Den Kohl heile rüberbringen...

Mathematische Methoden der Transportoptimierung

Eröffnungsvortrag von Prof. Dr. Martin Grötschel, Berlin

Bei der Entwicklung vieler mathematischer Teilgebiete, insbesondere der Diskreten Mathematik (Kombinatorik, Graphentheorie), der Optimierung (ganzzahlige Programmierung) und des Operations Research, haben Transportprobleme eine wichtige Rolle gespielt. So ist etwa Eulers berühmtes Königsberger Brückenproblem, das den Anlaß zur Entwicklung der Graphentheorie gab, ein (Unterhaltungs-) Transportproblem. Die Lineare Programmierung, das wahrscheinlich meistgenutzte und erfolgreichste mathematische Optimierungsverfahren, wurde unabhängig voneinander Ende der dreißiger Jahre in der ehemaligen Sowjetunion zur Lösung ökonomischer Planungs- und Verteilungsprobleme und Ende des zweiten Weltkrieges in den USA zur Lösung von Transport- und Logistikaufgaben entwickelt.

Doch die Beschäftigung mit Transportproblemen reicht noch viel weiter zurück. Die ersten tauchten schon vor über 1200 Jahren in dem Buch "Propositiones ad acuendos iuvenes" auf, der ältesten Sammlung mathematischer Aufgaben in lateinischer Sprache. Sie entstand im achten Jahrhundert nach Christus und wird Alcuin von York zugeschrieben. Alcuin war einer der führenden Gelehrten seiner Zeit und Berater Karls des Großen am fränkischen Hof. Die meisten seiner Aufgaben entstammen noch älteren chinesischen, ägyptischen, griechischen oder anderen Quellen. Alcuins vier "Flußüberquerungsprobleme" scheinen jedoch neu zu sein. Sie stellen die ältesten bekannten Transportprobleme in der mathematischen Literatur dar.

Alcuins Flußüberquerungsprobleme hatten keinen Einfluß auf die Entwicklung der Mathematik; die Probleme waren schlicht Denksportaufgaben. Tatsächlich weisen die Aufgaben aber bereits alle Charakteristika großer realer Transportprobleme auf, wie sie heute betrachtet und gelöst werden. Aus heutiger Sicht hätten die Flußüberquerungsprobleme deshalb der Ausgangspunkt zur Entwicklung des Operations Research und der ganzzahligen Programmierung sein können. Wir weisen dies anhand von Alcuins Problem 18 über einen Wolf, eine Ziege und einen Kohlkopf nach: *18. "Propositio de lupo et capra et fasciculo cauli."*

Ein Mann mußte einen Wolf, eine Ziege und einen Kohlkopf über einen Fluß bringen und konnte nur ein Schiff finden, das nicht mehr als zwei Gegenstände tragen konnte. Es war ihm aber vorgeschrieben, daß er sie alle unverletzt hinüberbringen sollte. Sage, wer es kann, wie er sie unverletzt hinüberbringen konnte.

