Zuordnungsprobleme - ein Streifzug durch die kombinatorische Optimierung

Rainer E. Burkard

Das Heiratsproblem

In der ersten Hälfte unseres Jahrhunderts wurde folgendes Problem studiert: Wir betrachten eine Gruppe von n Mädchen und n jungen Herren. Jedes Mädchen hat zumindest einen jungen Mann zum Freund. Können n Paare heiraten, so daß jedes der Mädchen einen ihrer Freunde heiratet?

Dieses Problem läßt sich graphisch auf folgende Weise darstellen: Jedem Mädchen ordnen wir einen Punkt M_i, $i = 1, \ldots, n$, zu. Jedem jungen Herren ordnen wir einen Punkt H_j, $j = 1, \ldots, n$, zu. Ist H_j ein Freund von M_i, dann verbinden wir M_i und H_j durch eine Linie (vgl. Abbildung 7.1, Farbabbildungen: S. 150). Mathematisch gesehen handelt es sich in Abbildung 7.1 um einen *paaren Graphen* $(M, H; E)$ mit den *Knotenmengen* M und H und der *Kantenmenge* E. Im folgenden wollen wir immer annehmen, daß die Anzahl $|M|$ der Knoten in M gleich der Anzahl $|H|$ der Knoten in H ist. Jede Kante $e \in E$ verbindet einen Knoten $v \in M$ mit einem Knoten $w \in H$, in unserem Falle ein Mädchen mit einem jungen Herren. Wenn die Kante e die Knoten v und w verbindet, sagen wir, die Kante e *inzidiert* mit den Knoten v und w, bzw. *trifft* diese Knoten. Eine *Heirat* oder *Zuordnung* ist nun eine Teilmenge $Z \subseteq E$, so daß jeder Knoten in M und in H mit einer Kante in Z inzidiert, aber kein Knoten zwei oder mehr Kanten in Z trifft.

Im Jahr 1935 zeigte Philipp Hall [Hall, 1935], daß eine Heirat genau dann existiert, wenn jede Teilmenge von k Mädchen, $k = 1, 2, \ldots, n$, zusammen mindestens k Freunde hat *(Hall-Bedingung)*. In unserem Beispiel von Abbildung 7.1 (Farbabbildungen: S. 150) haben die beiden Mädchen M_2 und M_5 zusammen nur den einen Freund H_1, den sie nicht beide gleichzeitig heiraten können. Daher existiert in diesem Beispiel keine Heirat. Würde jedoch M_2 auch mit H_5 befreundet sein, so könnten die folgenden Paare heiraten: M_1 heiratet H_4, M_2 heiratet H_5, M_3 heiratet H_2, M_4 heiratet H_3, und M_5 heiratet H_1. Aus dieser Beschreibung einer Heirat erkennt man, daß sich Heiraten auch durch sogenannte *Permutationen* darstellen lassen. Statt M_1 heiratet H_4 schreiben wir kurz $1 \longrightarrow 4$. In dieser Schreibweise erhält man weiters

$2 \longrightarrow 5$, $3 \longrightarrow 2$, $4 \longrightarrow 3$ und $5 \longrightarrow 1$. Wie man sieht, wird jedem Index i eines Mädchens M_i der Index j eines jungen Mannes H_j zugeordnet, so daß alle Indizes vorkommen, aber keiner doppelt. Eine solche Zuordnung nennt man *Permutation* der Menge $\{1, 2, \ldots, 5\}$. Die obige Permutation läßt sich nun als $\varphi(1, 2, 3, 4, 5) = (4, 5, 2, 3, 1)$ schreiben. Jede Heirat entspricht einer Permutation und umgekehrt. Im Falle, daß alle jungen Leute miteinander befreundet sind, gibt es $n!$ (sprich: n Fakultät) verschiedene Heiraten (Permutationen). Dabei ist $1! = 1$, $2! = 2$, $3! = 2! \cdot 3 = 6$, $4! = 3! \cdot 4 = 24$ und allgemein $n! = (n-1)! \cdot n$.

So einfach die Hall-Bedingung auch formuliert werden kann, so schwierig ist sie in konkret gegebenen Freundschaftssystemen direkt zu überprüfen. Es müssen dazu alle Teilmengen der Knotenmenge M überprüft werden. Wenn M n Elemente hat, müssen also $2^n - 1$ viele Teilmengen überprüft werden. Man sagt, daß man einen exponentiellen Aufwand dafür treiben muß.

Ein interessantes, schnelles Verfahren, das mit großer Wahrscheinlichkeit feststellt, ob in einem gegebenen Freundschaftssystem eine Heirat existiert, basiert auf einem berühmten Ergebnis des Graphentheoretikers W. T. Tutte [Tutte, 1947]. Gibt es $m = |E|$ Freundschaftsbeziehungen (Kanten im zugehörigen paaren Graphen), so werden für jede dieser Freundschaftsbeziehungen zufällig zwei Zahlen zwischen 1 und $4m^2$ bestimmt. Aus diesen zufällig bestimmten Zahlen wird dann auf schnelle Weise der numerische Wert einer Größe (der Determinante der Tutte-Matrix) bestimmt. Diese Größe ist ungleich Null, wenn das Freundschaftssystem eine Heirat besitzt. Sollte sich bei der Rechnung herausstellen, daß die Größe den Wert 0 hat, wiederholt man die Berechnung mit anderen, zufällig erzeugten Zahlen. Wird auf diese Weise bei r Wiederholungen der Rechnung mit zufällig erzeugten Zahlen immer der Wert 0 ermittelt, so gibt es mit großer Wahrscheinlichkeit keine Heirat. Die Fehlerwahrscheinlichkeit p, daß es doch eine Heirat gibt, ist kleiner als $1/m^r$, wobei m die Anzahl der Kanten (Freundschaftsbeziehungen) ist und r die Anzahl angibt, wie oft die Berechnung der Determinante der Tutte-Matrix wiederholt wurde. Dieses Verfahren liefert nur die Aussage, ob es eine Heirat gibt, oder nicht. Es liefert jedoch keine Hinweise dafür, wer eventuell wen heiraten soll. Im nächsten Abschnitt werden wir ein Verfahren beschreiben, das es uns ermöglicht, auf rasche Weise möglichst viele Paare miteinander zu verheiraten.

