combinatorial Optimization Problems*, volume 16 of *Series C: Game Theory, Mathematical Programming and Operations Research*. Kluwer Academic Publishers, Boston, Dordrecht, London, 1997.
- [29] Dantzig, G. B. Maximization of a Linear Function of Variables Subject to Linear Inequalities. In Koopman, T. C., editor, *Activity Analysis of Production and Allocation*, pages 339–347. John Wiley and Sons, New York, 1951.

Literaturverzeichnis

- [30] Dantzig, G. B. and M. N. Thapa. *Linear Programming I: Introduction*. Springer Series in Operations Research. Springer Verlag, New York, Berlin, Heidelberg, 1997.
- [31] Derks, J. and J. Kuipers. On the Core of Routing Games. *International Journal of Game Theory*, 26:193–206, 1997.
- [32] Derks, J. and J. Kuipers. Implementing the Simplex Method for Computing the Prenucleolus of Transferable Utility Games. Research Paper, Department of Mathematics, University of Maastricht, April 1997. <http://www.math.unimaas.nl/personal/jeand/home1.htm>, (Februar 2004).
- [33] Dijkstra, E. W. A Note on Two Problems in Connection with Graphs. *Numerische Mathematik*, 1:269–271, 1959.
- [34] Domschke, W. and A. Drexl. *Einführung in Operations Research*. Springer Verlag, Berlin, Heidelberg, New York, 5. Aufl., 2002.
- [35] Dragan, I. A Procedure for Finding the Nucleolus of a Cooperative n –Person Game. *Zeitschrift für Operations Research*, 25:119–131, 1981.
- [36] Dubey, P. The Shapley Value as Aircraft Landing Fees-Revisited. *Management Science*, 28(8):869–874, 1982.
- [37] Engeln-Müllges, G. and F. Reutter. *Numerik-Algorithmen*. VDI-Verlag, Düsseldorf, 8. Aufl., 1996.
- [38] Fargó, F., J. Szép, and F. Szidarovszky. *Introduction to the Theory of Games: Concepts, Methods, Applications*, volume 32 of *Nonconvex Optimizations and its Applications*. Kluwer Academic Publishers, Boston, Dordrecht, London, 1999.
- [39] Faulhaber, G. R. Cross-Subsidization: Pricing in Public Enterprises. *American Economic Review*, 65(5):966–977, 1975.

- [40] Frank, M. *Entwicklung und Anwendung einer integrierten Methode zur Analyse von betriebsübergreifenden Energieversorgungskonzepten*. PhD thesis, Universität Fridericiana zu Karlsruhe, 2003.
- [41] Frey, B. S. Eine spieltheoretische Analyse der Machtverteilung im schweizerischen Bundesrat. *Schweizerische Zeitschrift für Volkswirtschaft und Statistik*, 104:155–169, 1968.
- [42] Friedman, J. W. *Game Theory with Application to Economics*. Oxford University Press, Oxford, New York, 1986.
- [43] Galil, Z. Application of Efficient Mergeable Heaps for Optimization Problem on Trees. *Acta Informatica*, 13:53–58, 1980.
- [44] Gass, S. I. *Linear Programming*. McGraw-Hill, New York, London, 3. Aufl., 1969.
- [45] Gillies, D. B. Solutions to General Non-Zero-Sum Games. *Annals of Mathematical Studies*, 40:47–87, 1959.
- [46] Goldfarb, D. and M. J. Todd. Linear programming. In Nemhauser, G. L., A. H. G. Rinnooy Kan, and M. J. Todd, editors, *Optimization*, volume 1 of *Handbooks in Operations Research and Management Science*, chapter II, pages 73–170. Elsevier Science Publishers B. V., North-Holland, Amsterdam, New York, 1991.
- [47] Granot, D. and G. Huberman. Minimum Cost Spanning Tree Games. *Mathematical Programming*, 21:1–18, 1981.
- [48] Granot, D., M. Maschler, G. Owen, and W. R. Zhu. The Kernel/Nucleolus of a Standard Tree Game. *International Journal of Game Theory*, 25:219–244, 1996.
- [49] Güth, W. *Spieltheorie und ökonomische (Bei)Spiele*. Springer Verlag, Berlin, Heidelberg, New York, 2. Aufl., 1999.

