Wie schnell arbeitet das Simplexverfahren normalerweise?

Oder: Das Streben nach (stochastischer) Unabhängigkeit

Karl Heinz Borgwardt

1972 stand die Welt der mathematischen Optimierung unter Schock, weil für das wichtigste Optimierungsverfahren, das Simplexverfahren zur Lösung linearer Optimierungsprobleme, mit den Klee-Minty-Polytopen Beispielprobleme gefunden worden waren, bei denen Varianten dieses Verfahrens einen exponentiellen Rechenaufwand benötigen. Dies stand in krassem Gegensatz zu den bis dahin gemachten – äußerst positiven – Erfahrungen mit diesem Algorithmus und dessen Geschwindigkeit.

Infolgedessen setzte eine Forschungsbewegung ein, die eine Welle von Publikationen und von wertvollen Ergebnissen hervorbrachte. Es ging darum, analytisch nachzuweisen, dass die entdeckten Probleme „Ausreißer" sind und dass das Simplexverfahren normalerweise viel besser ist als es diese Worst-Case-Resultate besagen. Damit begab man sich auf das Feld der probabilistischen Analyse von Algorithmen. Dort werden bei Unterstellung von Verteilungsannahmen für Probleme Erwartungswerte für den Rechenaufwand analytisch ermittelt oder abgeschätzt. Dieser Aufsatz beschäftigt sich mit dieser Forschungsrichtung in Bezug auf das Simplexverfahren, geht aber aus von einer grundsätzlichen Erörterung des Konzepts der probabilistischen Analyse von Algorithmen.

Im Zeitraum von 1975–1998 wurden viele derartige Untersuchungen zum Average-Case-Verhalten des Simplexverfahrens angestellt. Maßgeblich involviert in diese Forschungsrichtung der analytischen Ermittlung von Erwartungswerten und durchschnittlichen Rechenzeiten waren unter anderem Stephen Smale (Fields-Medaille 1966), Richard Karp (Fulkerson-Preis 1979), Michael Todd (George Dantzig Prize 1988, John von Neumann-Prize 2003), Ilan Adler und Nimrod Megiddo (Lanchester Prize 1988) und der Verfasser dieses Artikels (Lanchester Prize 1982). In all diesen Analysen konnte festgestellt werden, dass die ermittelten Erwartungswerte sehr klein waren.

Ab Beginn des neuen Jahrhunderts entwickelte sich noch eine andere Sicht auf die Beurteilung. Man wollte nun vermeiden, dass der Aufwand bei schweren Problemen dadurch verschleiert werden könnte, dass viele leichte Probleme diese ausgleichen. Anders gesehen, wollte man wissen, ob (oder ob nicht) sich die schweren Probleme irgendwo häufen. Dazu unterzog man die einzelnen festen Probleme leichten Störungen und mittelte dann in diesem kleinen Streubereich. Man war also gespannt darauf zu sehen, ob sich bei dieser Art von Mittelung schon eine Mäßigung des Rechenaufwands erkennen lässt. Diese Forschungsrichtung läuft unter dem Schlagwort Glättungsanalyse (Smoothed Analysis).

Es war das Verdienst von Dan Spielman, zusammen mit Shang-Hua Teng als erste dieses Prinzip erfolgreich auf das Simplexverfahren angewendet zu haben und bewiesen zu haben, dass auch aus dieser neuen Sicht das Normalverhalten besser als das Worst-Case-Verhalten ist. Dafür erhielt Dan Spielman 2010 den Rolf Nevanlinna-Preis.

In diesem Aufsatz sollen die Vorzüge, die Nachteile, aber auch die Missdeutbarkeiten und die manchmal überzogene Erwartungshaltung bei probabilistischen Analysen von Algorithmusgeschwindigkeiten erläutert werden. Dies geschieht konkret am Fall der probabilistischen Analyse des Simplexverfahrens, an deren Entwicklung der Autor maßgeblich beteiligt war. Hier wird gezeigt, wie weit man in den Forschungsbemühungen kommen konnte und welche systematischen Barrieren mathematischer Art einer weitergehenden Durchleuchtung im Wege stehen. Es wird verdeutlicht, dass der Analyseerfolg weitgehend davon abhängt, ob die verschiedenen ineinandergreifenden Einflussfaktoren auf den algorithmischen Verlauf voneinander separiert und dann einzeln analysiert werden können. Dass das möglich ist, ist nämlich weder selbstverständlich noch in der Praxis einfach durchführbar.

1 Was soll, will und kann die Probabilistische Analyse?

Wir gehen von einem Rechenverfahren aus, das zur Lösung konkreter Einzelprobleme (sogenannten Instanzen) I einer Problemart Π eingesetzt werden kann und das nachweislich für jedes I aus Π nach einer gewissen, von I abhängigen, Rechenzeit die gewünschte Lösung liefert. Wie üblich wollen wir im Folgenden anstelle der Rechenzeit die Anzahl der elementaren Rechenbefehle betrachten. Die in der Komplexitätstheorie oft beachtete Darstellungsgröße der darin verarbeiteten Zahlen (also die Ziffernzahl oder die Kodierungslänge) spielt hier keine Rolle.

Natürlicherweise geht man davon aus, dass in der Regel mit Zunahme der Anzahl von zu verarbeitenden Zahlen der Rechenaufwand wächst. Deshalb teilt man die Menge der Probleminstanzen I vom Typ Π ein in Kategorien, die nach Größe der Datenmenge sortieren, wie $\Pi_n = \{I \mid I$ ist Instanz von Π und enthält n Zahlen$\}$.

Innerhalb einer solchen n-Kategorie können nun Instanzen mit höchst unterschiedlichem Rechenaufwand $R(A, I)$ auftreten (A steht für Algorithmus, I für die Instanz).

DOI 10.1515/dmvm-2014-0037

Der klassische, von extremer Vorsicht geprägte Ansatz zur Beurteilung des Algorithmus A ist die *Worst-Case-Analyse*. Man versucht dabei, für jedes n das Maximum über $\{R(A, I) \mid I \in \Pi_n\}$ zu erkennen, und beurteilt nach dem Verlauf dieser Maximalwerte in Abhängigkeit von n.

Aber diese Kennzahl bzw. der Verlauf dieser Kennzahlen gibt oft keinen stichhaltigen Eindruck vom normalen Verhalten des Algorithmus. Um Informationen darüber erhalten zu können, müsste man erstmal eine klare Vorstellung davon haben, welche Instanzen $I \in \Pi_n$ real zu lösen sind und wie oft diese auftreten. In der Sprache der Wahrscheinlichkeitstheorie müsste man also wissen, wie die realen I über Π_n verteilt sind. Dies weiß niemand, aber viele haben dazu individuelle Vorstellungen. Basierend auf einer klaren Vorstellung wäre man vielleicht in der Lage, über Zufallsexperimente den Rechenaufwand statistisch zu evaluieren. Jedoch entzieht sich diese Methodik der empirischen Austestung einer Ausdehnung in höhere Dimensionen und das so gewonnene Datenergebnismaterial liefert meist keinen Einblick in die wirklichen Zusammenhänge und erlaubt deshalb auch keine zutreffende qualitative Deutung.

Für den Mathematiker stellt sich nun die folgende Herausforderung:

Kann ich bei Unterstellung der von mir angenommenen Verteilung der Instanzen in Π_n eine Analyse von probabilistischen Kenngrößen (Erwartungswerte, Varianzen usw.) analytisch gewinnen?

Dass dies meist nur unter außergewöhnlichen Umständen mit Ja beantwortet werden kann, ist offensichtlich. Und deshalb setzt ein rekursiver (eigentlich so nicht gewollter) Selektionsprozess ein:

Unser Mathematiker sucht nach Verteilungen oder Verteilungsmodellen, die er analysieren kann. Also tritt die Frage nach der „realen Verteilung" (die ja niemand kennt) in den Hintergrund. Das Ergebnis der – wenn überhaupt – erfolgreichen Analyse basiert also auf einer Unterstellung, die sowohl vom Produzent als auch vom Abnehmer akzeptiert sein muss.

Hat man sich auf ein einheitliches stochastisches Modell geeinigt, dann kann ein Wettbewerb zwischen verschiedenen Analysen um eine genauere Approximation der gewünschten Größen (z. B. Verkleinerung von Oberschranken) stattfinden. Hüten muss man sich aber vor einem Herumbasteln am stochastischen Modell mit dem Ziel, die Oberschranken zu senken. Allzuoft prägt dann das stochastische Modell selber das Ergebnis und man untersucht in Wirklichkeit nicht mehr die Güte des Algorithmus. Wie eminent sich solche stochastischen Annahmen auswirken können, wird in den folgenden Abschnitten verdeutlicht. Aber selbst bei Festhalten an einem stochastischen Modell ist die Akzeptanz noch nicht gesichert. Nun setzt nämlich die Kritik an dem gewählten Modell aus Sicht der Anwender ein. Zu rechtfertigen ist nun, dass das verwendete Modell etwas mit Realproblemen und deren Auftrittshäufigkeiten zu tun hat.

Hinzu kommt noch ein anderer Effekt, der auch in den folgenden Abschnitten noch konkret verdeutlicht wird. Man tut sich oft schwer damit, eine Analyse basierend auf dem Einsatz von Algorithmus A in Reinform erfolgreich durchzuführen. Oft wird nun der Ausweg gewählt, den Algorithmus so zu variieren, dass der variierte Algorithmus $\bar{A}$ tatsächlich analysierbar wird. Dessen Übereinstimmung mit A liegt dann vor allem (und manchmal nur noch) darin, dass er ebenfalls das Problem $I \in \Pi_n$ erfolgreich löst.

Bei all diesen Kritikpunkten gegen die genauen qualitativen und quantitativen Ergebnisse der Analyse sollten aber deren Verdienste nicht kleingeredet werden. Durch die intensive mathematische Analyse gelingt es oft, die Ursachen für ein typischerweise auftretendes oder ein kritisch verlaufendes Verhalten des Algorithmus zu erkennen. Abseits von quantitativen Auswertungen liefert dies eine Fülle von nachvollziehbaren Einsichten und Plausibilitäten. Zumeist steht am Ende von vielerlei Bemühungen der Eindruck, dass der Algorithmus unter all den verschiedenen unterstellten Verteilungen immer ganz passabel funktioniert.

2 Lineare Optimierung mit dem Simplexverfahren

Der Autor hat sich in den vergangenen Jahrzehnten intensiv mit der probabilistischen Analyse des Rechenaufwands beim Lösen Linearer Optimierungsprobleme befasst. Deshalb soll das im vorigen Abschnitt in genereller Form Angesprochene nun in seiner Bedeutung für dieses Forschungsfeld konkretisiert und verdeutlicht werden.