Möglichst große Paarungen

Wir wollen in diesem Abschnitt zunächst der folgenden Frage nachgehen: Gegeben sei ein paarer Graph, also ein Freundschaftssystem. Was ist die *maximale* Anzahl von Paaren, die miteinander verheiratet werden können (möglichst große Paarung)? Im Beispiel der Abbildung 7.1 (Farbabbildungen: S. 150) war diese Anzahl 4. Es wird sich herausstellen, daß diese Aufgabe *effizient* lösbar ist. Dies heißt, daß die Anzahl der Rechenschritte nicht exponentiell mit der Anzahl der Mädchen und Herren wächst. Dazu formulieren wir unser Heiratsproblem in ein sogenanntes *maximales Flußproblem* um. Zu diesen Zweck erweitern wir den paaren Graphen durch zwei zusätzliche Knoten, eine *Quelle* q und eine *Senke* s. Nun verbinden wir q durch gerichtete Kanten (*Pfeile*) mit den Knoten in M. Ferner orientieren wir alle Kanten zwischen M und H so, daß die Pfeile von den Knoten in M zu den Knoten in H zeigen. Schließlich verbinden wir alle Knoten von H durch Pfeile mit der Senke s, wobei alle diese Pfeile zur Senke s zeigen (vgl. Abbildung 7.2, Farbabbildungen: S. 150).
Auf diese Weise haben wir ein *Netzwerk* gewonnen. In diesem Netzwerk werden wir einen *Fluß* einführen, der von der Quelle ausgeht und in die Senke einmündet. Dieser Fluß wird dadurch beschrieben, daß den Pfeilen Zahlen zugeordnet werden. In den Knoten von M und H müssen dann die folgenden *Flußerhaltungsgleichungen* erfüllt sein:

> *Die Summe der Kantenflüsse, die in einen Knoten hineinführen ist gleich der Summe der Kantenflüsse, die aus diesem Knoten herausführen.*

Die Bedingung, daß eine Person nur *eine* andere Person heiraten kann, kann nun durch *Kapazitäten* auf den Pfeilen modelliert werden: In unserem Beispiel bekommt jeder Pfeil, der von der Quelle q zu einem Knoten $v \in M$ führt, die Kapazität $c(q, v) = 1$. Analog erhält jeder Pfeil, der von einem Knoten $w \in H$ zur Senke s führt, die Kapazität $c(w, s) = 1$. Für unseren Fluß auf den Pfeilen fordern wir, daß die einzelnen Werte auf den Pfeilen nicht die vorgegebenen Kantenkapazitäten überschreiten (Kapazitätsrestriktionen). Gesucht wird nun ein Fluß, sodaß die gesamte von der Quelle zur Senke gesandte Menge möglichst groß wird. Dieses Problem spielt in Anwendungen eine wichtige Rolle und wird als *maximales Flußproblem* bezeichnet. Um 1956 entwickelten L. R. Ford und D. R. Fulkerson [Ford und Fulkerson, 1956] den ersten Algorithmus zu seiner Lösung. Besonders in den letzten Jahren hat sich die Forschung intensiv mit diesem

Problem auseinandergesetzt und immer schnellere Lösungsverfahren dafür gefunden.

Wie hängt nun ein maximaler Fluß, d.h. eine Lösung des maximalen Flußproblems, mit einer größtmöglichen Paarung zusammen? Zunächst zeigt sich, daß man im betrachteten speziellen Netzwerk immer einen maximalen Fluß finden kann, der auf den Pfeilen entweder den Wert 0 oder 1 und keine Werte dazwischen annimmt. Um eine Paarung zu finden, braucht man nur die Pfeile betrachten, die von einem Knoten in M zu einen Knoten in H führen und den Flußwert 1 haben. Genau diese zeigen an, welches Mädchen in einer größtmöglichen Paarung welchen Herrn heiraten soll. Die Anzahl der möglichen Paare wird durch die Anzahl der Kanten angegeben, die von der Quelle ausgehen und den Flußwert 1 haben, siehe Abbildung 7.3 (Farbabbildungen: S. 151).
Hopcroft und Karp [Hopcroft und Karp, 1973] haben gezeigt, daß auf diese Weise eine größtmögliche Paarung in $K \cdot m\sqrt{n}$ Rechenschritten gefunden werden kann, wenn das Freundschaftssystem m Kanten hat. Dabei ist K eine fest vorgegebene Konstante, die nur vom gewählten Rechenverfahren, nicht aber von den Problemgrößen m und n abhängt. Natürlich kann man dieses Verfahren auch verwenden um festzustellen, ob es eine Heirat gibt. Man hat dazu ein maximales Flußproblem zu lösen, bei dem der maximale Flußwert, d.h. die Summe der Flüsse auf den Pfeilen, die aus der Quelle herausführen, gleich n ist. Nachdem diese Frage in $K \cdot m\sqrt{n}$ Schritten beantwortet werden kann, hat man auf diese Weise ein recht schnelles Verfahren zur Lösung des Heiratsproblemes zur Hand. Man erhält dabei nicht nur die Antwort, ob es eine Heirat gibt, sondern auch den Vorschlag für eine tatsächliche solche Heirat. Wir werden nun im nächsten Abschnitt zeigen, daß derartige Heiratsprobleme auch interessante andere Anwendungen haben, etwa in der Kommunikation über Satelliten.