Literaturverzeichnis

- [50] Hamers, H., F. Klijn, T. Solymosi, S. Tijs, and D. Vermeulen. On the Nucleolus of Neighbor Games. *European Journal of Operational Research*, 146:1–18, 2003.
- [51] Hart, S. and A. Mas-Colell. Potential, Value, and Consistency. *Econometrica*, 57(3):589–614, 1989.
- [52] Herrmann, D. *Algorithmen Arbeitsbuch*. Addison-Wesley, Bonn, München, Paris, 1995.
- [53] Holler, M. J. and G. Illing. *Einführung in die Spieltheorie*. Springer Verlag, Berlin, Heidelberg, New York, 4. Aufl., 2000.
- [54] Huberman, G. The Nucleolus and the Essential Coalitions. In Benhoussan, A. and J. L. Lions, editors, *Analysis and Optimization of Systems*, Proceedings of the Fourth International Conference on Analysis and Optimization of Systems, pages 416–422. Springer Verlag, Berlin, Dezember 1980.
- [55] Jegher, A. *Der Einfluss von institutionellen, entscheidungspolitischen Faktoren auf die Gesetzgebungstätigkeit der Schweizerischen Bundesversammlung*. Inauguraldissertation, Universität Bern. Verlag Paul Haupt, Bern, Stuttgart, Wien, 1999.
- [56] Jianhua, W. *The Theory of Games*. Oxford Mathematical Monographs. Tsinghua University Press, Clarendon Press, Beijing, Oxford, 1988.
- [57] Karmarkar, N. A New Polynomial-Time Algorithm for Linear Programming. *Combinatorica*, 4:373–395, 1984.
- [58] Kaufmann, M. and W. H. Tack. Koalitionsbildung und Gewinnaufteilung bei strategisch äquivalenten 3–Personen-Spielen. *Zeitschrift für Sozialpsychologie*, 6:227–245, 1975.
- [59] Kern, W. and D. Paulusma. Matching Games: The Least Core and the Nucleolus. Memorandum 1541, Faculty of Mathematical Sciences, University of Twen-

te, www.math.utwente.nl/publicaties/2000/1541.pdf (04.03.2004), September 2000.

- [60] Khachian, L. G. A Polynomial Algorithm for Linear Programming. *Doklady Akademii Nauk USSR*, 244:1093–1096, 1979. Engl.: Soviet Mathematics Doklady 20, 191–194.
- [61] Klee, V. and G. J. Minty. How Good is the Simplex Algorithm. In O. Shisha, editor, *Inequalities III*, pages 159–175. Academic Press, 1972.
- [62] Klijn, F., D. Vermeulen, H. Hamers, T. Solymosi, S. Tijs, and J. P. Villar. Neighbour Games and the Leximax Solution. CentER Discussion Paper 99110, Tilburg University, Netherlands, 1999.
- [63] Knieps, G. Kostenaufteilung bei Mehrzweckprojekten. *Schweizerische Zeitschrift für Volkswirtschaft und Statistik*, 2:151–174, 1988.
- [64] Kohlberg, E. On the Nucleolus of a Characteristic Function Game. *SIAM Journal on Applied Mathematics*, 20(1):62–66, 1971.
- [65] Kohlberg, E. The Nucleolus as a Solution of a Minimization Problem. *SIAM Journal on Applied Mathematics*, 23(1):34–39, 1972.
- [66] Kopelowitz, A. Computation of the Kernel of Simple Games and the Nucleolus of N –Person Games. Research Program in Game Theory and Mathematical Economics, Dept. of Math., The Hebrew University of Jerusalem, RM 31, 1967.
- [67] Krelle, W. and H. P. Künzi. *Lineare Programmierung*. Verlag Industrielle Organisation, Zürich, 1958.
- [68] Krus, L. and P. Bronisz. Cooperative Game Solution Concepts to a Cost Allocation Problem. *European Journal of Operational Research*, 122:258–271, 2000.
- [69] Kruskal, J. B. On the Shortest Spanning Subtree of a Graph and the Travelling Salesman Problem. *Proc. Amer. Math. Soc.*, 7:48–50, 1956.