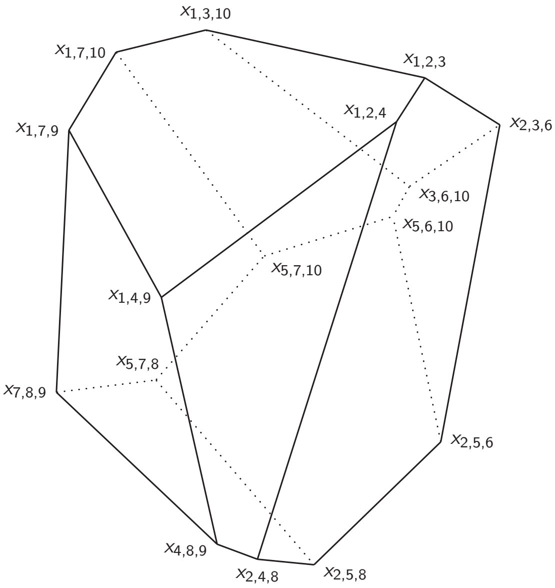
Man will also folgendes Problem lösen:

$$
(LP)\quad
\begin{array}{lll}
\text{maximiere} & v^T x & \text{als Zielfunktion} \\
\text{unter} & \text{den Restriktionen} & \\
 & a_1^T x \le b^1 & \\
 & \vdots & \text{bzw. } Ax \le b \\
 & a_m^T x \le b^m & \\
\text{wobei} & v, x, a_1, .., a_m \in \mathbb{R}^n, & b \in \mathbb{R}^m.
\end{array}
$$

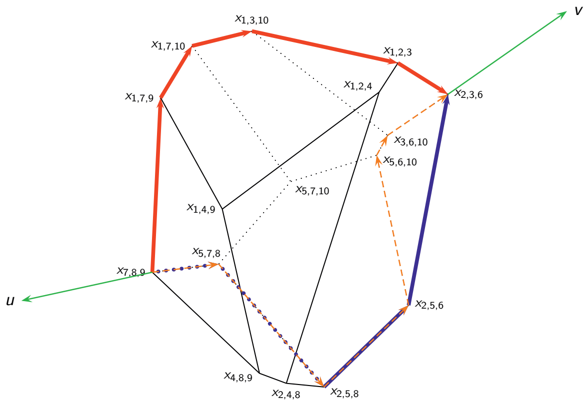
Also ist hier n die Dimension oder die Anzahl der Entscheidungsvariablen, m ist die Anzahl der Restriktionen, und grundsätzlich sei $m \ge n$.
So entsteht ein Polyeder $P(A, b) = \{x \in \mathbb{R}^n \mid Ax \le b\}$ als Zulässigkeitsbereich für die Wahl des optimalen Variablenvektors x_{opt}.

Es können folgende Fälle vorkommen:

- das Polyeder ist leer, weil das Restriktionensystem nicht erfüllbar ist;
- $v^T x$ (die Zielfunktion) lässt sich in $P(A, b)$ unbegrenzt steigern;
- in $P(A, b)$ gibt es Punkte x_{opt} mit maximalem $v^T x$-Wert.



Jede Ecke ist bestimmt durch drei Restriktionen, die dort straff werden.



Von der Optimalecke zur Richtung u führen mehrere Simplexpfade (unterschiedlicher Länge) zur v-Optimalecke. Der rote Pfad entspricht dem Schatteneckenalgorithmus.

Es wird von einem Lösungsverfahren erwartet, dass es in den ersten beiden Fällen diesen Sachverhalt meldet und im letzten Fall einen der Optimalpunkte benennt.

Wir unterstellen die Nichtentartung der Problemstellung (in (A, b) seien alle quadratischen Untermatrizen von vollem Rang und die a_i sowie die Zielrichtung v seien in allgemeiner Lage).

Wenn nun noch gilt $P(A, b) \neq \emptyset$, dann besitzt $P(A, b)$ wegen $m \geq n$ eine Ecke.
Und dann verläuft ausgehend von dieser Ecke x_0 ein (oder mehrere) Kantenpfad, wobei auf jeder Kante der Zielwert $v^T x$ strikt ansteigt und der

- entweder mit einer ins Unendliche verlaufenden Kante endet (Unbeschränktheit).
  Die Ecke, an der diese Kante beginnt, kann als Abbruchecke genutzt werden.
- oder bei einer Ecke (x_{opt}) endet, für die $v^T x$ optimal ist.

Damit man diese Methode einsetzen kann, ist entscheidend, dass sich bei Polyedern $P(A, b)$, die eine Ecke besitzen und auf denen die $v^T x$-Werte beschränkt bleiben, unter den Optimalpunkten eine Ecke befindet.

Diese iterierte Wanderung von Ecke über Kante zu einer Nachbarecke bis hin zur Optimal- oder Abbruchecke ist das Wesen des Simplexalgorithmus.

Jeder Wechsel von einer Ecke zu einer Nachbarecke (also über eine Kante) hat arithmetisch folgende Bedeutung:

Die erste Ecke x_Δ werde bestimmt durch eine n-elementige Indexmenge $\{\Delta^1, \Delta^2, \ldots, \Delta^n\} \subset \{1, \ldots, m\}$ als Lösung des Gleichungssystems

$a_{\Delta^1}^T x = b^{\Delta^1}, \ldots, a_{\Delta^n}^T x = b^{\Delta^n}$.

Für die Zulässigkeit von x_Δ ist entscheidend, dass auch für alle $k \notin \Delta$ gilt $a_k^T x_\Delta \leq b^k$.

Nun wird ein Eckenwechsel dadurch ausgelöst, dass für *ein einzelnes* Δ^i die Bedingung $a_{\Delta^i}^T x = b^{\Delta^i}$ gelockert wird zu $a_{\Delta^i}^T x < b^{\Delta^i}$ unter Beibehaltung der anderen $n-1$ Gleichungen. Dadurch wird eine Bewegung auf einer Kante weg von x_Δ initialisiert .

Diese Bewegung muss aber dort enden, wo $P(A, b)$ verlassen werden würde. Dies ist erkennbar daran, dass nun eine vorher lockere Restriktion $a_j^T x_\Delta \leq b^j$ mit $j \notin \Delta$ die Gleichung $a_j^T x_{neu} = b^j$ erfüllt. x_{neu} ist dann die Ecke am anderen Kantenende und erfüllt die Gleichungen

$$
\begin{aligned}
&a_{\Delta^1}^T x_{neu} = b^{\Delta^1}, \ldots, a_{\Delta^{i-1}}^T x_{neu} = b^{\Delta^{i-1}}, \quad a_j^T x_{neu} = b^j, \\
&a_{\Delta^{i+1}}^T x_{neu} = b^{\Delta^{i+1}}, \ldots, a_{\Delta^n}^T x_{neu} = b^{\Delta^n} \quad .
\end{aligned}
$$

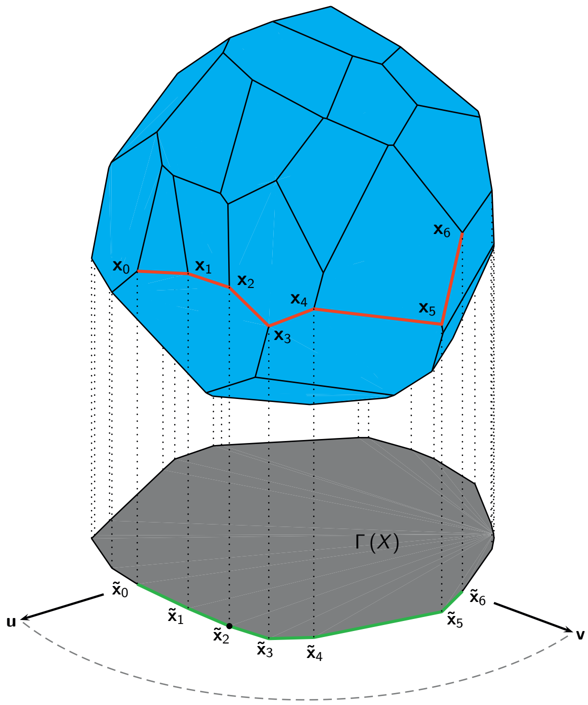
Für das Weitere wird zur Vereinfachung der Darstellung jeweils $\Delta = \{1, 2, \ldots, n\}$ unterstellt.

Diese Eckenwechsel (Pivotschritte) werden fortgesetzt, bis (wie vorher beschrieben) eine Optimal- oder Abbruchecke erreicht ist. Da von jeder Ecke (bei Nichtentartung) n Kanten ausgehen und ein Teil davon die Zielfunktion verbessert, steht diese Kantenmenge zur Auswahl und es erfordert eine feste Regel (eine Variante), um systematisch jeweils eine Kante zum Weiterkommen zu benutzen.

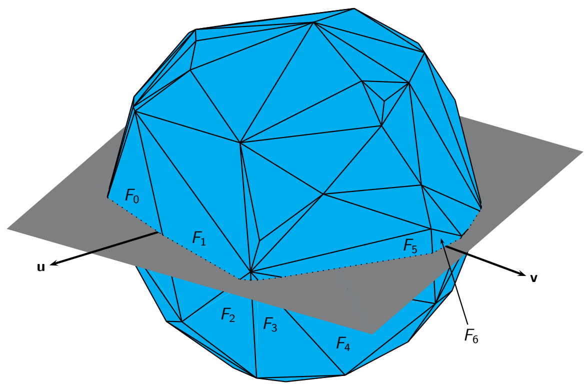
Nun sei noch klargestellt, dass zum Auffinden der erwähnten Startecke des Hauptproblems ein leicht modifiziertes Problem (statt LP) gelöst werden muss. Dessen Lösung geschieht nach dem gleichen erwähnten Prinzip. Das modifizierte Problem besitzt aber den Vorzug, dass man dafür eine Ausgangsecke kennt. Es besitzt jedoch den Nachteil, dass seine Lösung nur irgendeine Ecke des Originalpolyeders liefert, nicht etwa die Optimalecke. Diese so durch die „Phase *I*“ gelieferte Ecke wird dann als Startecke für die beschriebene „Phase *II*“ benutzt.

Die gebräuchlichsten Varianten (der Auswahl der nächsten Kante) orientieren sich an ganz unterschiedlichen Kriterien, wie z. B.