Eine Anwendung bei der Kommunikation über Satelliten

Bei der Nachrichtenübermittlung über Satelliten hat sich inzwischen auch die digitale Technik durchgesetzt. Bei der Übertragung eines Telefongespräches von Europa nach Amerika werden zunächst die Daten binär kodiert und in der sendenden Erdfunkstelle zwischengespeichert. Dann werden sie in ganz kurzen Datenstößen an den Nachrichtensatelliten übermittelt. Dort werden sie von einem sogenannten Transponder empfangen, verstärkt und auf einer anderen Frequenz zur empfangenden Erdfunkstelle in Amerika weitergeleitet (vgl. Abbildung 7.4, Farbabbildungen: S. 151).

An einem solchen System nehmen mehrere Erdfunkstellen gleichzeitig teil. Dafür stehen im Satelliten mehrere Transponder zur Verfügung. Ein Transponder kann aber in einem gewissen Zeitintervall jeweils nur ein Paar von Erdfunkstellen miteinander verbinden. Dabei entsteht folgendes Problem: Die n Erdfunkstellen $E_1, E_2, \ldots, E_n$ in Europa sollen Nachrichten an n Erdfunkstellen $A_1, A_2, \ldots, A_n$ in Amerika übermitteln. Für die Übermittlung der Daten von E_i an A_j werden genau t_{ij} Zeiteinheiten benötigt. Im Satelliten stehen n Transponder zur Verfügung. Auf welche Weise kann die Übermittlung der gesamten Daten in möglichst kurzer Zeit erfolgen? Die Schwierigkeit, die mit dieser Vorgehensweise verbunden ist, besteht darin, daß eine Erdfunkstelle in Europa nicht gleichzeitig Daten an verschiedene Erdfunkstellen in Amerika übermitteln kann. Zu jedem Zeitpunkt sind durch die Transponder an Bord des Satelliten die Erdfunkstellen in Europa mit festen Empfangsstationen in Amerika verbunden. Aber diese Zuordnung einzelner Stationen zueinander kann sich nach einer kurzen Zeit ändern. Jede solche Zuordnung entspricht einer Heirat aus dem ersten Abschnitt, die jedoch nur eine kurze Zeit andauert und dann durch eine neue Zuordnung (Heirat) ersetzt wird. Das *Zeitschlitz-Zuordnungsproblem* in der Nachrichtentechnik kann also folgenderweise formuliert werden: Gegeben sind Übermittlungszeiten t_{ij} von E_i, $i = 1, 2, \ldots, n$, nach A_j, $j = 1, 2, \ldots, n$. Gesucht werden Zuordnungen φ_k zwischen den Stationen E_i und A_j sowie Zeiten λ_k, die angeben, wie lange die Zuordnung φ_k andauern soll, daß insgesamt alle Daten in der kürzestmöglichen Zeit übermittelt werden können.

Um dieses Problem zu lösen, können wir folgenderweise vorgehen. Zunächst sei t^* die längste Zeit, die von einer einzelnen Station zum Senden oder Empfangen der Daten benötigt wird. Das Verfahren, das wir nun angeben, wird eine Lösung ermitteln, die garantiert, daß in t^* Zeiteinheiten die Daten *aller* Stationen übermittelt werden. Falls erforderlich, erhöhen wir einige Übermittlungszeiten, daß jede teilnehmende Station in Summe die Zeit t^* zum Senden oder Empfangen ihrer Daten benötigt. Wir bilden nun einen paaren Graphen mit den Knoten E_i, $i = 1, 2, \ldots, n$, und A_j, $j = 1, 2, \ldots, n$. Zwei Knoten E_i und A_j sind miteinander verbunden, wenn t_{ij} positiv ist. Aufgrund eines Satzes von Birkhoff [Birkhoff, 1946] weiß man nun, daß dieser Graph sicher eine Heirat besitzt. So eine Heirat, die einer Permutation φ_1 entspricht, kann nach Abschnitt *Möglichst große Paarungen* in $K \cdot m\sqrt{n}$ Schritten gefunden werden, wobei m die Anzahl der positiven Zahlen t_{ij}