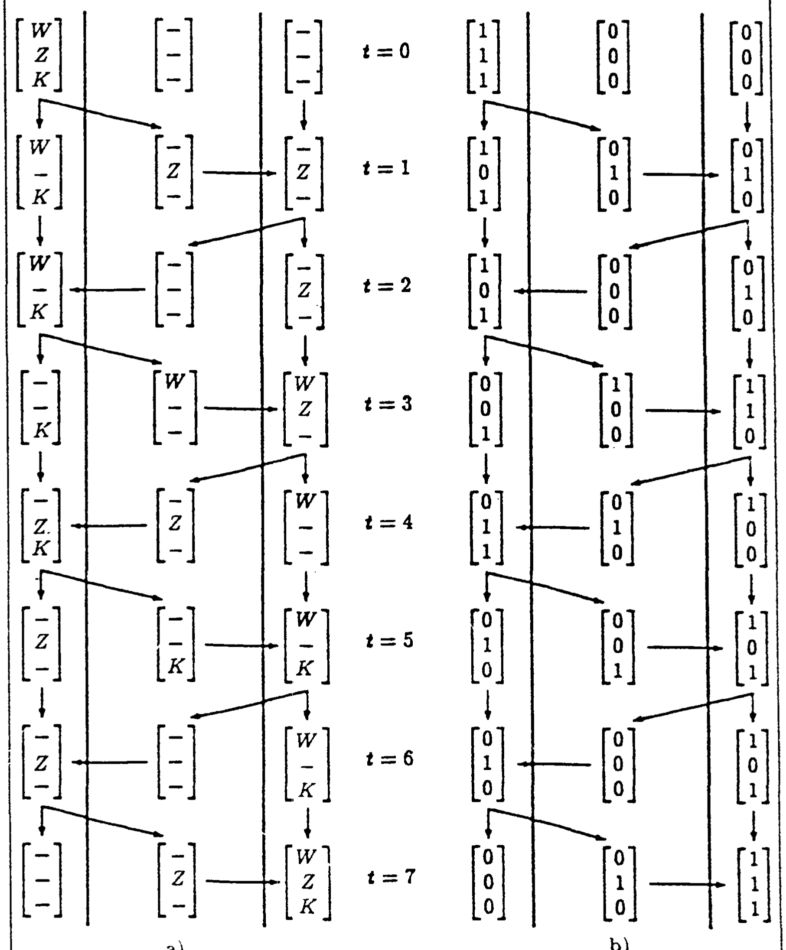

Abbildung 1. Alcuins Lösung: "Ich bringe zuerst die Ziege hinüber und lasse den Wolf und den Kohlkopf zurück. Dann fahre ich zurück und bringe den Wolf hinüber, lasse ihn drüben und bringe die Ziege zurück. Dann lasse ich die Ziege aus dem Boot und bringe den Kohlkopf hinüber, fahre nochmal zurück und bringe die Ziege hinüber. So wird die Kahnfahrt ohne Schaden ausgeführt, und ohne daß eines das andere verschlingt."

$$\text{Minimiere} \quad \sum_{t=0}^{T}\sum_{i=1}^{3} x(t,i)$$

unter den Nebenbedingungen

$$\begin{array}{rcll}
x(0) &=& (1,1,1) & \\
y(0) &=& (0,0,0) & \\
z(0) &=& (0,0,0) & \\
\left.\begin{array}{rcl} x(t+1) &=& x(t)-y(t+1) \\ z(t+1) &=& z(t)+y(t+1) \end{array}\right\} & & & \text{falls } 0 \le t \le T \text{ gerade ist} \\
\left.\begin{array}{rcl} x(t+1) &=& x(t)+y(t+1) \\ z(t+1) &=& z(t)-y(t+1) \end{array}\right\} & & & \text{falls } 0 \le t \le T \text{ ungerade ist} \\
y(t,1)+y(t,2)+y(t,3) &\le& 1 & \text{für alle } 0 \le t \le T \\
\left.\begin{array}{r} x(t,1) + x(t,2) \le 1 \\ x(t,2) + x(t,3) \le 1 \end{array}\right\} & & & \text{falls } 0 \le t \le T \text{ ungerade ist} \\
\left.\begin{array}{r} -z(t,1) + z(t,2) + z(t,3) \le 1 \\ z(t,1) + z(t,2) - z(t,3) \le 1 \end{array}\right\} & & & \text{falls } 0 \le t \le T \text{ gerade ist} \\
z(T) &=& (1,1,1) & \\
0 \le x(t,i), y(t,i), z(t,i) &\le& 1 & \text{für alle } 0 \le t \le T, 1 \le i \le 3.
\end{array}$$

Abbildung 2

Durch eine sorgfältige Analyse dieses Problems mit modernen mathematischen Methoden wollen wir dem Zuhörer eine allgemeinverständliche und unterhaltsame Einführung in die moderne Theorie der ganzzahligen Programmierung geben.

Der erste Schritt besteht aus der Herleitung einer mathematischen Formulierung des Wolf-Ziege-Kohl-Problems als ganzzahliges Programm. Dazu gehen wir von der in der linken Hälfte von Abbildung 1 gezeigten Lösung aus, die Alcuin für sein "Kohl-Problem" angegeben hat. Dort symbolisiert ein W, Z bzw. K die Anwesenheit des Wolfes, der Ziege bzw. des Kohlkopfes und ein Strich '-' ihre Abwesenheit. Die Abbildung zeigt, wie man sukzessive Wolf, Ziege und Kohlkopf von der linken zur rechten Seite des Flusses verschifft, so daß alle "heile" rechts ankommen. Ersetzt man nun die W, Z und K durch eine 1 an der entsprechenden Stelle und schreibt 0 statt '-', so erhält man die in der rechten Hälfte von Abbildung 1 gezeigte Lösung. Diese erlaubt es uns nun, mit Wölfen, Ziegen und Kohlköpfen zu "rechnen". Weiter bezeichnen wir mit x(t), y(t) bzw. z(t) die, wie man sagt, Inzidenzvektoren der Wolf-Ziege-Kohlkopf-Konfiguration auf der linken Flußseite, auf dem Boot bzw. auf der rechten Flußseite nach t Flußüberquerungen. Die Anfangs- und Endbedingungen (t=0 bzw. t=T) und die Interpretation von Alcuins Phrase "unverletzt" führen dann nach einigen zusätzlichen Überlegungen und Umformulierungen zu der in Abb. 2 gezeigten mathematischen Formulierung des Wolf-Ziege-Kohlkopf-Problems:

Für die Lösung dieses ganzzahligen Programms diskutieren wir anschließend verschiedene Techniken. Wir zeigen die Vielseitigkeit und Flexibilität dieser Art der mathematischen Modellierung, indem wir erläutern, wie auf (zwar nicht mechanische aber) einfache Weise Variationen und Erweiterungen des Problems behandelt werden können. Wir erklären, wie ein solches ganzzahliges Programm mit Hilfe von linearer Programmierung angegangen und schließlich im Rahmen eines Schnittebenenverfahrens und/oder eines Branch & Bound-Verfahrens gelöst werden kann. Speziell zeigen wir, wie und warum man mit Hilfe von ganzzahliger Programmierung eine mathematisch beweisbar beste Lösung oder (falls das Verfahren aus Zeitmangel oder anderen Gründen vorzeitig abgebrochen wird) eine Lösung, die höchstens einen beweisbaren Prozentsatz schlechter als die (unbekannte) Optimallösung ist, erhalten kann. In anderen Worten, wir führen vor, wie und warum solche Verfahren Lösungen von beweisbarer Güte liefern. Die von uns dabei diskutierten Techniken sind genau die Verfahren, die heute zur Lösung von großen realen Transportproblemen verwendet werden.

Zum Abschluß unseres Vortrages schließen wir dann den Bogen von Alcuin zu den Transportproblemen von heute, indem wir noch zwei derartige Probleme - an denen wir derzeit arbeiten - vorstellen: die Behindertenbeförderung in Berlin und die Busumlaufplanung in Hamburg.

Die Stadt Berlin betreibt eine Flotte von etwa hundert Spezialbussen zur Beförderung von Rollstuhlfahrern. Jeder Bus kann gleichzeitig zwischen zwei und vier Rollstuhlfahrer transportieren. Zur Zeit nehmen etwa 23.000 Behinderte diesen Service in Anspruch. Täglich werden zwischen 1.000 und 1.500 Transporte durchgeführt. Das Problem, das wir hier untersucht haben, bestand aus der täglichen Zuordnung der individuellen Transporte auf die Busse, so daß einerseits eine möglichst pünktliche Abholung der Personen gewährleistet ist und anderseits die dabei anfallenden Kosten so gering wie möglich sind. Wir haben dieses Problem als ein spezielles ganzzahliges Programm, genauer, als ein sogenanntes Set-Partitioning Problem, formuliert, dessen Matrix aus etwa 1.000 bis 1.500 Zeilen besteht und zwischen 100.000 und einer Million Spalten enthält. Dieses Problem können wir derzeit zwar noch nicht optimal lösen, doch gelingt es uns, in wenigen Minuten CPU-Zeit Näherungslösungen zu bestimmen, die um höchstens 10% bis 20% vom (unbekannten) Minimum abweichen. Eine erste Version unseres Systems ist seit Juni 1995 in Berlin im Einsatz und führte zu einer Kostenersparnis von etwa 25% im Vergleich zu der früher "von Hand" durchgeführten Zuordnung.

Die Busumlaufplanung besteht aus folgendem Problem: ein Transportunternehmen (in unserem Fall der öffentliche Nahverkehr der Stadt Hamburg) hat sich unter Berücksichtigung diverser Nebenbedingungen und Einschränkungen für die Routen und Fahrpläne der Busse entschieden. Das Unternehmen betreibt mehrere Depots und Bustypen. Die Aufgabe besteht nun darin, die Busse so den einzelnen Einsätzen zuzuordnen, daß zum einen jede Fahrt von einem "zulässigen" Bustyp bedient wird und zum anderen die Flottengröße und die Anzahl der Leerfahrten so klein wie möglich ist. Auch dieses Problem läßt sich wieder als ein spezielles ganzzahliges Programm formulieren. Diesmal ist es ein sogenanntes ganzzahliges Mehrgüterflußproblem. Für das Problem der Stadt Hamburg enthält dieses Programm etwa 30 Millionen Variablen. Eine optimale Lösung können wir in etwa 2 Tagen berechnen. Testrechnungen mit Daten verschiedener Verkehrsbetriebe zeigen, daß die Einsparungsmöglichkeiten in Bezug auf die Anzahl benötigter Busse und die Kosten für die notwendigen Leerfahrten beträchtlich sind.

LITERATUR

Borndörfer, R., Grötschel, M. und Löbel, A.: Alcuin's transportation problems and integer programming, erscheint in: P.L. Butzer et al (Hrsg.), "Colloquium Carolus Magnus", 1996.

Butzer, P.L. und Lohrmann, D. (Hrsg.): "Science in Western and Eastern Civilization in Carolingian Times", Birkhäuser Verlag, Basel, 1993.

Grötschel, M. und Lovasz, L.: Combinatorial Optimization, in: R. Graham, M. Grötschel und L. Lovasz (Hrsg.), "Handbook of Combinatorics", Elsevier Science B. V., Amsterdam, 1995, Chapter 28, 1541 - 1597.

Grötschel, M., Lovasz, L. und Schrijver, A.: "Geometric Algorithms and Combinatorial Optimization", Springer Verlag, Berlin, 1988.

Padberg, M.: "Linear Optimization and Extensions", Springer Verlag, Berlin, 1995.

Schrijver, A.: "Theory of Linear and Integer Programming", John Wiley & Sons Ltd., Chichester, 1989.