- am Zielfunktionsfortschritt, den die Benutzung dieser Kante bringen wird (*größte Verbesserung*)



Primale Sicht auf den Schatteneckenalgorithmus



Duale Sicht auf den Schatteneckenalgorithmus

- am Abweichungswinkel zwischen der Zielrichtung v und der Kantenrichtung (*steilster Anstieg*)
- an der Darstellung der Zielrichtung v durch die an der aktuellen Ecke straffen (d.h. mit Gleichheit erfüllten) Restriktionen
  $v = \sum_{i=1}^{n} \xi^i a_{\Delta^i}$ (*„Regel von Dantzig“*)
- an den Originalindizes der straffen Restriktionen (*„Regel von Bland“*).

Eine für die probabilistische Analyse entscheidende Sonderrolle spielt hier die parametrische Variante von Gass und Saaty [9] (mittlerweile auch bekannt unter dem Namen Schatteneckenalgorithmus). Diese soll wegen ihrer für das Hiesige wesentlichen Bedeutung näher erläutert werden. Die Bedeutung fußt auf dem sogenannten Polarkegelsatz bzw. dem Lemma von Farkas:

Eine Ecke x_Δ ist dann optimal, wenn die Restriktionsvektoren $a_{\Delta^1}, \ldots, a_{\Delta^n}$ zu den dort straffen Restriktionen die Zielrichtung konisch erzeugen, d.h. wenn gilt:
v liegt im Kegel dieser Vektoren bzw. in $\sum_{i=1}^{n} \xi^i a_{\Delta^i} = v$ sind alle $\xi^i \geq 0$.

Nun betrachtet man die vorliegende Startecke als Optimalecke zu einer Zielrichtung u, die man als konische Kombination aus $a_{\Delta^1}, \ldots, a_{\Delta^n}$ gewinnen kann, also $u = \sum_{i=1}^{n} \xi^i a_{\Delta^i}$ mit $\xi^i \geq 0$ für $1 \leq i \leq n$.

Wir sind uns aber bewusst, dass nicht $u^T x$, sondern $v^T x$ optimiert (maximiert) werden sollte.

Deshalb betrachtet man nun Mischzielrichtungen $w_\lambda = (1-\lambda)u + \lambda v$ mit $\lambda \in [0, 1]$. Während bei $\lambda = 0$ und $w_0 = u$ die Startecke für w_0 optimal ist, wird diese Ecke ihre Optimalität allmählich einbüßen, wenn wir λ steigern. Ab einem $\bar{\lambda} \geq 0$ ist dann eine andere Ecke optimal und dieser Wechsel vollzieht sich etliche Male, bis $\lambda = 1$ und $w_1 = v$ erreicht werden kann. Es zeigt sich, dass jeder solche Wechsel der Optimalecke der Wanderung von der bisherigen Optimalecke zu einer Nachbarecke entspricht, so dass diese Wanderung einen Simplexpfad von der $u^T x$-Optimalecke zur $v^T x$-Optimalecke ergibt. Der Aufwand für einen solchen Wechsel ist leicht bestimmbar ($O(m \cdot n)$). Deshalb ist man brennend daran interessiert, wie viele Eckenwechsel diese Wanderung erfordert. Das kann man somit auch daran ablesen, wie viele Ecken denn bei der λ-Steigerung temporär für gewisse w_λ die Optimalstellung innehaben. Der parametrische Algorithmus folgt mit seiner Eckenwanderung genau dieser Folge von Optimalecken.

Der Autor hat in seiner Dissertation 1977 eine geometrische Bedeutung dieses Prozesses erkannt. Jede erreichte Ecke ist ja bezüglich einer Zielrichtung $w_\lambda = (1-\lambda)u + \lambda v$ extremal. Projiziert man nun das Polyeder $P(A, b)$ auf die Ebene Span (u, v), so bildet sich ein zweidimensionaler Schatten. Die erwähnte Ecke wird wegen ihrer Extremalität auf eine Ecke des Schattens projiziert. Für diese Art von Ecken hat der Autor deshalb den Namen Schattenecken und für den obigen Algorithmus den Namen Schatteneckenalgorithmus geprägt [3, 4].

Folglich würde eine entdeckte Oberschranke für die Zahl der Schattenecken auch eine Oberschranke für die Simplexschritte auf dem Weg von Start- zu Zielecke abgeben.

Dem Beobachter dieser Bemühungen um probabilistische Resultate, die in den folgenden Kapiteln beschrieben werden, stellt sich die Frage, weshalb eine solche Analyse ausgerechnet für den Schatteneckenalgorithmus und nicht auch für andere Varianten funktioniert. Denn es ist schon auffällig, dass beim Rotationssymmetrie-Modell (Kapitel III) beim Umklapp-Modell (Kapitel IV) und bei der Glättungsanalyse (Kapitel V) tatsächlich nur Ergebnisse über den Schatteneckenalgorithmus vorliegen.

Der Grund liegt im erkennbar klaren geometrischen Konzept des Schatteneckenalgorithmus, das sich mühelos auf die Betrachtung der Einzelkandidaten (das sind in diesem Fall die Ecken bzw. Basislösungen) herunterbrechen lässt.

Hier wird nämlich jeder Kandidat (das ist etwa die Kombination aus $a_1, a_2, \dots, a_n$ bzw. die Lösung x_Δ des entsprechenen Gleichungssystems) daraufhin abgefragt, ob simultan

- x_Δ Ecke von $P(A, b)$ ist bzw. ob $\operatorname{Conv}(a_1, \dots, a_n)$ Facette von $\operatorname{Conv}(a_1, \dots, a_m, 0)$ ist
- x_Δ eine Schattenecke ist bzw. ob $\operatorname{Conv}(a_1, \dots, a_n)$ und $\operatorname{Span}(u, v)$ sich schneiden.

Damit ergibt sich die Schatteneckenzahl direkt aus der Anzahl der Kandidaten, die beides gleichzeitig erfüllen. Also setzt sich diese „Summe" aus den Einzelbeobachtungen zusammen. Den Erwartungswert bekommt man entsprechend, wenn man die Wahrscheinlichkeit dafür ermittelt, dass unser typischer Kandidat $\Delta = \{1, \dots, n\}$ beide Bedingungen gleichzeitig erfüllt.

Bei anderen Varianten des Simplexalgorithmus gibt es nur eine dynamische Charakterisierung mit Hilfe der Beobachtung des ganzen Pfades. Ob hier eine Ecke oder ein Kandidat $(a_1, \dots, a_n)$ vom Simplexpfad besucht wird, kann nicht allein aus der Beobachtung der Daten dieses Kandidaten erkannt werden. Vielmehr hängt dies entscheidend davon ab, wie der Verlauf des Simplexpfades war, bevor das v^Tx-Niveau unseres Kandidaten erreicht war. Wurden dabei etwa keine Nachbarecken unseres Kandidaten erfasst, dann bleibt auch unser Kandidat außen vor. Es türmen sich also hier enorme Abhängigkeiten zwischen allen Daten auf, die viel zu komplex sind, als dass man sie (bisher) in den Griff hätte bekommen können.

3 Probabilistische Analyse für den Schatteneckenalgorithmus unter dem Rotationssymmetrie-Modell

Wenn wir also bei der Variante Schatteneckenalgorithmus die Frage nach der Zahl der Ecken auf dem Simplexpfad schlichtweg ersetzen können durch die Frage nach der Anzahl von Schattenecken, dann bietet sich die Stochastische Geometrie zur Beantwortung an.

Um stochastische Informationen über diese Anzahl zu gewinnen, müssen wir uns zunächst auf ein stochastisches Modell festlegen. Dazu setzen wir (in diesem Abschnitt) das sogenannte Rotationssymmetrie-Modell ein.

Rotationssymmetrie-Modell
Zur Erzeugung der Probleminstanzen von

$$(LP1) \quad \text{maximiere } v^Tx \quad \text{unter } a_1^Tx \le 1, \dots, a_m^Tx \le 1$$

seien die Zufallsvektoren $v, a_1, \dots, a_m$ und ein Hilfsvektor u jeweils über $\mathbb{R}^n \setminus \{0\}$
- stochastisch unabhängig
- identisch
- rotationssymmetrisch

erzeugt.

(Die Festlegung der rechten Seite b^i auf 1 ist bedeutungslos, solange $b^i \ge 0$ gilt. In diesem Fall kann diese Normierung durch Anpassung der a_i erfolgen.)

Will man stochastische Geometrie betreiben, dann empfiehlt es sich, den Dualraum zu betrachten, in dem die Zufallsvektoren $v, a_1, \dots, a_m$ direkt erzeugt werden. Dies ist günstiger als im Primalraum, wo das Polyeder $P(A, b)$ und die Vektoren x leben, Studien anzustellen, die nur indirekt durch die Wahl der $v, a_1, \dots, a_m$ geprägt werden.

Die folgenden beiden Zusammenhänge besorgen die erforderliche Übersetzung

(a) x_Δ (bestimmt durch $a_1^Tx_\Delta = 1, \dots, a_n^Tx_\Delta = 1$) ist genau dann Ecke von $P(A, b) = \{x \mid a_1^Tx \le 1, \dots, a_m^Tx \le 1\}$, wenn $\operatorname{Conv}(a_1, \dots, a_n)$ Facette (Randsimplex) von $\operatorname{Conv}(0, a_1, \dots, a_m)$ ist.
(b) Die Ecke x_Δ von $P(A, b)$ ist genau dann Schattenecke bei Projektion auf $\operatorname{Span}(u, v)$, wenn $\operatorname{Conv}(a_1, \dots, a_n) \cap \operatorname{Span}(u, v) \neq \emptyset$.