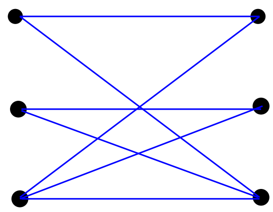
ist. Wie lange soll nun diese Zuordnung andauern? Dazu bestimmen wir

$$\lambda_1 = \min_{1 \leq i \leq n} t_{i,\varphi_1(i)}.$$

Dies heißt, daß λ_1 die kleinste der Zahlen $t_{1,\varphi_1(1)}, t_{2,\varphi_1(2)}, ..., t_{n,\varphi_1(n)}$ ist. Dadurch wird gewährleistet, daß während λ_1 Zeiteinheiten die Stationen E_i an die Stationen $A_{\varphi_1(i)}$, $i = 1, 2, ..., n$, Nachrichten übermitteln und keine der Stationen nicht arbeitet. Wir lassen nun diese Stationen ihre Daten in dieser Zeit übermitteln. Danach haben alle Stationen den Zeitbedarf von $t^* - \lambda_1$ Zeiteinheiten für die Übermittlung bzw. den Empfang der verbliebenen Daten. Nun aber können wir das gleiche Verfahren wie oben anwenden: Wir stellen wieder den zugehörigen paaren Graphen auf, der nun mindestens eine Kante weniger besitzt, da im ersten Zeitintervall ein Paar von Sendern und Empfängern bereits alle Daten austauschen konnte. In diesem paaren Graphen wird nun eine neue Zuordnung φ_2 bestimmt und für diese Zuordnung die Zeitdauer λ_2 analog wie oben ermittelt. Da in jedem Schritt die Anzahl der Kanten des paaren Graphen um mindestens 1 abnimmt, bricht das Verfahren nach endlich vielen Schritten ab. Am Ende des Verfahrens sind alle Daten übermittelt. Die Konstruktion im oben beschriebenen Rechenverfahren garantiert, daß $\lambda_1 + \lambda_2 + \ldots = t^*$ gilt.

Beispiel.

$$(t_{ij}) = \begin{pmatrix} 3 & 0 & 3 \\ 0 & 4 & 2 \\ 3 & 2 & 1 \end{pmatrix} \qquad \text{mit} \qquad t^* = 6.$$

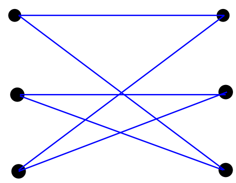


Dabei steht das Element t_{ij} in der i-ten Zeile und j-ten Spalte des obigen Schemas, z.B. $t_{32} = 2$. Im ersten Schritt des Verfahrens wählen wir die Zuordnung $\varphi_1(1, 2, 3) = (1, 2, 3)$ und erhalten

$$\lambda_1 = \min\{t_{11}, t_{22}, t_{33}\} = \min\{3, 4, 1\} = 1.$$

Nachdem diese Werte von den Anfangswerten abgezogen wurden, erhält man

$$(t_{ij}) = \begin{pmatrix} 2 & 0 & 3 \\ 0 & 3 & 2 \\ 3 & 2 & 0 \end{pmatrix} \qquad \text{mit} \qquad t^* = 6.$$

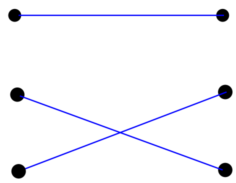


Da $t_{33} = 0$ ist, enthält der neue paare Graph nicht mehr die Kante von E_3 nach A_3. Wir können als neue Zuordnung φ_2 die Zuordnung $\varphi_2(1,2,3) = (3,2,1)$ wählen und erhalten

$$\lambda_2 = \min\{t_{13}, t_{22}, t_{3,1}\} = \min\{3,3,3\} = 3.$$

Damit erhalten wir als neue Daten

$$(t_{ij}) = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 0 & 2 \\ 0 & 2 & 0 \end{pmatrix} \qquad \text{mit} \qquad t^* = 12.$$



Daher lautet die letzte Zuordnung $\varphi_3(1,2,3) = (1,3,2)$ und $\lambda_3 = 2$. Damit haben wir einen Weg gefunden, alle Daten innerhalb eines Zeitintervalles von $\lambda_1 + \lambda_2 + \lambda_3 = 6$ Zeiteinheiten zu übermitteln.

Das obige Rechenverfahren liefert im allgemeinen sehr viele verschiedene Zuordnungen φ_k, die nur sehr kurze Zeiten λ_k verwendet werden. Dies macht dieses Verfahrens in der Praxis unbrauchbar. Möchte man eine optimale Lösung erhalten, die nur n Zuordnungen verwendet, so erhält man ein sehr schweres Problem, das bis jetzt grundsätzlich nur durch systematisches Ausprobieren aller Möglichkeiten lösbar ist. Es gibt nämlich in der diskreten algorithmischen Mathematik zwei große Problemklassen $\mathcal{P}$ und *$\mathcal{NP}-$vollständig*. Die Probleme der Komplexitätsklasse $\mathcal{P}$ zeichnen sich dadurch aus, daß es für sie Lösungsverfahren gibt, die nur polynomiell viele Rechenschritte in den (binär kodierten) Eingangsdaten verwenden. Während sich also die Probleme in der Klasse $\mathcal{P}$ durch effiziente Verfahren lösen lassen, kennt man kein solches Verfahren für die Probleme in der Klasse *$\mathcal{NP}-$vollständig*. Würde man ein Problem in dieser Klasse

$\mathcal{NP}-$*vollständig* effizient lösen können, so könnte man *alle* Probleme in dieser Klasse effizient lösen. Nach dem heutigen Kenntnisstand ist es aber ziemlich unwahrscheinlich, daß es schnelle Lösungsverfahren für die Probleme in der Klasse $\mathcal{NP}-$*vollständig* gibt. Man ist daher zum Lösen solcher Probleme auf systematisches Durchsuchen von großen Teilmengen aller Lösungen angewiesen, was einen exponentiellen Aufwand, gemessen an den Eingangsdaten, mit sich bringt.