Literaturverzeichnis

- [70] Kuipers, J., T. Solymosi, and H. Aarts. Computing the Nucleolus of Some Combinatorially Structured Games. *Mathematical Programming, Ser. A*, 88:541–563, 2000.
- [71] Lachmann, W. *Volkswirtschaftslehre 1*. Springer Verlag, Berlin, Heidelberg, New York, 3. Aufl., 1997.
- [72] Lawler, E. L. A Procedure for Computing the K Best Solutions to Discrete Optimization Problems and its Application to the Shortest Path Problem. *Management Science*, 18(7):401–405, 1972.
- [73] Lawler, E. L. *Combinatorial Optimization: Networks and Matroids*. Dover Publications, Mineola, New York, 2001.
- [74] Legros, P. Computation of the Nucleolus of Some Bilateral Market Games. *International Journal of Game Theory*, 16:1–14, 1987.
- [75] Lijphart, A. *Democracies. Patterns of Majoritarian and Consensus Government in Twenty-One Countries*. Yale University Press, New Haven, 1984.
- [76] Linder, W. *Schweizerische Demokratie: Institutionen – Prozesse – Perspektiven*. Haupt, Bern, Stuttgart, Wien, 1999.
- [77] Littlechild, S. C. A Simple Expression for the Nucleolus in a Special Case. *International Journal of Game Theory*, 3(1):21–29, 1974.
- [78] Littlechild, S. C. and G. H. Thompson. Aircraft Landing Fees: A Game Theory Approach. *The Bell Journal of Economics*, 8(1):186–204, 1977.
- [79] Littlechild, S. C. and G. Owen. A Simple Expression for the Shapley Value in a Special Case. *Management Science*, 20:370–372, 1973.
- [80] Luce, R. D. and H. Raiffa. *Games and Decisions: Introduction and Critical Survey*. John Wiley & Sons, Inc., New York, London, Sydney, 7. Aufl., 1967.

- [81] Maschler, M. The Bargaining Set, Kernel, and Nucleolus. In Aumann, R. J. and S. Hart, editors, *Handbook of Game Theory*, volume 1, pages 591–667. Elsevier Science Publishers, Amsterdam, 1992.
- [82] Maschler, M., B. Peleg, and L. S. Shapley. Geometric Properties of the Kernel, Nucleolus, and Related Solution Concepts. *Mathematics of Operations Research*, 4(4):303–338, 1979.
- [83] Maschler, M., J. Potters, and S. Tijs. The General Nucleolus and the Reduced-Game property. In Binmore, K., A. Kirman, and P. Tani, editors, *Frontiers of Game Theory*, pages 117–132. The Mit Press, Cambridge, London, 1993.
- [84] Megiddo, N. On the Nonmonotonicity of the Bargaining Set, the Kernel, and the Nucleolus of a Game. *SIAM Journal on Applied Mathematics*, 27(2):355–358, 1974.
- [85] Megiddo, N. Computational Complexity of the Game Theory Approach to Cost Allocation for a Tree. *Mathematics of Operations Research*, 3(3):189–196, 1978.
- [86] Moulin, H. *Axioms of Cooperative Decisions Making*. Economic Society Monographs, Nr. 15. Cambridge University Press, Cambridge, New York, Port Chester, 1988.
- [87] Moulin, H. *Cooperative Microeconomics: A Game-Theoretic Introduction*. Prentice Hall/Harvester Wheatsheaf, London, New York, 1995.
- [88] Nash, J. F. Non-Cooperative Games. *Annals of Mathematics*, 54:286–295, 1951.
- [89] Neidhard, L. Schweizerische Logik der Machtverteilung. Nach der Ersatzwahl in den Bundesrat – ein Blick auf die Regeln. *Neue Zürcher Zeitung*, 13. Dezember 2000.