Da es nur $\binom{m}{n}$ Auswahlmengen der Art $\{a_{\Delta^1}, \dots, a_{\Delta^n}\} \subset \{a_1, \dots, a_m\}$ gibt, lässt sich die erwartete Anzahl von Schattenecken errechnen als

$$EW\ (\text{Anzahl Schattenecken}) = \binom{m}{n} P(\operatorname{Conv}(a_1, .., a_n) \text{ ist Facette und } \operatorname{Conv}(a_1, .., a_n) \cap \operatorname{Span}(u, v) \neq \emptyset).$$

Da hier $a_1, \dots, a_n$ zufällig erzeugt sind, berücksichtigen wir deren Verteilungsfunktionen F in der folgenden Integraldarstellung :

$$EW(S) = \binom{m}{n} \int_{\mathbb{R}^n} \dots \int_{\mathbb{R}^n} P([\operatorname{Conv}(a_1, .., a_n) \text{ Facette}] \wedge [\operatorname{Conv}(a_1, .., a_n) \cap \operatorname{Span}(u, v) \neq \emptyset])dF(a_1) \dots dF(a_n)$$

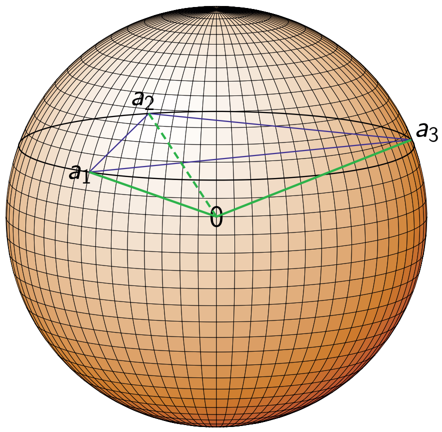
Es stellt sich heraus, dass das erste Ereignis, nur noch von der Lage von $a_{n+1}, \dots, a_m$ abhängt, sobald $a_1, \dots, a_n$ einmal festliegen. Genauso ist für das zweite Ereignis dann nur noch entscheidend, wie u und v liegen. Somit können wir in der Integraldarstellung aufspalten

$$EW(S) = \binom{m}{n} \int_{\mathbb{R}^n} \dots \int_{\mathbb{R}^n} P(\operatorname{Conv}(a_1, \dots, a_n) \text{ Facette}) \cdot P(\operatorname{Conv}(a_1, .., a_n) \cap \operatorname{Span}(u, v) \neq \emptyset) dF(a_1) \dots dF(a_n).$$

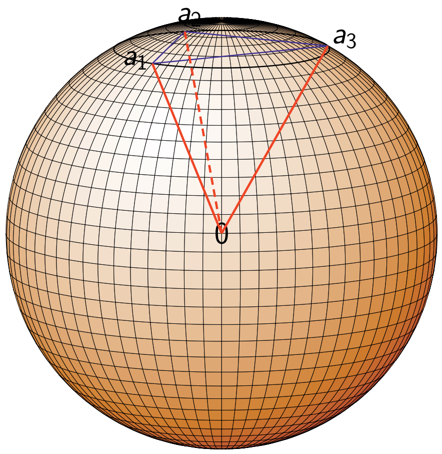
Man beachte aber, dass, solange sich $a_1, \dots, a_n$ ändern, *sicher nicht* gilt:

$$P([a_1, .., a_n \text{ def. Facette}] \wedge [\operatorname{Span}(u, v) \cap \operatorname{Conv}(..) \neq \emptyset]) = P(a_1, .., a_n \text{ def. Facette}) \cdot P(\operatorname{Span}(u, v) \cap \operatorname{Conv}(..) \neq \emptyset).$$

Um dies plausibel zu machen, betrachte man die Gleichverteilung auf der Vollkugel des $\mathbb{R}^n$.



Bei tiefer Lage des Dreiecks aus a_1, a_2, a_3 *öffnet sich* $\mathrm{Cone}(a_1, a_2, a_3)$ *stark, die Schnittwahrscheinlichkeit ist demnach groß. Gleichzeitig gibt es oberhalb extrem viel Platz für Punkte* a_i*, die die Facetteneigenschaft zerstören könnten. Die beiden Ereignisse sind also negativ korreliert.*



Bei hoher Lage des Dreiecks aus a_1, a_2, a_3 *öffnet sich* $\mathrm{Cone}(a_1, a_2, a_3)$ *nur schwach, die Schnittwahrscheinlichkeit ist demnach klein. Gleichzeitig gibt es oberhalb kaum Platz für Punkte* a_i*, die die Facetteneigenschaft zerstören könnten. Die beiden Ereignisse sind also negativ korreliert.*

$a_1, \ldots, a_n$ legen einen Simplex fest, der wiederum als affine Hülle eine Hyperebene definiert. Die Facetteneigenschaft erfordert, dass sich alle übrigen Vektoren $a_{n+1}, \ldots, a_m$ unterhalb dieser Hyperebene ansiedeln, d. h. in dem Halbraum, der auch den Ursprung enthält. Es liegt auf der Hand, dass bei Gleichverteilung aller Punkte auf der Vollkugel dieses Ereignis hochwahrscheinlich ist, wenn der Ursprung einen sehr großen Abstand zur Hyperebene hat, und sehr unwahrscheinlich wird, wenn der Abstand zwischen beiden klein ist. Aber genau umgekehrt ist es bei der Schnittbedingung. Hier muss ja der Kegel $\mathrm{Cone}(a_1, \ldots, a_n)$ von $\mathrm{Span}(u, v)$ geschnitten werden. Liegt nun die Hyperebene als Träger von $a_1, \ldots, a_n$ nahe beim Nullpunkt, dann können $a_1, \ldots, a_n$ sehr weit voneinander entfernt sein. Das heißt, der betrachtete Kegel kann sich sehr weit öffnen, was die Schnittwahrscheinlichkeit in die Höhe treibt. Hat aber die Hyperebene einen großen Abstand vom Nullpunkt, dann müssen sich die n Punkte auf einer sehr kleinen $n-1$-dimensionalen Kreisscheibe ansiedeln. Infolgedessen kann sich nur ein kleiner Kegel öffnen und die Schnittwahrscheinlichkeit wird extrem klein.

Dieser gegenläufige Effekt (diese negative Korrelation) sorgt nun dafür, dass das genannte Integral ziemlich klein bleibt. (Hätte man hier die oben beschriebene Produktform bzw. die Unabhängigkeit, dann wären die Oberschranken fulminant höher ausgefallen). Obige Unterscheidung hat zu Entstehungszeiten zu erheblichen Fehlinterpretationen, Missverständnissen und Anzweiflungen geführt. Wir werden im Späteren über andere Verteilungsmodelle reden, bei denen die hier zuweilen fälschlich unterstellte Unabhängigkeit wirklich zutrifft, was psychologisch einige Erklärungen für diese damaligen Auseinandersetzungen liefert. Nun ergab die korrekte Auswertung dieser negativen Korrelation für die Integraldarstellung folgendes Ergebnis

Fundamentalsatz (Schattenecken im RSM) [7]
Die erwartete Anzahl von Schattenecken bei m Restriktionen und n Entscheidungsvariablen bei Projektion auf $\mathit{Span}(u, v)$ ist für alle rotationssymmetrischen Verteilungen nicht größer als

$$m^{\frac{1}{n-1}} n^2 \cdot \mathrm{Const.}$$

Man mache sich an dieser Stelle klar, dass bisher nur der Eckenwanderungsweg von der Startecke zur Abbruchecke unter die Lupe genommen worden ist. Aber zunächst verfügen wir ja noch gar nicht über die Startecke bzw. die Optimalecke zu einer frei gewählten Zielrichtung u.

Und hier kommt man in ein Dilemma. Im obigen Fundamentalsatz über die Schatteneckenanzahl wird nämlich verlangt, dass u unabhängig von den übrigen Problemdaten zu wählen ist. Um dann eine Schattenecke zu bekommen, muss man eine Ecke x_0 haben, die $u^T x$ maximiert. Würde man aber zuerst das $u^T x$-Optimum suchen, dann wäre das Problem eigentlich nur verlagert worden von v auf u und keineswegs leichter geworden. Würde man u umgekehrt bestimmen aus $\mathrm{Cone}(a_{\Delta^1}, \ldots, a_{\Delta^n})$, wobei $a_{\Delta^1}, \ldots, a_{\Delta^n}$ die straffen Restriktionsvektoren an einer irgendwie gewonnenen Startecke sind, dann hätte man die Erfordernis der unabhängigen Wahl von u gegenüber $a_1, \ldots, a_m$ verletzt und deshalb hätte man keinen rigoros richtigen Beweis mehr. Der Leser sei darauf hingewiesen, dass sich diese Komplikation durch alle folgenden Kapitel dieses Aufsatzes schleppt.

Hier eröffnete sich ein Ausweg, der zwar vom effizientesten (Praktiker-)Verhalten abweicht, aber für einen rigorosen Beweis sorgt. Der sogenannte Dimensionssteigerungsalgorithmus löst das $v^T x$-Maximierungsproblem, indem er nacheinander den Schatteneckenalgorithmus zur Lösung von Problemen mit wachsender Dimension einsetzt. Dazu lösen wir sukzessive die folgenden Probleme in den Stufen k, wobei k von 1 bis n läuft.

$$\begin{array}{lll} & \text{maximiere} & v^T x \\ (\text{Stufe } k) & \text{unter} & a_1^T x \leq 1, \ldots, a_m^T x \leq 1 \\ & & x_{k+1} = 0, \ldots, x_n = 0 \end{array}$$

Man kann auf diese Weise den Optimalpunkt der k-ten Stufe dazu verwenden, um die $(k+1)$-te Stufe zu starten. Denn dieser Optimalpunkt der k-ten Stufe liegt auf einer Schattenkante des $(k+1)$-ten Problems bei Projektion auf Span(e_{k+1}, v). Man beachte, dass diese beiden Vektoren das Gebot der Unabhängigkeit von den anderen Eingabedaten erfüllen. Deshalb kann man dort wie gehabt den Schatteneckenalgorithmus einsetzen, um das $(k+1)$-te Ergebnis zu erhalten. Die letzte Stufe führt uns (wenn nicht vorher schon abgebrochen worden war) zum Optimalpunkt des Hauptproblems. Dadurch, dass man nun n Stufen durchläuft, schwillt die Oberschranke, die ja alle Schattenecken berücksichtigt, zunächst einmal um einen Faktor n an.

Man hat nach [7] und [6] folglich für alle Dimensionspaare (m, n):

$$EW(\text{Schatteneckenzahl bei Dimensionssteigerung}) \leq m^{\frac{1}{n-1}} \cdot n^3 \cdot \text{ Const.}$$

Jedoch kann man zumindest asymptotisch (d.h. bei $m \gg n$) nachweisen, dass der Korrekturbedarf in den höheren Stufen immer kleiner wird. Weil nämlich die Nachbesserung ja nur die letzte Komponente v^{k+1} der Zielrichtung betrifft, wird nur noch ein mit k immer kleiner werdender Anteil der vorhandenen Schattenecken für Simplexschritte genutzt, so dass man bei $m \gg n$ eine Abschätzung folgender Art hat [12]

$$EW(\text{benutzte Anzahl Ecken bei Dimensionssteigerung}) \leq m^{\frac{1}{n-1}} \cdot n^{\frac{5}{2}} \cdot \text{ Const.}$$

Es bleibt zu klären, wie man mit Problemen umgeht, bei denen die rechten Seiten b^i auch negativ sein können und bei denen die b^i unabhängig von den a_i gewählt werden. Leider lassen sich unsere Transformationen in den Dualraum und unsere Skalierungen dann nicht mehr direkt durchziehen. Wir können aber einen rigorosen Beweis für folgendes Vorgehen führen.