Um nun doch zu einer praktisch brauchbaren Lösung für das Zeitschlitzzuordnungsproblem zu kommen, haben Lewandowski, Liu und Liu [Lewandowski et al., 1983] vorgeschlagen, an Bord des Satelliten $2n$ Zuordnungen fix zu verdrahten und nur diese Zuordnungen zu verwenden. Wie in Burkard [Burkard, 1991] gezeigt wurde, ist dieses Problem dann wieder effizient lösbar, denn es führt auf die Lösung eines *linearen Zuordnungsproblems.* Für nähere Einzelheiten sei auf [Burkard, 1991] verwiesen.

Lineare Zuordnungsprobleme

Kehren wir zu unserem Heiratsproblem zurück. Wir lassen nun jedes Mädchen M_i angeben, wie gerne es den Herrn H_j hat. Dieser Beliebtheitskoeffizient sei c_{ij}, der umso größer ist, je lieber das Mädchen M_i den jungen Herren H_j hat. Sind alle n^2 Koeffizienten c_{ij}, $i, j = 1, 2, \ldots, n$, gegeben, kann man nach einer Heirat (Zuordnung) φ suchen, die das Gesamtglück der heiratenden Paare maximiert. Das heißt, es ist eine Zuordnung φ zu finden, die die *Zielfunktion*

$$c_{1,\varphi(1)} + c_{2,\varphi(2)} + \ldots + c_{n,\varphi(n)}$$

maximiert. Dieses Problem heißt *lineares Zuordnungsproblem*. Meist wird die Zielfunktion in einem linearen Zuordnungsproblem nicht maximiert, sondern minimiert. Wir können aber leicht ein Maximierungsproblem in ein Minimierungsproblem überführen. Dazu gehen wir so vor: Sei C eine Konstante, die größer als jeder einzelne der Koeffizienten c_{ij} ist. Man definiert $d_{ij} := C - c_{ij}$ als *Verdruß*, der bei einer Verbindung von M_i mit H_j entsteht. Äquivalent zum Maximierungsproblem von oben ist die Minimierung des Gesamtverdrusses bei einer Heirat, nämlich

$$\text{Minimiere } d_{1,\varphi(1)} + d_{2,\varphi(2)} + \ldots + d_{n,\varphi(n)}.$$

Zur Lösung linearer Zuordnungsprobleme gibt es mehrere schnelle Methoden, die auf Ideen der linearen Optimierung oder auf Verfahren zur Lösung von Flußproblemen in Graphen (vgl. Abschnitt *Möglichst große Paarungen*) beruhen. So können diese Probleme etwa mit $K \cdot n^3$ Rechenoperationen gelöst werden, wobei K wieder eine Konstante ist, die nicht von den Eingangsdaten des Problemes, sondern vom verwendeten Lösungsverfahren abhängt.

Auch in der Praxis kommen lineare Zuordnungsprobleme immer wieder vor: Im letzten Abschnitt wurde bereits erwähnt, daß das Zeitschlitzzuordnungsproblem bei fest verdrahteten Zuordnungen an Bord des Satelliten durch ein lineares Zuordnungsproblem gelöst werden kann. Brogan [Brogan, 1989] beschreibt, wie man n anfliegende Objekte im Raum lokalisieren und verfolgen kann. Dazu wird von 2 Radarstationen aus der Winkel gemessen, unter denen man diese Objekte zum Zeitpunkt t_0 sieht. Diese Winkel bestimmen Geraden im Raum, auf denen sich die Objekte zur Zeit t_0 befinden. Die von der ersten Radarstation ermittelten Geraden seien $g_1, g_2, ..., g_n$, die von der zweiten Radarstation ermittelten Geraden seien $h_1, h_2, ..., h_n$. Was man nun nicht weiß, ist, welche der Geraden g_i und h_j dasselbe Objekt bestimmen. Um dies herauszufinden, löst man ein lineares Zuordnungsproblem mit den Koeffizienten c_{ij}, die als kürzeste Abstände zwischen den Geraden g_i und h_j definiert werden. (Man beachte, daß aufgrund von Meßfehlern sich die Geraden, die dasselbe Objekt bestimmen, nicht in einem Punkte schneiden müssen). Auf diese Weise erhält man die Paarung der richtigen Geraden, die jeweils das gleiche Objekt bestimmen und damit kann man auch die Koordinaten der Objekte zum Zeitpunkt T_0 bestimmen. Wiederholt man dieses Verfahren für einen Zeitpunkt t_1 kurze Teit später, so kann man auch die Koordinaten der Objekte zu diesem Zeitpunkt ermitteln. Löst man nun ein weiteres lineares Zuordnungsproblem mit den Koeffizienten d_{ij}, wobei d_{ij} die Entfernung des i-ten Objektes zum Zeitpunkt t_0 zum j-ten Objekt zum Zeitpunkt t_1 angibt, dann kann man die Bahn der Objekte rekonstruieren und die Objekte somit auf ihrer Flugbahn verfolgen.