Literaturverzeichnis

- [90] Neumann, K. and M. Morlock. *Operations Research*. Carl Hanser Verlag, München, Wien, 1993.
- [91] Niemeyer, G. *Einführung in die lineare Planungsrechnung mit ALGOL- und FORTRAN-Programmen*. Walter de Gruyter & Co, Berlin, 1968.
- [92] O. V. Absolut unbestrittene Richterwahlen. *Neue Zürcher Zeitung*, page 18, Nr. 294, 18. Dezember 2003.
- [93] O. V. Sonderbeilage Wahlen 2003. *Neue Zürcher Zeitung*, Nr. 214, 16. September 2003.
- [94] Orshan, G. The Prenucleolus and the Reduced Game Property: Equal Treatment Replaces Anonymity. *International Journal of Game Theory*, 22:241–248, 1993.
- [95] Osborne, M. J. and A. Rubinstein. *A Course in Game Theory*. The MIT Press, Cambridge, Massachusetts, 1994.
- [96] Owen, G. A Note on the Nucleolus. *International Journal of Game Theory*, 3:101–103, 1974.
- [97] Owen, G. *Game Theory*. Academic Press, San Diego, New York, Boston, 3. Aufl., 1995.
- [98] Owen, G. *Discrete Mathematics and Game Theory*, volume 22 of *Series C: Game Theory, Mathematical Programming and Operations Research*. Kluwer Academic Publishers, Boston, Dordrecht, London, 1999.
- [99] Padberg, F. and H. Kütting. *Lineare Algebra: Eine elementare Einführung*. BI-Hochschultaschenbücher. Vol. 649. Wissenschaftsverlag, Mannheim, Wien, Zürich, 1991.
- [100] Parsons, T. W. *Introduction to Algorithms in Pascal*. John Wiley & Sons, Inc., New York, Chichester, Toronto, 1994.

- [101] Peleg, B. An Inductive Method for Constructing Minimal Balanced Collections of Finite Sets. *Naval Research Logistics Quarterly*, 12:155–162, 1965.
- [102] Peleg, B. and P. Sudhölter. *Introduction to the Theory of Cooperative Games*, volume 34 of *Series C: Game Theory, Mathematical Programming and Operations Research*. Kluwer Academic Publishers, Boston, Dordrecht, London, 2003.
- [103] Potters, J. and H. Reijnierse. The \mathcal{B} –Nucleolus of TU-Games. *Games and Economic Behaviour*, 24:77–96, 1998.
- [104] Potters, J. A. M. An Axiomatization of the Nucleolus. *International Journal of Game Theory*, 19:365–373, 1991.
- [105] Potters, J. A. M., J. H. Reijnierse, and M. Ansing. Computing the Nucleolus by Solving a Prolonged Simplex Algorithm. *Mathematics of Operations Research*, 21(3):757–768, 1996.
- [106] P. Potters, J. A. M. and Sudhölter. Airport Problems and Consistent Allocation Rules. *Mathematical Social Sciences*, 38:83–102, 1999.
- [107] Prim, R. C. Shortest Connection Networks and Some Generalizations. *Bell Syst. Techn.*, 36:1389–1401, 1957.
- [108] Raghavan, T. E. S. Some Recent Algorithms For Finding the Nucleolus of Structured Cooperative Games. In T. Parthasarathy et al., editor, *Game Theoretical Applications to Economics and Operations Research*, volume 18 of *Series C: Game Theory, Mathematical Programming and Operations Research*, chapter V, pages 207–238. Kluwer Academic Publishers, Boston, Dordrecht, London, 1997.
- [109] Ransmeier, J. S. *The Tennessee Valley Authority: A Case Study in the Economics of Multiple Purpose Stream Planning*. Vanderbilt University Press, Nashville, 1942.