Löse zunächst mit dem beschriebenen Dimensionssteigerungsalgorithmus das Problem in der (bis jetzt bearbeitbaren) Einheits-Form, also

$$(LP1) \quad \max v^T x \text{ unter } a_i^T x \leq 1 \text{ für } i = 1, \dots, m$$

und bewahre dessen Optimalecke.

Dann können wir durch Einführung einer zusätzlichen Variable x^{n+1} eine Stufe an den Dimensionssteigerungsalgorithmus anhängen, bei der wir verlangen

$$\max x^{n+1} \text{ unter } a_i^T x + x^{n+1}(1 - b^i) \leq 1 \; \forall i = 1 \dots m$$

Dies führt zu folgender Beobachtung

- bei $x^{n+1} = 0$ liegt immer noch das Original mit Restriktionen $a_i^T x \leq 1$ vor
- bei $x^{n+1} = 1$ erfüllen wir die Anforderungen des erweiterten Problems mit allgemeinem b^i.

Das passt aber in unser Schema vom Dimensionssteigerungsalgorithmus als Stufe $n+1$. Als letztes soll sich also x^{n+1} von 0 lösen dürfen. Ist durch Steigerung von x^{n+1} der Wert 1 nicht erreichbar, dann ist das Originalproblem nicht zulässig. Ist aber durch Steigerung von x^{n+1} der Wert 1 erreichbar, dann ist man mit $(x^1, \dots, x^n)$ zulässig. Hat man auch für diese letzte Stufe den Schatteneckenalgorithmus benutzt, dann ist man beim Erreichen von $x^{n+1} = 1$ gleich beim optimalen x angelangt.

Eine Verteilungsannahme für die Werte $b^1, \dots, b^m$, für die auf diese Art ein Resultat gewonnen wurde, ist z. B.

- $a_1, \dots, a_m$ sowie v seien nach dem Rotationssymmetrie-Modell verteilt.
- die $b^1, \dots, b^m$ seien davon und untereinander unabhängig jeweils gleichverteilt auf $[-1, 1]$.

Daraus entsteht dann eine Art Zylinderverteilung für die Vektoren (a_i, b^i) (Rotationssymmetrie in den ersten n Komponenten, Gleichverteilung in der letzten).

Während für die ersten n Stufen die obigen Ergebnisse (Rotationssymmetrie) Bestand haben, muss die letzte Stufe wegen ihrer andersartigen Verteilung anders analysiert werden. Aber auch für diese letzte Stufe wurde bestätigt, dass (bei $m \gg n$) gilt [10]

$$EW(\text{durchlaufene Anzahl von Schattenecken}) \leq m^{\frac{1}{n-1}} \cdot n^{\frac{5}{2}} \cdot \text{ Const.}$$

Unter Hinzunahme dieser letzten Stufe lassen sich deshalb auch die Probleme mit beliebigen b^i unter unserem Modell lösen mit einem Gesamtaufwand von

$$EW(\text{Gesamtaufwand}) \leq m^{\frac{1}{n-1}} \cdot n^{\frac{5}{2}} \text{ Const.}$$

Nun wird der Praktiker entgegenhalten: Ich suche mir in einer ersten Phase eine Startecke (das ist irgendwie vergleichbar mit einem Durchlauf des Schatteneckenalgorithmus). Zu dieser so erhaltenen Startecke bestimme ich ein u, das diese Ecke $u^T x$-optimal macht. Danach optimiere ich in Phase II mit dem Schatteneckenalgorithmus die Zielfunktion $v^T x$. Da beides vergleichbare Optimierungsprobleme sind, schätze ich den Aufwand hierfür auf ca. $m^{\frac{1}{n-1}} n^2$. Obwohl mannigfache Testläufe diese Hypothese irgendwie bestätigen, kann man nicht davon sprechen, dass diese Oberschranke bewiesen wäre. Wir stellen zwei essentielle Verletzungen der Voraussetzungen für den Beweis fest:

- Das vom Praktiker aufgestellte Phase-I-Problem entspricht nicht den Anforderungen des Rotationssymmetrie-Modells.
- Die nachträgliche Bestimmung der Zielrichtung u aus dem Polarkegel der gewonnenen Startecke (also aus Cone$(a_{\Delta^1}, \dots, a_{\Delta^n})$) verletzt offensichtlich die essentiellen Unabhängigkeits-Anforderungen an das Problem im Rotationssymmetrie-Modell.

Also befindet man sich hier tatsächlich in einer Situation, wo man – wie angedeutet – einen um einen Faktor $n^{\frac{1}{2}}$ (bei $m \gg n$) erhöhten Laufzeitbedarf für einen rigorosen Beweis in Kauf nehmen muss.

4 Stochastische Unabhängigkeit der Ungleichungausrichtungen

Parallel zum Rotationssymmetriemodell wurde ein gänzlich verschiedenes Modell, das Umklapp-Modell (Sign-Invariance-Modell) für Average-Case-Analysen herangezogen [1, 2, 11, 14, 17].

Hier unterstellt man, dass der Zielvektor v und die Restriktionsvektoren $a_1, \dots, a_m$ zunächst einmal festgelegt sind (mit der kleinen Einschränkung der Nichtentartung). Nun besteht die Variation der Probleme darin, dass für jede Restriktionsungleichung und unabhängig voneinander entschieden wird, ob diese Restriktion von der Form $a_i^T x \leq b^i$ oder $a_i^T x \geq b^i$ sein soll. Beide Versionen sollen mit Wahrscheinlichkeit $\frac{1}{2}$ auftreten. Die untersuchten und zu lösenden Probleme sind also von der Art

maximiere	$v^T x$
unter	$a_1^T x \leq\geq b^1, \dots, a_m^T x \leq\geq b^m$
sowie	$P(\leq) = \frac{1}{2} = P(\geq)$ für jede Einzelrestriktion.

Auf diese Weise generiert eine Festlegung der Restriktionsvektoren schon 2^m gleichberechtigte Partnerprobleme. Bei der Average-Case Analyse werden jetzt fiktiv alle diese Probleme gelöst, ihre Schrittzahlen werden aufaddiert und danach wird durch 2^m geteilt.

Bestechend an diesem Ansatz ist die Tatsache, dass (solange die a_i nichtentartet festgelegt wurden) sich für jede Festlegung der gleiche Mittelwert ergibt: Man erspart sich also hier alle Berücksichtigungen der Variation dieser a_i, b^i. Diese angesprochene Gleichheit ist eine Folge der altbekannten Ergebnisse von Buck über Hyperebenenarrangements im $\mathbb{R}^n$. Danach ergibt sich bei Vorgabe von m Hyperebenen (nichtentartet) eine Partition des Raumes in Zellen, deren Anzahl immer gleich ist, nämlich $\binom{m}{0} + \binom{m}{1} + \binom{m}{2} + \dots + \binom{m}{n}$ [8].

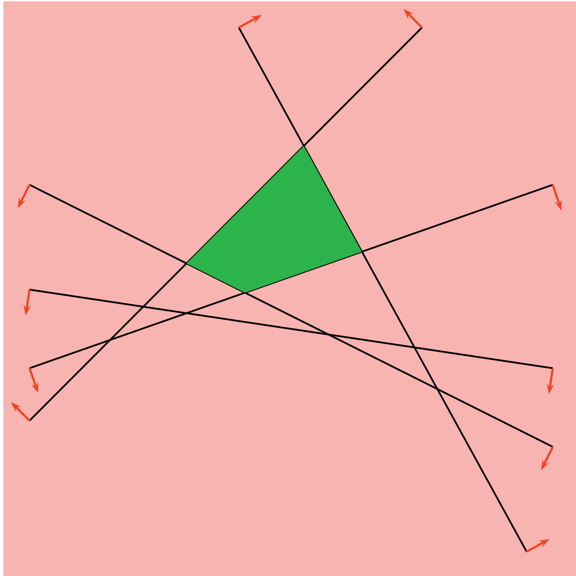
Diese Zellen entsprechen genau denjenigen Umklapp-Probleminstanzen (unter den 2^m möglichen), die überhaupt zulässige Punkte x besitzen. Man beachte aber, dass $\binom{m}{0} + \binom{m}{1} + \cdots + \binom{m}{n} \leq 2^m$ bei $m \geq n$ und dass der Quotient

$$\frac{1}{2^m} \cdot \left[\binom{m}{0} + \cdots + \binom{m}{n}\right]$$

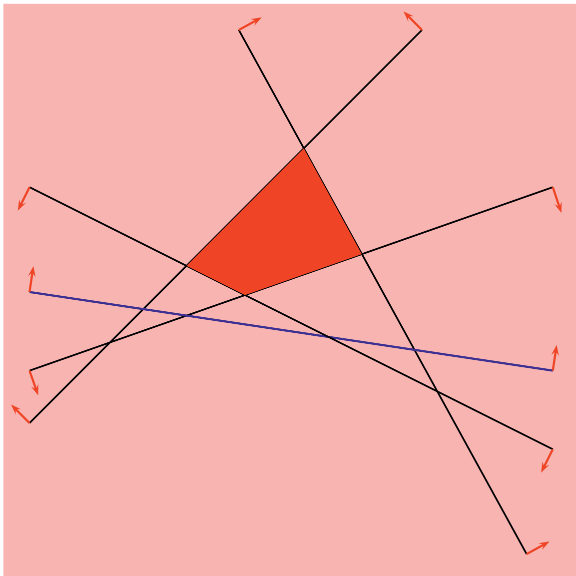
dramatisch gegen 0 driftet, wenn bei festem n die Restriktionszahl m gegen ∞ geht.

Dies bedeutet, dass allein schon bedingt durch das angesetzte stochastische Modell nur in einem verschwindend kleinen Anteil der Probleminstanzen überhaupt noch eine Phase II stattfindet.

Der entsprechende Effekt wird auch noch dann erzielt, wenn man auf die zulässigen Probleme konditioniert. Denn es lässt sich zeigen, dass die Gesamtzahl der Ecken



Bei 5 Hyperebenen entstehen 16 Zellen. Wenn die Ausrichtungen alle passen, wird eine Zelle zulässig.