Wir können aber auch eine Modifikation des linearen Zuordnungsproblems betrachten. Anstelle wie oben die Summe der einzelnen Verdrußkoeffizienten zu minimieren, können wir nach einer Heirat suchen, die den maximalen Einzelverdruß so klein als möglich macht. Dies führt auf sogenannte Engpaß-Zuordnungsprobleme der Form

$$\text{Minimiere } \max(d_{1,\varphi(1)}, d_{2,\varphi(2)}, \ldots, d_{n,\varphi(n)}).$$

Diese Engpaß-Zuordnungsprobleme sind schneller lösbar als die analogen Zuordnungsprobleme, in denen eine Summe minimiert werden soll. Auch im obigen Objekt-Lokalisierungsproblem würde es mehr Sinn machen, Engpaß-Zuordnungsprobleme anstelle gewöhnlicher linearer Zuordnungsprobleme zu lösen. Darüber hinaus treten Engpaß-Zuordnungsprobleme etwa in folgendem Zusammenhang auf: Zur Erledigung von n Arbeiten stehen n Leute zur Verfügung, die aber nicht alle gleich gut für die einzelnen Arbeiten geeignet sind. Es sei d_{ij} der Zeitbedarf, den die i-te Person zur Erledigung der j- ten Aufgabe benötigt. Um nun alle Aufgaben in der kürzestmöglichen Zeit zu erledigen, löst man ein Engpaß-Zuordnungsproblem mit den Koeffizienten d_{ij}.

Rundreiseprobleme

Im vorigen Abschnitt wurde erwähnt, daß sich lineare Zuordnungsprobleme effizient lösen lassen. Hier betrachten wir nun ein Zuordnungsproblem mit einer zusätzlichen Bedingung. Wir lassen nicht mehr alle Permutationen als Lösungen zu, sondern nur mehr die sogenannten *zyklischen Permutationen.* Eine Permutation φ heißt zyklisch, wenn man durch $1, \varphi(1), \varphi(\varphi(1)), \varphi(\varphi(\varphi(1))), ...$ alle Zahlen $1, 2, ..., n$ erhält. So ist etwa die Permutation $\varphi_1(1,2,3,4) = (2,3,4,1)$ zyklisch, während die Permutation $\varphi_2(1,2,3,4) = (2,1,4,3)$ nicht zyklisch ist. Letztere Permutation zerfällt vielmehr in 2 Zyklen $1 \to 2 \to 1$ und $3 \to 4 \to 3$.

Zyklische Permutationen spielen bei sogenannten *Rundreiseproblemen* eine tragende Rolle. Gegeben seien n Objekte, z.B. Punkte im Raum oder Städte in Europa. Ferner sei d_{ij} die Entfernung zwischen dem i-ten und j-ten Punkt (bzw. der i-ten und j-ten Stadt). Wir suchen nun eine Tour (Rundreise), die alle Punkte genau einmal besucht und dann zum Ausgangspunkt zurückkehrt. So eine Tour oder Rundreise entspricht einer zyklischen Permutation. Fordern wir außerdem, daß die Tour eine minimale Gesamtlänge hat, so müssen wir das folgende *Rundreiseproblem* lösen:

> *Man finde eine zyklische Permutation φ, die $d_{1,\varphi(1)} + d_{2,\varphi(2)} + \ldots + d_{n,\varphi(n)}$ minimiert.*

Auch Rundreiseprobleme haben zahlreiche interessante Anwendungen, sei es die Erstellung von optimalen Touren für die tägliche Zeitungszustellung, das Leeren von Briefkästen oder die optimale Bestimmung einer Produktionsreihenfolge für Aufträge auf einer Maschine, die zwischen der Bearbeitung zweier verschiedener Aufträge gereinigt werden muß (vgl. Abbildung 7.5, Farbabbildungen: S. 152).

Obwohl das Rundreiseproblem formal einem linearen Zuordnungsproblem sehr ähnlich sieht, ist es viel schwerer zu lösen. Es gehört nämlich der Klasse der $\mathcal{NP}-$*vollständigen* Probleme an, für die nur Verfahren bekannt sind, die eine große Teilmenge der zulässigen Lösungen systematisch absuchen.

Man kann sich aber die Frage stellen: Gibt es Bedingungen für die Koeffizienten d_{ij} eines Rundreiseproblems, so daß das entsprechende Problem effizient lösbar wird? Und tatsächlich kennt man eine ganze Reihe von solchen Bedingungen, siehe etwa den jüngst erschienenen Übersichtsartikel von Burkard et al. [Burkard et al., 1998b]. Eine solche Bedingung ist etwa die *Monge Bedingung*, benannt nach dem großen französischen Geometer Gaspard Monge (1746–1818). Monge stellte bei Schanzarbeiten an Verteidigungswällen fest, daß es günstiger ist, wenn die Karren direkt an den Wall heranfahren, ohne sich zu kreuzen. Überträgt man diese Bedingung auf Rundreiseprobleme, so erhält man

Monge Bedingung

Die Koeffizienten d_{ij} erfüllen die Mongebedingung wenn gilt

$$d_{ij} + d_{kl} \leq d_{il} + d_{kj} \text{ für } 1 \leq i < k \leq n, 1 \leq j < l \leq n.$$

Natürlich erfüllt nicht jedes Rundreiseproblem so eine Bedingung. Aber wenn diese Bedingung erfüllt ist, kann man das Rundreiseprobem in $K \cdot n$ Schritten lösen, wobei die Konstante K unabhängig von den Problemdaten ist.