Literaturverzeichnis

- [110] Rawls, J. *A Theory of Justice*. Oxford University Press, New York, 1971.
- [111] Reischuk, K. R. *Komplexitätstheorie: Grundlagen*. Vol. 1. G. Teubner, Stuttgart, 2. Aufl., 1999.
- [112] Rosenmüller, J. Spieltheorie. In *Handwörterbuch der Wirtschaftswissenschaften (HdWW)*, volume 7, pages 177–196. G. Fischer, J. C. B. Mohr, Vandenhoeck, Tübingen, New York, Stuttgart, 1988.
- [113] Sankaran, J. K. On Finding the Nucleolus of an N-Person Cooperative Game. *International Journal of Game Theory*, 19:329–338, 1991.
- [114] Saxer, M. Kampfansage an die Konkordanz (Kommentar). *Neue Zürcher Zeitung*, 27./28. November 1999.
- [115] Schaback, R. and H. Werner. *Numerische Mathematik*. Springer Verlag, Berlin, Heidelberg, New York, 4. Aufl., 1993.
- [116] Schmeidler, D. The Nucleolus of a Characteristic Function Game. *Journal of Applied Mathematics*, 17:1163–1170, 1969.
- [117] Schrijver, A. *Theory of Linear and Integer Programming*. Wiley - Interscience Series in Discrete Mathematics. John Wiley & Sons, Chichester, New York, Brisbane, 1986.
- [118] Schwarz, H. R. *Numerische Mathematik*. B. G. Teubner, Stuttgart, 4. Aufl., 1997.
- [119] Seitz, W., M. Schneider, and R. Frey. Nationalratswahlen 2003. Statistik der Schweiz, Bundesamt für Statistik der Schweiz, Neuchâtel, 2003.
- [120] Selten, R. Die konzeptionellen Grundlagen der Spieltheorie einst und jetzt. Bonn Econ Discussion Paper 2/2001, Universität Bonn, Januar 2001.
- [121] Shamir, R. The Efficiency of the Simplex Method: A Survey. *Management Science*, 33:301–334, 1987.

- [122] Shapley, L. and M. Shubik. Quasi-Cores in a Monetary Economy with Non-convex Preferences. *Econometrica*, 34:805–827, 1966.
- [123] Shapley, L. S. On Balanced Sets and Cores. *Naval Research Logistics Quarterly*, 14:453–460, 1967.
- [124] Shapley, L. S. and M. Shubik. A Method for Evaluating the Distribution of Power in a Committee System. *American Political Science Review*, 48:787–792, 1954.
- [125] Shapley, L. S. and M. Shubik. The Assignment Game I: The Core. *International Journal of Game Theory*, 1:111–130, 1972.
- [126] Shubik, M. *Game Theory in the Social Sciences: Concepts and Solutions*. The MIT Press, Cambridge, Mass., 2. Aufl., 1983.
- [127] Shubik, M. The Cooperative Form, the Value, and the Allocation of Joint Costs and Benefits. In Young, H. P., editor, *Cost Allocation: Methods, Principles, Applications*, pages 79–94. North-Holland, Amsterdam, New York, Oxford, 1985.
- [128] Snijders, C. Axiomatization of the Nucleolus. *Mathematics of Operations Research*, 20(1):189–196, 1995.
- [129] Sobolev, A. I. Characterization of Optimality Principles in Cooperative Games by Functional Equations. *Mathematical Methods in the Social Sciences*, 6:94–151, 1975.
- [130] Solymosi, T. *On Computing the Nucleolus of Cooperative Games*. PhD thesis, University of Illinois at Chicago, 1993.
- [131] Solymosi, T., H. Aarts, and T. Driessen. On Computing the Nucleolus of a Balanced Connected Game. *Mathematics of Operations Research*, 23(4):983–1009, 1998.