Dreht man bei einer vorher redundanten Restriktion die Ausrichtung um, dann gibt es gar keine Zulässigkeit mehr.

aller Zellen nicht größer ist als $2^n\binom{m}{n}$. Dann entwickelt sich die Relation

$$\frac{2^n\binom{m}{n}}{\binom{m}{0} + \binom{m}{1} + \cdots + \binom{m}{n}}$$

und dies wird auch klein bei $m \gg n$.

Der Grund hierfür liegt im Wesentlichen in der Tatsache begründet, dass bei $m \gg n$ eine zusätzliche Hyperebene eine vorher bestehende Zelle zumeist verfehlt, also nicht mehr teilt. Dann gibt es in diesem Fall zwei gleichwahrscheinliche Möglichkeiten aufgrund der Ausrichtung dieser Restriktion

– die Zulässigkeit dieser Zelle wird vernichtet,

– die Zelle bleibt zulässig und komplett unversehrt (die neue Hyperebene hat also nichts zur Eckenvermehrung bei dieser Zelle und insgesamt beigetragen).

So weit hat unser stochastisches Modell also die Zeichen bereits gesetzt. Man beachte: wir haben bisher noch gar nicht festgelegt, welche Variante des Simplexalgorithmus eingesetzt wird.

Auch hier erwies sich der Schatteneckenalgorithmus als einzige auswertbare Option.

Es ergab sich als Resultat der Mittelwertbildung über die in Phase II (der Optimierungsphase) durchgeführten Pivotschritte (= Zahl der Schattenecken) ein Durchschnitt von nicht mehr als n Schritten pro zulässiger Zelle [1, 2, 11, 17].

Noch haben wir aber nicht erörtert, wie man jeweils eine Startecke (in Phase I) findet. Dazu haben [1, 2] einen sogenannten *Constraint-By-Constraint*-Algorithmus eingesetzt.

Dieser Algorithmus verläuft in $m-n$ Stufen. In jeder Stufe wird dafür gesorgt, dass eine weitere Restriktion erfüllt ist. Zur Erklärung seien gerade die $\leq$-Richtungen aktuell.

Zunächst sind nur n Restriktionen erfüllt, wir agieren also auf $X^{(n)} = \{x \mid a_1^T x \leq b^1, \dots, a_n^T x \leq b^n\}$.

Nun versucht man ausgehend von der einzigen Ecke von $X^{(n)}$ die Restriktion $a_{n+1}^T x \leq b^{n+1}$ auch noch einzuhalten. Ist dies nicht erreichbar, dann kann man schon wegen Unzulässigkeit abbrechen.

Andernfalls gehen wir zur nächsten Restriktion über und wiederholen das Verfahren.

So wird also in $X^{(k)} = \{x \mid a_1^T x \leq b^1, \dots, a_k^T x \leq b^k\}$ in Zielrichtung a_{k+1} minimiert, bis $a_{k+1}^T x \leq b^k$ erreicht ist.

Bezeichnend ist, dass dazu in jeder Stufe der parametrische Algorithmus eingesetzt wird. Ausgehend von einer einmal geschickt lexikographisch gewählten Zielrichtung $u \in \text{Cone}(a_1, \dots, a_n)$, die an der einzigen Ecke von $X^{(n)}$ optimiert wird, und mit Hilfe der Zielrichtung a_{k+1} kann dann für jede Stufe eine Schatten-Projektionsebene festgelegt und damit jeweils der Schatteneckenalgorithmus eingesetzt werden.

Man nutzt hier die Cooptimalitätseigenschaft des parametrischen Algorithmus aus. Dies bedeutet, dass für jedes erreichte Niveau bzgl. $a_i^T x$ der auf dem Schatteneckenpfad erreichte Punkt den besten $u^T x$-Wert aufweist.

Deshalb erreicht dieses Verfahren also im Fall der Zulässigkeit gleich die $u^T x$-optimale Ecke von $X^{(m)} = P(A, b)$. Und von dort aus kann die Phase II starten. Im Fall der Unzulässigkeit bricht eine der Stufen ab.

In [1, 2] konnten die Autoren so nachweisen, dass auf diese Weise der Gesamtaufwand im Erwartungswert die Schranke von $2(n+1)^2$ nicht überschreitet.

Also verhindert hier eigentlich das Modell (und nicht die Qualität des Algorithmus) das Anwachsen der Schrittzahlen mit m. Die Autoren konstatieren: „These results cannot show a good behaviour of the algorithm for $m \gg n$, but they are a consequence of the stochastic model per se" [1].

Ein anderer interessanter Aspekt ist aber bei diesem Modell zu beobachten (ganz im Gegensatz zum Rotationssymmetrie-Modell). Hier sind nämlich die Ereignisse

– $a_1, \dots, a_n$ definieren über $a_1^T x = b^1, \dots, a_n^T x = b^n$ eine Ecke von $X^m = P(A, b)$
– der Polarkegel dieser Basislösung enthält die Zielrichtung u, also $u \in \text{Cone}(a_1, \dots, a_n)$

tatsächlich stochastisch unabhängig voneinander (denn hier liegen $a_1, \dots, a_n$ ja von vornherein fest).

Hat man nämlich $\bar{x}$ ermittelt als Lösung von $a_1^T \bar{x} = b^1, \dots, a_n^T \bar{x} = b^n$, dann ist für die Eigenschaft der Basislösung unwichtig, ob $a_i^T x \leq b^i$ oder $a_i^T x \geq b^i$ $(i \leq n)$ gefordert wurde.

Nun aber entscheiden über die primale Zulässigkeit von $\bar{x}$ die Ausrichtungen der restlichen Ungleichungen, nämlich $a_i^T x \leq b^i$ oder $a_i^T x \geq b^i$ für $i = n+1, \dots, m$. Um aber zu wissen, ob diese Basislösung dual zulässig ist, muss man für eine Richtung v (die Zielrichtung) wissen, ob sie im Polarkegel von $\bar{x}$ liegt. Dieser hat formal die Beschreibung $\text{Cone}(\pm a_1, \dots, \pm a_n)$, wobei $+$ zu wählen ist bei Ausrichtung $\leq$ und $-$ bei Ausrichtung $\geq$. Da im Nichtentartungsfall genau eine Ausrichtungskombination zum Erfolg führt, entscheiden also die Ausrichtungen zu $a_1, \dots, a_n$ über die duale Zulässigkeit. Dagegen wird über die primale Zulässigkeit von den Ausrichtungen zu $a_{n+1}, \dots, a_m$ entschieden. Die Definition des Modells sorgt dann für die stochastische Unabhängigkeit, da ja die Ausrichtungen unabhängig voneinander zu wählen sind.

5 Glättungsanalyse des Simplexverfahrens

Anfang des neuen Jahrhunderts kam eine ganz andersartige Philosophie oder Sicht der Dinge bezüglich der normalen Laufzeit-Bewertung auf. Unter dem Namen „Smoothed Analysis" (Glättungsanalyse) sollte die folgende Frage geklärt werden:

Wenn alle Daten des Problems (also $a_1, \dots, a_m$) geringfügig Gauß-verteilt gestört werden und das Problem jeweils unter allen solchen Störungskonstellationen gelöst würde, wird sich dann eine signifikante Glättung ergeben? Das heißt, werden diese Erwartungswerte erkennbar tiefer liegen als die extrem hohen Aufwandswerte, die man bei festen Problemen in Kauf nehmen muss (Klee–Minty-Probleme)? Wenn man will, kann man dies als eine andere Art von Average-Case-Analyse ansehen.

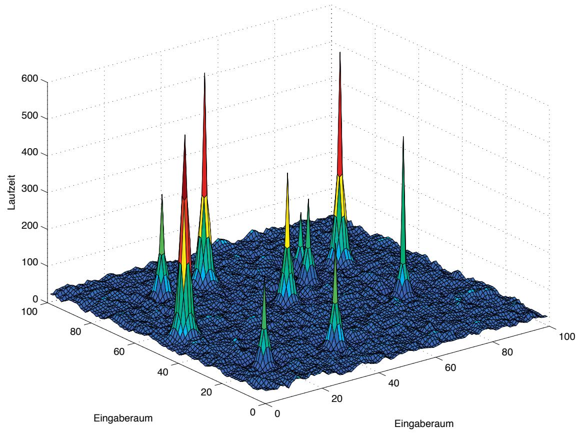
Eigentlich zu lösen ist also das feste Originalproblem

$$(\overline{LP}) \quad \begin{array}{ll} \text{maximiere} & \bar{v}^T x \\ \text{unter} & \bar{a}_1^T x \leq \bar{b}^1, \dots, \bar{a}_m^T x \leq \bar{b}^m \\ \text{mit} & \bar{a}_1, \dots, \bar{a}_m \in \mathbb{R}^n, \bar{b}^1, \dots, \bar{b}^m \in \mathbb{R}. \end{array}$$

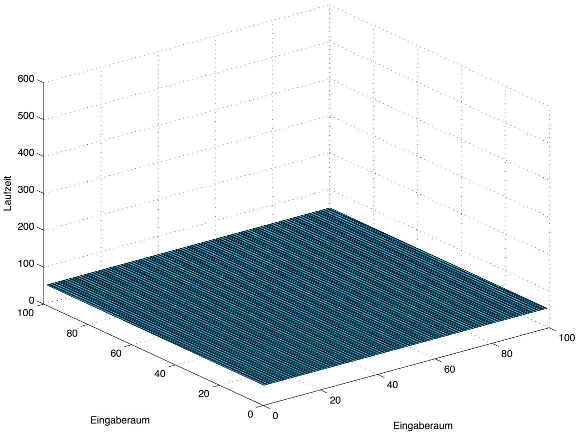
Nun sollen an den Vektoren $\bar{a}_1, \dots, \bar{a}_m$ und an den $\bar{b}^i$ Störungen der Form $N(\bar{a}_i, \sigma)$ (also normal verteilt um $\bar{a}_i$ bzw. $\bar{b}^i$ mit Streuung σ) angebracht werden, so dass jeweils tatsächlich gelöst wird

$$\begin{array}{ll} \text{maximiere} & \bar{v}^T x \\ \text{unter} & a_1^T x \leq b^1, \dots a_m^T x \leq b^m \\ \text{mit} & (a_i, b^i) \in N((\bar{a}_i, \bar{b}^i), \sigma). \end{array}$$

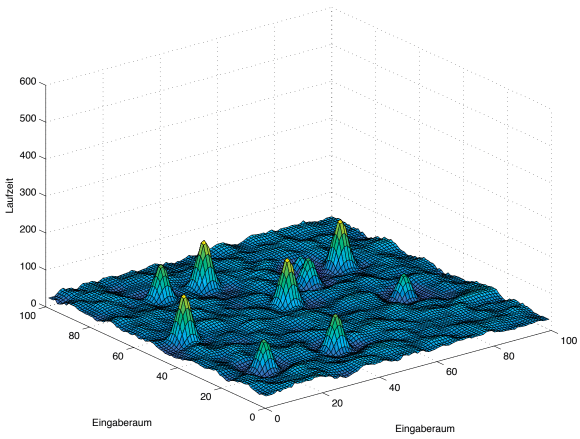
Der erwartete Aufwand, der bei dieser Art Mittelung entsteht, orientiert sich also nicht mehr an einer Gesamtverteilung, sondern konkret an diesem festen Originalproblem. Es besteht die Hoffnung, dass man mit geringfügig wachsendem σ schon die extremen Ausschläge von Festproblemen vermeiden (besänftigen) kann. Und man hofft, dass für diesen Glättungseffekt eine gleichmäßige Oberschranke für alle Originalprobleme gefunden werden kann. Es ist deshalb angemessen und unvermeidlich, dass man bei der Gütemessung der Glättung neben den üblichen Dimensionen m und n auch die Störungsintensität einbezieht. Dies sollte konsequenterweise mit σ^{-1}, also indirekt proportional zur Streuung geschehen. Die Verfechter dieser Sicht auf die Bewertung der Algorithmuslaufzeit führen folgende Argumente ins Feld:



Bei einzelnen Problemen ergeben sich extreme Ausschläge der Rechenzeit.