Quadratische Zuordnungsprobleme

Nachdem wir im 1.Abschnitt Heiraten betrachtet haben, wenden wir uns nun der Sitzordnung an der Hochzeitstafel zu. Dazu kann man das folgende mathematische Modell verwenden. Wir nehmen an, daß n Gäste kommen und n Plätze an der Tafel zur Verfügung stehen. Von einem Platz an der Tafel aus kann man sich gut mit den Nachbarplätzen unterhalten und einigermaßen gut auch mit gegenüberliegenden und übernächsten Nachbarplätzen (vgl. Abbildung 7.6, Farbabbildungen: S. 152).

Wir messen die Kommunikationsmöglichkeit durch nichtnegative Kommunikationskoeffizienten a_{ij}, $i, j = 1, 2, ..., n$. Dabei drückt a_{ij} aus, wie gut man sich zwischen den Plätzen i und j unterhalten kann. Je besser, umso höher ist dieser Koeffizient. Ferner sollen die Größen b_{kl}, $k, l = 1, 2, ..., n$, beschreiben, wie gerne sich die $k-$te Person mit dem $l-$ten Gast der Hochzeitstafel unterhalten würde. Gesucht wird nun eine Zuordnung φ der Plätze an der Tafel zu den Gästen, daß die Gesamtrunde möglichst fröhlich ist. Dies führt auf das Modell

$$\text{Maximiere } \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} \cdot b_{\varphi(i),\varphi(j)}$$

Dieses Problem heißt *quadratisches Zuordnungsproblem* und spielt in vielerlei Anwendungen eine sehr wichtige Rolle. Dabei wird es meist als Minimierungsproblem formuliert. Im folgenden werden einige Anwendungen des quadratischen Zuordnungsproblems angeführt:

Entwurf von Schreibmaschinentastaturen ([Burkard und Offermann, 1977])

Es soll eine Schreibmaschinentastatur entwickelt werden, die es gestattet, Texte in bestimmten Sprachen möglichst schnell zu schreiben. n sei die Anzahl von Zeichen, die auf der Tastatur plaziert werden sollen. Die Koeffizienten a_{ij} beschreiben, wie schnell es möglich ist, nach der Taste i die Taste j zu drücken. Die Größen b_{kl} beschreiben die Häufigkeit, wie oft die Zeichenpaare k, l im Text auftreten. Die Zuordnung φ belegt jede Taste mit einem Zeichen, so daß die Gesamtsumme $\sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} \cdot b_{\varphi(i),\varphi(j)}$ minimal wird. Diese Gesamtsumme entspricht aber genau der durchschnittlichen Zeitdauer, die zum Schreiben eines Textes nötig ist.

Campusplanung ([Dickey und Hopkins, 1972])

Auf einem neuen Gelände soll ein neuer Universitätscampus mit n Gebäuden errichtet werden. Dazu stehen am Gelände n Bauplätze zur Verfügung. Jedes der Gebäude hat eine spezielle Funktion: Bibliothek, Institutsgebäude, Mensa, Studentenwohnheim etc. Die Entfernungen zwischen den Bauplätzen werden durch die Größen a_{ij} beschrieben. Die Koeffizienten b_{kl} geben an, wie oft in einer Woche Leute zwischen einem Gebäude der Funktion k und einem Gebäude der Funktion l hin- und hergehen. Gesucht ist eine Zuordnung der einzelnen Gebäude zu den Bauplätzen, so daß die wöchentlich zurückgelegte Gesamtentfernung aller Betroffenen minimiert wird. Man sucht also eine Zuordnung φ der Bauplätze zu den Gebäuden, so daß $\sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} \cdot b_{\varphi(i),\varphi(j)}$ minimiert wird.

Weitere Anwendungen des quadratischen Zuordnungsproblems betreffen die Anordnung von Schaltelementen auf einer Kontrolleinheit, um die Ermüdung der Augen des Bedienungspersonals zu minimieren, den Entwurf von gut geplanten Spitälern, das zeitliche Reihen von archäologischen Funden, die optimale Reihung von Staffelläufern, die optimale Planung paralleler Produktionslinien oder etwa die Analyse organisch-chemischer Verbindungen.

Quadratische Zuordnungsprobleme sind leider sehr schwer optimal zu lösen. Sie gehören nämlich auch der Klasse der $\mathcal{NP}$*−vollständigen* Probleme an. Eines der größten quadratischen Zuordnungsprobleme wurde im Jahr 1997 an der ETH in Zürich optimal gelöst, und zwar für $n = 25$. Dazu wurde wochenlang die Rechenkraft paralleler Großcomputer ausgenützt.

Man kann sich nun die Frage stellen, warum gerade quadratische Zuordnungsprobleme so schwer zu lösen sind. Eine Teilantwort darauf gibt eine Theorie, die von Burkard und Fincke [Burkard und Fincke, 1983] entwickelt wurde und die das Verhalten von großen kombinatorischen Optimierungsproblemen beschreibt. Während etwa bei Rundreiseproblemen die Werte für die beste und für die schlechteste Lösung immmer weiter auseinandergehen, wenn die Problemgröße zunimmt, hat man bei quadratischen Zuordnungsproblemen gerade das umgekehrte Verhalten. Man kann zeigen, das bei zufällig erzeugten Problemen mit wachsender Problemgröße alle Lösungen fast sicher einen konstanten Wert annehmen. Das bedeutet, daß sich die Zielfunktionswerte der einzelnen Lösungen nur sehr wenig unterscheiden. Daher ist es so schwierig, aus all den Lösungen, die fast den gleichen Wert haben, die wirklich beste herauszufinden. Andrerseits kann man dieses seltsame Phänomen aber auch positiv bewerten: Da die Lösungen fast alle den optimalen Wert ergeben, findet man sehr rasch sehr gute Lösungen, ohne jedoch zu wissen, ob sie wirklich optimal sind. Dies ist für die Praxis aber meist gar nicht so wichtig. Viel wichtiger ist es, rasch viele gute Lösungen zu finden und dies ist nach dieser Theorie umso eher möglich, je größer das Problem ist. Tatsächlich stützen die numerischen Ergebnisse von vielen Rechentests diese theoretischen Vorhersagen.