Literaturverzeichnis

- [132] Solymosi, T. and T. E. S. Raghavan. An Algorithm for Finding the Nucleolus of Assignment Games. *International Journal of Game Theory*, 23:119–143, 1994.
- [133] Solymosi, T., T. E. S. Raghavan, and S. Tijs. Computing the Nucleolus of Cyclic Permutation Games. *European Journal of Operational Research*, 162:270–280, 2005.
- [134] Straffin, P. D and J. P. Heaney. Game Theory and the Tennessee Valley Authority. *International Journal of Game Theory*, 10(1):35–43, 1981.
- [135] Strebel, D. Rechtsrutsch auch an den Gerichten. *Neue Zürcher Zeitung am Sonntag*, page 9, 14. Dezember 2003.
- [136] Sudhölter, P. The Modified Nucleolus: Properties and Axiomatizations. *International Journal of Game Theory*, 26:147–182, 1997.
- [137] Sutter, M. Voting and Voting Power in the Stability Pact. *Homo Oeconomicus*, 15:521–542, 1999.
- [138] Suzuki, M. and M. Nakayama. The Cost Assignment of the Cooperative Water Ressource Development: A Game Theoretical Approach. *Management Science*, 22:1081–1086, 1976.
- [139] Syslo, M. M., N. Deo, and J. S. Kowalik. *Discrete Optimization Algorithms With Pascal Programs*. Prentice-Hall, New Jersey, 1983.
- [140] Tijs, S. H., T. Parthasarathy, J. A. M. Potters, and V. Rajendra Prasad. Permutations Games: Another Class of Totaly Balanced Games. *OR Spektrum*, 6:119–123, 1984.
- [141] Vanderbei, R. J. *Linear Programming: Foundations and Extensions*. Kluwer Academic Publishers, Boston, London, Dordrecht, 2. Aufl., 1997.
- [142] Varian, H. R. *Grundzüge der Mikroökonomik*. Oldenbourg, München, Wien, 5. Aufl., 2001.

- [143] von Neumann, J. and O. Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, Princeton, 1944.
- [144] Wallmeier, E. *Der f -Nukleolus und ein dynamisches Verhandlungsmodell als Lösungskonzepte für kooperative n -Personenspiele*. Dissertationsnachdruck, Westfälische Wilhelm-Universität Münster, Institut für Mathematische Statistik, Skripte zur Mathematischen Statistik, Nr. 5, 1983.
- [145] Walukiewicz, S. *Integer Programming*, volume 46 of *Mathematics and its Applications*. Kluwer Academic Publishers, Dordrecht, Boston, London, 1990.
- [146] Wildgrén, M. Voting Power in the EC Decision Making and the Consequences of two Different Enlargements. *European Economic Review*, 38:1153–1170, 1994.
- [147] Wissler, W. *Unternehmenssteuerung durch Gemeinkostenzuteilung: Eine Spieltheoretische Untersuchung*. Gabler Verlag, Deutscher Universitäts-Verlag, Wiesbaden, Mannheim, 1997.
- [148] Wolff, R. Faire Methoden der Kostenallokation. *Das Wirtschaftsstudium (WISU)*, 8–9/04:1101–1106, 2004.
- [149] Wolff, R. A Computational Approach to the Coincidence of Egalitarian Solutions for Cost-Sharing Games. *International Game Theory Review (IGTR)*, 8(1):169–183, 2006.
- [150] Wolsey, L. A. The Nucleolus and the Kernel for Simple Games or Special Valid Inequalities for 0 – 1 Linear Integer Programs. *International Journal of Game Theory*, 5(4):227–238, 1976.
- [151] Wolsey, L. A. *Integer Programming*. Series in Discrete Mathematics and Optimization. John Willey & Sons, Inc., New York, Chichester, Weinheim, 1998.

Literaturverzeichnis

- [152] Young, H. P. Methods and Principles of Cost Allocation. In Young, H. P., editor, *Cost Allocation: Methods, Principles, Applications*, pages 3–29. North-Holland, Amsterdam, New York, Oxford, 1985.
- [153] Young, H. P. Monotonic Solutions of Cooperative Games. *International Journal of Game Theory*, 14(2):65–72, 1985.
- [154] Young, H. P. Cost Allocation. In R. J. Aumann and S. Hart, editors, *Handbook of Game Theory with Economic Applications*, volume II, pages 1193–1235. Amsterdam, 1994.
- [155] Young, H. P. *Equity: In Theory and Practice*. Princeton University Press, Princeton, New Jersey, 1994.
- [156] Zimmermann, W. *Operations Research - Quantitative Methoden zur Entscheidungsvorbereitung*. R. Oldenbourg Verlag, München, Wien, 4. Aufl., 1989.
- [157] Zumsteg, S. M. *Non-Cooperative Aspects of Cooperative Game Theory and Related Computational Problems*. Dissertation, Eidgenössische Technische Hochschule (ETH), Zürich, 1995.

URLs (Universal Resource Locators)

- <http://www.parlament.ch>
- <http://www.lindo.com>
- <http://www.math.unimaas.nl/PERSONAL/jeand/home1.htm>