Eine Average-Case Analyse ergibt einen Mittelwert über dem ganzen Raum der Probleme.



Die Glättungsanalyse mittelt in der Nähe eines jeden Festproblems und behebt so die extremem Ausschläge.

– Im Gesamtbereich aller Problemstellungen wird der einzelne Anwender es oft mit einem oder einzelnen Typen zu tun haben. Viele denkbare Problemstellungen bzw. Datensetzungen haben keinen realen Anwendungshintergrund (auch wenn eine Abgrenzung zwischen realen und fiktiven nicht möglich ist).
– Das eigentlich gleiche Problem wird oft mit unterschiedlicher Technologie und unterschiedlicher Modellierungsgenauigkeit gelöst bzw. beschrieben.
– Es soll vermieden werden, dass die Effekte bei „ernstzunehmenden" realen Problemstellungen ausgeglichen oder verwässert werden durch fiktive Probleme, die rein aus Verteilungs- oder Symmetriegründen den ersten formal gleichgestellt sind.

Die Basis für alle im Folgenden besprochenen Aufwandsuntersuchungen für die Lösung des Gesamtproblems bildet das Fundamentaltheorem, das eine Oberschranke dafür angibt, wie viele Ecken des Zufallspolyeders $\{x \mid Ax \leq 1\}$ bei Projektion auf eine (wirklich feste) Ebene $\mathrm{Span}(\bar{u}, \bar{v})$ zu Ecken des Bildes dieser Projektion werden. Also sind wir wieder bei der Anzahl der Schattenecken.

Fundamentalheorem bei Glättung [15, 16]
Seien alle $\bar{a}_i$ von der Norm $||\bar{a}_i|| \leq 1$ und alle $\bar{b}^i = 1$. Dann ist der Erwartungswert der Schatteneckenanzahl bei Projektion auf $\mathrm{Span}(\bar{u}, \bar{v})$ durch $D(m, n, \sigma)$ gleichmäßig nach oben beschränkt, wobei

$$D(m, n, \sigma) := \frac{(58.888.678) \cdot m \cdot n^3}{\left[\mathrm{Min}\ \{\sigma, \frac{1}{3\sqrt{n\ \ln m}}\}\right]^6}.$$

Setzt man beispielsweise σ so an, dass sich im Nenner Gleichheit der beiden Größen in der Minimierungsklammer ergibt, dann hat man $D(m, n, \sigma) \approx$ Const. $\cdot\, mn^3 \cdot n^3 \cdot (\ln m)^3$.

Die Herleitung dieses Ergebnisses ist enorm kompliziert und spielt in immenser Weise mit dem Instrumentarium der Eigenschaften von verknüpften und approximierten Normalverteilungen. Eine Verdeutlichung davon kann daher hier nicht vorgenommen werden.

Die einschränkende Bedingung $||\bar{a}_i|| \leq 1$ erweist sich bei der Algorithmusanalyse als sehr hinderlich, weil man ja eine gleichmäßige Oberschranke für alle Probleme und alle Spielarten der a_i sucht. Jedoch lassen sich nach diesbezüglicher Relaxierung durch Skalierung auch noch dann

Beschränkungen retten, wenn die obige Bedingung verletzt ist. Allerdings fallen dann die Oberschranken entsprechend höher aus. Erlaubt man auch höhere Normen der $\bar{a}_i$, dann ist in obiger Formel σ zu ersetzen durch

$$\frac{\sigma}{\text{Max}\{\| \bar{a}_i \| | i = 1, \dots, m\}}.$$

Erlaubt man auch Vektorenmengen $\{a_1, \dots, a_m\}$, die mit unterschiedlichen Covarianzmatrizen M_i verteilt sind, dann braucht man zur grundsätzlichen Erhaltung der obigen Abschätzung, dass die Eigenwerte der M_i zwischen σ^2 und $\frac{1}{9n \cdot \ln m}$ liegen. Erlaubt man, dass die b^i positiv bleiben, aber von 1 abweichen, dann ist σ nun zu ersetzen durch die (nun noch kleinere) Größe

$$\frac{\sigma \cdot \text{Min}\{b^1, \dots, b^n\}}{\text{Max}\{\| a_1 \|, \dots, \| a_m \|\} \cdot \text{Max}\{b^1, \dots, b^m\}}$$

Wie zu erkennen ist, treibt jede Relaxierung der Voraussetzungen aus dem Fundamentaltheorem die Oberschranken nach oben. Und da eine gleichmäßige Oberschranke angestrebt wird, muss man alle denkbaren Konstellationen abdecken. Außerdem ergeben sich die beschriebenen „Überschreitungen" selbst aus günstigen Anfangsdaten, wenn man zur Algorithmusanalyse des Phasenwechsels bzw. des Gesamtalgorithmus kommt.

Betrachten wir nun die Problem-Gesamtbearbeitung und deren Analyse unter Zuhilfenahme des Fundamentaltheorems. Denn dazu braucht man zunächst eine Startecke und dazu ein festes $\bar{u}$. Im Gegensatz zum früher beschriebenen Dimensionssteigerungsalgorithmus kommt man in diesem Fall nicht automatisch an der $\bar{u}$-optimalen Ecke vorbei und kann auch nicht mehr davon ausgehen, dass die Ankunftsecke nach Phase I unabhängig von (A, b) ist – es ist nicht einmal mehr selbstverständlich, dass die erzeugten Störungen dieser Ankunftsecke nach Abschluss der Phase I dem normalverteilten Störungsbild entsprechen, wie es im Fundamentaltheorem von den Daten verlangt wird. So wird u auch noch abhängig von $a_1, \dots, a_n$, da die $a_1, \dots, a_n$ ja stochastisch zum Wackeln gebracht werden. Es kann z. B. geschehen, dass $\bar{u} \in \text{Cone}(\bar{a}_1, \dots, \bar{a}_n)$, aber $u \notin \text{Cone}(a_1, \dots, a_n)$.

Natürlich könnte man sich, ausgehend von der Phase-I-Ecke, durch Optimierung von $\bar{u}^T x$ die „richtige" Startecke besorgen. Dies hätte die Problematik aber nur von $\bar{v}^T x$ auf $\bar{u}^T x$ verlagert.

Deshalb bereinigen Spielman und Teng die obigen Komplikationen durch angemessene Erhöhung ihrer Oberschranken. Um einen analysierbaren Lösungsalgorithmus zu bekommen, werden zwei Schritte ausgeführt. Zunächst wird ein zulässiges Hilfsproblem künstlich so konstruiert, dass man für eine gewünschte Indexmenge (z. B. $\Delta = \{1, \dots, n\}$) mit Hilfe der Restriktionen $a_1, \dots, a_n$ eine Startecke erhält. Da Δ in aller Regel von vornherein noch keine Ecke liefert, werden alle anderen Restriktionen $a_{n+1}^T x \leq b^{n+1}, \dots, a_m^T x \leq b^m$ relaxiert, d. h. ihre Oberbegrenzungen werden erhöht oder die a_j werden skaliert. Die künstlichen b^i werden positiv gewählt, um vom Fundamentaltheorem profitieren zu können.

Weil die notwendige Stärke dieser Relaxierungen von vornherein nicht bekannt ist, werden hierzu vorsichtigerweise Größen verwendet, die aus den Eigenwerten/ Konditionszahlen der Ausgangsmatrix entnommen werden können, was wiederum die Oberschranken anhebt.

In einem ersten Schritt löst man dieses eben geschaffene Hilfsproblem, bedenkt aber, dass eigentlich das Hauptproblem (evtl. auch mit negativen b^i) zu lösen wäre. Unbeschränktheit – auch für das Hauptproblem – wird schon in diesem ersten Schritt sichtbar.

Nun parametrisiert / interpoliert man zwischen den relaxierten Versionen der a_i, b^i mit positiven b^i und den originalen Versionen. Auch dafür eignet sich der parametrische Algorithmus (in einer dualen Form). (Dies entspricht der Vorgehensweise, die schon am Ende von Kapitel 3 beschrieben worden war). Vorteilhaft dafür ist, dass auf diese Weise das Fundamentaltheorem greift. Dies ist dann der zweite Schritt, der die Originallösung liefert.

Hierbei hing im ersten Schritt zwar nicht die Zielrichtung v, wohl aber die Anfangsrichtung u von (A, b) ab. Im Gegensatz dazu entspricht der zweite Schritt durchaus den Anforderungen des Fundamentaltheorems. Jedoch führen die Skalierungen im ersten Schritt und der Phasenwechsel zum Anschwellen der Schranken aus dem Fundamentaltheorem. Auch das Wackeln von u lässt sich auffangen, aber dazu müssen wiederum die Schwellen angehoben werden.