Ausblick

In den vorhergehenden Abschnitten haben wir gesehen, daß es zwei wichtige Klassen von kombinatorischen Optimierungsaufgaben gibt: jene der Komplexitätsklasse $\mathcal{P}$, die durch effiziente Verfahren gelöst werden können, und jene der $\mathcal{NP}-$*vollständigen* Probleme, die sich bis jetzt einer effizienten Lösung hartnäckig widersetzen. Zu den Forschungsaufgaben zählt bei beiden Klassen die Entwicklung von guten Lösungsverfahren. Bei der zweiten Klasse ist man insbesondere auch an guten Näherungsverfahren interessiert. Weitere Forschungsgesichtspunkte betreffen das Identifizieren von schnell lösbaren Spezialfällen $\mathcal{NP}-$schwerer Probleme, das Verhalten von Problemen bei wachsender Problemgröße und natürlich vor allem die Anpassung der Modelle an die Gegebenheiten der Praxis.

Für ein tieferes Eindringen in den Themenkreis dieses Artikels sei die interessierte Leserin oder der interessierte Leser auf das Buch von D. Jungnickel: *Graphen, Netzwerke und Algorithmen*, Bibliographisches Institut Wissenschaftsverlag, Mannheim, 1994 hingewiesen. Wer sich für den neuesten Stand der Forschung auf dem Gebiet der Zuordnungsprobleme interessiert, sei auf die beiden Übersichtsartikel [Burkard und Çela, 1999] und [Burkard et al., 1998a] im *Handbook of Combinatorial Optimization* hingewiesen.

Literatur

[Birkhoff, 1946] Birkhoff, G., (1946). Tres observaciones sobre el algebra lineal. *Rev. univ. nac. Tucum n (A)* 5, 147–151.

[Brogan, 1989] Brogan, W. L., (1989). Algorithm for ranked assignments with application to multiobject tracking. *Journal of Guidance* 12, 357–364.

[Burkard, 1991] Burkard, R. E., (1991). Time Division Multiple Access systems and matrix decomposition. In *Proceedings of the Fourth European Conference on Mathematics in Industry*, von H. J. Wacker und W. Zulehner, Hrsg., Teubner, Stuttgart, 35–46.

[Burkard und Çela, 1999] Burkard, R. E. und Çela, E. (1999). Linear Assignment problems and extensions. Bericht Nr.127, SFB Optimierung und Kontrolle, Technische Universität Graz, Juni 1998. Erscheint im *Handbook of Combinatorial Optimization*, Kluwer, Dordrecht, 1999.

[Burkard et al., 1998a] Burkard, R. E., Çela, E., Pardalos, P. M. und Pitsoulis, L. S. (1999). The Quadratic Assignment Problem. *Handbook of Combinatorial Optimization*, vol. 3, 241–337, Du D. Z. and Pardalos, P. M., Hrsg. Kluwer, Dordrecht, 1998.

[Burkard et al., 1998b] Burkard, R. E., Deĭneko, V. G. van Dal, R., van der Veen, J. und Woeginger G. J. (1998). Well solvable special cases of the travelling salesman problem: a survey. *SIAM Review* 40, 1998, 496–546.

[Burkard und Fincke, 1983] Burkard, R. E. und Fincke, U. (1983). The asymptotic probabilistic behaviour of the quadratic sum assignment problem. *Zeitschrift für Operations Research* 27, 73–81.

[Burkard und Offermann, 1977] Burkard, R. E. und Offermann, J. (1977). Entwurf von Schreibmaschinentastaturen mittels quadratischer Zuordnungsprobleme. *Zeitschrift für Operations Research* 21, B121–B132.

[Dickey und Hopkins, 1972] Dickey, J. W. und Hopkins, J. W. (1972). Campus building arrangement using TOPAZ. *Transportation Research* 6, 59–62.

[Ford und Fulkerson, 1956] Ford, L. R. und Fulkerson, D. R. (1956). Maximal flow through a network. *Canadian J. Math* 8, 399–404.

[Hall, 1935] Hall, Ph. (1935). On representatives of subsets. *J. London Math. Soc.* 10, 26-30.

[Hopcroft und Karp, 1973] Hopcroft, J. E. und Karp, R. M. (1973). An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. *SIAM Journal on Computing* 2, 225–231.

[Karmarkar et al., 1993] Karmarkar, N., Karp, R., Lipton, R., Lovász, L. und Luby, M. (1993). A Monte-Carlo algorithm for estimating the permanent. *SIAM J. Comput.* 22, 284–293.

[Lewandowski et al., 1983] Lewandowski, J. L., Liu J. W. S. und Liu, C. L. SS/TDMA time slot assignment with restricted switching modes. *IEEE Transactions on Communications*, COM-23, 149-154.

[Tutte, 1947] Tutte, W.T. (1947). The factorization of linear graphs. *J. London Math. Soc.* 22, 107–111.