Als Alternative zu diesem mehrmaligen Schranken-Heben käme auch eine Randomisierung in Frage. Die Hoffnung ist dabei, dass man hier nur die Schranke so weit wie nötig anheben muss. Insbesondere legt man sich hier nicht auf eine Wunschbasis wie $a_1, \dots, a_n$ fest, sondern probiert eine (komplexitätstheoretisch noch vertretbare) Vielzahl von Anfangsbasen aus. Davon verwendet man diejenige, die die geringste Erhöhung der Schranken verursachen würde. Es wird dabei stochastisch ausgenutzt, dass durch das Wackeln der Vektoren a_i (im Gegensatz zu den festen $\bar{a}_i$) eine relativ hohe Wahrscheinlichkeit dafür besteht, dass der kleinste Eigenwert der A-Teilmatrix von 0 erkennbar abweicht und dass die Konditionszahl relativ harmlos wird. Schließlich werden auch die b^i beliebig aus $\mathbb{R}$ gewählt und gestört. Legt man die Ergebnisse der stochastischen Glättungsanalyse alle betont pessimistisch aus (also so, dass alle Oberschrankenerhöhungen notwendig sind), dann kommt Vershynin für die Spielman-Teng-Laufzeitschranke auf eine Größenordnung von

$$m^{86} n^{55} \sigma^{-30}.$$

Vershynin hat in [18] die Abschätzungsmethodik für sein Fundamentaltheorem neu gestaltet. Darin erreicht er für $\sigma \leq \frac{1}{6\sqrt{n \ln m}}$ eine Oberschranke von $Cd^3 \sigma^{-4}$.

Zusätzlich hat er die Algorithmusdurchführung modifiziert. Zunächst einmal werden die Übergänge vom Hilfsproblem zum Originalproblem so gestaltet, dass die parametrische Interpolation simultan auf die Zielfunktion und auf die rechten Seiten b^i zugreift. Die Variation der rechten Seite wird dabei aufgefangen durch eine zusätzliche Variable. Man vermeidet dadurch Verzerrungen von gaußverteilten Störungen beim Phasenwechsel.

Um das Startproblem in den Griff zu bekommen, geht Vershynin einen anderen Weg als Spielman/Teng. Er verzichtet auf die Auswahl einer Wunschbasis und damit auch auf die Relaxierung der restlichen Restriktionen. Stattdessen fügt er n Restriktionen

$a_{m+1}^T x \leq b^{m+1}, \dots, a_{m+n}^T x \leq b^{m+n}$

hinzu und macht diese so restriktiv, dass diese eine Ecke generieren. Diese Ecke dient als Startecke und das Barizentrum dieser Vektoren wird die Startrichtung $\bar{u}$. Von dieser fiktiven Ecke aus wird nun der parametrische Algorithmus gestartet.

Man gelangt dann zu einer Optimal- oder Abbruchecke. Diese aus dem modifizierten Problem gewonnene Abbruchecke wird dann daraufhin überprüft, ob sie nichts mehr mit den künstlichen (hinzugefügten) Restriktionen zu tun hat (ob diese also dort alle locker sind). Ist dies der Fall, dann ist das Problem ja gelöst.

Andernfalls (also wenn die erreichte Abbruchecke noch straffe hinzugefügte Restriktionen impliziert) muss der Lösungsprozess als misslungener Test verworfen werden. Nun wird ein erneuter Test durchgeführt, mit anders gewählten Vektoren $a_{m+1}, \dots, a_{m+n}$. Um zu erreichen und um zu zeigen, dass diese Tests nicht oft scheitern, geht er wie folgt vor:

Er bestimmt gleichverteilt über der Einheitskugel eine Richtung $\tilde{u}$. Nun bestimmt er $a_{m+1}, \dots, a_{m+n}$ so, dass $||a_i - \tilde{u}||$ für alle i sehr klein ist und dass gilt $\tilde{u} \in \text{Cone}(a_{m+1}, \dots, a_{m+n})$. Zum Nachweis der Erfolgswahrscheinlichkeit betrachtet er den sogenannten *numbset* des Problems. Dieser besteht aus dem Bereich/Anteil der Einheitskugel, in den man Restriktionen hinzufügen kann, ohne dass dies auf das Ergebnis Einfluss haben würde. Es ist klar, dass dieser *numbset* immer mindestens die Hälfte der Einheitskugel umfasst (er enthält einen Halbraum). Man kann also sicher sein, dass man mit Wahrscheinlichkeit nahe bei $\frac{1}{2}$ alle a_{m+i} im *numbset* untergebracht hat.

Und dies würde zum Erfolg des Tests führen: Eine tatsächliche Unterschranke dafür, dass all diese Vektoren im *numbset* liegen, ist $P = \frac{1}{4}$.

Wiederholt man diese Prozedur, so ergibt sich eine Oberschranke für die erwartete Anzahl der nötigen Testläufe (in diesem Falle 4). Somit wurde ein randomisierter Algorithmus eingesetzt, der (im schlechten Falle) nicht funktionieren muss, dessen nachhaltige Wiederholung aber mit hoher Wahrscheinlichkeit zum Ziel führt. Dadurch, dass es hier zu weniger Systembrüchen (Phasenwechseln) und zu weniger Verteilungsanpassungen kommt, verläuft hier die Kombination der verschiedenen Gaußanalysen und die Anwendung des Fundamentaltheorems über Zusatzabschätzungen viel harmloser.

Satz [18]
Für diesen beschriebenen randomisierten Algorithmus gilt eine gleichmäßige Oberschranke von der Größenordnung

$$O(\ln^2 m \cdot \ln(\ln m)(n^3\sigma^{-4} + n^5 \ln^2 m + n^9 \ln^4 n)).$$

Hier hat man angesichts der Aufgabenstellung die Randomisierung akzeptiert.

Ohne Randomisierung (dafür aber mit Hilfe des umständlicheren Dimensionssteigerungsalgorithmus) hat Emanuel Schnalzger aufbauend auf den Ergebnissen von Spielman & Teng sowie von Vershynin nachweisen können, dass mit diesem Algorithmus ein erwarteter geglätteter Aufwand von

$$\text{Const}\left(\frac{1}{\sigma^2} + \frac{(n+1)^4}{\sigma^4} + (\ln m)^2(n+1)^6\right)$$

verbunden ist [13].

Diskutabel ist nun, ob man neben Datenzufälligkeit auch noch Verfahrenszufälligkeit (Randomisierung) akzeptiert.

Und hier schließt sich der Kreis: Was wollte ich eigentlich – was soll ich – was kann ich?

Literatur

[1] Adler, I., R. M. Karp, R. Shamir (1987). A Simplex Variant Solving an $m \times d$ Linear Program in $O(min(m^2, d^2))$ Expected Number of Steps. *Journal of Complexity* **3** 372–387.

[2] Adler, I., N. Megiddo (1985). A Simplex Algorithm where the Average Number of Steps is Bounded Between two Quadratic Functions of the Smaller Dimension. *Journal of the ACM* **32** 871–895.

[3] Borgwardt, K.H.(1977). *Untersuchungen zur Asymptotik der mittleren Schrittzahl von Simplexverfahren in der linearen Optimierung*, Dissertation Universität Kaiserslautern.

[4] Borgwardt, K.H.(1982a). Some Distribution-Independent Results About the Asymptotic Order of the Average Number of Pivot Steps of the Simplex Method. *Mathematics of Operations Research* **7** 441–462.

[5] Borgwardt, K.H.(1982b). The Average Number of Pivot Steps Required by the Simplex-Method is Polynomial, *Zeitschrift für Operations Research* **26** 157–177.

[6] Borgwardt, K.H. (1987). *The Simplex Method, A Probabilistic Analysis*, Springer Verlag, Heidelberg.

[7] Borgwardt, K.H. (1999). A sharp upper bound for the expected number of shadow-vertices in LP-polyhedra under orthogonal projection on two-dimensional planes, In *Mathematics of Operations Research*, Vol.24, Nr. 4, 925–984

[8] Buck, R.C. (1943). Partition of Space, In *American Mathematical Monthly 50* 541–544

[9] Gass, S. & Saaty, Th. (1955). The Computational Algorithm for the Parametric Objective Function, Naval Research Logistics, Quarterly 2, 39–45

[10] Göhl, M. (2013). Der durchschnittliche Rechenaufwand des Simplexverfahrens unter einem verallgemeinerten Rotationssymmetrie-Modell, Dissertation, Universität Augsburg.

[11] Haimovich, M.(1983).*The Simplex Algorithm is Very Good! - On the Expected Number of Pivot Steps and Related Properties of Random Linear Programs*. Report Columbia University, New York.

[12] Höfner, G. (1995). *Lineare Optimierung mit dem Schatteneckenalgorithmus –Untersuchungen zum mittleren Rechenaufwand und Entartungsverhalten*, Dissertation Universität Augsburg.

[13] Schnalzger, E. (2014). Interner Report Universität Augsburg: *Smoothed Analysis des Dimensionssteigerungsalgorithmus*.

[14] Smale, S.(1983). On the Average Speed of the Simplex Method of Linear Programming. *Mathematical Programming* **27** 241–262.

[15] Spielman, D. & Tng, Sh.H. (2004). Smoothed Analysis of Algorithms: Why the Simplex Algorithm usually takes Polynomial Time, In *Journal of the ACM* **51(3)**, 385–463.

[16] Spielman, D. & Tng, Sh.H. (2008). Smoothed Analysis of Algorithms: Why the Simplex Algorithm usually takes Polynomial Time, 7. Version, Februar 2008, Report.

[17] Todd, M.J. (1986). Polynomial Expected Behavior of a Pivoting Algorithm for Linear Complimentarity and Linear Programming Problems. *Mathematical Programming* **35** 173–192.

[18] Vershynin, R. (2009). Beyond Hirsch conjecture: walls on random polytopes and smoothed complexity of the simplex method, In *SIAM Journal on Computing* **39(2)**, 646–678.

Prof. Dr. Karl Heinz Borgwardt, Lehrstuhl für Diskrete Mathematik, Optimierung und Operations Research, Institut für Mathematik, Universität Augsburg, 86135 Augsburg
borgwardt@math.uni-augsburg.de

1949 in Landstuhl/Pfalz geboren, Studium der Mathematik und Betriebswirtschaftslehre in Saarbrücken, 1973–1979 Wissenschaftlicher Mitarbeiter an der TU Kaiserslautern dort Promotion 1977 und Habilitation 1985. 1979–1984 Planungsabteilung der Deutsche Bank AG Zentrale (Frankfurt), 1982 Lanchester Prize für den Nachweis der Polynomialität der mittleren Schrittzahl beim Simplexverfahren, 1984–2014 Professor für Optimierung und Operations Research am Institut für Mathematik der Universität Augsburg. Forschungsgebiete: Probabilistische Analyse von Algorithmen, Lineare Optimierung, Heuristiken, Stochastische Polyedertheorie